

DIAView

Структура системи

Веббер
2019/04/12



Контур

- Терміни та поняття DIAView
- Середовище розробки
- Основні функції та кнопки
- Як почати проект

призначення

У цьому розділі ви дізнаєтеся....

- ... докладніше про терміни та поняття DIA.
- ... Детальніше про середовище розробки.
- ... Детальніше про основні функції та кнопки.
- ... Детальніше про те, як почати проект.

- Терміни та поняття DIAView
- Середовище розробки
- Основні функції та кнопки
- Як почати проект

Стабільний промисловий комунікаційний дизайн

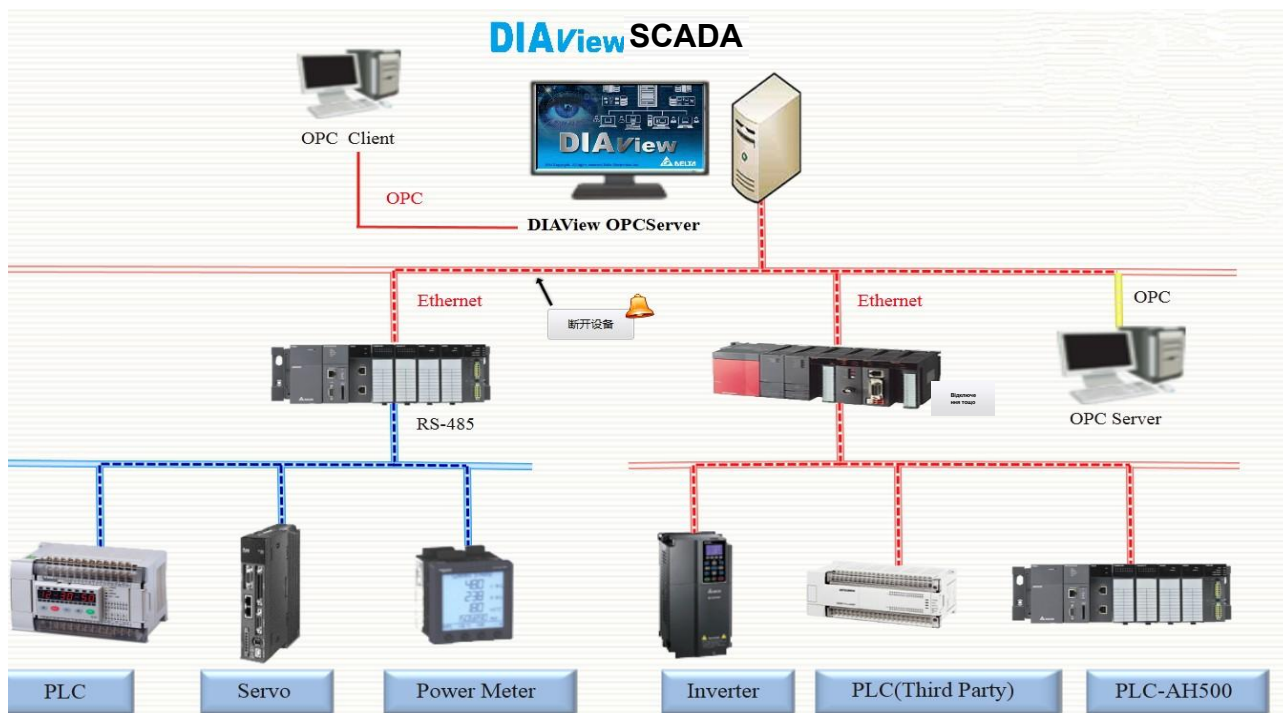


Схема підключення зв'язку системи SCADA

Вступ до DIAView

➤ Демонструвати

- Проект — це набір автоматизованої прикладної системи, що включає інтерфейс графічного вікна, комунікацію вводу-виводу, сигналізацію, змінну та інші частини.

➤ Середовище розробки та виконання

- Програмне забезпечення DIAView складається з двох частин.

➤ Середовище розробки

- Виконання розробки дизайну проекту, в т.ч проектування графічного інтерфейсу, налаштування параметрів та конфігурування функцій тощо.

➤ Середовище виконання

- Середовище забезпечує інтерфейс спостереження та контролю в реальному часі.

Вступ до DIAView

➤ **ІО зв'язок, канал, обладнання**

- Зв'язок вводу/виводу стосується збору даних про обладнання на місці та використання зв'язку для введення/виведення інформації в змінні.

➤ **Варіативний словник, змінні**

- Змінні - це значення інженерних елементів, які можуть змінюватися в будь-який час; Вони є важливими учасниками рукостискання системних даних. Словник змінних — це набір змінних у проекті, який виконує групове керування змінними

➤ **вікна**

- Вікно є основою для реалізації візуалізації в реальному часі – воно надає різні елементи малювання, включаючи основні графічні інструменти, блоки керування вікном.

➤ **Властивості**

- Властивості описують особливості або характерну інформацію. Наприклад ім'я, розмір і колір фону тощо.

Вступ до DIAView

➤ **Анімація**

- Анімація відноситься до процесу встановлення властивостей графічного об'єкта, зміни відповідно до змін у змінних даних під час виконання проекту.

➤ **Подія**

- Подія відноситься до процесу визначення властивостей графічного об'єкта, включаючи розмір, колір і положення, які можна змінювати за допомогою операцій миші або клавіатури.

➤ **сигналізація**

- Сигнали тривоги стосуються пов'язаних сповіщень, які запускаються, щоб нагадувати пов'язаному персоналу звернути увагу, коли певні дані або стан системи перевищують або нижче попередньо встановленого значення.

➤ **Змінні історичних записів**

- Відноситься до збереження змінених записів певних значень змінних у словнику змінних і визначає метод запису для статистики та аналізу даних.

Вступ до DIAView

➤ Програма користувача («програма умов» і «програма часу»)

- Програма умов стосується виконання функцій програми, коли набір умов відповідає змінним. Програма часу стосується виконання пов'язаних функцій програми кожен певний період часу.

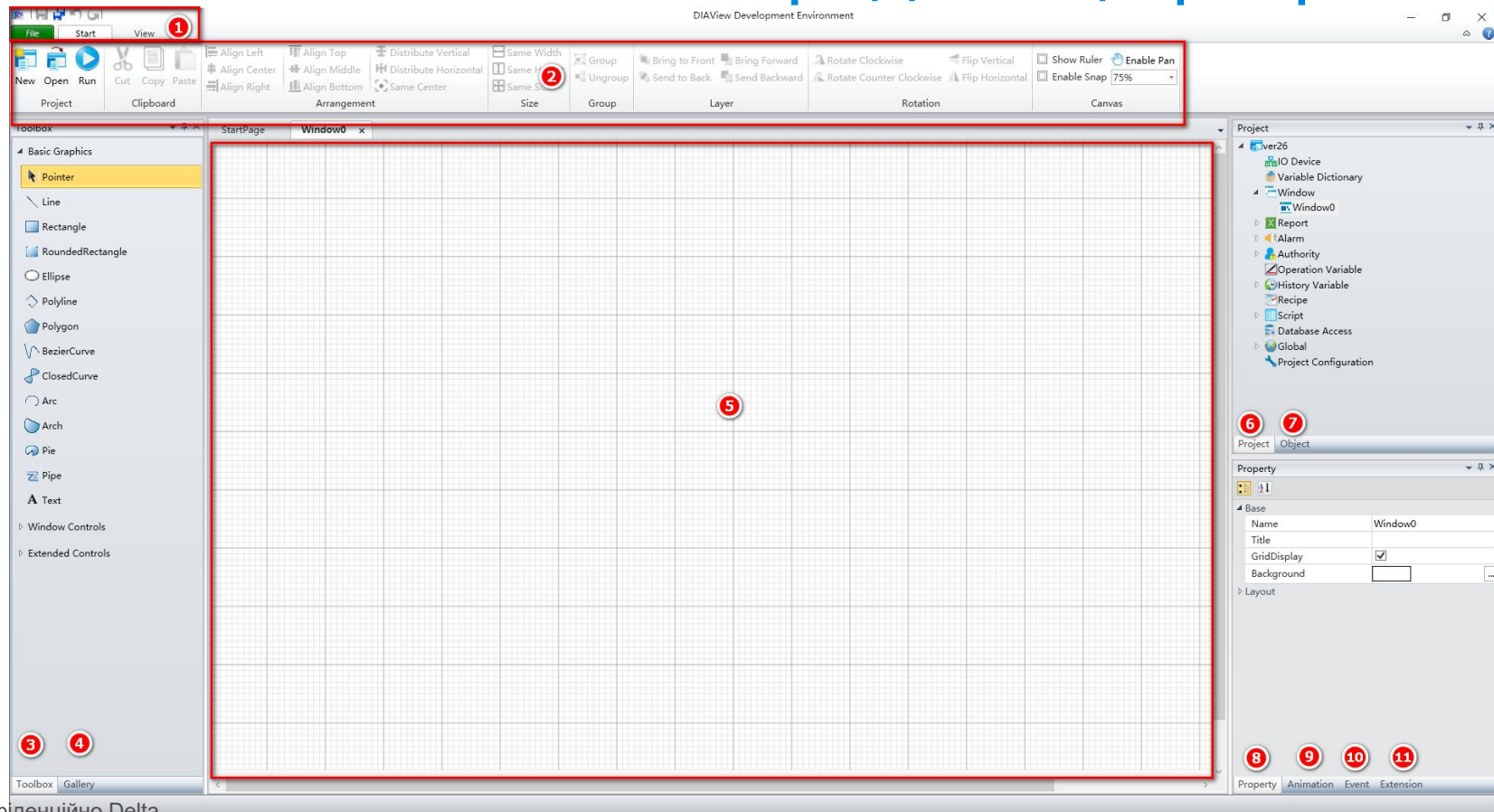
➤ Інженерна конфігурація

- Інженерна конфігурація включає налаштування бази даних, налаштування екрана виконання та налаштування початкового екрана.

- Терміни та поняття DIAView
- Середовище розробки
- Основні функції та кнопки
- Як почати проект

Середовище розробки

DIAView Development Environment



Середовище розробки

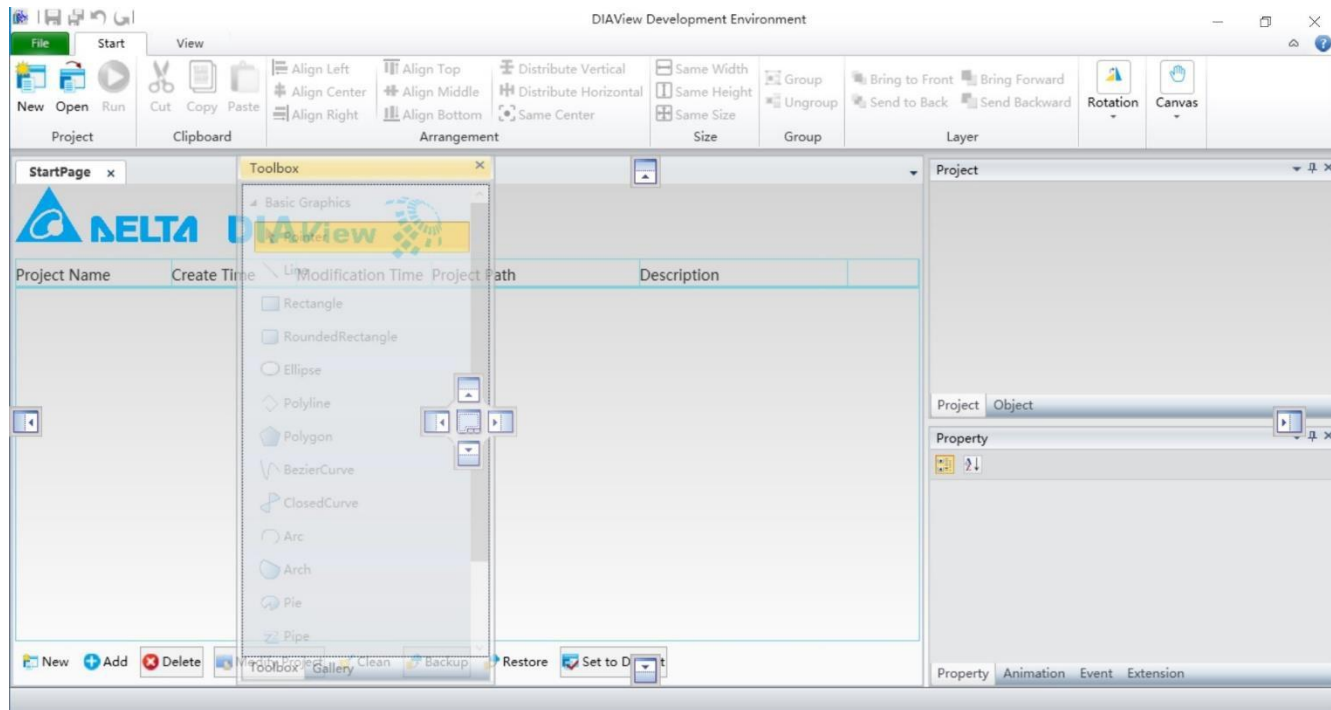
1. Рядок меню	Надає різні базові операційні функції для розробки проекту.
2. Кнопки інструментів	Кнопки швидкого доступу для робочих команд
3. Ящик для інструментів	Надає часто використовувані основні графічні елементи та блоки керування для малювання.
4. Галерея	Часто використовувані графічні елементи вбудовані в систему та контейнери для користувачів вільно розширювати персоналізовану графіку.
5. Область для блокнота	Область для управління проектами, малювання та редагування графічних зображень

Середовище розробки

6. Вікно проекту	Вікно відображення індексу дерева проекту; демонструє різні композиції проекту та надає різні портали для роботи і функції конфігурації.
7. Вікно браузерa об'єктів	Відображає всі члени об'єкта у вікні.
8. Вікно властивостей	Відображає властивості об'єкта; надає робочий портал для змінити властивості.
9. Вікно налаштування анімації	Портал для налаштування анімації для об'єктів.
10. Вікно налаштування події	Портал для налаштування подій для об'єктів.
11. Розширення	Інші функції

Середовище розробки

- Вікна середовища розробки програмного забезпечення DIAView також можна змінювати, перетягуючи їх за допомогою миші, як показано на малюнку нижче:

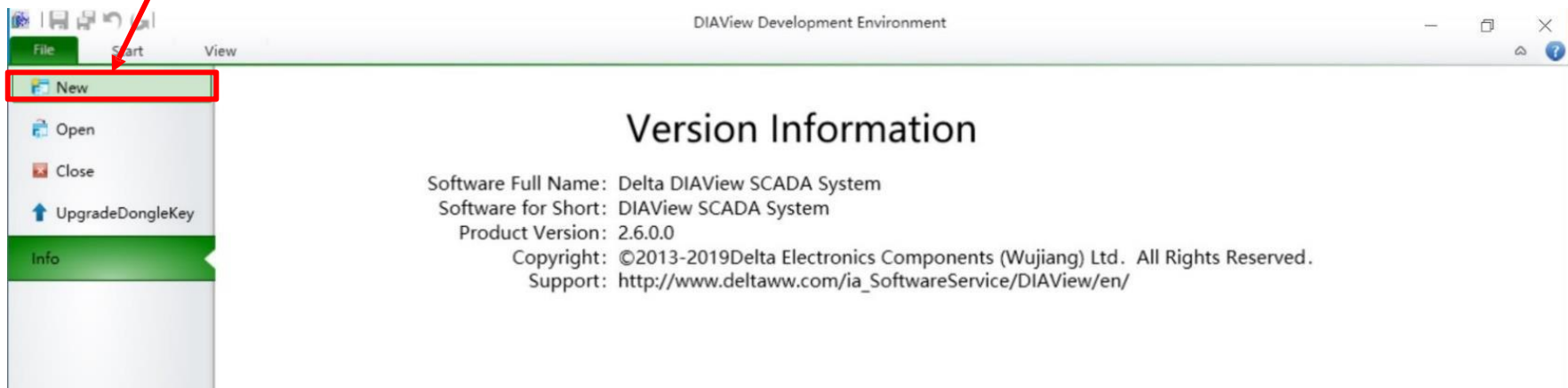


Середовище розробки

➤ Додавання проектів

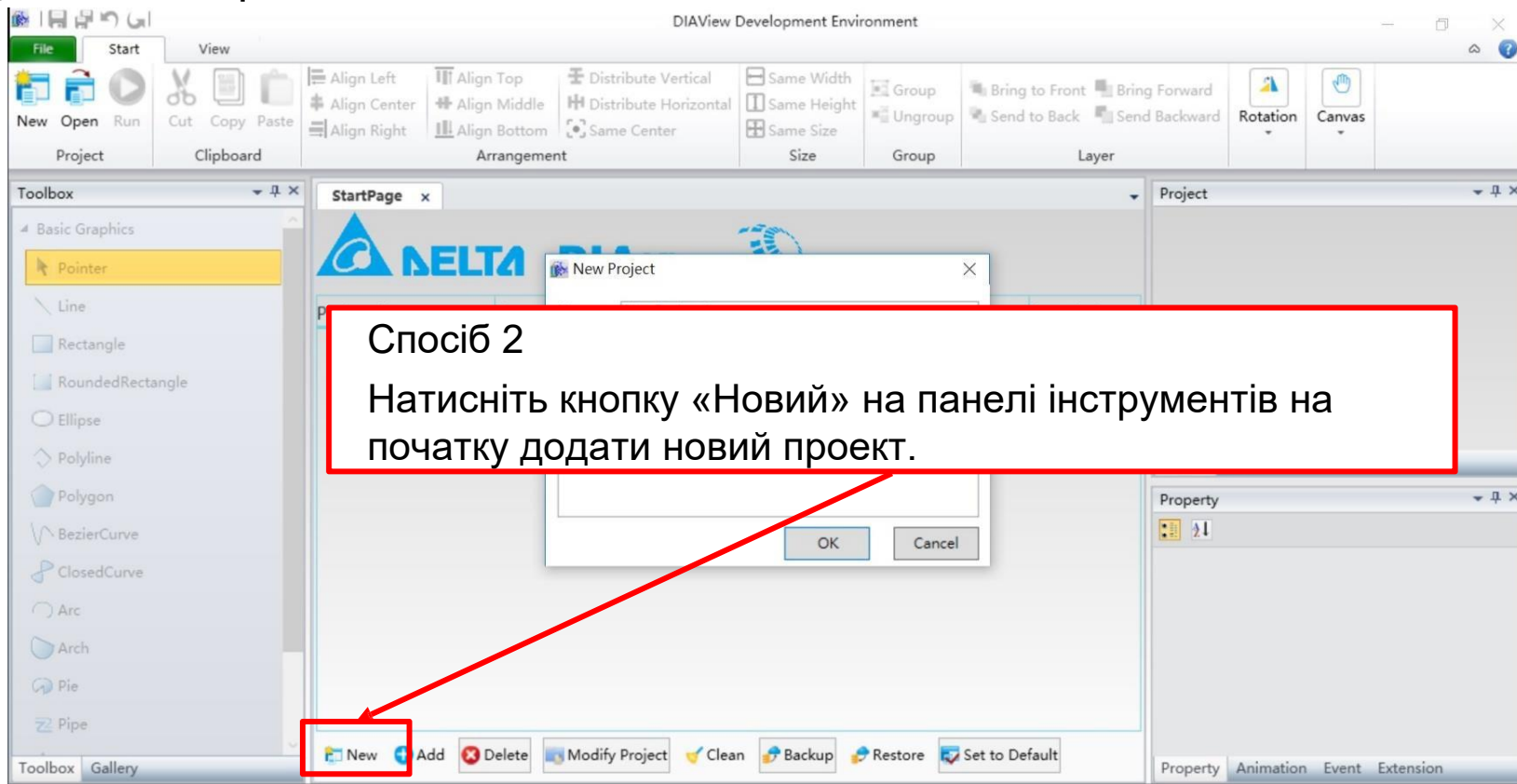
Спосіб 1

Виберіть меню «Файл» і натисніть «Новий».



Середовище розробки

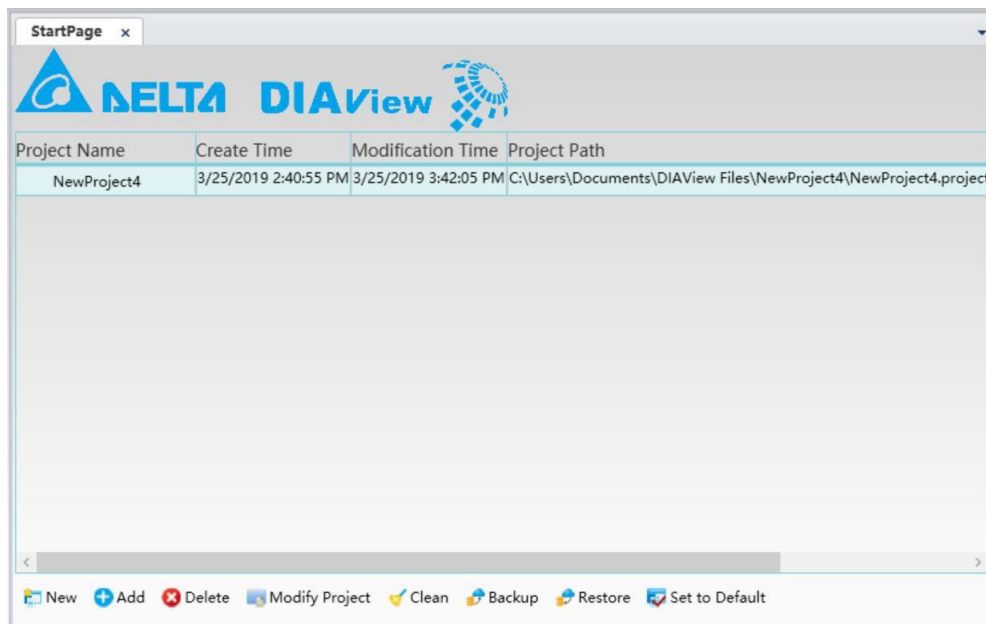
➤ Додавання проектів



Середовище розробки

➤ Управління проектами

1. Управління проектами означає виконання централізованого управління.
2. Відкрийте середовище розробки програмного забезпечення DIAView і запустіть
3. сторінка буде відкрита на головній інтерфейс за замовчуванням. Операції такі
4. як додати новий, додати, видалити, очистити, створити резервну копію, відновити та встановити проект як попередні налаштування
5. проекту тощо можна виконати на початкова сторінка.



Середовище розробки



➤ **НОВИЙ:**

- Додає новий проект; створений проект буде додано до списку проектів на початковій сторінці та відкрито у вікні проекту.

➤ **ДОДАТИ:**

- Додає проект, який уже існує за вказаним шляхом, до списку проектів початкової сторінки. Натисніть кнопку «Додати» або клацніть меню правою кнопкою миші, щоб відкрити вікно вибору папки.

➤ **ЧИСТИЙ:**

- Очистить інформацію про проект у списку початкових сторінок. Буде розчищено проект файл уже видалено, але все ще існує в списку проектів.

➤ **Резервне копіювання:**

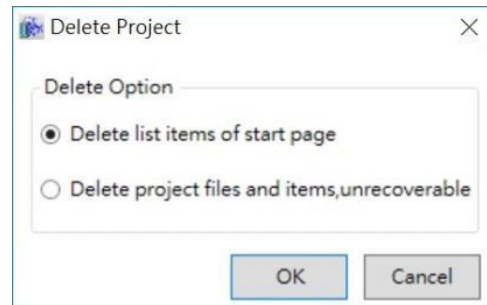
- Резервне копіювання вибраного проекту. Резервний файл проекту матиме розширення «.projectzip», як показано на малюнку нижче:

Середовище розробки



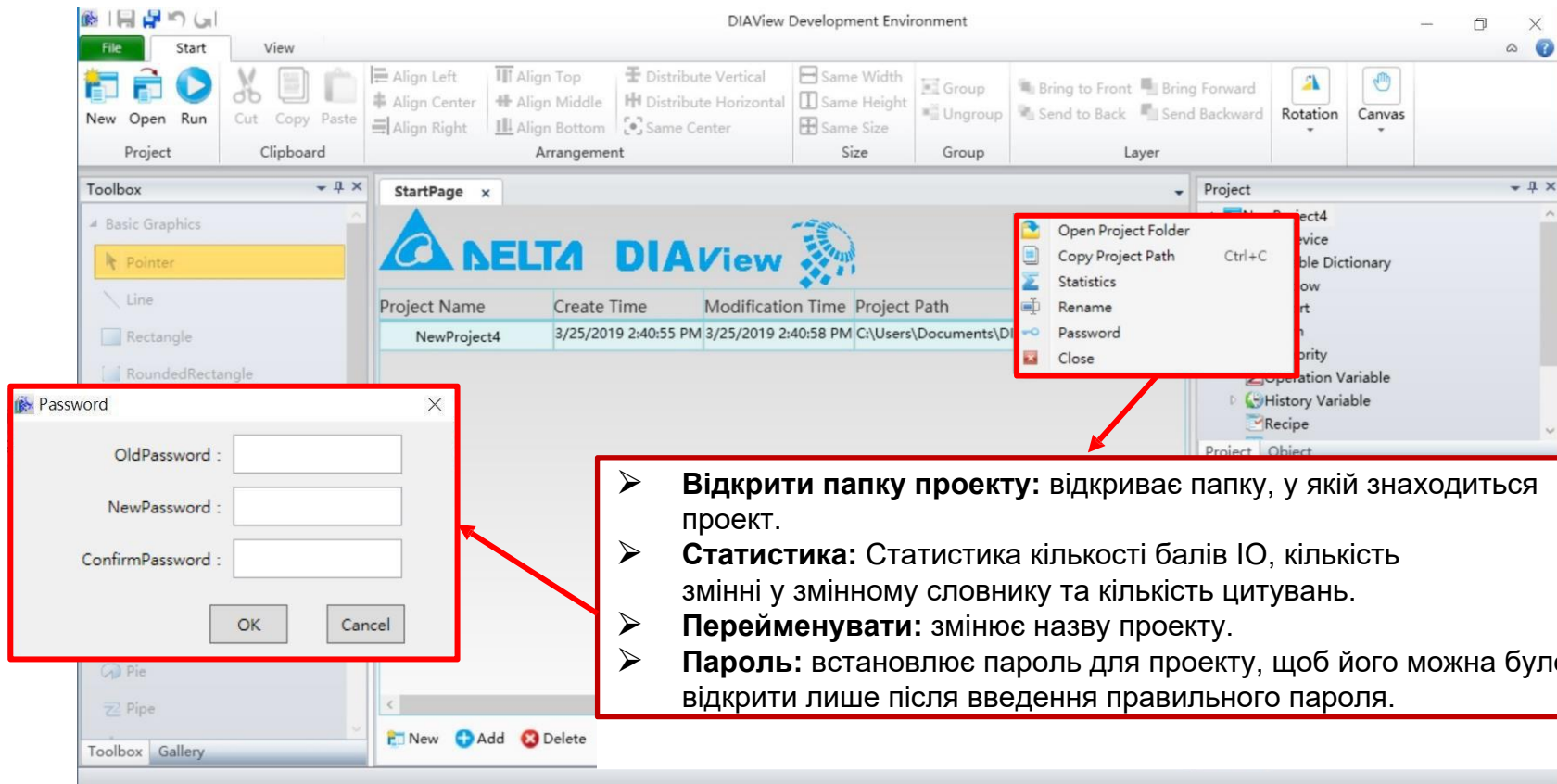
➤ Видалити:

- Видаляє вибраний проект зі списку проектів початкової сторінки. Натисніть кнопку «Видалити» або клацніть правою кнопкою миші меню, щоб відкрити поле вибору видалення проекту.
- Є два варіанти видалення:
 1. Видалити елементи списку початкової сторінки:
Видаляє інформацію про проект із самого початку сторінки, але файл проекту не видаляється.
 2. Видалити файл і елемент проекту, неможливо відновити: видаляє проект зі списку початкових сторінок, а також повністю видаляє файл проекту.



Середовище розробки

- Клацніть правою кнопкою миші назву проекту



The screenshot shows the DIAView Development Environment interface. The main window displays a project named 'NewProject4' in a table with columns: Project Name, Create Time, Modification Time, and Project Path. A context menu is open over the project name, listing actions: Open Project Folder, Copy Project Path (Ctrl+C), Statistics, Rename, Password, and Close. A red arrow points from the 'Password' option in the context menu to a 'Password' dialog box. The dialog box has three input fields: OldPassword, NewPassword, and ConfirmPassword, with OK and Cancel buttons.

➤ **Відкрити папку проекту:** відкриває папку, у якій знаходиться проект.

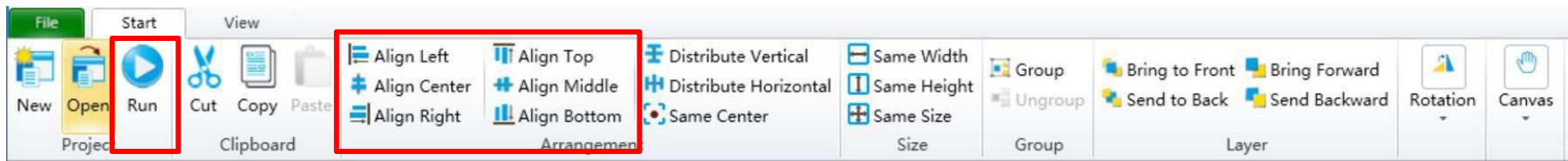
➤ **Статистика:** Статистика кількості балів ІО, кількість змінні у змінному словнику та кількість цитувань.

➤ **Перейменувати:** змінює назву проекту.

➤ **Пароль:** встановлює пароль для проекту, щоб його можна було відкрити лише після введення правильного пароля.

Середовище розробки с

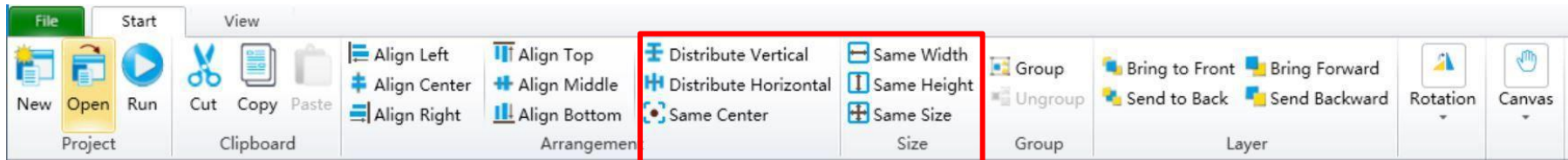
➤ Меню Пуск



	Execution	Executes the project currently opened in the development environment.
	Align left	Aligns the left side of two or more selected graphic objects in the sketchpad to the left side of the graphic object that was selected first.
	Вирівняти по центру	Aligns the geometric center of two or more selected graphic objects in the sketchpad to the same vertical line of the geometric center of the graphic object that was selected first.
	Align right	Aligns the right side of two or more selected graphic objects in the sketchpad to the right side of the graphic object that was selected first.
	Align top	Aligns the top side of two or more selected graphic objects in the sketchpad to the top side of the graphic object that was selected first.
	Вирівняти посередині	Aligns the geometric center of two or more selected graphic objects in the sketchpad to the same horizontal line of the geometric center of the graphic object that was selected first.
	Align bottom	Aligns the bottom side of two or more selected graphic objects in the sketchpad to the bottom side of the graphic object that was selected first.

Середовище розробки с

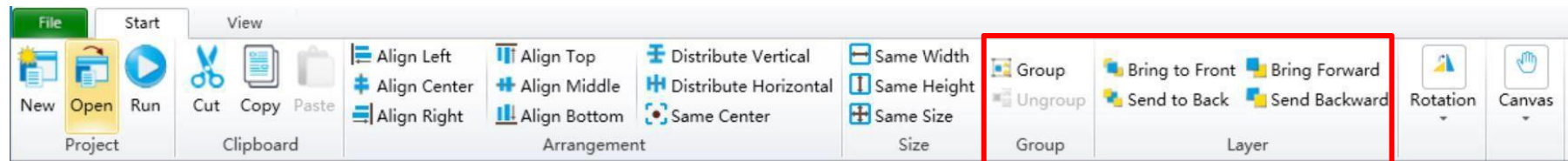
➤ Меню Пуск



	Vertical distribution	Distributes the geometric center of three or more selected graphic objects in the sketchpad evenly in the vertical direction.
	Horizontal distribution	Distributes the geometric center of three or more selected graphic objects in the sketchpad evenly in the horizontal direction.
	Same center	Aligns the center of two or more selected graphic objects in the sketchpad to the center of the graphic object that was selected first.
	Same width	Make two ore more selected graphic objects in the sketchpad have the same width.
	Same height	Make two or more selected graphic objects in the sketchpad have the same height.
	Same size	Make two or more selected graphic objects in the sketchpad have the same width and height.

Середовище розробки

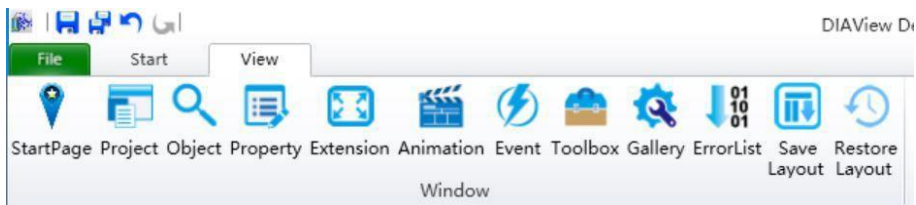
➤ Меню Пуск



	Надіслати назад	Places the selected graphic object at the bottom layer when there are multiple graphic objects stacked together in the sketchpad.
	На передній план	Places the selected graphic object at the top layer when there are multiple graphic objects stacked together in the sketchpad.
	Надіслати назад	Moves the selected graphic object down one layer when there are multiple graphic objects stacked together in the sketchpad.
	Висувати	Moves the selected graphic object up one layer when there are multiple graphic objects stacked together in the sketchpad.
	Група	Groups the selected multiple graphic objects into a new graphic when there are multiple graphic objects in the sketchpad.
	Розгрупувати	Splits the group graphic into constituent graphic objects.

Середовище розробки

➤ Меню Пуск



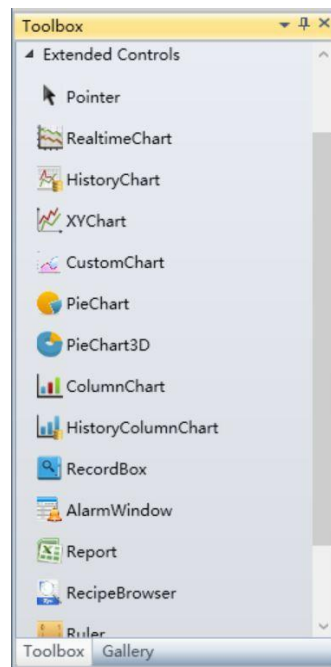
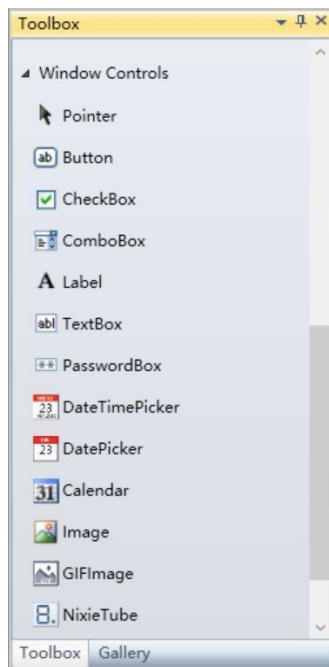
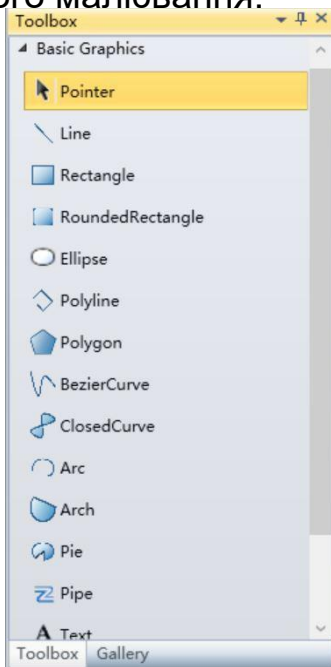
Команда	Опис функції
Стартова сторінка	Відкриває вікно початкової сторінки.
Демонструвати	Відкриває вікно папки проекту.
Об'єкт	Відкриває вікно браузера об'єктів.
Власність	Відкриває вікно властивостей.
Розширення	Відкриває вікно розширення.
Анімація	Відкриває вікно анімації.
Подія	Відкриває вікно події.
Ящик інструментів	Відкриває панель інструментів.
Галерея	Відкриває графічну бібліотеку.
Список помилок	Відкриває список помилок
Зберегти макет	Збережить макет, який ви налаштували раніше
Відновити макет	Відновити вихідний макет

- Терміни та поняття DIAView
- Середовище розробки
- Основні функції та кнопки
- Як почати проект

Основні функції та кнопки

➤ Знайомство з вікном інструментів

Вікно інструментів програмного забезпечення DIAView відноситься до панелі інструментів у вікні розробки графіки. Вони забезпечують основні графічні елементи, графічні блоки керування та часто використововувані графічні компоненти та моделі, необхідні для графічного малювання.



Основні функції та кнопки

➤ Знайомство з галереєю

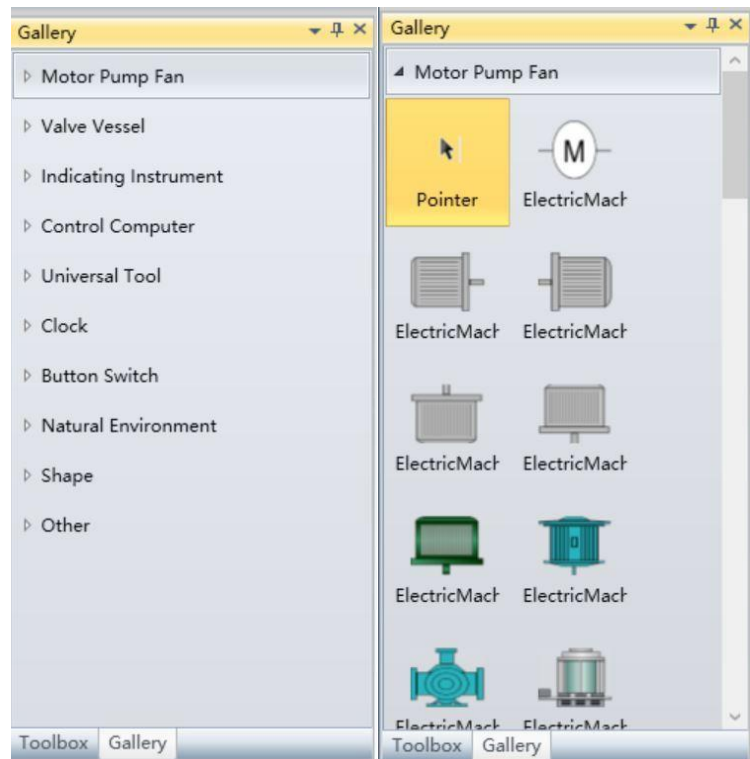
Креслення також є основою для розробки проекту SCADA.

Ми пропонуємо багато одиниць, включаючи вентилятори, посудини, інструменти, годинники тощо.

Більшість одиниць мають анімацію, пов'язану зі змінними.

Використовуючи наш інструментарій галереї, ви можете створити проект набагато швидше та зручніше. Моделі вентиляторів мотопомпи в «Галереї» представлені зліва.

Просто натисніть галерею ліворуч, а потім натисніть позицію в області розробки, щоб додати одиницю.

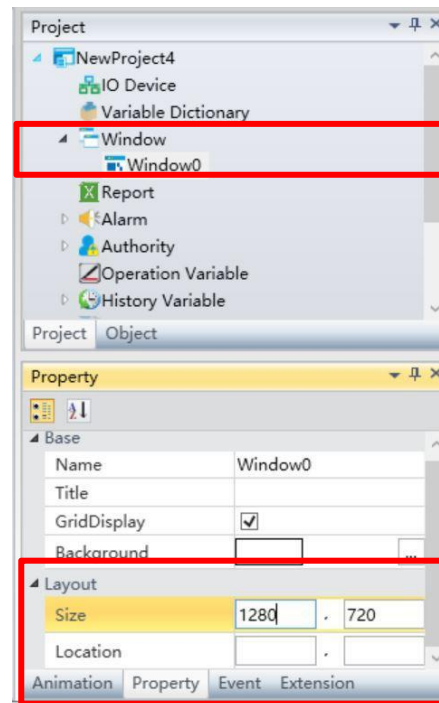


Основні функції та кнопки

➤ Вікно проекту

Вікно проекту – це вікно відображення індексу дерева проекту.

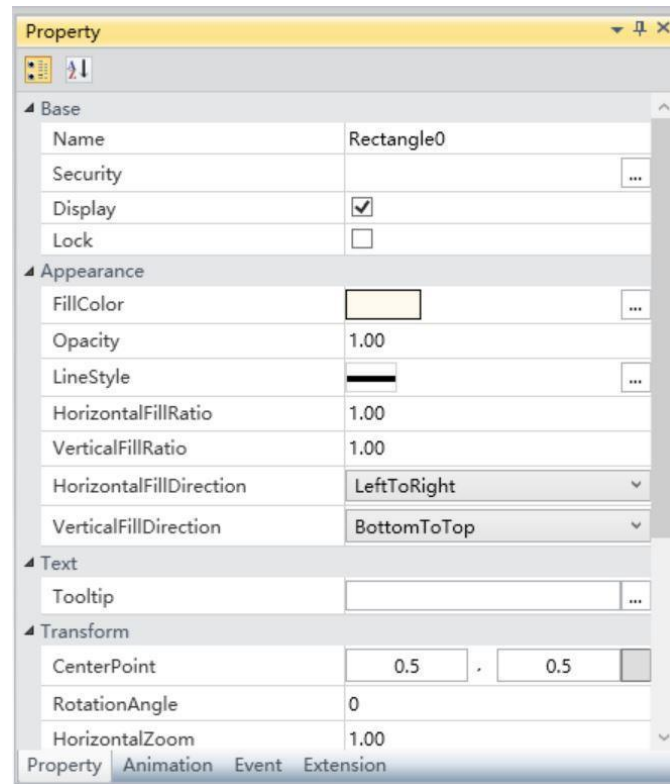
Він має різні функціональні вузли. Наприклад, ви можете створити своє вікно у вузлі «Вікно», а потім встановити властивість (розмір, розташування) у вікні властивостей нижче.



Основні функції та кнопки

➤ Власність

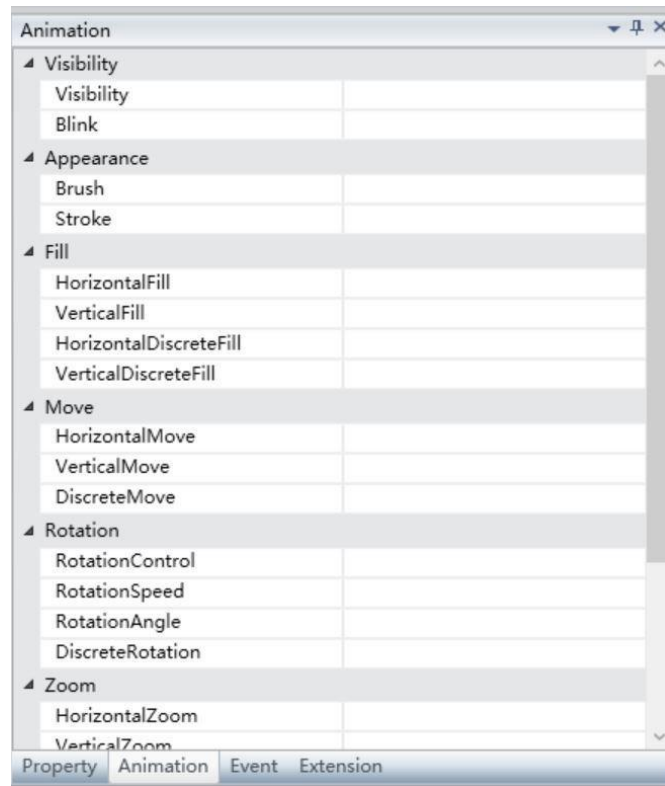
Вікно властивостей керує пов'язаними властивостями графічних об'єктів. Коли на скетчборді вибрано будь-яку графіку, це вікно відображатиме її відповідні властивості в реальному часі. Властивості включають назву, положення, розмір і колір тощо графічного об'єкта. Користувач може змінювати та налаштовувати значення властивостей у діалоговому вікні властивостей, як показано на малюнку



Основні функції та кнопки

➤ Анімація

Вікно анімації використовується для керування інформацією про конфігурацію анімації графічних об'єктів. Користувачі можуть додавати або видаляти анімацію для графіки. Конфігурація анімації вимагає підключення до пов'язаних властивостей і змінних даних, як показано на малюнку.



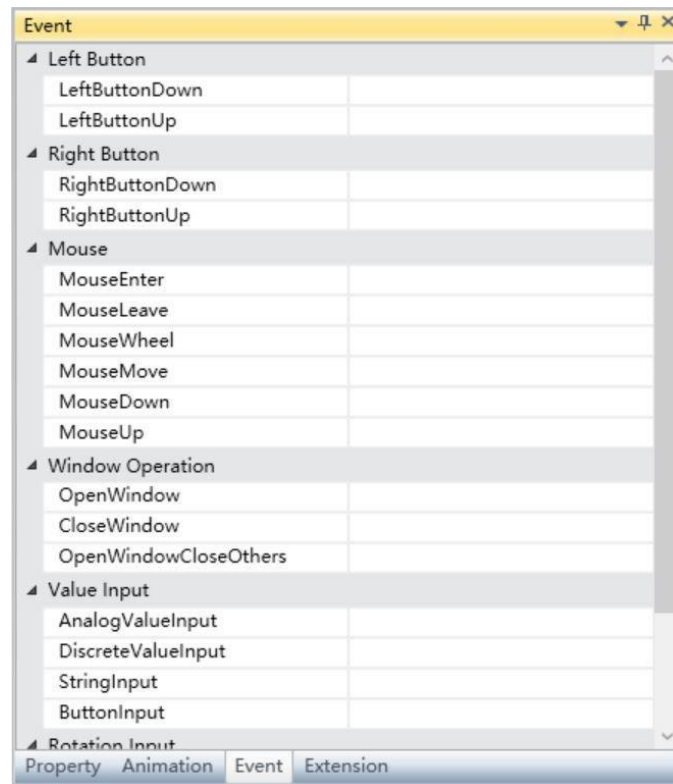
Основні функції та кнопки

➤ Подія

Вікно події використовується для керування інформацією про конфігурацію події графічних об'єктів.

Користувачі можуть додавати або видаляти графічні події. Подія

Конфігурація вимагає підключення до пов'язаних подій миші та клавіатури та змінних даних, як показано на малюнку.



- Терміни та поняття DIAView
- Середовище розробки
- Основні функції та кнопки
- Як почати проект

Як почати проект

➤ Основні етапи розробки проекту

- **Крок 1:** запустіть середовище розробки програмного забезпечення DIAView.
- **Крок 2:** Додайте проект: натисніть «Додати елемент проекту з меню «Файл» або «Пуск» у головному інтерфейсі середовища розробки та налаштуйте інформацію, включаючи назву проекту та шлях до зберігання проекту тощо, після чого система згенерує та відобразити початковий індекс дерева у вікні «Проект».
- **Крок 3:** Визначте змінні: Змінні – це носії інформації, які використовуються для запису змін певних даних програмного забезпечення DIAView в реальному часі. Він динамічно відображає джерело даних і форму відображення контрольованих даних на місці. Тому всі змінні, що беруть участь в інформаційному рукостисканні проекту, повинні бути визначені.

Як почати проект

➤ Основні етапи розробки проекту

- **Крок 4. Створіть IO Communication:**

Створіть фізичний канал даних між верхньою та нижньою машинами програмного забезпечення DIAView та обладнанням моніторингу та керування на місці та встановіть інформацію, включаючи метод підключення, параметри зв'язку та змінні зв'язку тощо.

- **Крок 5 Створіть вікно:**

Це означає створення графічного інтерфейсу для системи. Намалюйте графіку для аналогічних зображень на місці та налаштуйте відповідну інформацію, включаючи властивості та діаграми тощо.

Як почати проект

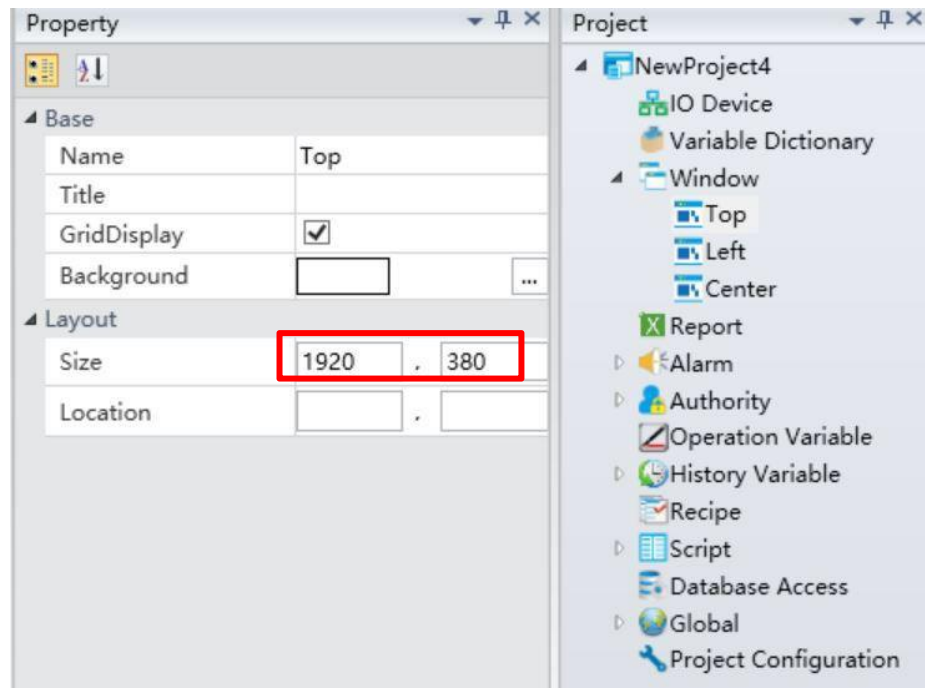
➤ Основні етапи розробки проекту

- Крок 5 Створіть вікно:

Наприклад, ми хочемо створити проект, у якому велике вікно розділено на 3 маленькі вікна (верхнє, ліворуч, центр).

Розмір великого вікна становить **1920*1080.**

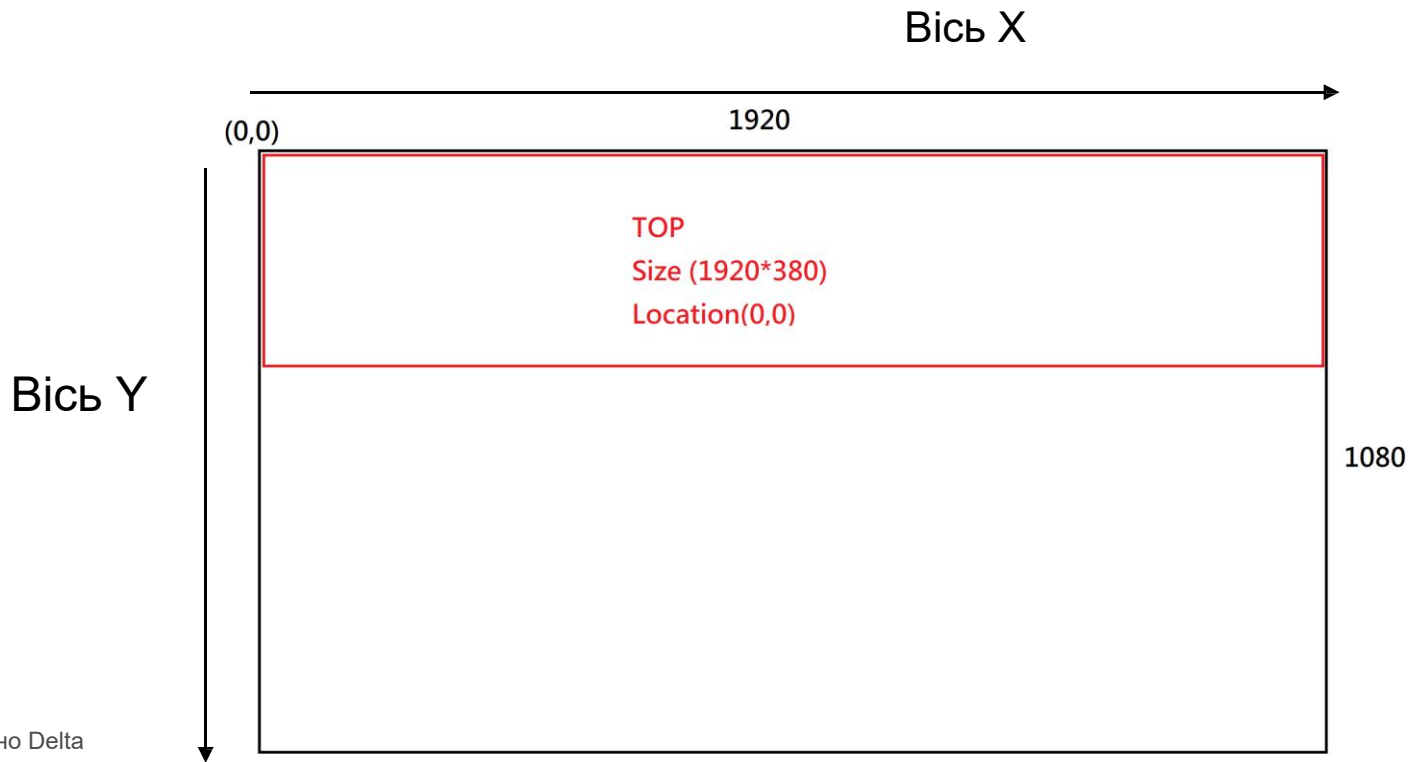
Спочатку ми повинні зобразити структуру.



Як почати проект

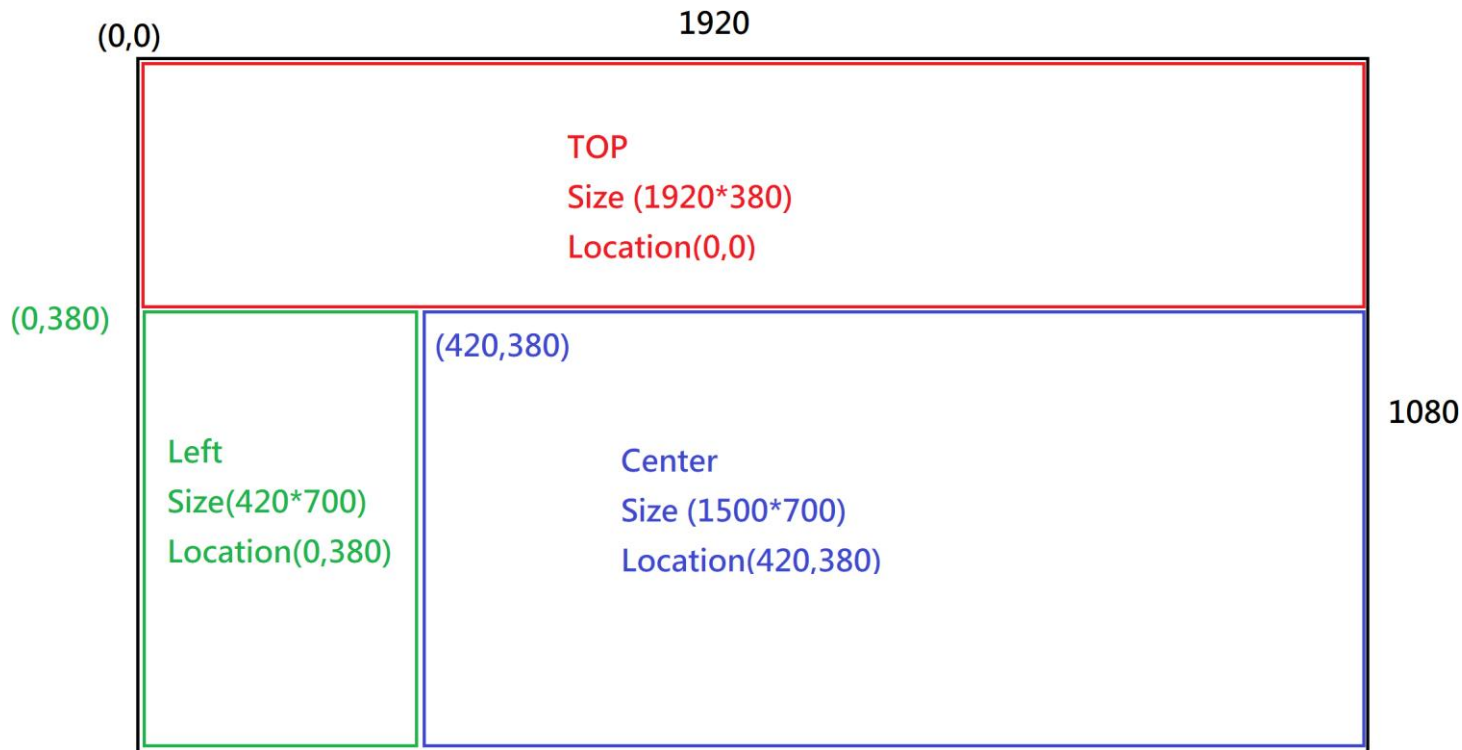
- Крок 5 Створіть вікно:

Координати обчислюються від верхнього лівого кута (0,0)



Як почати проект

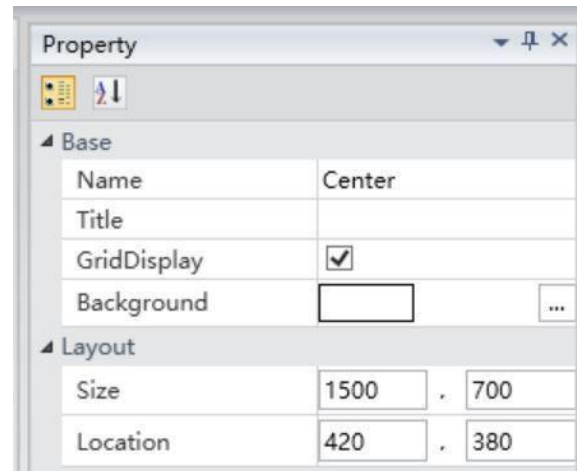
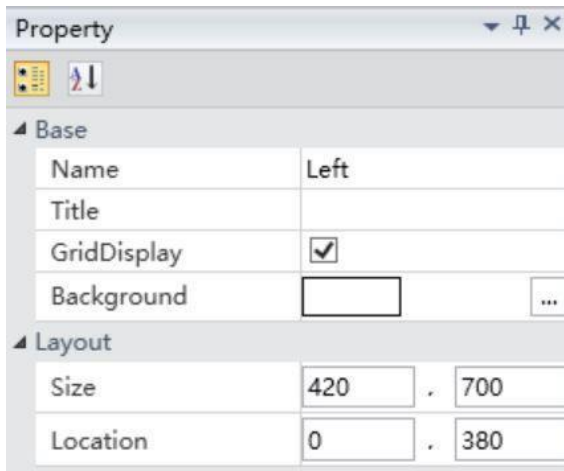
- Крок 5 Створіть вікно:



Як почати проект

- **Крок 5 Створіть вікно:**

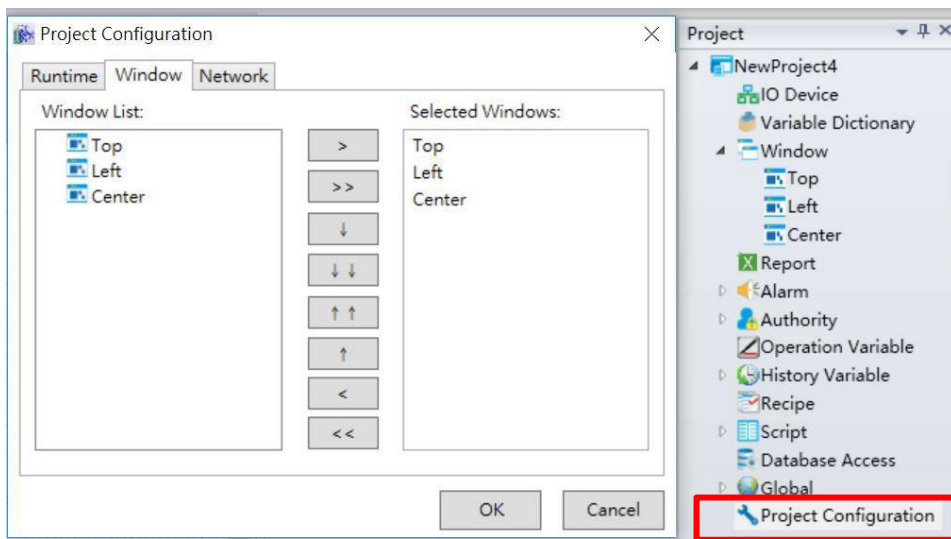
Слідуючи макету, ми можемо створити два ліві вікна. Майте на увазі, що розташування означає верхній лівий кут вікон.



Як почати проект

- Крок 6 Конфігурація проекту:

Встановлює відповідні параметри для з'єднання між програмним забезпеченням DIAView і сторонніми базами даних для досягнення зв'язку між даними проекту в реальному часі та іншими комерційними базами даних, а також установлює параметри виконання проекту та початковий екран.



Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Встановіть посилання на
зовнішні пристрої

Веббер
2019/08/26



Контур

- Канал і пристрої
- Водій
- Статус зв'язку
- Симулятор

призначення

У цьому розділі ви дізнаєтеся....

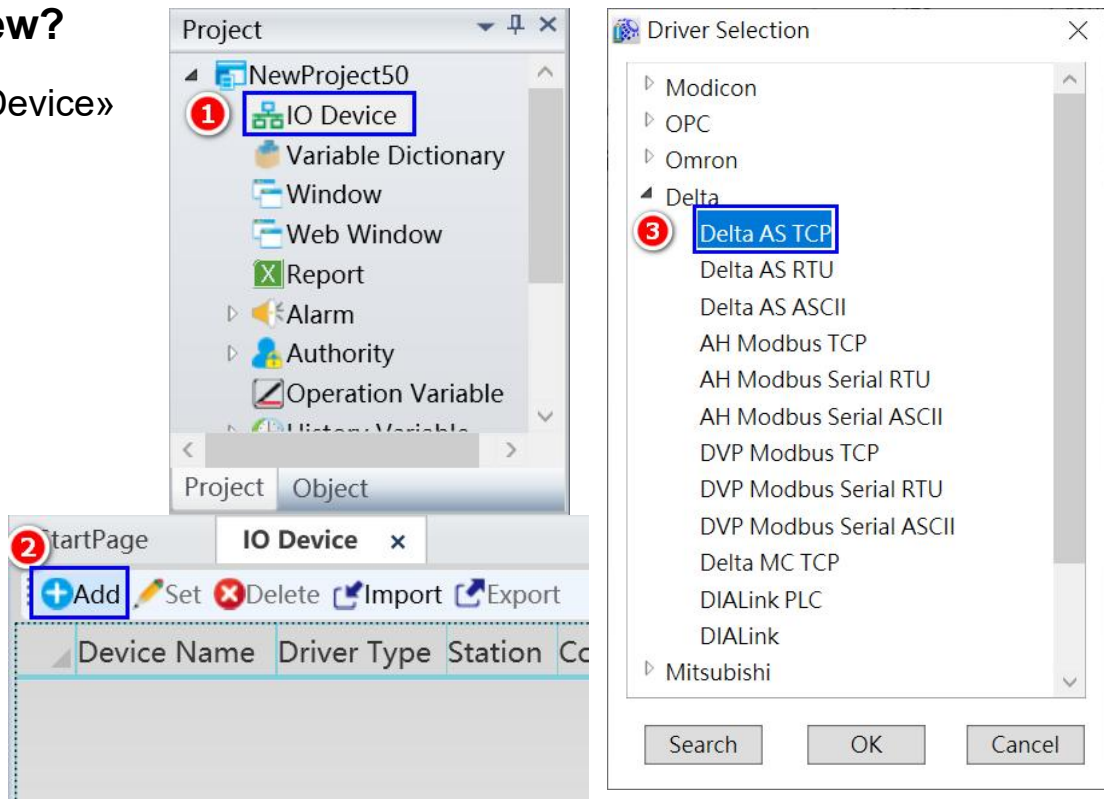
- ... докладніше про канал і пристрої DIAView
- ... докладніше про драйвер, який підтримує DIAView
- ... докладніше про те, як налаштувати з'єднання з пристроєм

- Канал і пристрої
- Водій
- Статус зв'язку
- Симулятор

Канал і пристрої

➤ Як підключити пристрій в DIAView?

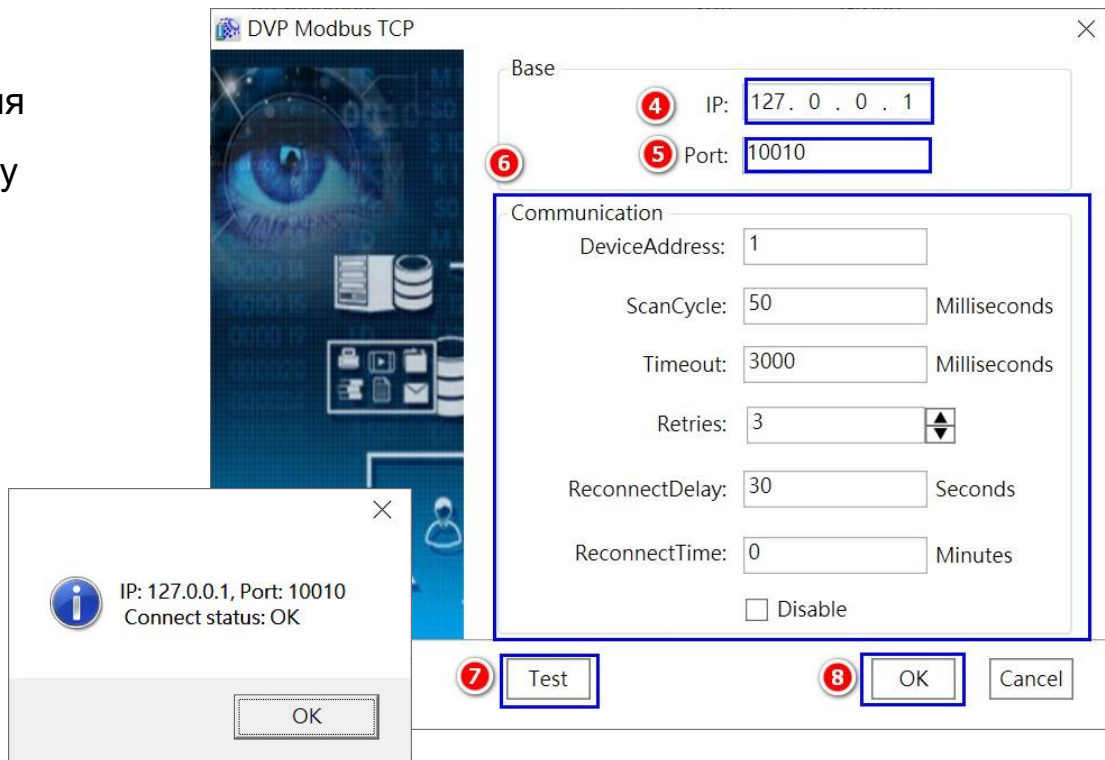
- ① Клацніть лівою кнопкою миші «IO Device»
- ② Натисніть «Додати»
- ③ Виберіть драйвер, який потрібно підключити.



Канал і пристрої

➤ Як підключити пристрій в DIAView?

- ④ Встановити IP-адресу
- ⑤ Встановити порт підключення
- ⑥ Встановити параметри зв'язку
- ⑦ Перевірте пінг на пристрій
- ⑧ Завершити налаштування



Канал і пристрої

➤ Параметри зв'язку:

Власність	опис
DeviceAddress	Номер адреси підключеного пристрою
ScanCycle	Цикл сканування даних зв'язку IO попередньо встановлено на 100 мілісекунд.
Час вийшов	Якщо пристрій не відповідає протягом тривалого періоду часу, буде визначено час очікування зв'язку відповідно до встановленої тривалості часу. Тривалість тайм-ауту попередньо встановлена на 1000 мілісекунд
Повторні спроби	Це кількість разів, коли система намагатиметься повторно підключитися до пристрою, якщо це буде тайм-аут зв'язку. Стандартна кількість повторень становить 3 рази.
ReconnectDelay	Цей параметр використовується для встановлення періоду затримки перед спробою повторного підключення, коли кількість спроб досягла ліміту та повторне підключення було невдалим. За замовчуванням час затримки повторного підключення становить 30 секунд.
ReconnectTime	За замовчуванням 0 хвилин: немає обмежень за часом, що означає, що він постійно намагатиметься повторно підключитися до пристрій; Інше значення: встановлює тривалість часу повторного підключення. Якщо система не може з'єднатися й отримати відповіді від пристрою протягом ліміту часу повторного підключення, вона скасує підключення до цього пристрою.
Вимкнути	Обладнання поточної конфігурації буде вимкнено для зв'язку з DIAView програмне забезпечення.

- Канал і пристрої
- Водій
- Статус зв'язку
- Симулятор

➤ Список драйверів:

Тип драйвера	приклад
Модікон	Modbus TCP, Modbus Serial RTU, Modbus Serial ASCII
OPC	OPC, OPCUA
Omron	FINS TCP, FINS ASCII, HostLink ASCII
Дельта	Серія Delta AS: TCP, RTU, ASCII Серія Delta AH, TCP 、 RTU 、 ASCII Серія Delta DVP: TCP RTU 、 ASCII, Delta MC TCP DIALink
Mitsubishi	Mitsubishi FX Serial, Mitsubishi ProFX Serial Mitsubishi Q Serial, Mitsubishi Q EtherNet Mitsubishi FX EtherNet
Siemens	S7300 TCP, S71200 TCP, S7300 MPI S7200 TCP, S7200 PPI, S7200 Smart TCP
Симулятор	Симулятор
Дельта вимірювач потужності	DPM-C530 Ethernet, DPM-C530 Serial
Роквелл	Rockwell Controllogix Ethernet

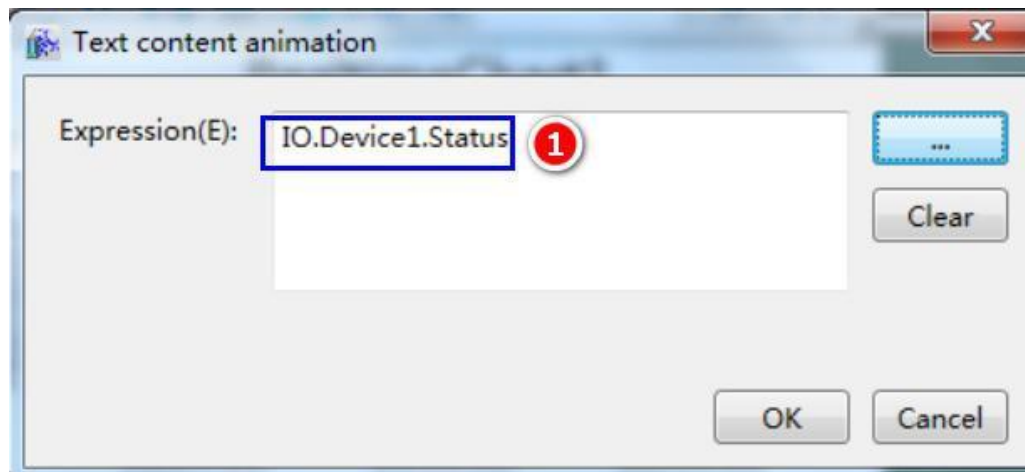
✗ Для налаштувань драйвера зверніться до посібника користувача CH5.8, CH5.9

- Канал і пристрої
- Водій
- Статус зв'язку
- Симулятор

Статус зв'язку

➤ Як перевірити стан зв'язку пристрою в проекті DIAView?

- ① Намалюйте базову «текстову» графіку в будь-якому вікні існуючого проекту та налаштуйте «Вираз (E)» у вікні анімації тексту, як показано на малюнку нижче:



Статус зв'язку

➤ Як перевірити стан зв'язку пристрою в проєкті DIAView?

- ② Статус зв'язку [пристрою] можна перевірити після виконання; будь ласка, зверніться до таблиці удару:

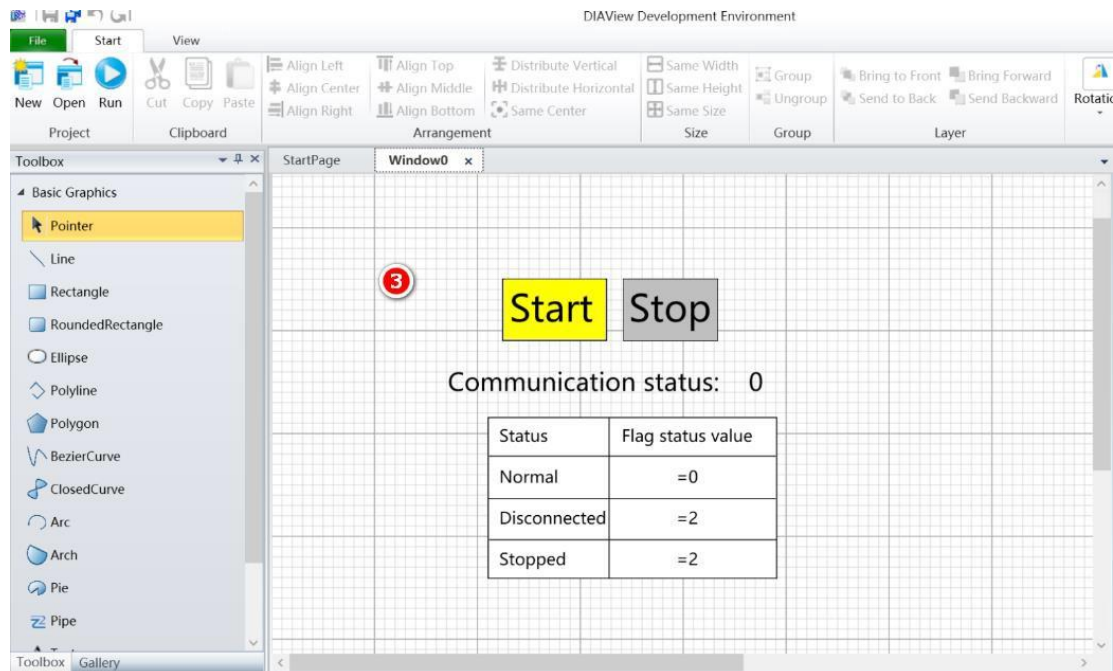
Статус	Значення статусу біта прапора (0 означає нормальний стан зв'язку, інші значення є ненормальними)
Спілкування нормальне	ІО.[пристрій].Статус = 0
Зв'язок відключено	ІО.[пристрій].Статус = 2
Зв'язок припинився	ІО.[пристрій].Статус = 2

Статус зв'язку

➤ Як перевірити стан зв'язку пристрою в проекті DIAView?

③ Намалюйте дві кнопки та два тексти у вікні існуючого проекту, як показано на малюнку

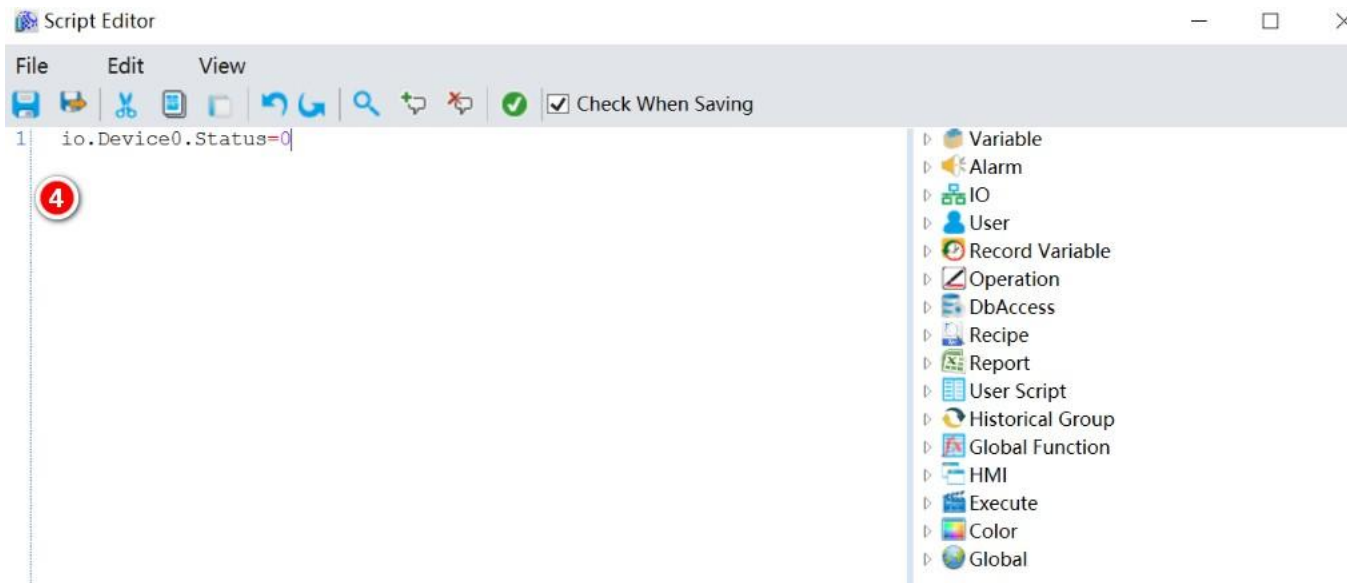
НИЖЧЕ:



Статус зв'язку

➤ Як перевірити стан зв'язку пристрою в проекті DIAView?

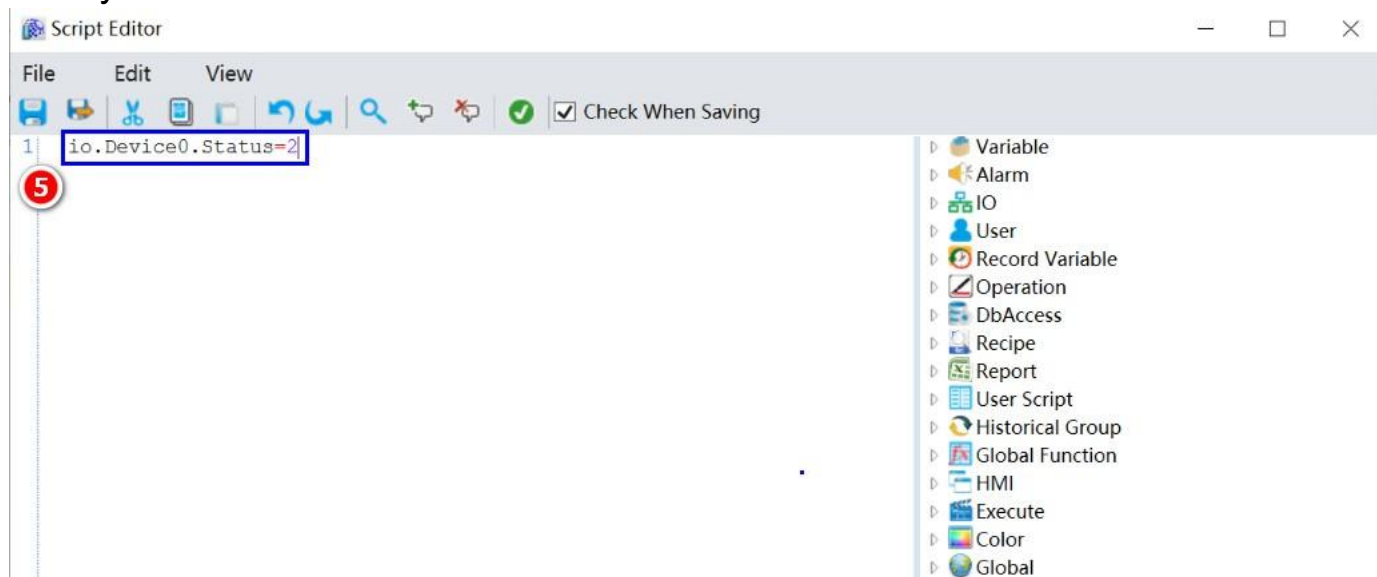
④ Налаштуйте подію «Клацання лівою кнопкою миші» для «кнопки «Пуск», як показано на малюнку нижче:



Статус зв'язку

➤ Як перевірити стан зв'язку пристрою в проекті DIAView?

⑤ Налаштуйте подію «Клацання лівою кнопкою миші» для «Кнопки зупинити», як показано на малюнку нижче :



Статус зв'язку

➤ Як перевірити стан зв'язку пристрою в проєкті DIAView?

⑥ Після виконання початковий стан зв'язку "Device0" є таким:



Communication status: 2

Status	Flag status value
Normal	=0
Disconnected	=2
Stopped	=2

Статус зв'язку

➤ Як перевірити стан зв'язку пристрою в проєкті DIAView?

⑦ Після натискання кнопки «Пуск» стан зв'язку «Device0» виглядає наступним чином:



Communication status: 0

Status	Flag status value
Normal	=0
Disconnected	=2
Stopped	=2

- Канал і пристрої
- Водій
- Статус зв'язку
- Симулятор

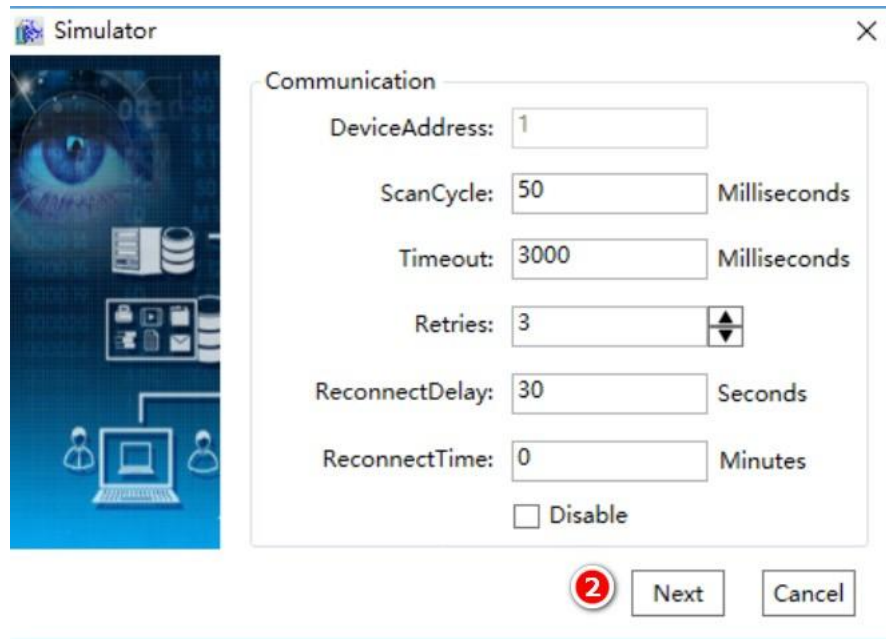
➤ Як створити зв'язок між DView та Simulator?

- ① Клацніть правою кнопкою миші вузол «IODevice» і виберіть «Новий пристрій», виберіть «симулятор» у вікні вибору драйвера



➤ Як створити зв'язок між DView та Simulator?

- ② Налаштуйте параметри зв'язку симулятора системи, можна зберегти значення за замовчуванням.



The screenshot shows the 'Simulator' window with a 'Communication' tab selected. The window contains several input fields for configuring communication parameters. A red circle with the number '2' is placed over the 'Next' button at the bottom right of the window.

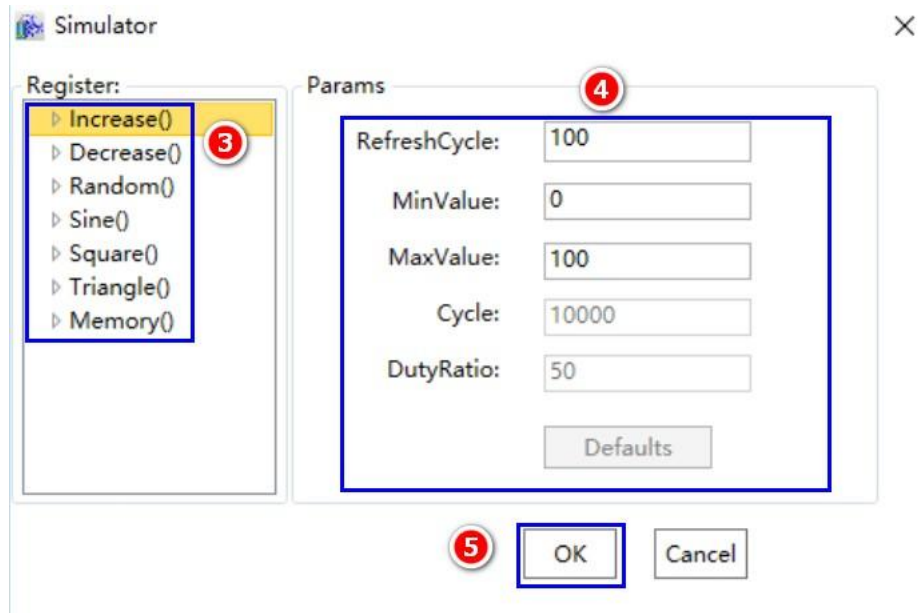
Parameter	Value	Unit
DeviceAddress	1	
ScanCycle	50	Milliseconds
Timeout	3000	Milliseconds
Retries	3	
ReconnectDelay	30	Seconds
ReconnectTime	0	Minutes

☐ Disable

2 Next Cancel

➤ Як створити зв'язок між DView та Simulator?

- ③ Виберіть тип реєстру симулятора
- ④ Встановити параметри реєстру
- ⑤ Натисніть «ОК»



➤ Параметри моделювання:

зареєструватися	Діапазон	опис
Збільшити	0~63	Збільшити
Зниження	0~63	Зниження
Випадковий	0~63	Випадковий
синус	0~63	Синусоїда
Майдан	0~63	Квадратна хвиля
Трикутник	0~63	Трикутна хвиля
Пам'ять	0~63	Пам'ять, читання-запис

- **Цикл оновлення:** цикл зміни даних
- **Мінімальне значення:** (значення типу Int32, мінімальне значення), -2 147 483 648
- **Максимальне значення:** (значення типу Int32, максимальне значення), 2 147 483 647
- **Цикл:** Частково використаний; цикл даних
- **Коефіцієнт заповнення:** відноситься до позиції позитивного імпульсу в періоді імпульсу (можна встановити лише для трикутника та квадрата)

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Середовище розробки
програмного забезпечення

Венді
2020/02/11



Контур

- Огляд змінного словника
- Інструмент керування словником змінних
- Функція правої кнопки миші групи змінних
- Сценарій змінного словника

призначення

Після цього розділу ви дізнаєтесь...

- ... Основні операції та керування змінними

- ... Функція групи змінних

- ... Скрипти змінного словника, які часто використовуються

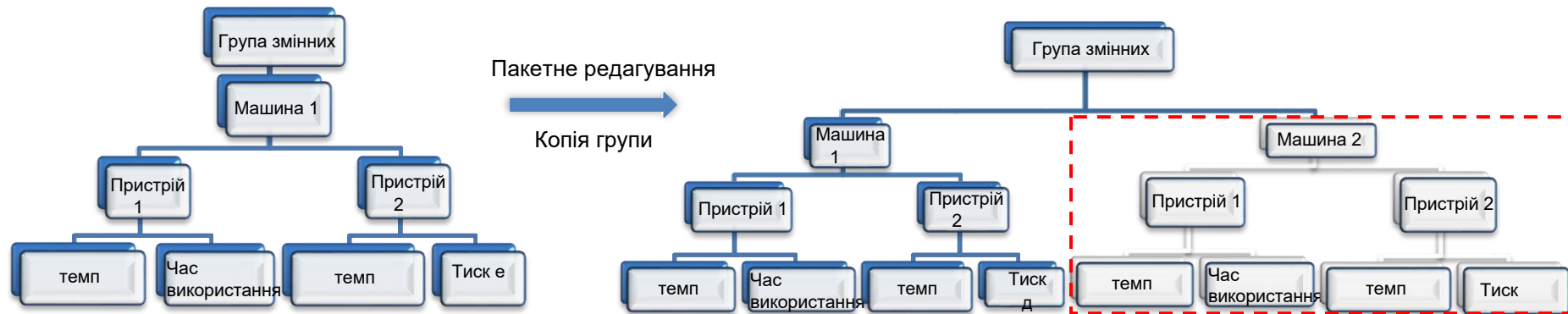
Контур

- Огляд змінного словника
- Інструмент керування словником змінних
- Функція правої кнопки миші групи змінних
- Сценарій змінного словника

Огляд змінного словника

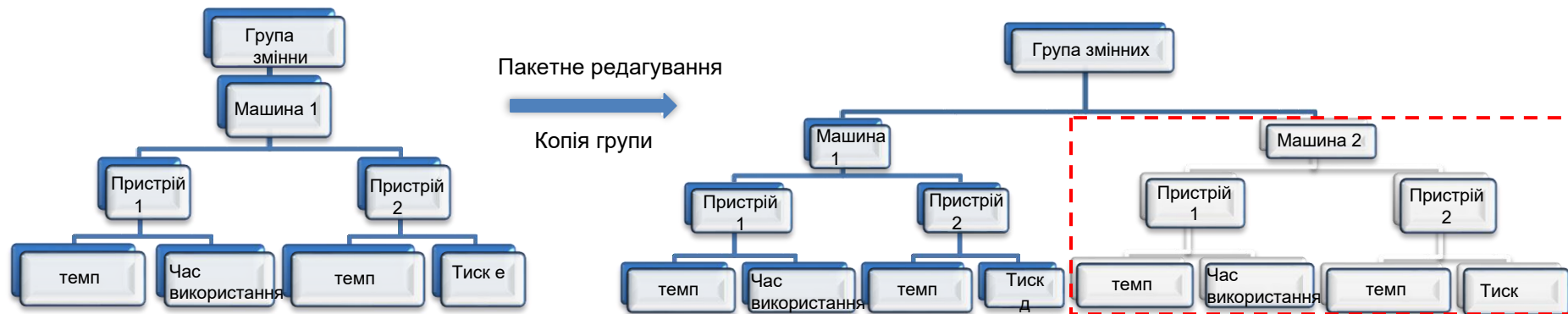
➤ Інженерні проекти вимагають різних типів змінних для участі у взаємодії з даними, щоб представляти та контролювати умови виробництва в реальному часі. Словник змінних — це набір змінних у проекті, а група змінних — це одиниця для групування та керування змінними.

- Швидко додавайте, видаляйте, вставляйте, пакетно додавайте, імпортуйте та експортуйте, змінюйте змінні
- Швидко додавайте, видаляйте, копіюйте, вставляйте, перейменовуйте, імпортуйте та експортуйте групи змінних для підвищення ефективності розробки



Огляд змінного словника

- Підтримка інтелектуальної заміни, перегляду довідкових матеріалів, статистики використання, перевірки помилок та інших функцій
- Операції групування та сортування впорядкованих змінних
- Підтримка керування змінними деревовидної структури та групування шпилів
- Додайте функцію домену розширення змінних, щоб реалізувати групування змінних



- Огляд змінного словника
- Інструмент керування словником змінних
- Функція правої кнопки миші групи змінних
- Сценарій змінного словника

➤ Робота інструменту керування словником змінних:

1. Вікно редагування змінного словника

- Редагувати такі дії, як додавання змінних, вставка, видалення, пакетне додавання, імпорт і експорт

2. Варіативний браузер

- Щоб відфільтрувати змінні за групою змінних, типом змінної, назвою змінної

3. Вікно виразу

- Встановлення умовних виразів в умовних програмах

➤ Основні операції зі змінними

додати	Додайте змінну, щойно додана змінна відображається в списку змінних в області редагування змінних
Вставка	Вставте змінну, щойно вставлена змінна відображається над вставленою змінною
Видалити	Видалити змінну
Пакетне додавання	Функція пакетного додавання може додавати кілька змінних одночасно
Імпорт	Імпортуйте файли конфігурації зовнішніх змінних, щоб підвищити ефективність конфігурації змінних
Експорт	Інформацію про змінну конфігурацію можна експортувати як файл Excel, який підтримує формати xls іxlsx
Копія	Копіювати змінні
Вирізати	Змінна зсуву
Вставити	Вставити змінні

StartPage Variable Dictionary x

+ Add
 + Insert
 ✖ Delete
 + Batch Add
 📄 Import
 📄 Export

	Name	Variable Type	Initial Value	Retentive Value	Minimum	Maximum	Deadband	Decimal Digit	Eng Units	Extended Domain	Description
1	Variable	Analog	0	☐	0	10000	0	0			
2	Variable1	Digital	False	☐	N/A	N/A	N/A	N/A	N/A		
3	Variable2	String		☐	N/A	N/A	N/A	N/A	N/A		

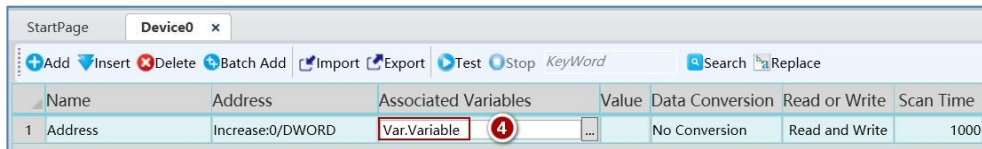
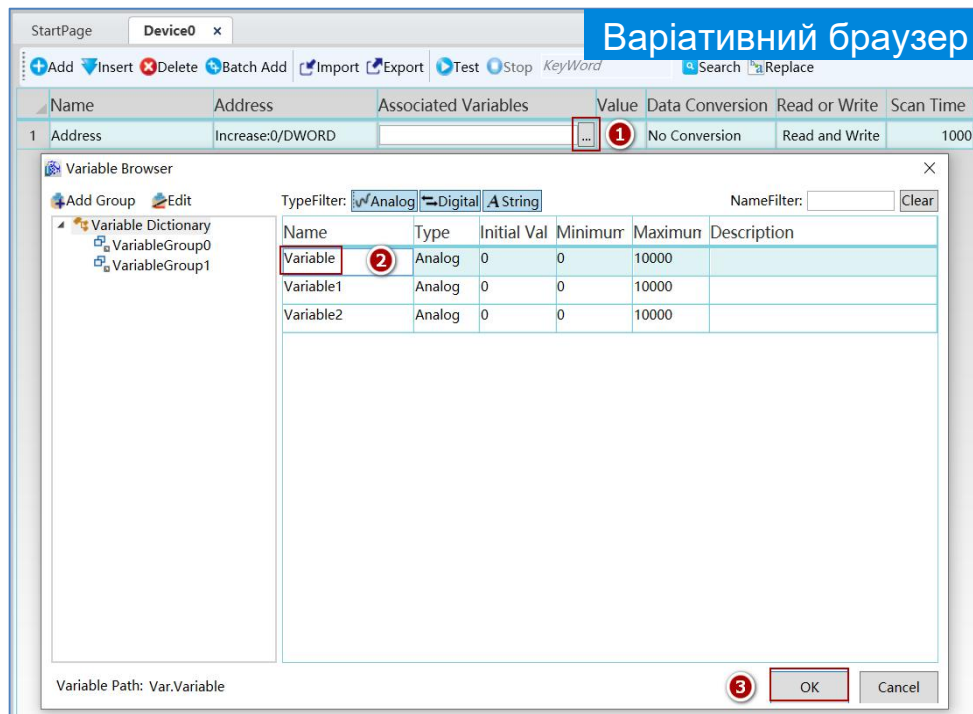
+	Add	Alt+A
+	Insert	Alt+I
✖	Delete	Del
+	Batch Add	Alt+B
💾	Save	Ctrl+S
📄	Copy	Ctrl+C
✂	Cut	Ctrl+X
📄	Paste	Ctrl+V
📄	Import	Ctrl+I
📄	Export	Ctrl+E

Вікно редагування змінного словника

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

➤ Приклад роботи браузера зі змінними:

- ① Натисніть кнопку Вибрати змінну, щоб відкрити браузер змінних
- ② Виберіть змінну, яку потрібно зв'язати
- ③ Натисніть "ОК"
- ④ Змінні в словнику змінних асоціації змінних ІО, асоціація успішна



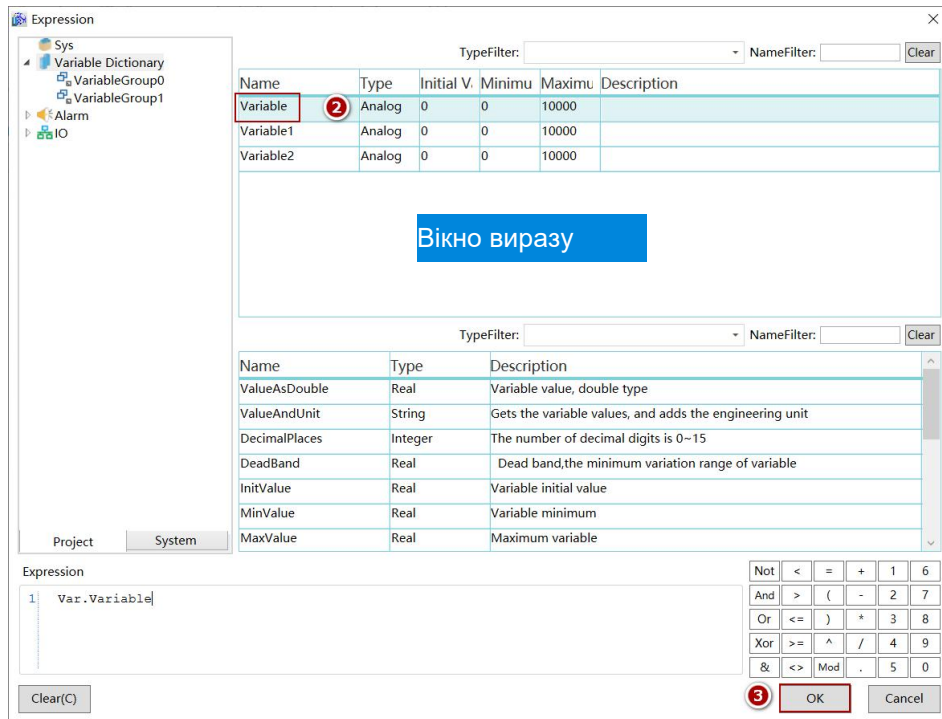
➤ **Варіативний браузер**

- Під час налаштування змінних у пов'язаному словнику змінних для змінних вводу/виводу, змінних сигналів, робочих змінних і історичних змінних використовується браузер змінних;
- Ось приклад змінних у словнику змінних, пов'язаних зі змінною ІО;

➤ Вікно виразу Приклад операції:

- Асоційовані змінні в словнику змінних для умовних виразів в умовних програмах

- Натисніть кнопку вибору, щоб відкрити вікно виразу
- Виберіть змінну, яку потрібно зв'язати
- Натисніть "OK"
- Умовний вираз пов'язаний зі змінною успіх



StartPage x Condition Script x						
+ Add Insert Delete						
Name	IsEnable	Expression	Trigger Mode	Time Interval (ms)	Script Content	
1 ConditionScript	✓		... 1 change	N/A		

StartPage x Condition Script x						
+ Add Insert Delete						
Name	IsEnable	Expression	Trigger Mode	Time Interval (ms)	Script Content	
1 ConditionScript	✓	Var.Variable 4	... Value change	N/A		

- Розширений домен є другим іменуванням словника змінних. Завдяки розширеному домену значення змінної можна отримати пакетами за допомогою сценарію. Назва розширеного домену: розширене доменне ім'я + "#" + серійний номер;

Сценарій розширеного домену	Приклади сценаріїв розширеного домену
- GetExtendedDomainVars: отримати змінні розширеного домену - GetExtendedDomainCount: отримати кількість змінних у розширеному домені	<pre>List = VarCmd.GetExtendedDomainVars(" test ",0) Count = VarCmd.GetExtendedDomainCount(" test ",0) For i = 0 Підрахувати - 1 GetObjectByName("Label10"&i).Text = CStr(List(i)) Далі</pre>

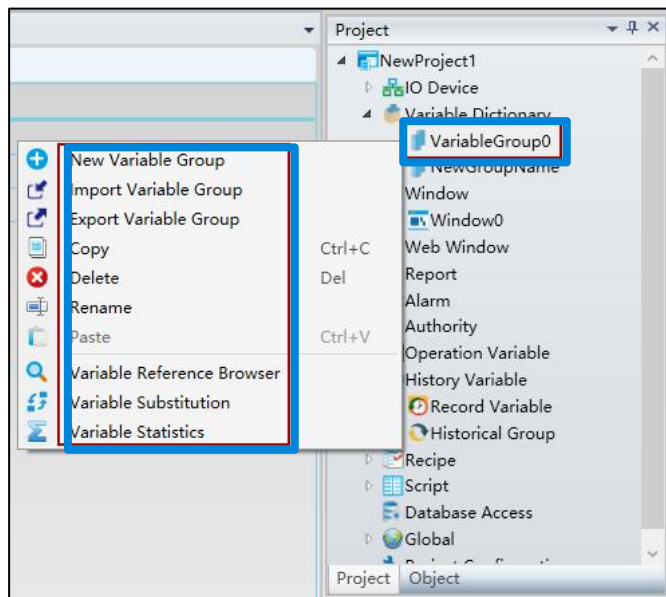
Variable Dictionary x										
StartPage + Add + Insert ✖ Delete + Batch Add 📄 Import 📄 Export										
Name	Variable Type	Initial Value	Retentive Value	Minimum	Maximum Value	Deadband	Decimal Digit	Eng Units	Extended Domain	Description
1 Variable	Analog	0	☐	0	10000	0	0		test#1	
2 Variable1	Analog	0	☐	0	10000	0	0		test#2	
3 Variable2	Analog	0	☐	0	10000	0	0		test#3	
4 Variable3	Analog	0	☐	0	10000	0	0		test#4	
5 Variable4	Analog	0	☐	0	10000	0	0		test#5	

- Огляд змінного словника
- Інструмент керування словником змінних
- Функція правої кнопки миші групи змінних
- Сценарій змінного словника

Функція клацання правою кнопкою миші групи змінних

- Група змінних призначена для керування групою змінних, які мають логічний зв'язок у проекті разом, наприклад, поділ групи змінних відповідно до функціонального модуля змінної в проекті та позиції змінної в проекті. Група змінних підтримує пакетну роботу та безелектродне групування.

● Функція клацання правою кнопкою миші групи змінних ● Опис функції групи змінних



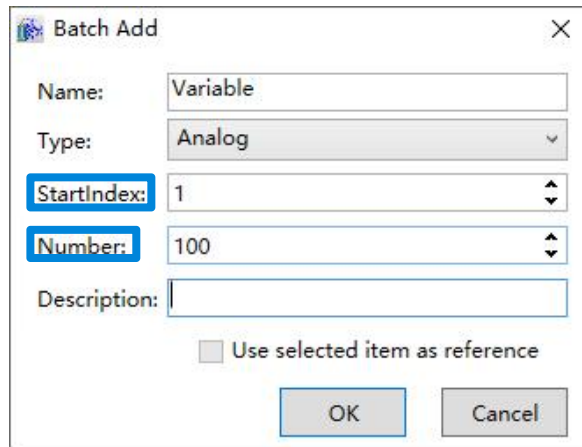
Імпорт групи змінних	Імпортувати групу змінних, імпортувати експортований файл Excel групи змінних
Експорт групи змінних	Експортована група змінних зберігається як файл Excel, який можна створити та змінити у файлі Excel
Копія	Скопіюйте групу змінних, DIAView автоматично скопіює кожен змінну, що міститься
Вставити	Вставте групи змінних
Переглядач змінних посилань	Перегляд довідників змінних, щоб переглянути всі змінні, які використовуються у «словнику змінних»
Заміна змінної	Замініть одну змінну в проекті іншою
Змінна статистика	Підрахуйте загальну кількість змінних, які використовуються в проекті, і використання змінних
Налаштування змінних параметрів	Налаштування параметра змінної, значення, яке відображається кожною змінною під час збою зв'язку вводу-виводу

Пакетна робота груп змінних

➤ DIAView підтримує пакетне створення змінних, копіювання та вставлення груп змінних, імпорт та експорт.

• Імпорт та експорт груп змінних

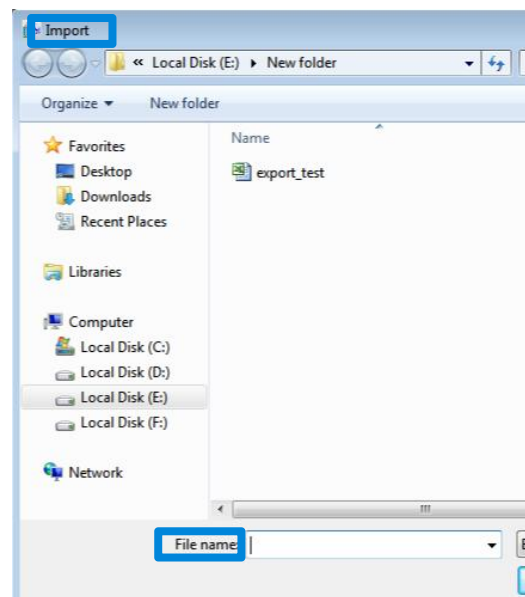
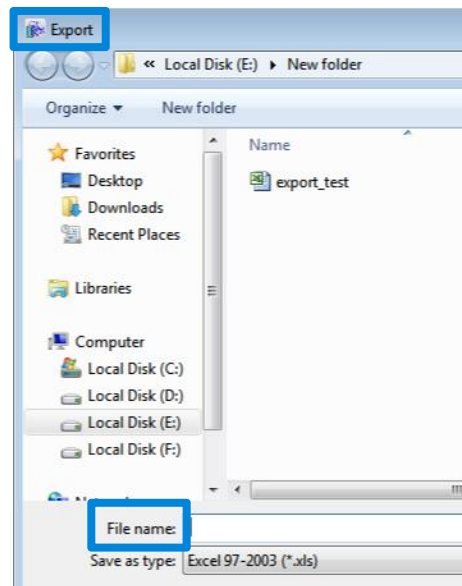
• Змінна нова партія



Batch Add dialog box with the following fields:

- Name: Variable
- Type: Analog
- StartIndex: 1
- Number: 100
- Description: (empty)
- ☐ Use selected item as reference
- OK button
- Cancel button

- **StartIndex:** початковий індекс додавання змінних у пакетах;
- **Кількість:** додавати змінні партіями;

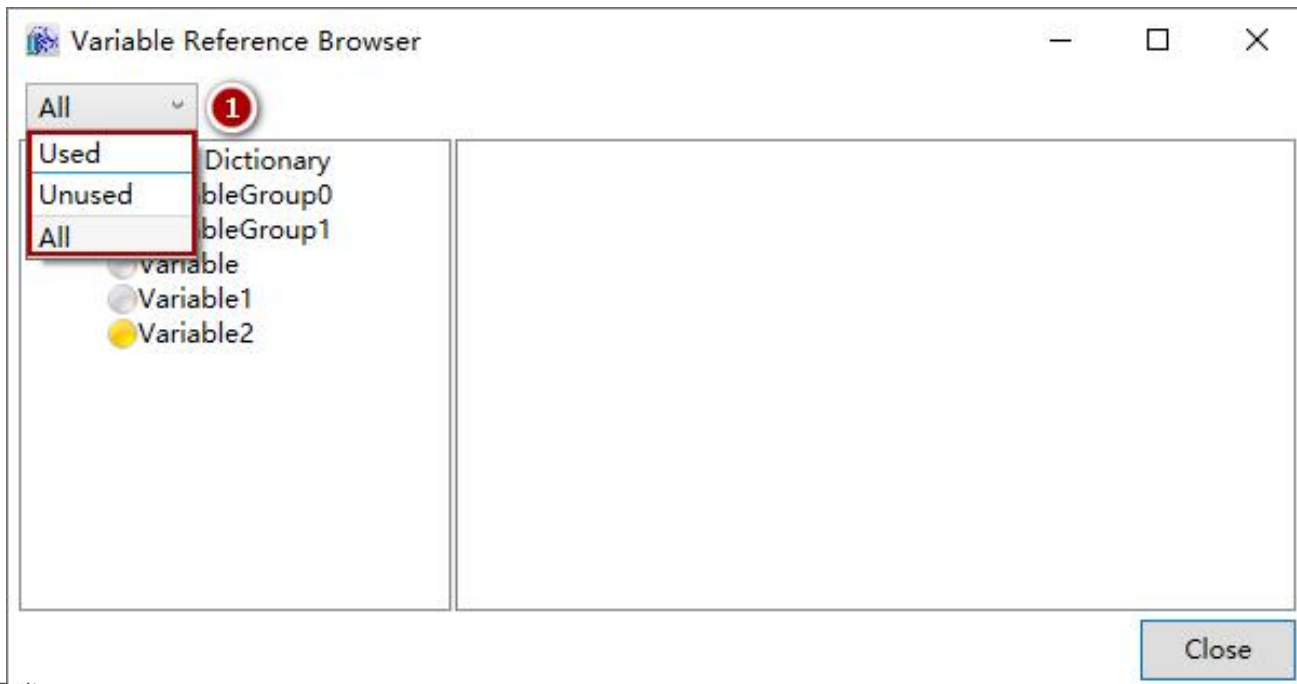


- **Ім'я файлу:** ім'я файлу, збережене змінною пакетного експорту, або ім'я файлу, вибране змінною пакетного імпорту;
- **Зберегти як тип:** тип файлу, підтримує формат Excel xls іxlsx за замовчуванням;

Посилання на змінну

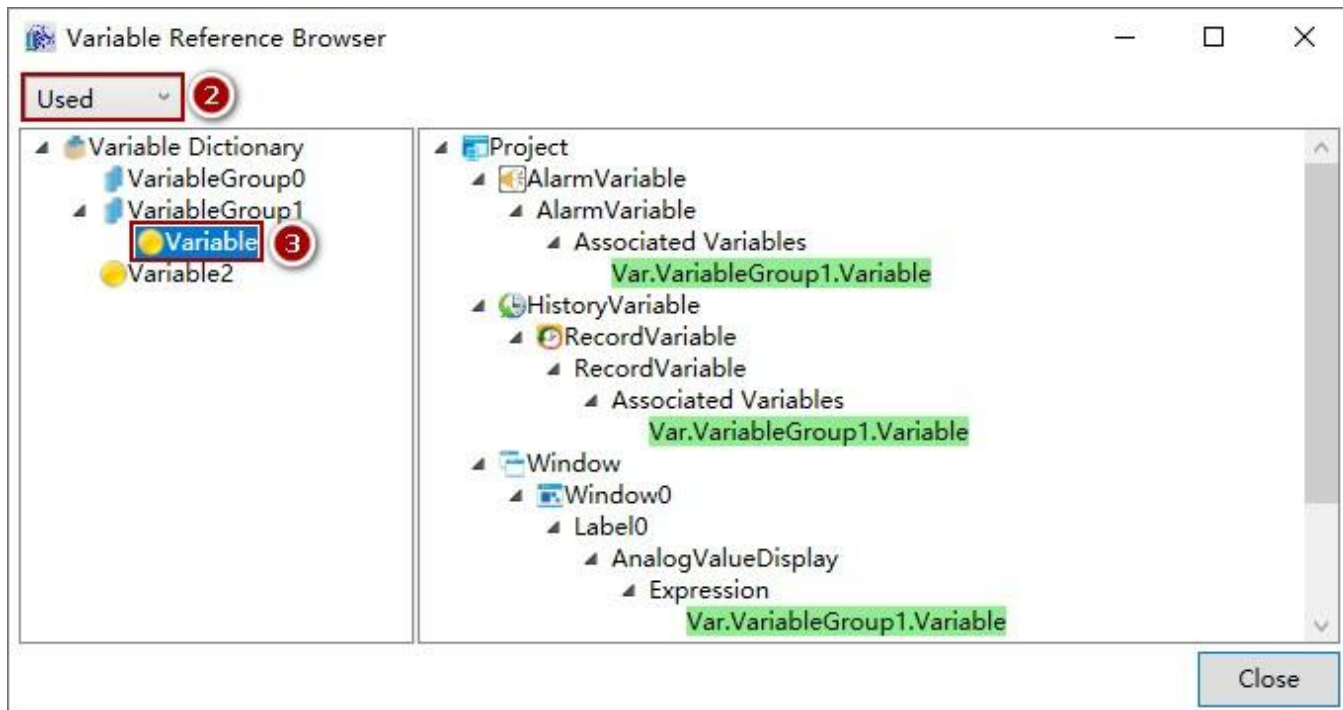
➤ Перегляд довідників змінних: перевірити стан усіх змінних у «словнику змінних»

① Фільтр "використані змінні", "невикористані змінні" та "всі змінні"



Посилання на змінну

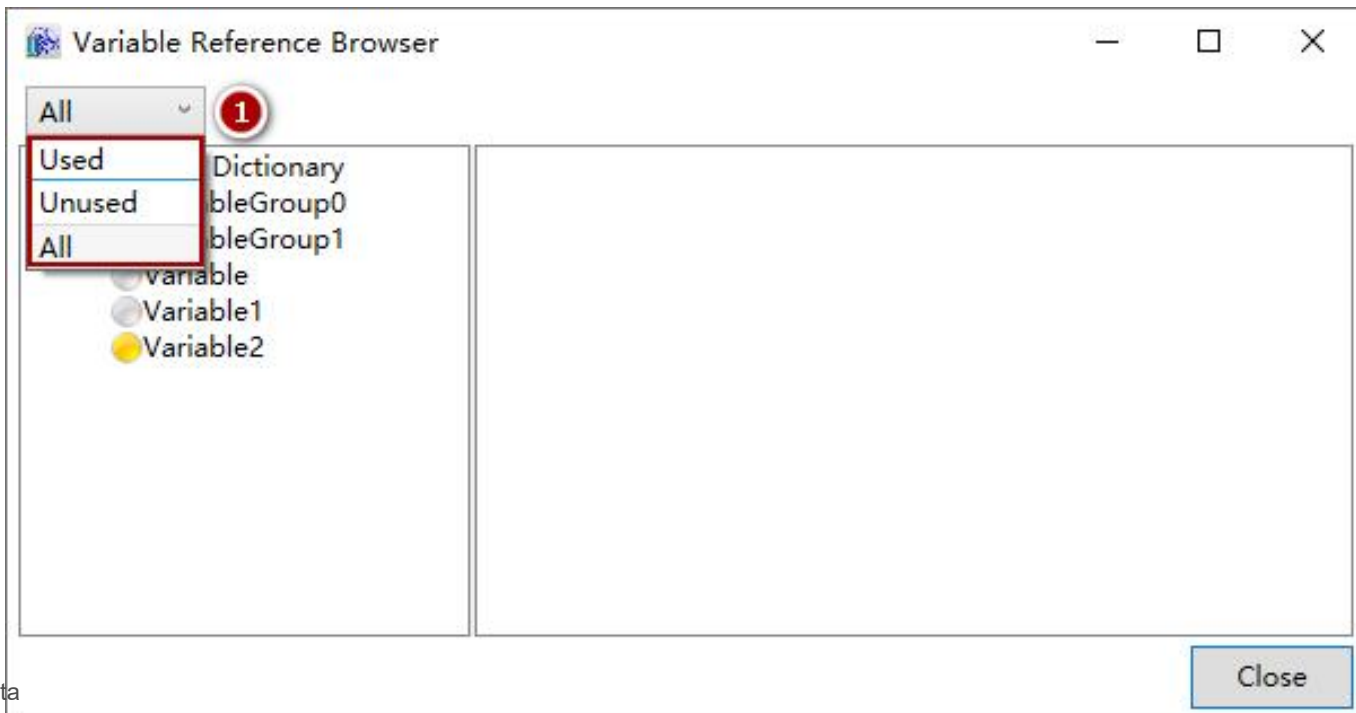
- ② Виберіть «Використані змінні»
- ③ Виберіть «Змінна», у правому вікні перелічено всю довідкову інформацію про змінну;



Посилання на змінну

➤ Перегляд довідників змінних: перевірити стан усіх змінних у «словнику змінних»

① Фільтр "використані змінні", "невикористані змінні" та "всі змінні"

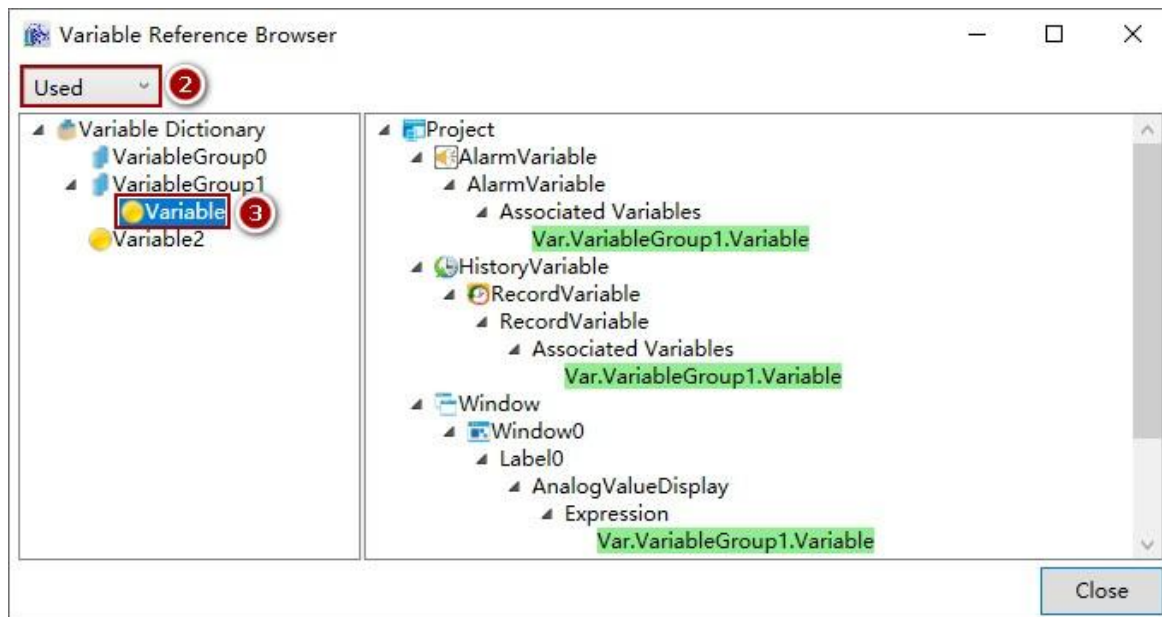


Посилання на змінну

➤ Заміна змінної: замініть одну змінну в проекті іншою.

② Виберіть «Використані змінні»

③ Виберіть «Змінна», у правому вікні перелічено всю довідкову інформацію про змінну;



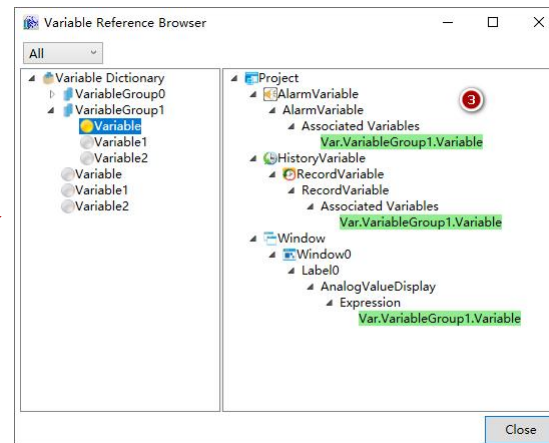
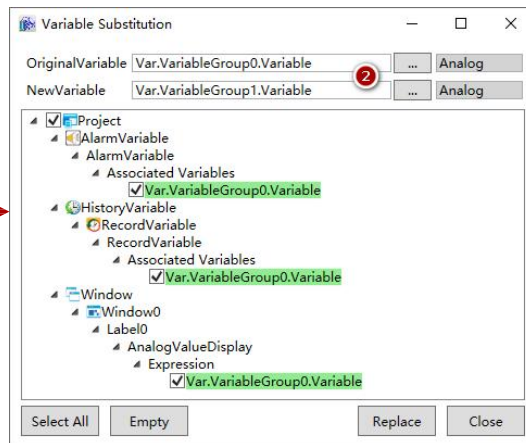
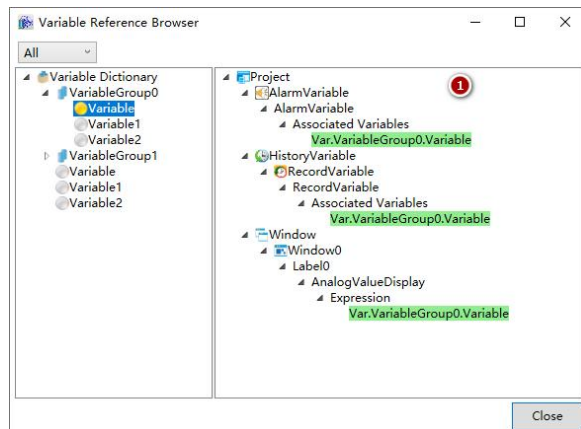
Заміна змінної

➤ Заміна змінної: замініть одну змінну в проекті іншою.

① Перевірте деякі змінні, частина змінної буде замінена іншою змінною в проекті

② Позначте «Вибрати все», щоб замінити всі змінні в проекті іншою змінною

③ Повна заміна змінної

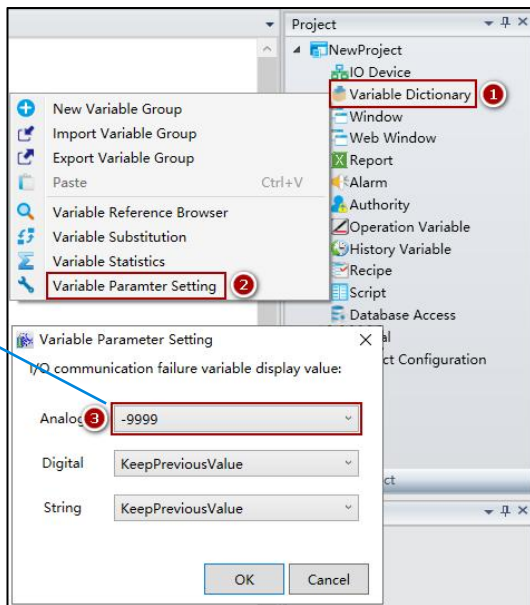


Налаштування змінних параметрів

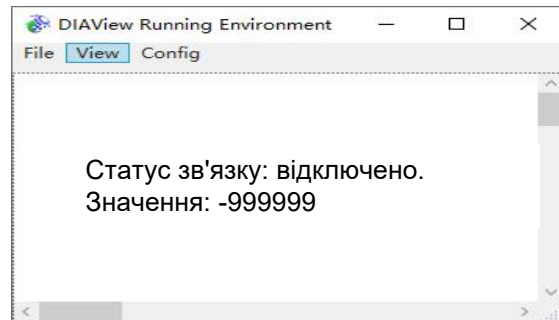
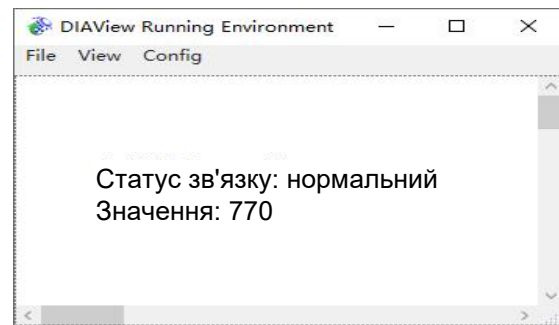
➤ Коли зв'язок вводу-виводу не вдається, значення, яке відображається кожною змінною

- Коли зв'язок між DIAView та ПЛК нормальний, змінна зчитує значення в ПЛК у реальному часі
- Коли зв'язок між DIAView та ПЛК розривається, змінна отримує встановлене значення помилки

Змінний тип	Значення помилки зв'язку IO
Аналоговий	Зберігати попереднє значення, 0, -9999
цифровий	Зберігати попереднє значення True, False
Рядок	Зберегти попереднє значення, ? ? ? ?



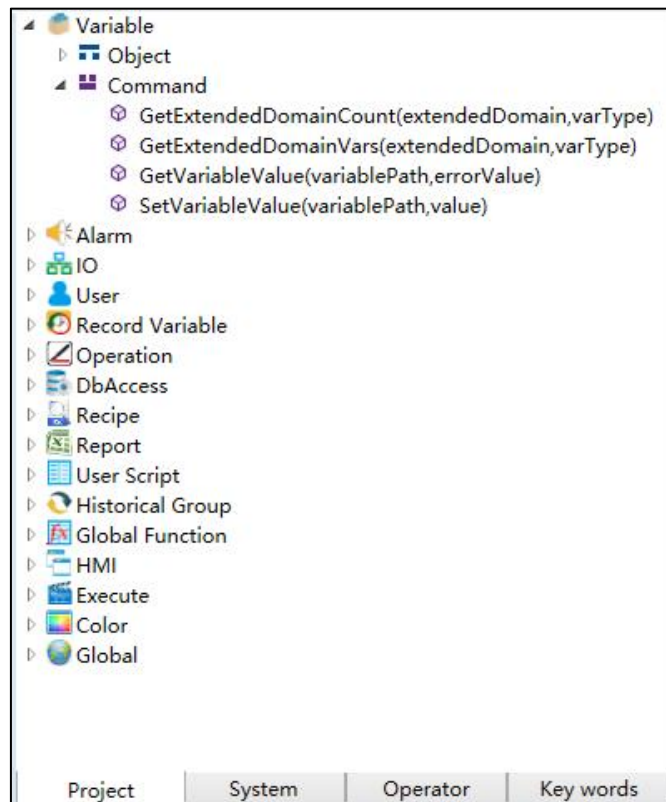
результат



Сценарій змінного словника

- Коли користувачі налаштовують події віконних кнопок, екранні програми та сценарії програм користувача, програмне забезпечення надає прості та зручні у використанні властивості сценаріїв і методи для виконання деяких спеціальних завдань і функцій.

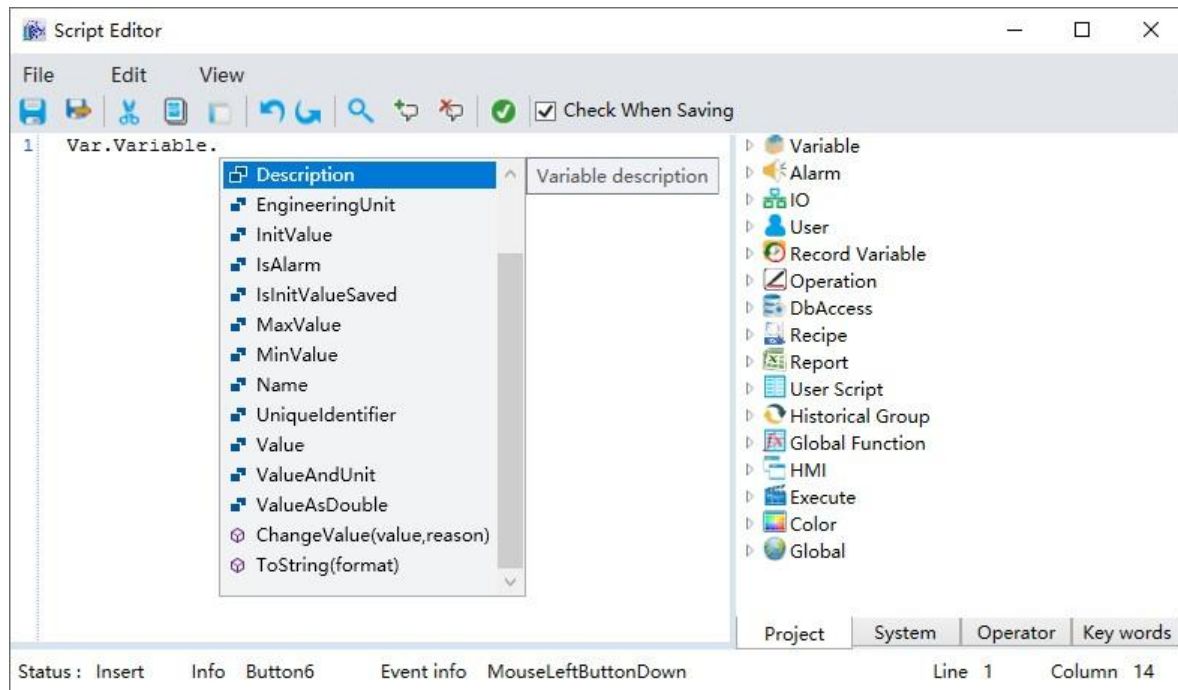
- **Змінний словник відкритих методів**



Сценарій змінного словника

➤ Список методів і властивостей змінних об'єктів

- **GetExtendedDomainCount:** отримати кількість змінних у вказаному розширеному домені та вказаний тип змінної;
- **GetExtendedDomainVars:** отримати набір змінних зазначеного розширеного домену та вказаний тип змінної;
- **GetVariableValue:** отримати значення змінної вказаного шляху до змінної;
- **SetVariableValue:** встановити значення змінної для вказаного шляху до змінної;



Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Історичні змінні

Венді
2020/02/28



Контур

- Огляд історії
- Конфігурація бази даних
- Конфігурація змінного запису
- Конфігурація запису групи історії
- Сценарій

призначення

Після цього розділу ви дізнаєтесь...

- ... Історична конфігурація бази даних

- ... Конфігурація змінного запису та сценарій

- ... Конфігурація та сценарій запису історичної групи

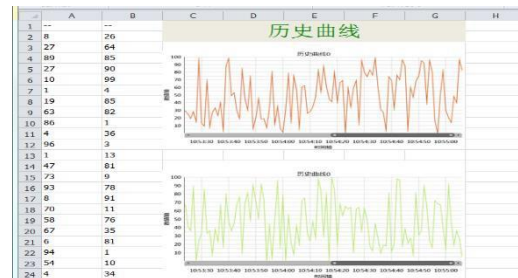
Огляд історії

- Різні дані будуть згенеровані під час роботи проекту.
- Ці дані дуже цінні для користувачів, щоб зрозуміти статус виробництва та історичний статус. Функція історичного запису забезпечує запис і запит цих даних.
 - Підтримка змінних записів і записів історичних груп;
 - Різні режими зберігання: запис часу , запис змін , загальне зберігання, зберігання різниці ;
 - Вбудований таймер для досягнення гнучкого та зручного зберігання та інтервалу запитів;
 - Підтримка My SQL, SQL Server, SQL Compact, Oracle та інших різних баз даних;
 - Підтримка відображення різноманітних історичних записів, таких як звіти та історичні криві ;

DIAView



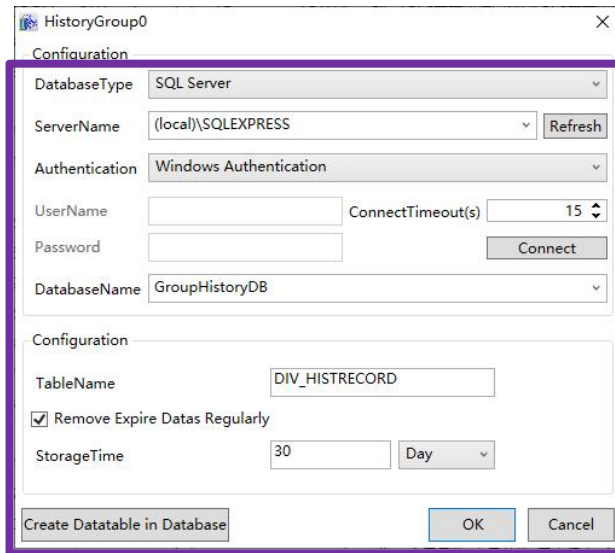
База даних



Конфігурація бази даних

➤ Дані історичного запису зберігаються у вказаній базі даних. Перед початком запису історії необхідно налаштувати запис змінної та груповий запис історії за допомогою бази даних.

- **DatabaseType** : Тип джерела даних, записи змінних підтримують SQL Server, а підтримка групи записів історії SQL Server Compact \ SQL Server \ Oracle \ My SQL
- **ServerName** : ім'я або IP-адреса сервера бази даних, до якого потрібно під'єднатися, натисніть кнопку спадного списку, щоб автоматично шукати сервер у мережі.
- **Автентифікація** : існує два методи автентифікації: автентифікація Windows, автентифікація SQL
- **Connect** : Перевірка підключення, щоб перевірити, чи можна підключитися до сервера бази даних
- **DatabaseName** : ім'я бази даних для використання
- **TableName** : назва таблиці даних, яка буде використовуватися
- **Створити таблицю даних у базі даних** : після натискання цієї кнопки DIAView створює таблицю в базі даних відповідно до конфігурації бази даних



HistoryGroup0

Configuration

DatabaseType: SQL Server

ServerName: (local)\SQLEXPRESS Refresh

Authentication: Windows Authentication

UserName: ConnectTimeout(s): 15

Password: Connect

DatabaseName: GroupHistoryDB

Configuration

TableName: DIV_HISTRECORD

☒ Remove Expire Datas Regularly

StorageTime: 30 Day

Create Datable in Database OK Cancel

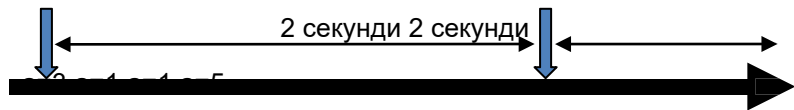
Конфігурація змінної запису

➤ Існує два режими запису змінних:

① **Час :**

Приклад: одиниця часу – 2 секунди.

запис a=3 запис a=5

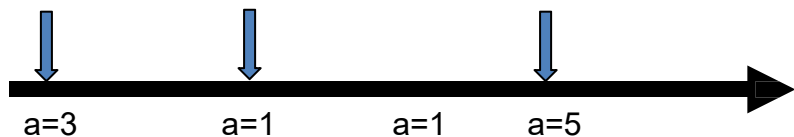


① **Змінено :**

Приклад:

запис a=3 запис a=1

запис a=5



StartPage

Record Variable x

Add Insert Delete Import Export

Name	Associated Variables	Mode	Timer	Deadband	Description
RecordVariable	Var.Variable	Timing	m1	N/A	
RecordVariable1	Var.Variable1	Changed	N/A	0	
RecordVariable2	Var.Variable2	Timing	m1	N/A	

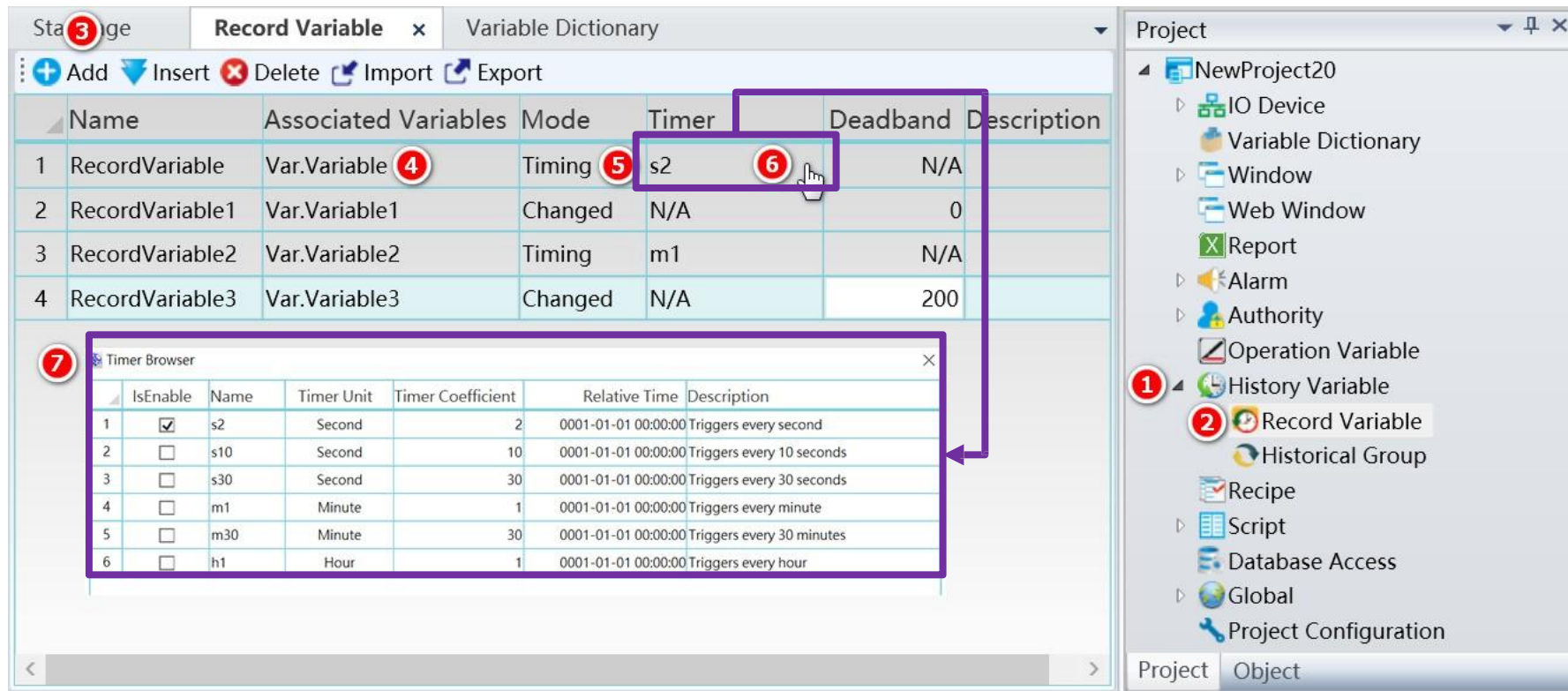
Timer Browser

	IsEnable	Name	Timer Unit	Timer Coefficient	Relative Time	Description
1	☐	s1	Second	1	0001-01-01 00:00:00	Triggers every second
2	☐	s10	Second	10	0001-01-01 00:00:00	Triggers every 10 seconds
3	☐	s30	Second	30	0001-01-01 00:00:00	Triggers every 30 seconds
4	☒	m1	Minute	1	0001-01-01 00:00:00	Triggers every minute
5	☐	m30	Minute	30	0001-01-01 00:00:00	Triggers every 30 minutes
6	☐	h1	Hour	1	0001-01-01 00:00:00	Triggers every hour

Add Insert Delete OK Cancel

Конфігурація змінної запису

① **Приклад режиму синхронізації** : як налаштувати запис RecordVariable кожні 2 секунди.



The screenshot shows the Delta software interface for configuring a Record Variable. The main window is titled 'Record Variable' and contains a table with the following data:

Name	Associated Variables	Mode	Timer	Deadband	Description
RecordVariable	Var.Variable ④	Timing ⑤	s2 ⑥	N/A	
RecordVariable1	Var.Variable1	Changed	N/A	0	
RecordVariable2	Var.Variable2	Timing	m1	N/A	
RecordVariable3	Var.Variable3	Changed	N/A	200	

A 'Timer Browser' dialog is open, showing a table of timers:

IsEnable	Name	Timer Unit	Timer Coefficient	Relative Time	Description
☒	s2	Second	2	0001-01-01 00:00:00	Triggers every second
☐	s10	Second	10	0001-01-01 00:00:00	Triggers every 10 seconds
☐	s30	Second	30	0001-01-01 00:00:00	Triggers every 30 seconds
☐	m1	Minute	1	0001-01-01 00:00:00	Triggers every minute
☐	m30	Minute	30	0001-01-01 00:00:00	Triggers every 30 minutes
☐	h1	Hour	1	0001-01-01 00:00:00	Triggers every hour

The 'Project' tree on the right shows the hierarchy of the project, with 'Record Variable' highlighted with a red circle 2.

Конфігурація змінної запису

② **Приклад зміненого режиму** : як налаштувати запис RecordVariable у зміненому режимі?

Sta 3 je x

Record Variable x

+

 Add

↓

 Insert

×

 Delete

↶

 Import

↷

 Export

	Name	Associated Variables	Mode	Timer	Deadband	Des
1	RecordVariable	Var.Variable	Timing	s1	N/A	
2	RecordVariable1	Var.Variable1 ④	Changed ⑤	N/A	⑥ 1	
3	RecordVariable2	Var.Variable2	Timing	m1	N/A	
4	RecordVariable3	Var.Variable3	Changed	N/A	3	

Project

- HistoricalVariables
 - IO Device
 - Variable Dictionary
 - Window
 - Web Window
 - Report
 - Alarm
 - Authority
 - Operation Variable
 - History Variable ①
 - Record Variable ②
 - Historical Group
 - Recipe
 - Script
 - Database Access
 - Global
 - Project Configuration

Project Object

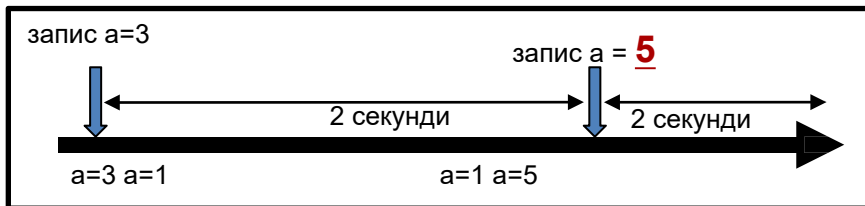
Конфігурація групи історії

➤ Запис історичної групи записує історичні змінні в групах, і кожна група може бути пов'язана з до 256 змінними.

- Спосіб зберігання:

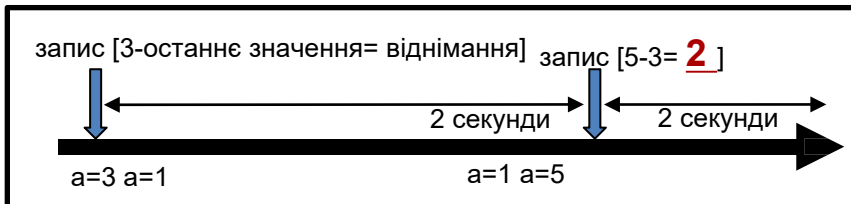
① **Загальне зберігання**: запис даних у реальному часі, створених змінними;

- Тригер часу
- Одиниця часу = 2 сек



① **Відняти пам'ять**: записати різницю між даними, згенерованими змінною, і останніми даних

- Тригер часу
- Одиниця часу = 2 сек



Конфігурація змінної запису

① Загальне зберігання

Star **4** e
HistoryGroup0 x

+ Add
Insert
Delete
Import
Export

	Name	Column Name	Expression	Variable Type	Storage Type	Description
1	RecordVariable	HistVariable	Var.Variable 5	Analog	General Store 6	
2	RecordVariable1	HistVariable1	Var.Variable1	Analog	Subtract Store	

Project

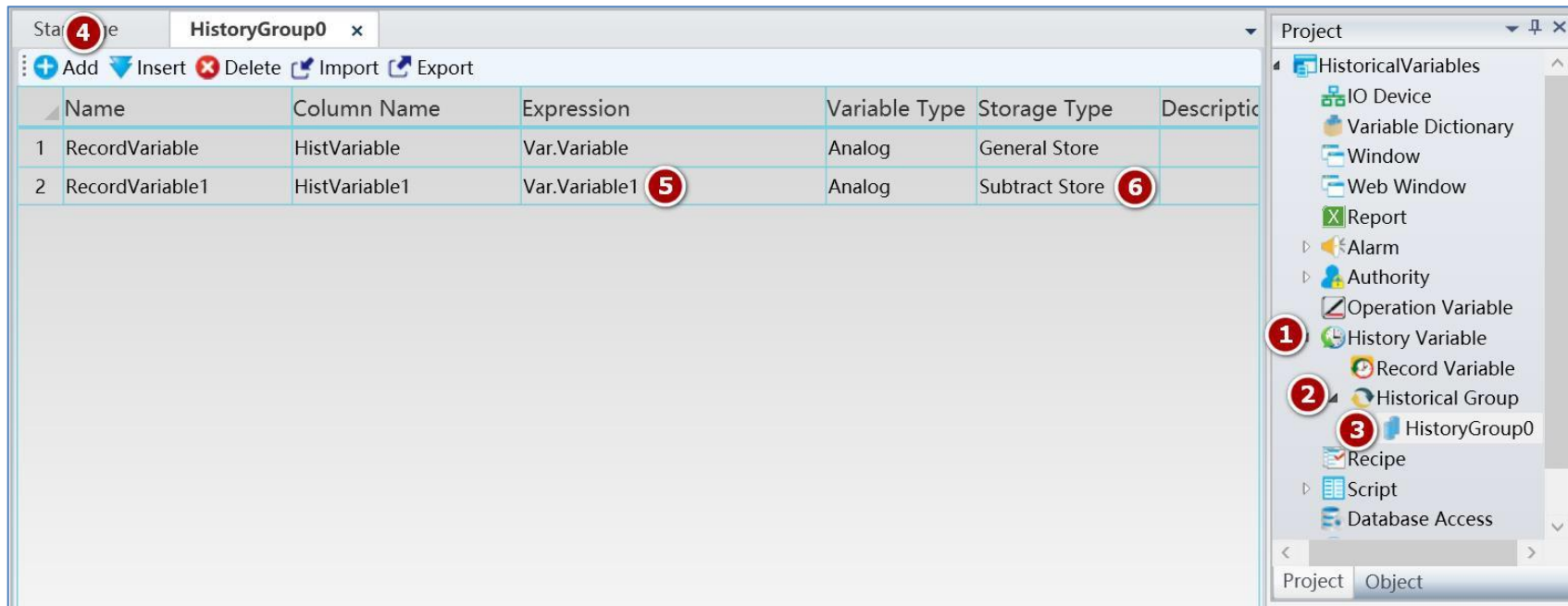
HistoricalVariables

- IO Device
- Variable Dictionary
- Window
- Web Window
- Report
- Alarm
- Authority
- Operation Variable
- 1** History Variable
- 2** Record Variable
- 3** Historical Group
- HistoryGroup0
- Recipe
- Script
- Database Access

Project
Object

Конфігурація змінної запису

② Відніміть зберігання:



Sta 4 je HistoryGroup0 x

+ Add Insert Delete Import Export

	Name	Column Name	Expression	Variable Type	Storage Type	Description
1	RecordVariable	HistVariable	Var.Variable	Analog	General Store	
2	RecordVariable1	HistVariable1	Var.Variable1	Analog	Subtract Store	

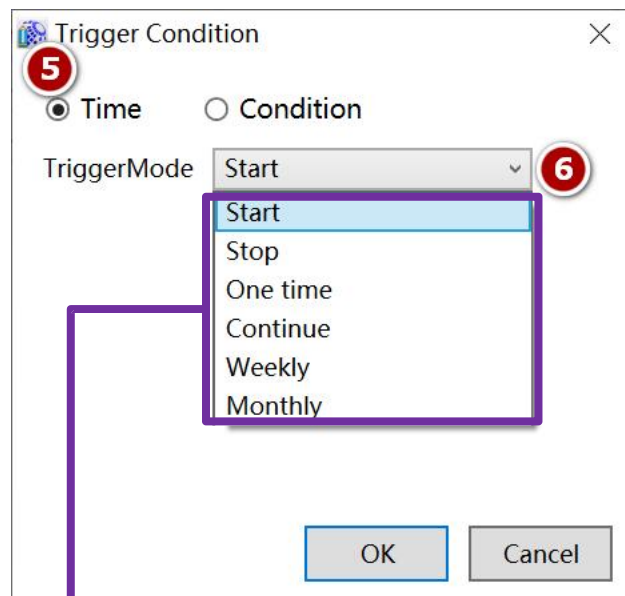
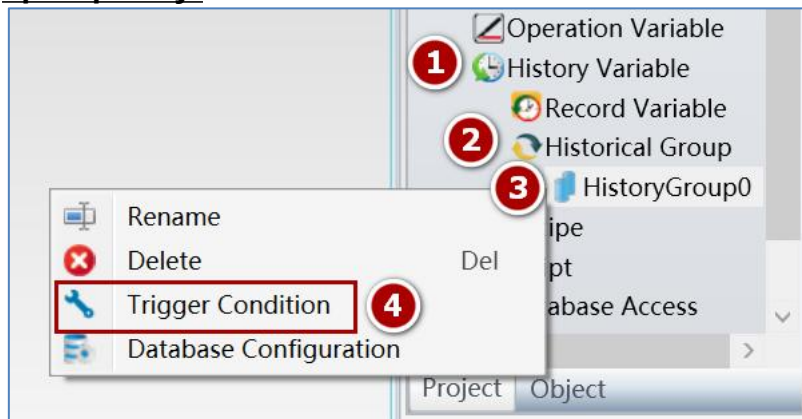
Project

- HistoricalVariables
 - IO Device
 - Variable Dictionary
 - Window
 - Web Window
 - Report
 - Alarm
 - Authority
 - Operation Variable
 - History Variable
 - Record Variable
 - Historical Group
 - HistoryGroup0
 - Recipe
 - Script
 - Database Access

Project Object

Конфігурація змінної запису

① Тригер часу:

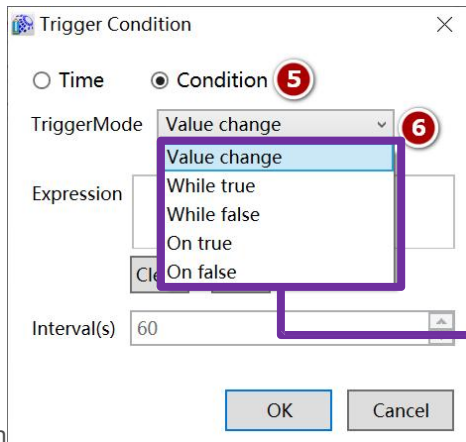
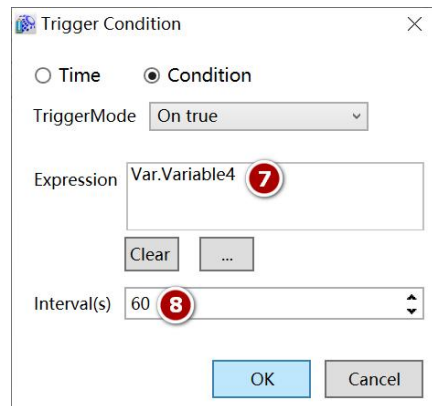
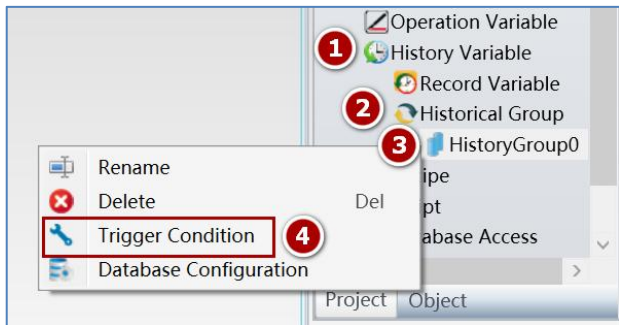


старт	Запис фрагмента даних під час запуску
СТІЙ	Записати частину даних після зупинки
Одного разу	Запис фрагмента даних у фіксований час
Продовжити	Безперервний запис даних
Щотижня	Записуйте дані постійно щотижня
Щомісяця	Записуйте дані постійно щомісяця

Конфігурація змінної запису

➤ Існує два способи запуску:

② Тригер умов:



Зміна значення	Спрацює, коли змінюється значення виразу
Поки правда	Спрацює, коли значення виразу змінюється на true
Поки фальшиві	Спрацює, коли значення виразу змінюється на false
По правді	Спрацює, коли значення виразу залишається істинним
На помилкові	Спрацює, коли значення виразу залишається false
Зміна значення	Спрацює, коли змінюється значення виразу

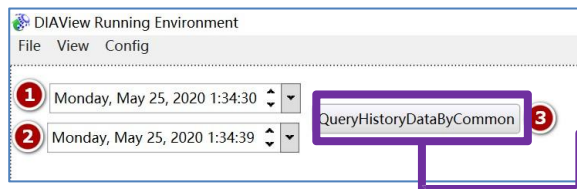
Сценарій запиту змінного запису

Приклад 1:

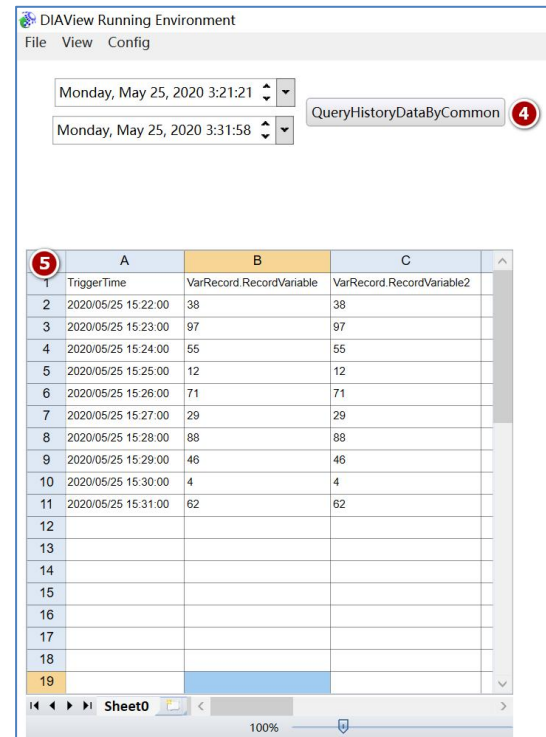
Запис змінної зчитується через сценарій і відображається у звіті.

- QueryHistoryDataByCommon(sheetIndex, умови, типи): запит на запис змінної
 - sheetIndex: індекс аркуша
 - умови: колекція змінних записів, розділених комами
 - типи: TriggeringTime, Value, також можна замінити на 0 і 1

Відповідно



```
DateTimePicker0.Value = Sys.StartTime
DateTimePicker1.Value = Sys.Now
Звіт про виклик0.SetWorkSheetStartTime(0,DateTime
Picker0.ValueTime)
Виклик Report0.SetWorkSheetEndTime(0,DateT
imePicker1.ValueTime)
Виклик Report0.SetVarRecordRulerName(0,"m1")
Виклик Report0.QueryHistoryDataByCommon(0,"Var
Record.RecordVariable,VarRecord.RecordVariabl
e,VarRecord.RecordVariable2","0,1,1")
```



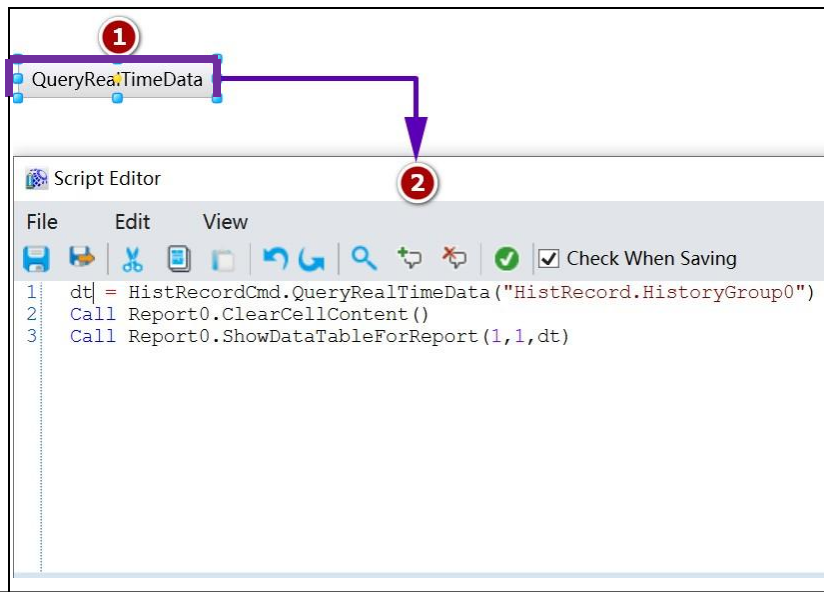
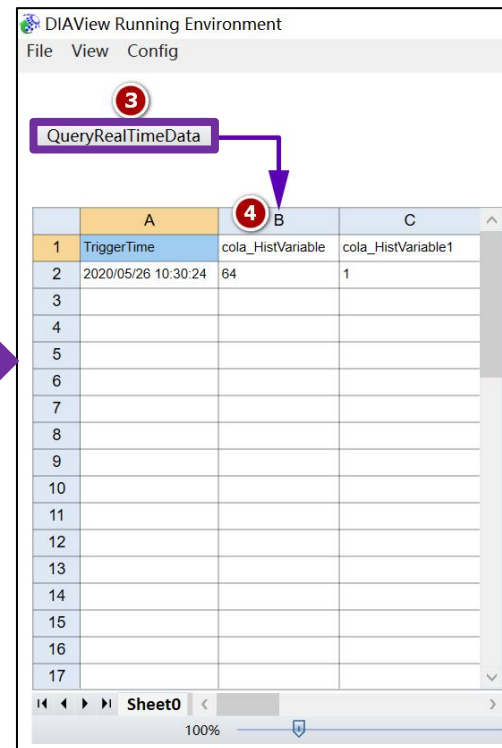
	A	B	C
1	TriggerTime	VarRecord.RecordVariable	VarRecord.RecordVariable2
2	2020/05/25 15:22:00	38	38
3	2020/05/25 15:23:00	97	97
4	2020/05/25 15:24:00	55	55
5	2020/05/25 15:25:00	12	12
6	2020/05/25 15:26:00	71	71
7	2020/05/25 15:27:00	29	29
8	2020/05/25 15:28:00	88	88
9	2020/05/25 15:29:00	46	46
10	2020/05/25 15:30:00	4	4
11	2020/05/25 15:31:00	62	62
12			
13			
14			
15			
16			
17			
18			
19			

Сценарій запису групи історії запитів

➤ приклад 2:

Зчитуйте дані в реальному часі групи історичних записів за допомогою сценарію та відображайте їх у звіті.

- QueryRealTimeData: запит даних у реальному часі групи історичних записів;

The screenshot shows the DIAView Running Environment. A red circle with the number 3 is placed over the 'QueryRealTimeData' function in the top toolbar. A red circle with the number 4 is placed over the 'B' column header in the data table. A purple arrow points from the toolbar icon to the column header.

	A	B	C
1	TriggerTime	cola_HistVariable	cola_HistVariable1
2	2020/05/26 10:30:24	64	1
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			

Сценарій запису групи історії запитів

➤ приклад 3:

Прочитайте дані групи історичних записів за допомогою сценарію та відобразити його зі звітом.

- QueryWithStorageItemName: Запит даних групи історичних записів, наприклад запит даних за останню годину;
- Використовуйте функцію DateAdd, щоб установити діапазон часу запиту
минула година;



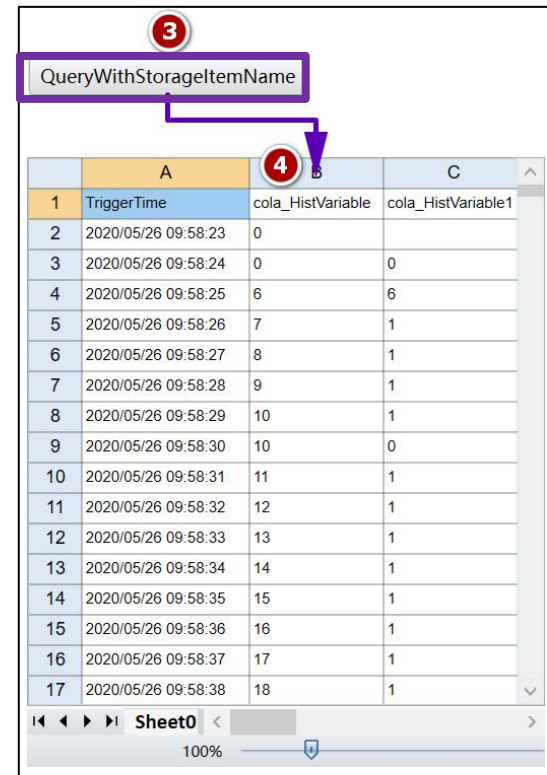
Script Editor

File Edit View


☒ Check When Saving

```

1 Dim StartTime
2 StartTime = DateAdd("h",-1,Now())
3 dt = HistRecordCmd.QueryWithStorageItemName("HistoryGroup0", "RecordVariable",RecordVariable1",StartTime, Sys.Now,-1)
4 Call Report1.ClearCellContent()
5 Call Report1.ShowDataTableForReport(1,1,dt)
    
```



	A	B	C
1	TriggerTime	cola_HistVariable	cola_HistVariable1
2	2020/05/26 09:58:23	0	
3	2020/05/26 09:58:24	0	0
4	2020/05/26 09:58:25	6	6
5	2020/05/26 09:58:26	7	1
6	2020/05/26 09:58:27	8	1
7	2020/05/26 09:58:28	9	1
8	2020/05/26 09:58:29	10	1
9	2020/05/26 09:58:30	10	0
10	2020/05/26 09:58:31	11	1
11	2020/05/26 09:58:32	12	1
12	2020/05/26 09:58:33	13	1
13	2020/05/26 09:58:34	14	1
14	2020/05/26 09:58:35	15	1
15	2020/05/26 09:58:36	16	1
16	2020/05/26 09:58:37	17	1
17	2020/05/26 09:58:38	18	1

Sheet0 100%

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Об'єкт

рубін

2020/02/27



Контур

- Концепції Toolbox
- Як малювати базову графіку
- Як намалювати елементи керування вікном
- Як намалювати Extend Controls

призначення

У цьому розділі ви дізнаєтеся....

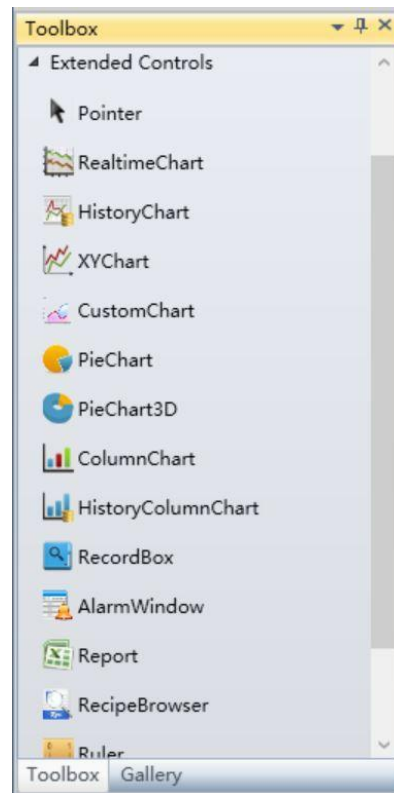
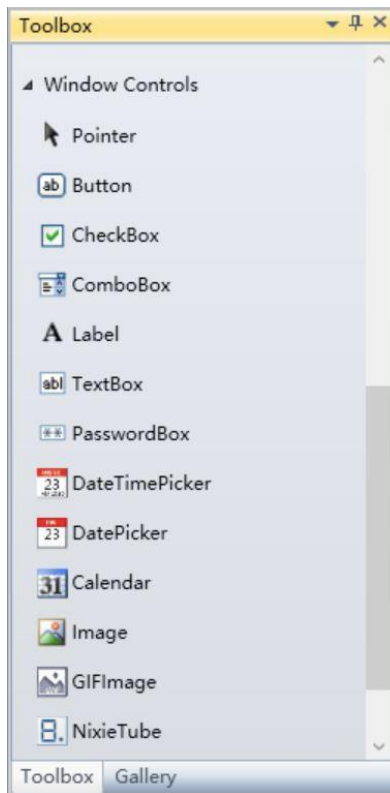
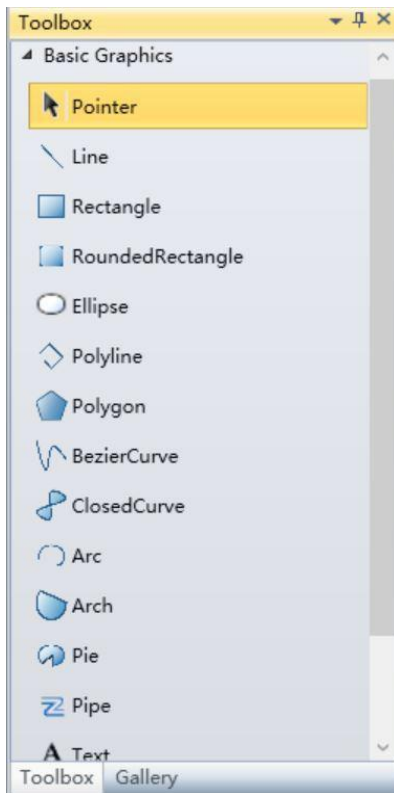
- ... докладніше про основну графіку докладніше
- ... про елементи керування вікном докладніше про
- ... елементи керування розширенням

Контур

- Концепції Toolbox
- Як малювати базову графіку
- Як намалювати елементи керування вікном
- Як намалювати Extend Controls

Знайомство з Toolbox

- Панель інструментів у вікні розробки графіки містить базову графіку, елементи керування вікнами та розширені елементи керування. Як показано на малюнку нижче:

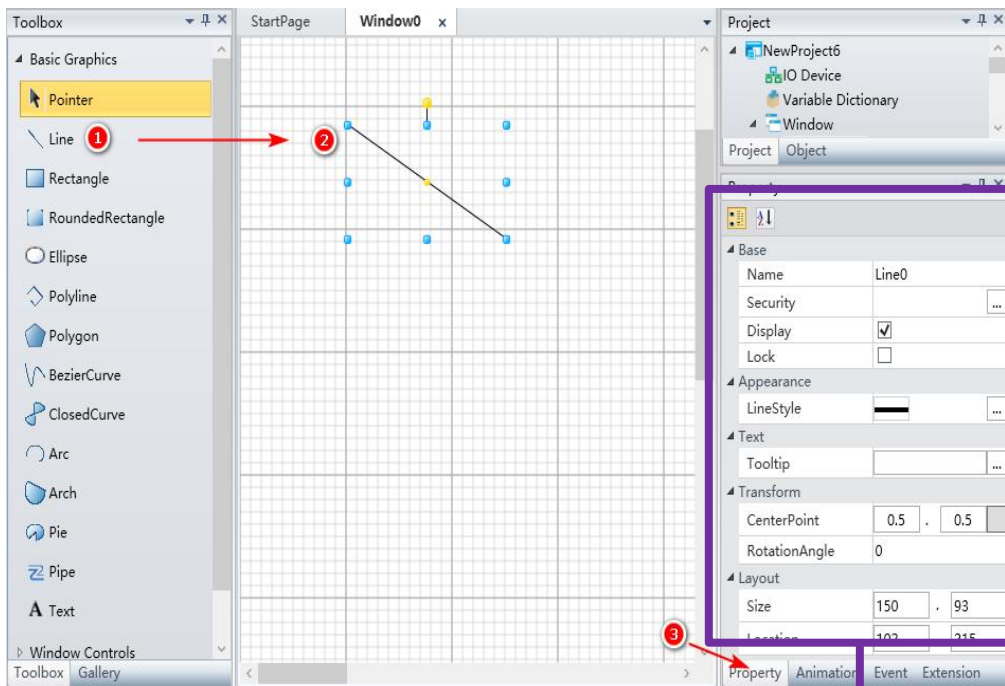


Контур

- Концепції Toolbox
- Як малювати базову графіку
- Як намалювати елементи керування вікном
- Як намалювати Extend Controls

Основна графіка-Line

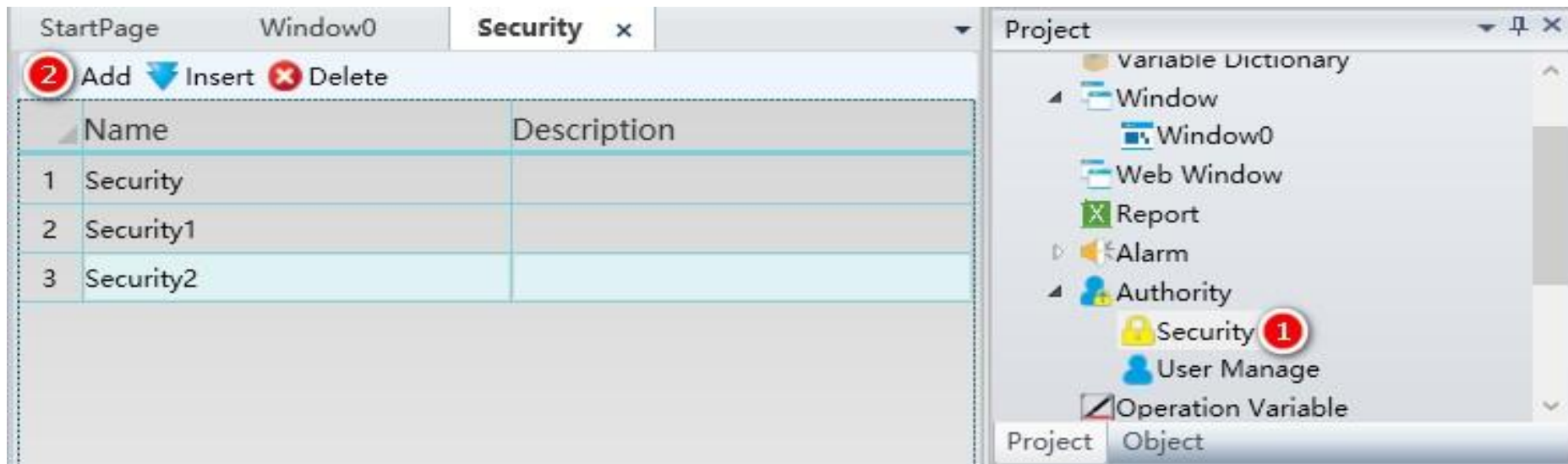
➤ Намалуйте Line0, відкрийте вікно її властивостей



- **Назва** : назва графіки у вікні; назви графічних об'єктів в одному вікні не можуть збігатися.
- **Безпека** : встановить повноваження роботи графіки та призначте належність графіки до однієї чи кількох зон безпеки для керування повноваженнями користувача.
- **Дисплей** : чи відображати графіку під час виконання.
- **Блокувати** : чи блокувати графіку; після блокування операції миші стануть недійсними.
- **LineStyle** : установить тип лінії, наприклад, стиль, товщину, колір тощо.
- **Підказка** : установить вміст підказки, який миша залишатиме на малюнку під час виконання
- **CenterPoint** : установить розташування центральної точки графіки
- **RotationAngle** : установить кут повороту графіки.
- **Розмір** : установить розмір графіки.
- **Розташування** : розташування графіки у вікні.

➤ Приклад зони безпеки:

(1) Створіть зону безпеки



StartPage Window0 Security x

2 Add Insert Delete

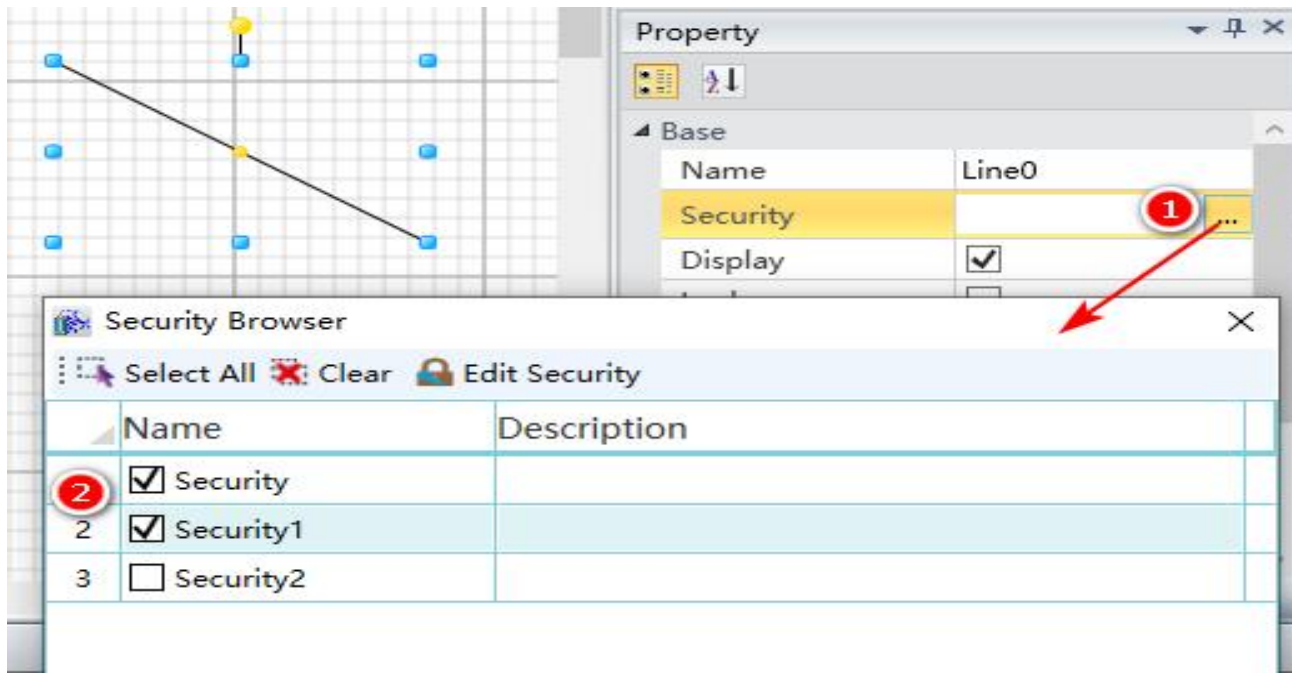
	Name	Description
1	Security	
2	Security1	
3	Security2	

Project

- variable Dictionary
- Window
 - Window0
- Web Window
- Report
- Alarm
- Authority
 - Security 1
 - User Manage
- Operation Variable

Project Object

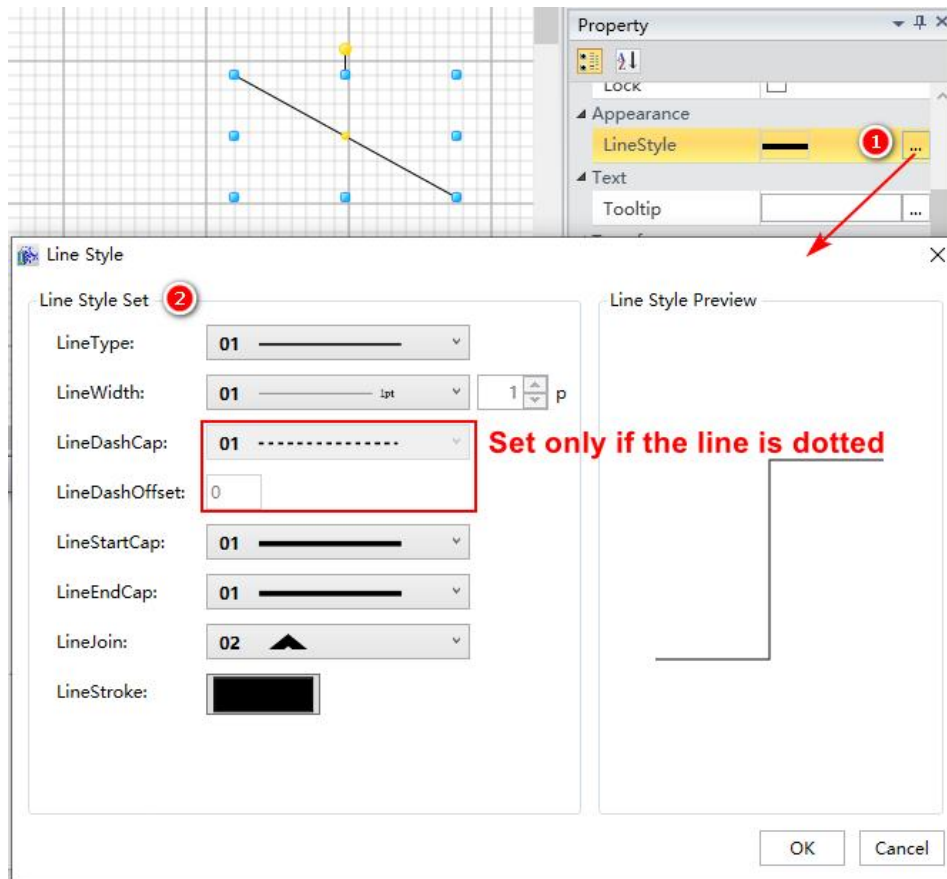
(2) Графічна пов'язана зона безпеки



The screenshot displays a graphical interface for configuring a security zone. On the left, a grid shows a line segment connecting two points, with several blue square markers. On the right, a 'Property' window is open, showing the 'Base' section with a table of properties. The 'Security' property is highlighted in yellow and has a red circle with the number '1' next to its value field. A red arrow points from this field to the 'Security Browser' window below. The 'Security Browser' window has a toolbar with 'Select All', 'Clear', and 'Edit Security' buttons. It contains a table with the following data:

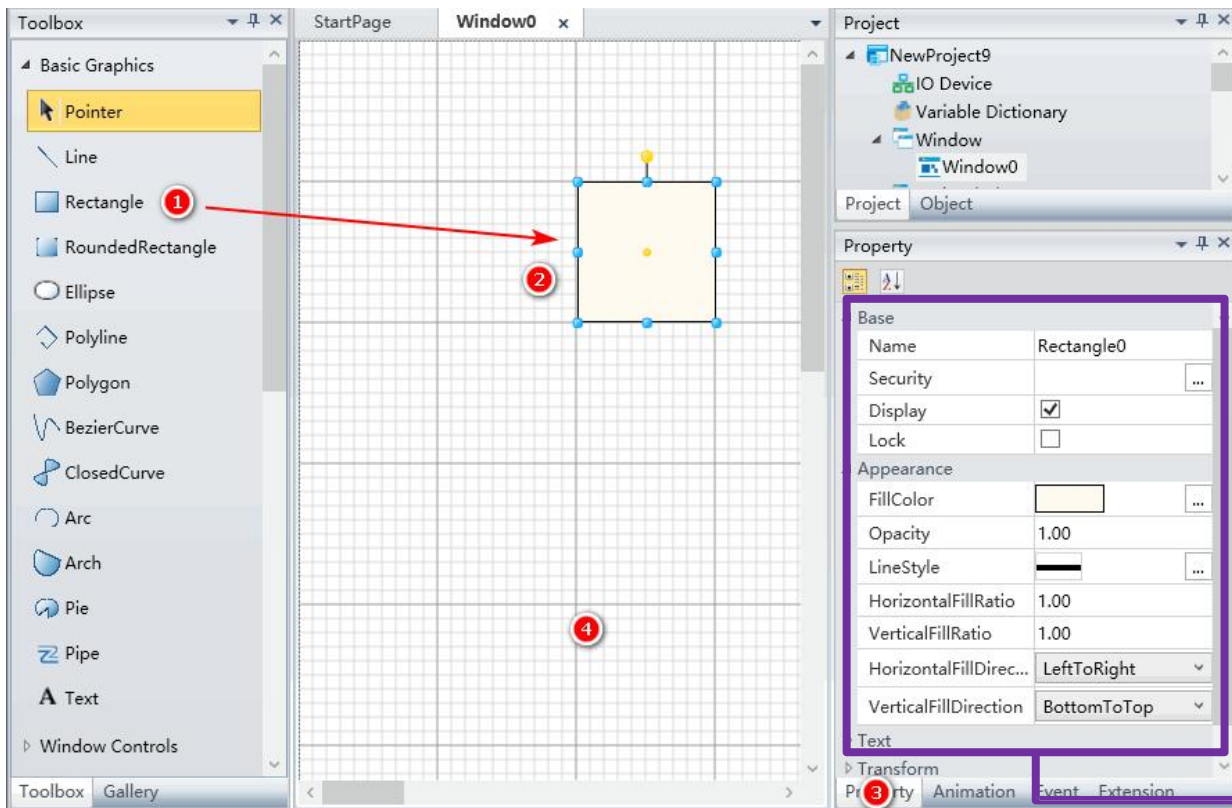
	Name	Description
2	☒ Security	
2	☒ Security1	
3	☐ Security2	

- Встановити стиль лінії графіки



Базова графіка-прямокутник

➤ Намалуйте Rectangle0, відкрийте вікно його властивостей

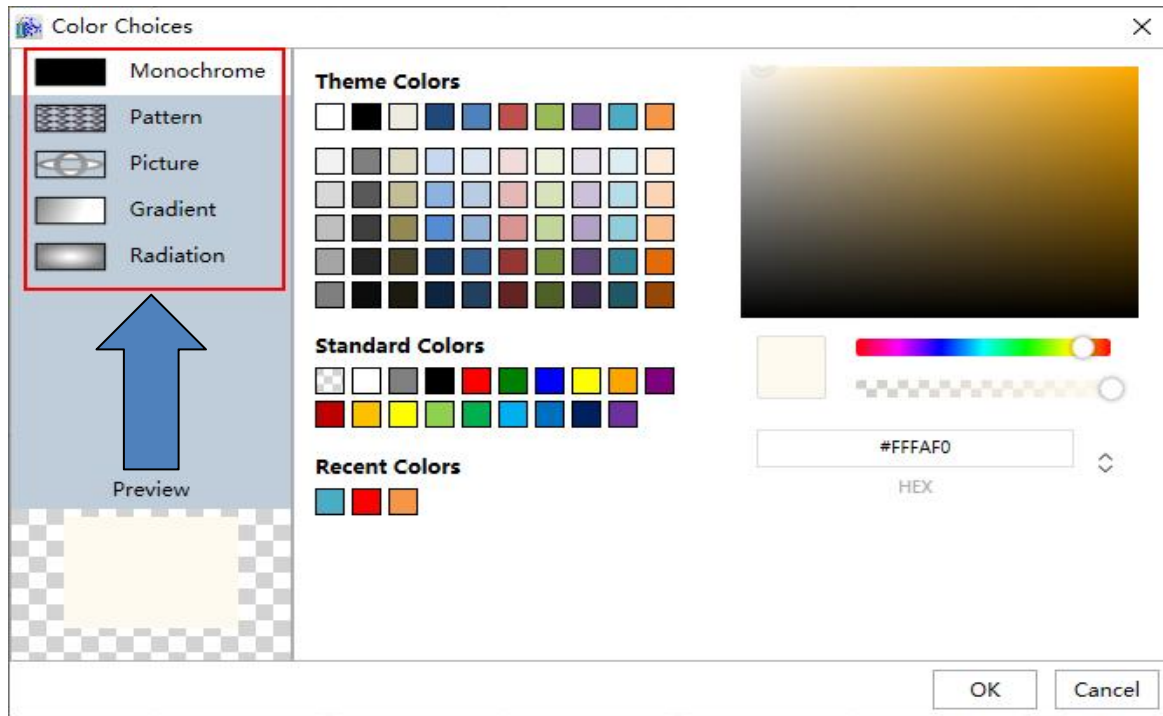


- **Колір заливки:** установіть колір заливки прямокутника
- **Непрозорість :** установіть прозорість прямокутника
- **LineStyle :** встановити тип лінії кордону для наприклад, стиль, товщина, колір тощо
- **HorizontalFillRatio:** Встановити коефіцієнт горизонтального заповнення прямокутника (0~1)
- **HorizontalFillDirection:** установіть горизонтальний напрямок заповнення прямокутника
- **VerticalFillRatio:** Установіть вертикальне заповнення частка прямокутника (0~1)
- **VerticalFillDirection:** установіть напрямок вертикального заповнення прямокутника

➤ Існує п'ять режимів колірної заливки в кольорах:

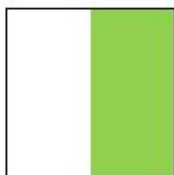
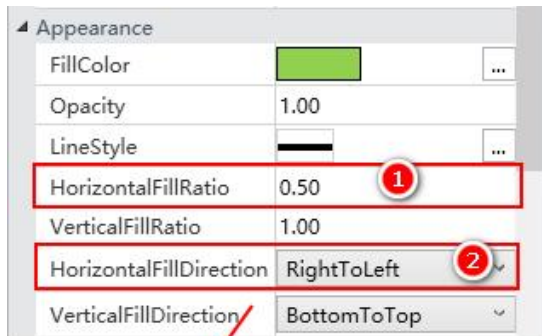
Як показано на малюнку нижче:

- ① Монохромна заливка
- ② Заливка візерунком
- ③ Заливка малюнка
- ④ Градієнтна заливка
- ⑤ Радіаційне заповнення



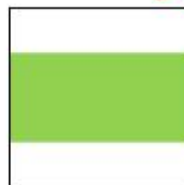
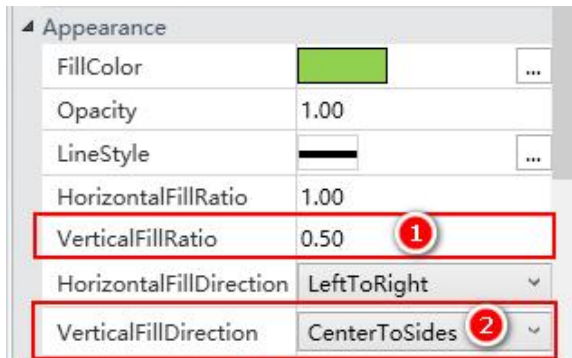
Коефіцієнт заповнення та напрямок

➤ Коефіцієнт заповнення та напрямок



3

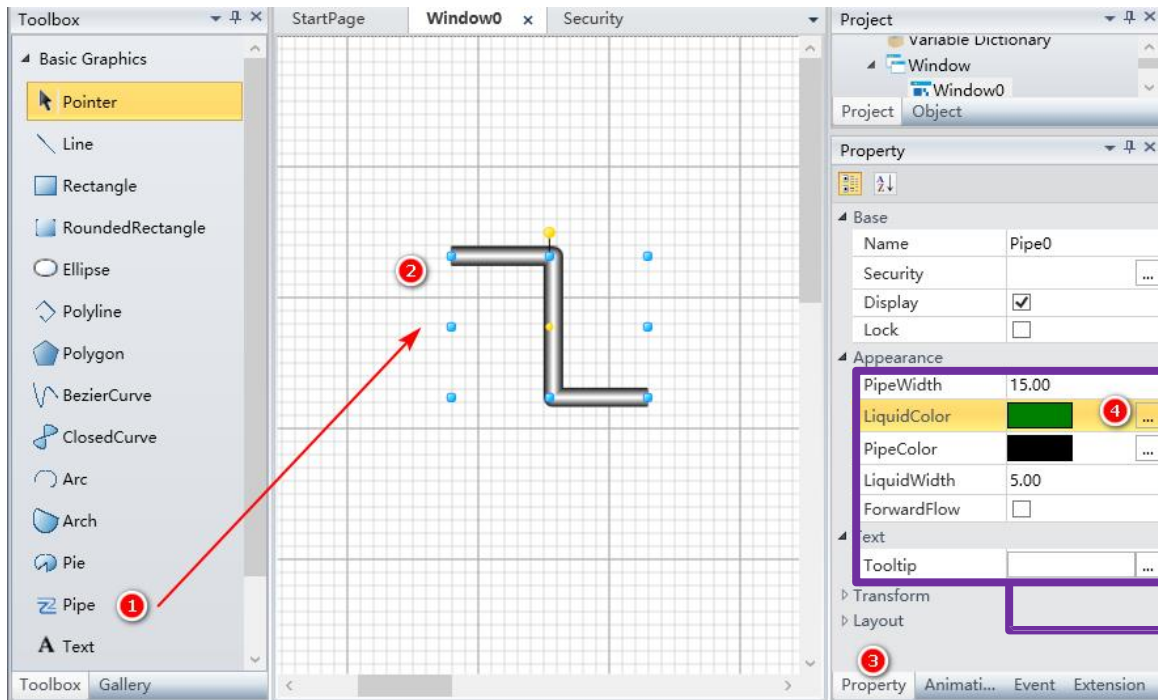
Горизонтальний коефіцієнт заповнення та напрямок



3

Коефіцієнт вертикального заповнення та напрямок

- Намалюйте Pipe0, відкрийте вікно його властивостей

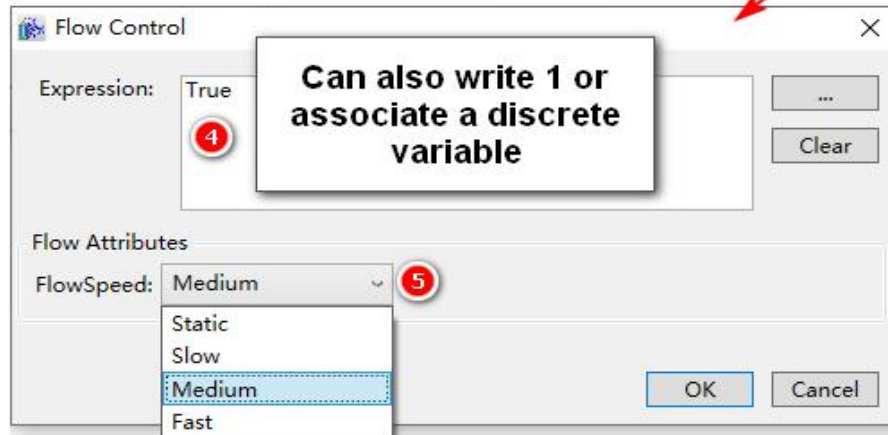
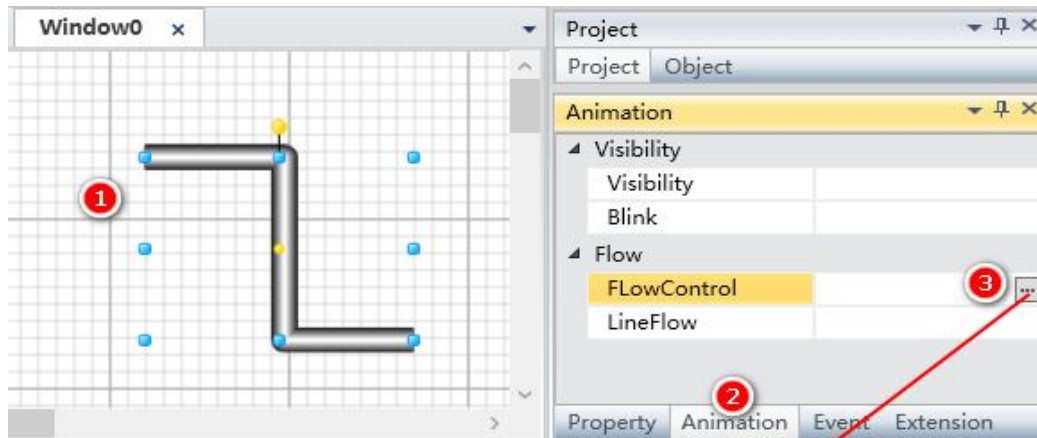


- **PipeWidth:** установіть ширину труби.
- **LiquidColor :** Установіть колір рідини в трубі.
- **PipeColor:** Установіть колір труба
- **LiquidWidth :** установіть ширину рідини в трубі
- **ForwardFlow:** Установіть напрямок потоку рідини в трубі.

Створіть плавну трубу

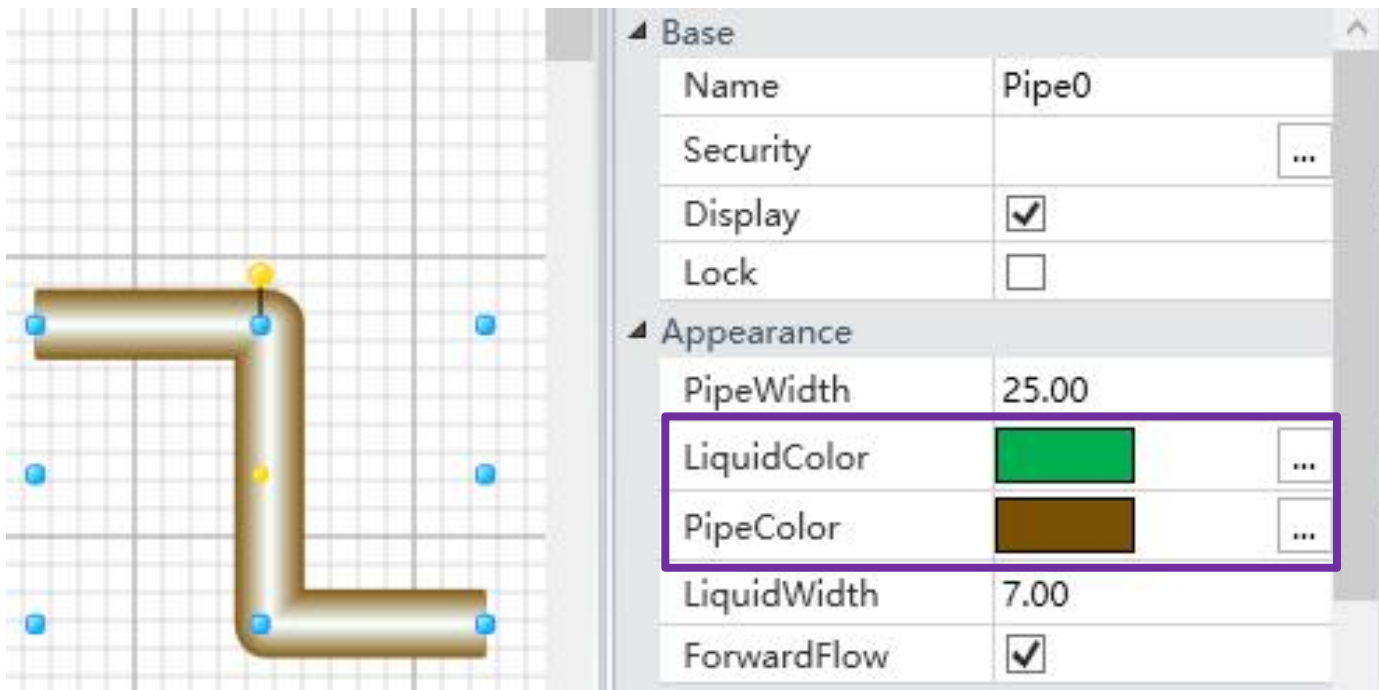
➤ Приклад текучої труби:

(1) Налаштуйте керування потоком
анімація труби.



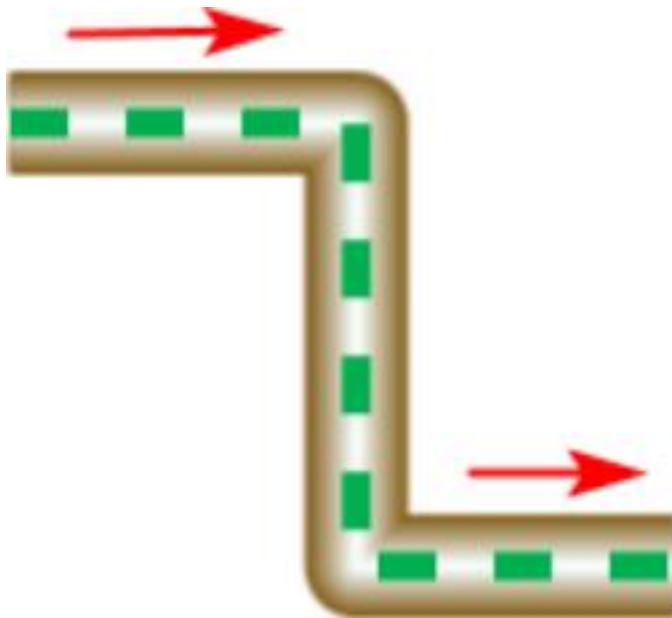
Створіть плавну трубу

(2) Встановити властивості труби

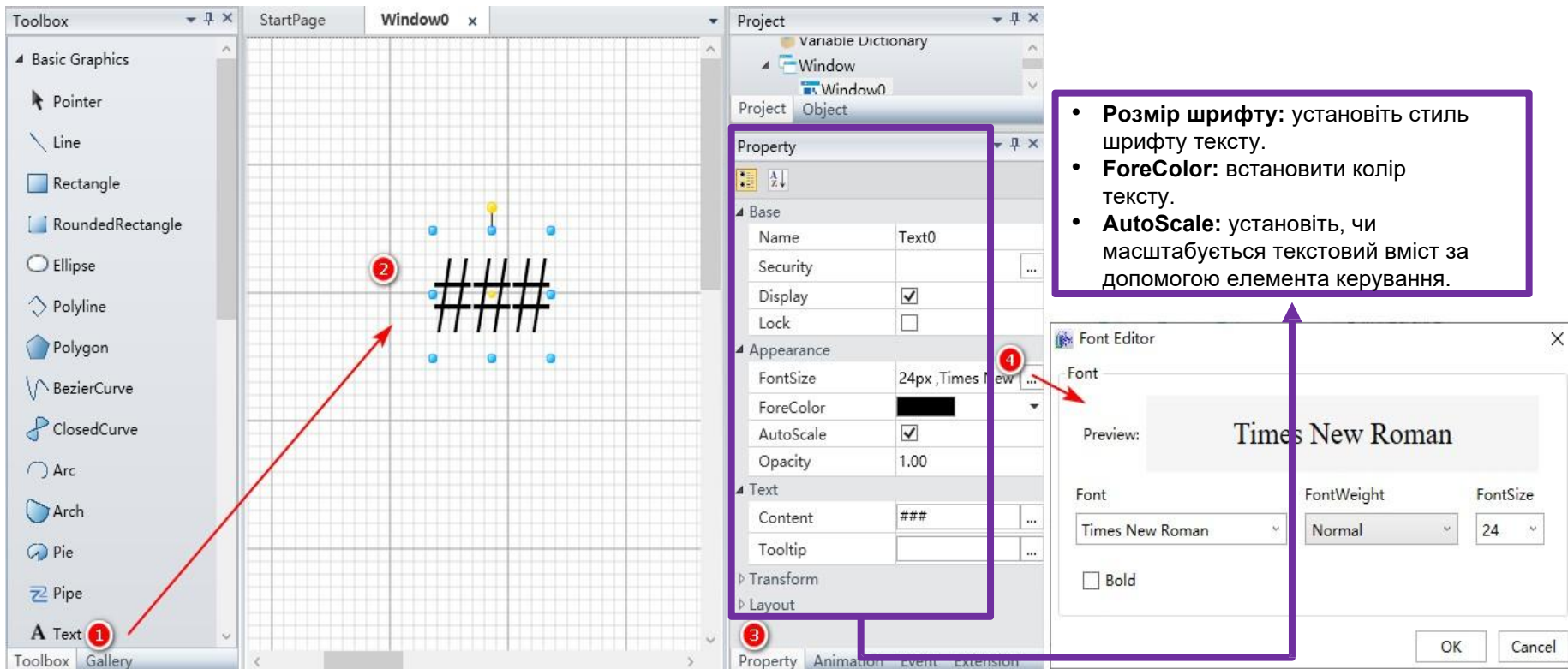


Створіть плавну трубу

(3) Запустіть проект.



➤ Намалюйте Text0, відкрийте вікно його властивостей



The screenshot displays the Delta software interface with the following components:

- Toolbox:** Located on the left, it contains various drawing tools. The 'Text' tool is highlighted with a red circle and labeled '1'.
- Canvas:** The central workspace shows a grid with a text object 'Text0' represented by a grid of hash symbols. A red arrow points from the 'Text' tool to this object, and a red circle with the number '2' is placed near the object.
- Property Window:** Located on the right, it shows the properties of the selected 'Text0' object. The 'Appearance' section is expanded, showing settings for 'FontSize' (24px), 'ForeColor' (black), and 'AutoScale' (checked). A red circle with the number '3' is placed near the 'Property' tab.
- Font Editor:** A dialog box is open, showing the 'Font' settings. The 'Font' dropdown is set to 'Times New Roman', and the 'FontSize' is set to '24'. A red arrow points from the 'Font' dropdown in the Property window to this dialog. A red circle with the number '4' is placed near the 'Font' dropdown in the Property window.
- Annotations:** A purple box highlights the 'Property' window and the 'Font Editor' dialog. A purple arrow points from the 'Font Editor' dialog to the 'Property' window.

Properties of Text0:

Section	Property	Value
Base	Name	Text0
	Security	
	Display	☒
	Lock	☐
Appearance	FontSize	24px, Times New Roman
	ForeColor	Black
	AutoScale	☒
	Opacity	1.00
Text	Content	###
	Tooltip	
Transform		
Layout		

Font Editor Settings:

Property	Value
Font	Times New Roman
FontWeight	Normal
FontSize	24
Bold	☐

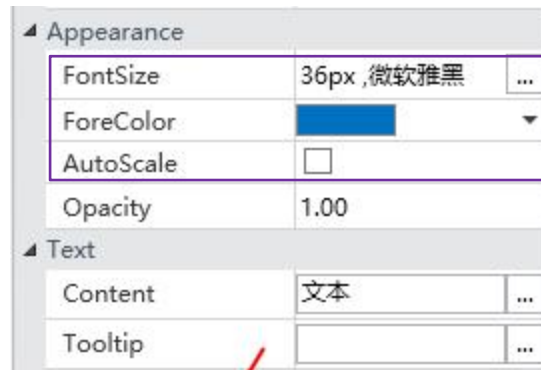
Встановіть властивості для тексту

➤ Встановити властивості тексту



文本

Розмір , колір і масштаб

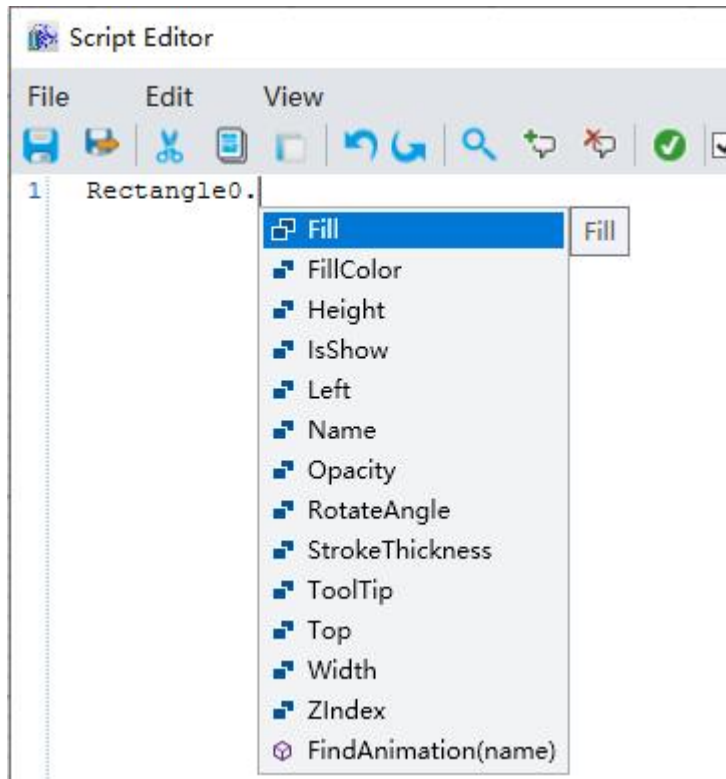


文本

Розмір , колір і масштаб

Сценарій базової графіки

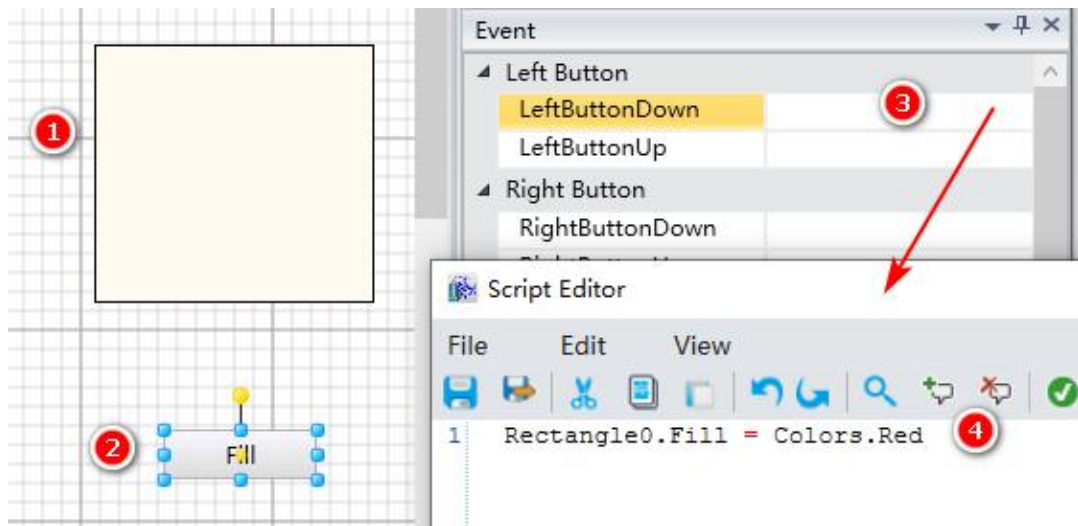
- Сценарій можна використовувати для встановлення властивостей елемента керування



Сценарій базової графіки

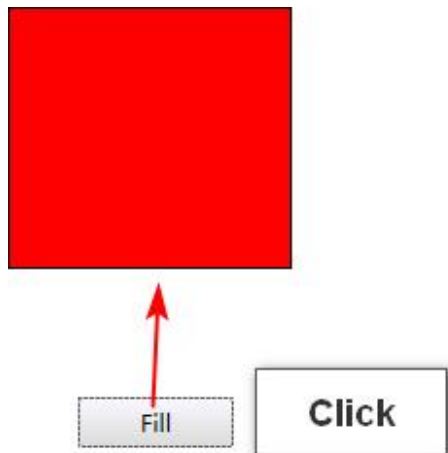
➤ Приклад сценарію заповнення:

(1) Створіть Rectangle0 і Button0 у Window0, налаштуйте LeftButtonDown подія кнопки0



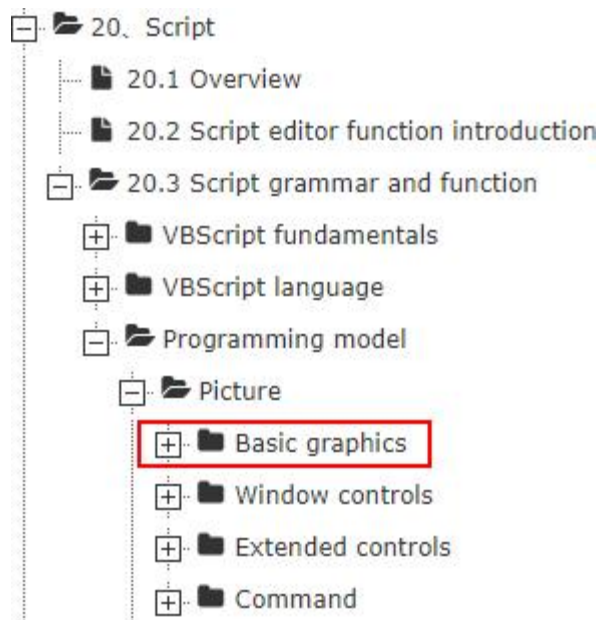
Сценарій базової графіки

(2) Запустіть проект



Сценарії базової графіки

Використання сценаріїв базової графіки див. у розділі « 20.3 Граматика сценаріїв і функції » в посібнику користувача. Як показано на малюнку нижче:



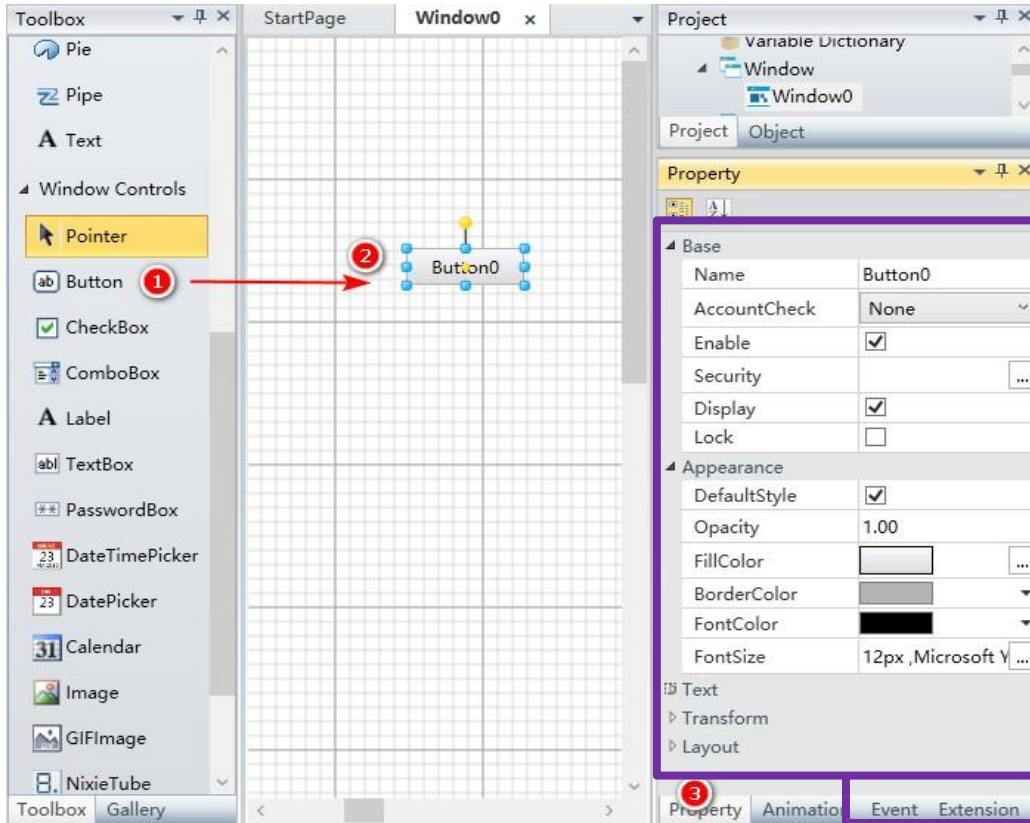
Контур

- Концепції Toolbox
- Як малювати базову графіку
- Як намалювати елементи керування вікном
- Як намалювати Extend Controls



Кнопка керування вікном

➤ Намалюйте Button0, відкрийте вікно його властивостей

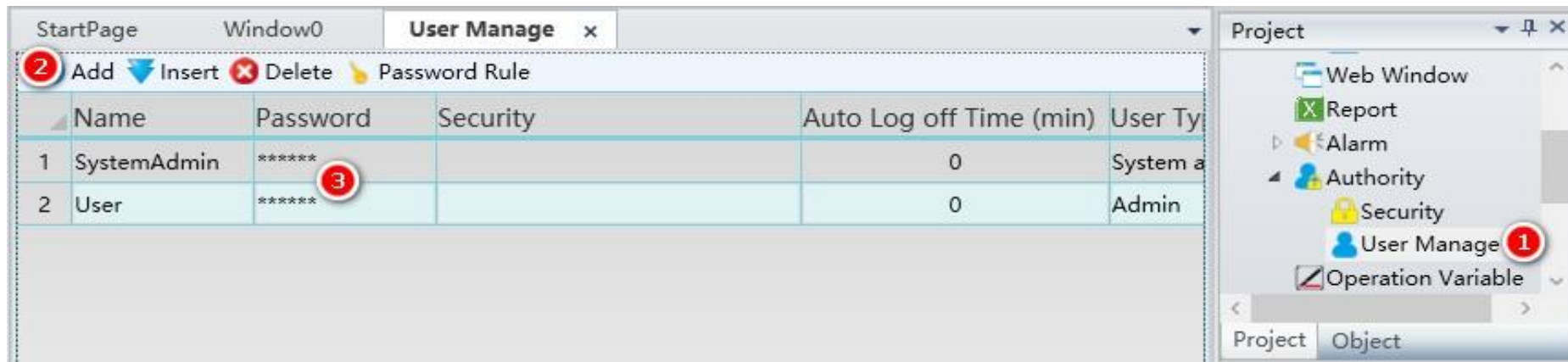


- **AccountCheck** : встановлює властивості підтвердження облікового запису під час виконання;
(1) Немає: пароль облікового запису користувача підтвердження не вимагається;
(2) Загальна перевірка: підтвердження пароля облікового запису потрібне для керування елементом керування в робочому середовищі (3) Подвійна перевірка: у робочому середовищі щоразу, коли ви використовуєте цей елемент керування, вам потрібно підтверджувати обліковий запис і пароль
- **Увімкнути** : встановлює доступність кнопки. Не позначено означає, що кнопка недоступна.
- **DefaultStyle** : встановить, чи використовувати стиль кнопки за замовчуванням

Властивість AccountCheck кнопки

➤ Приклад властивості AccountCheck:

(1) Створити користувача



The screenshot displays the 'User Manage' window. At the top, there are tabs for 'StartPage', 'Window0', and 'User Manage'. Below the tabs, there are buttons for 'Add', 'Insert', 'Delete', and 'Password Rule'. A table lists users with columns for Name, Password, Security, Auto Log off Time (min), and User Type. The table contains two rows: 'SystemAdmin' and 'User'. The 'User' row is highlighted. On the right side, there is a 'Project' tree view showing a hierarchy of objects: 'Web Window', 'Report', 'Alarm', 'Authority', 'Security', 'User Manage', and 'Operation Variable'. The 'User Manage' object is selected and highlighted with a red circle labeled '1'. A red circle labeled '2' is on the 'Add' button, and a red circle labeled '3' is on the password field of the 'User' row.

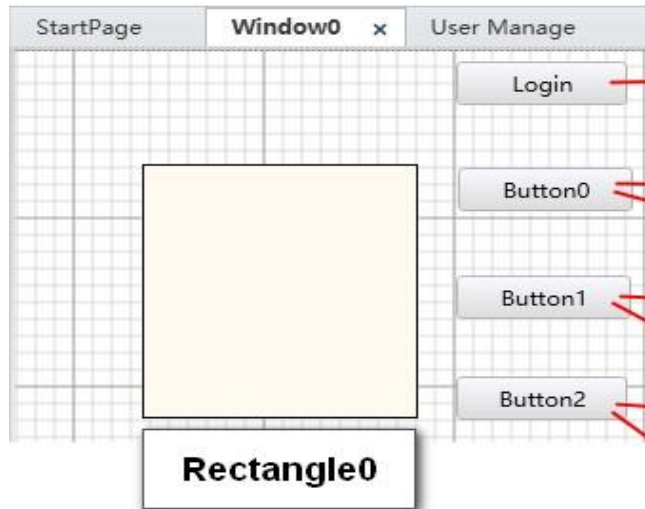
	Name	Password	Security	Auto Log off Time (min)	User Ty
1	SystemAdmin	*****		0	System a
2	User	*****		0	Admin

Project Tree:

- Web Window
- Report
- Alarm
- Authority
- Security
- User Manage
- Operation Variable

Властивість AccountCheck кнопки

(2) Створіть чотири кнопки у Window0, налаштуйте властивість AccountCheck і подія LeftButtonDown чотирьох кнопок

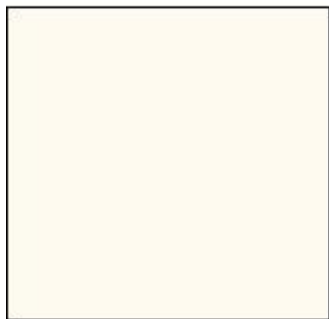


The event of LeftButtonDown

Button Name	AccountCheck Property	LeftButtonDown Event Action
Login	-	UserCmd.LoginBox()
Button0	None	Rectangle0.Fill = Colors.Red
Button1	GeneralCheck	Rectangle0.Fill = Colors.Green
Button2	DoubleCheck	Rectangle0.Fill = Colors.Blue

Властивість AccountCheck кнопки

(3) Запустіть проект



Button0

Button1

Button2

Button3

Властивість AccountCheck кнопки

(4) Перевірка облікового запису : немає

① Натисніть « Button0 », прямокутник0 стане червоним

(5) Перевірка облікового запису: Загальна перевірка

① Натисніть « Button1 » , спливаюче вікно з підказкою: спочатку увійдіть! Як показано на малюнку 1,

② Системний адміністратор або вхід користувача, потім натисніть « Button1 », прямокутник0 стає зеленим

(6) AccountCheck: подвійна перевірка

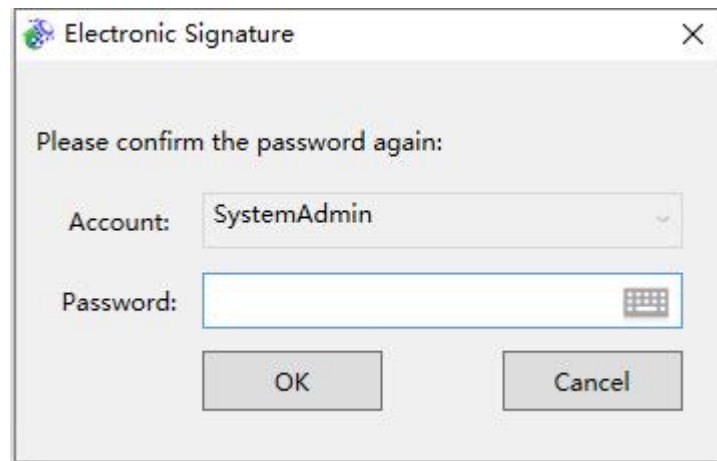
① Натисніть « Button2 » , спливаюче вікно з підказкою: спочатку увійдіть! Як показано на малюнку 1,

② Системний адміністратор або вхід користувача, потім натисніть « Кнопка 2 », спливаюче вікно з підказкою: повторіть введення пароля. Як показано на малюнку 2,

③ Введіть пароль ще раз і натисніть ОК, після чого прямокутник0 стане синім



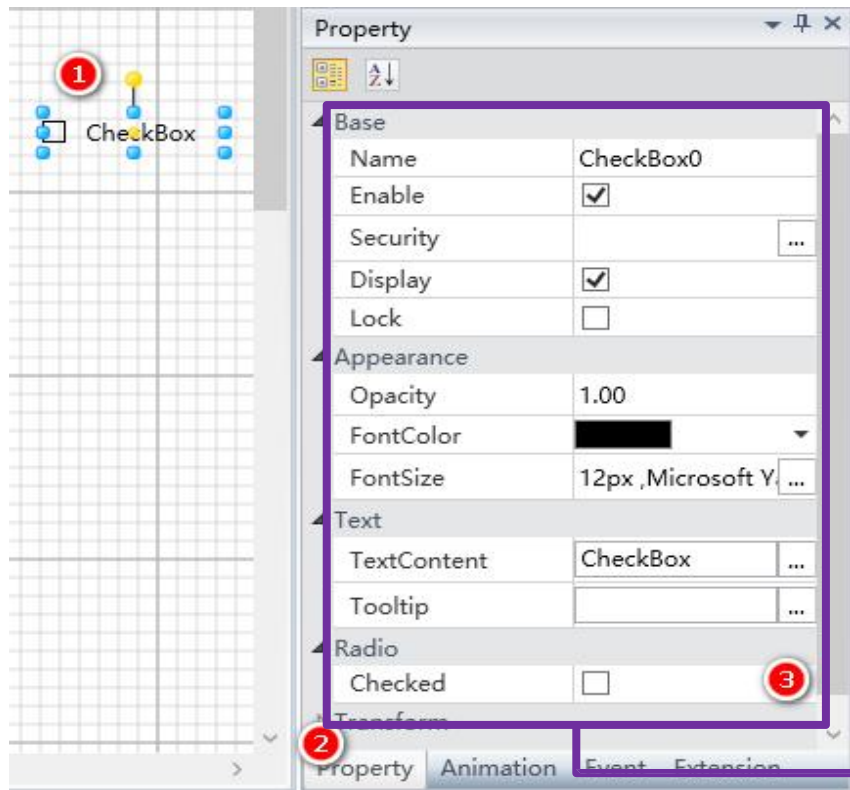
Фігура 1



малюнок 2

Елементи керування вікном - прапорець

- Намалуйте CheckBox0, відкрийте вікно його властивостей

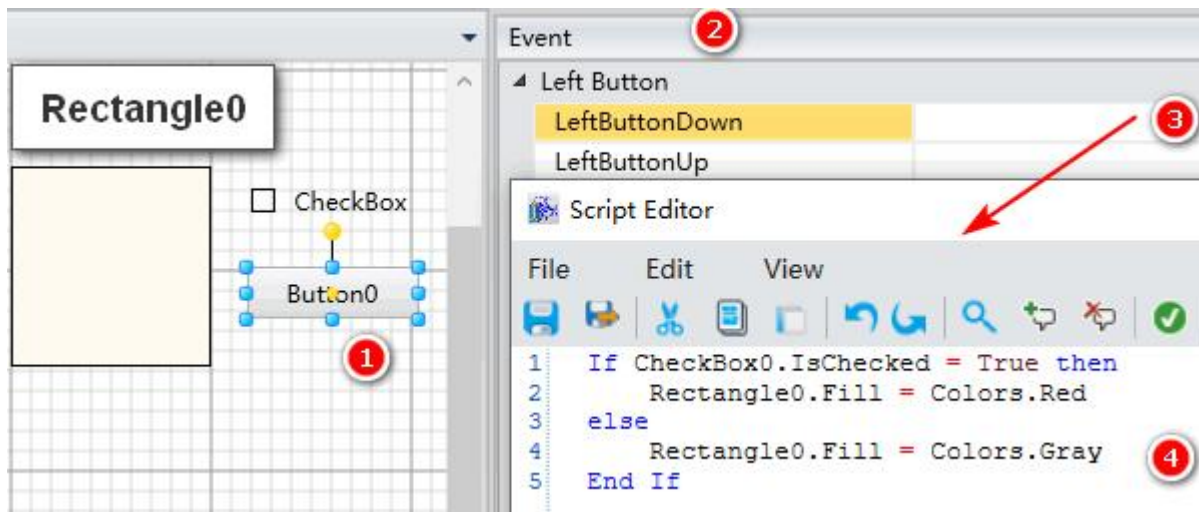


- **Розмір шрифту:** установіть стиль шрифту тексту.
- **ForeColor:** встановити колір тексту.
- **Позначено:** установіть, чи позначено прапорець

Елементи керування вікном - прапорець

➤ Перевірений приклад власності

(1) Створіть Checkbox0, Rectangle0, Button0 у Window0, налаштуйте подію LeftButtonDown для Button0



The screenshot displays a software development environment with two main panels. On the left is a design canvas with a grid background. It contains a yellow rectangle labeled 'Rectangle0', a checkbox labeled 'CheckBox', and a button labeled 'Button0'. A red circle with the number '1' is placed over the 'Button0' label. On the right is a panel with two sections. The top section, titled 'Event' (marked with a red circle '2'), shows a list of events for 'Left Button'. The 'LeftButtonDown' event is selected and highlighted in yellow (marked with a red circle '3'). The bottom section is a 'Script Editor' with a menu bar (File, Edit, View) and a toolbar. It contains a script with the following code:

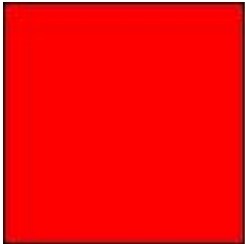
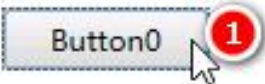
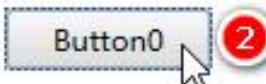
```

1  If CheckBox0.IsChecked = True then
2      Rectangle0.Fill = Colors.Red
3  else
4      Rectangle0.Fill = Colors.Gray
5  End If
  
```

A red arrow points from the 'LeftButtonDown' event in the top section to the script editor. A red circle with the number '4' is placed over the 'End If' line of the script.

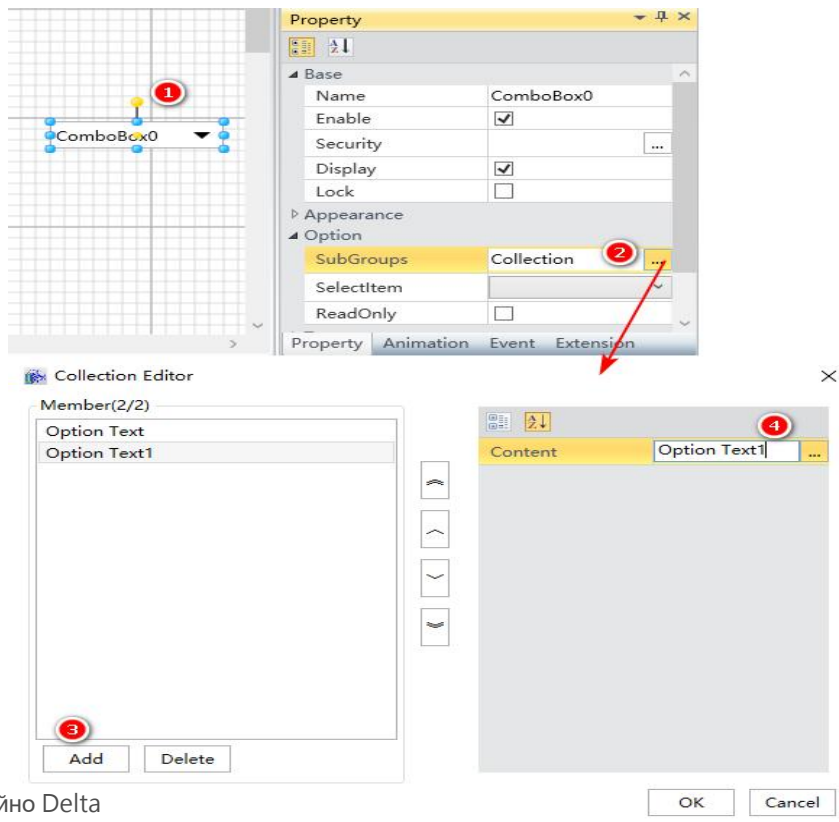
Керуюча подія CheckBox

(2) Запустіть проект

 ☒ CheckBox 	 ☐ CheckBox 
Червоний	Сірий

- ① Коли прапорець 0 позначено, прямокутник 0 стає червоним
- ② Якщо прапорець 0 знято, прямокутник 0 стає сірим

- Використання властивості SubGroups для додавання або видалення параметрів ComboBox



The screenshot illustrates the process of configuring a ComboBox element in the Delta software. The main window shows a grid with a ComboBox0 element. A red circle '1' highlights the element. A red circle '2' highlights the 'SubGroups' property in the 'Option' section of the 'Property' window. A red arrow points from the 'SubGroups' property to the 'Collection Editor' dialog. The 'Collection Editor' dialog shows a list of members with 'Option Text1' selected. A red circle '3' highlights the 'Add' button. A red circle '4' highlights the 'Content' property in the 'Option' section of the 'Property' window.

Елементи керування вікном - ComboBox

- Використання сценаріїв для додавання або видалення параметрів ComboBox

```
ComboBox0.AddItem( "AA" )
```

//Додати опцію AA

```
ComboBox0.AddItems( "BB,CC,DD" )
```

//Додайте 3 варіанти: BB, CC, DD

```
ComboBox0.ClearItems()
```

//Очищає всі параметри

```
ComboBox0.RemoveAtItem( 1 )
```

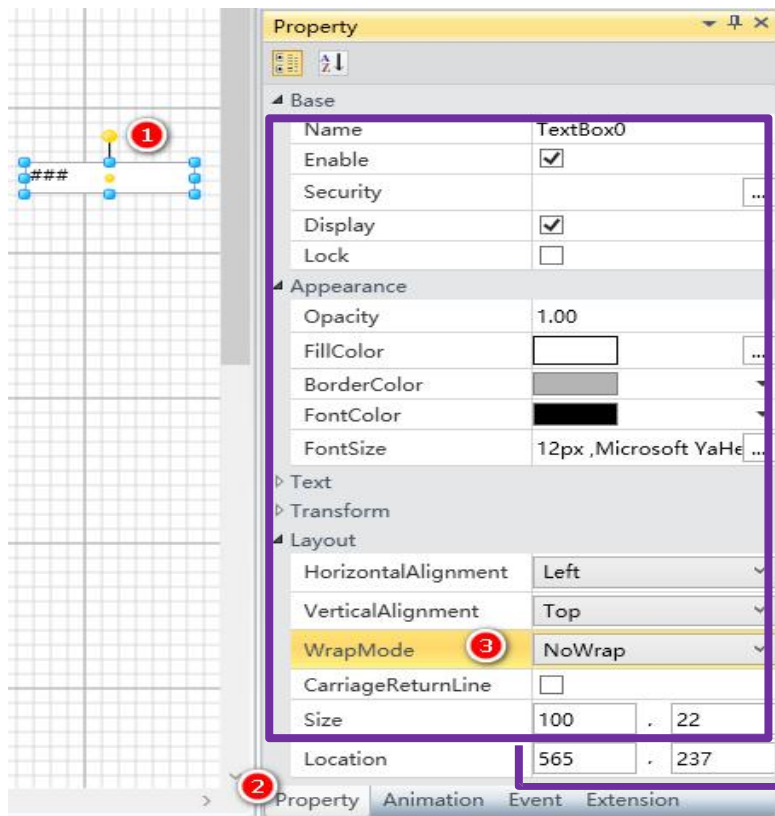
//Видалити опцію з індексом 1

```
ComboBox0.RemoveItem( "AA" )
```

//Видалити параметр AA

Елементи керування вікном - текстове поле

- Намалюйте TextBox0, відкрийте вікно його властивостей

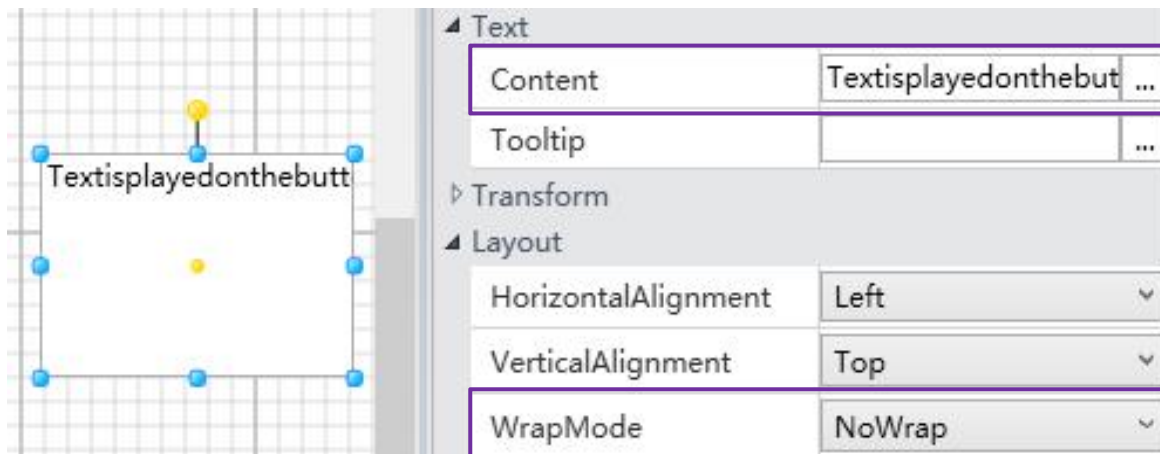


- **CarriageReturnLine** : натисніть клавішу enter для нового рядка
- **Горизонтальне вирівнювання**: ліворуч, по центру, праворуч
- **Вертикальне вирівнювання**: зверху, по центру, знизу
- **WrapMode**: NoWrap , WrapWithOverflow, Wrap

➤ WrapMode приклад 1 :

Введіть літери в текстовому полі 0 : текст, який відображається на кнопці, підтримує багатомовність

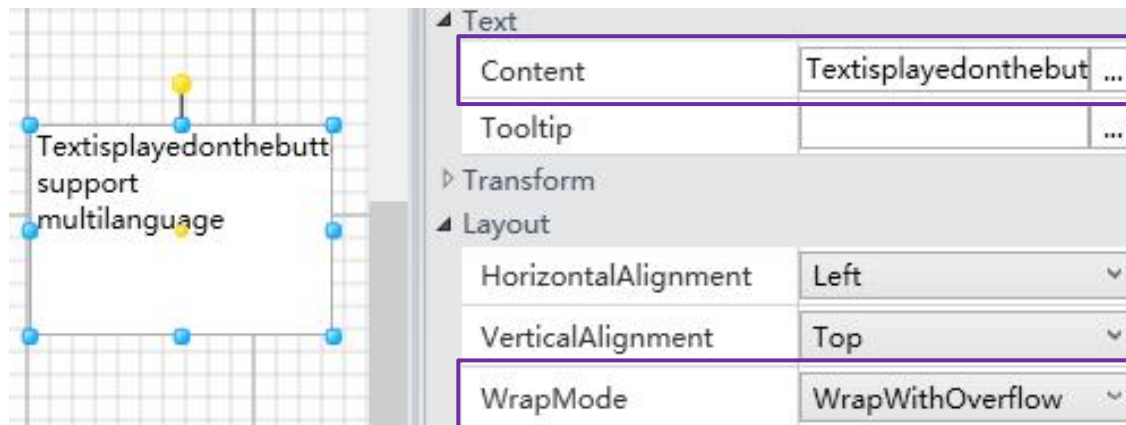
① Властивість WrapMode: NoWrap



➤ WrapMode приклад 2 :

Введіть літери в текстовому полі 0 : текст, який відображається на кнопці, підтримує багатомовність

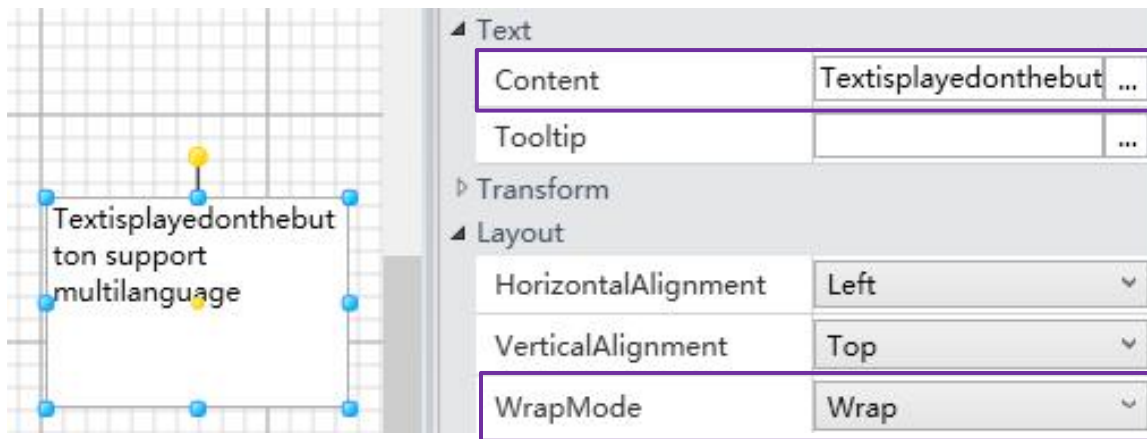
② Властивість WrapMode: WrapWithOverflow



➤ WrapMode приклад 3 :

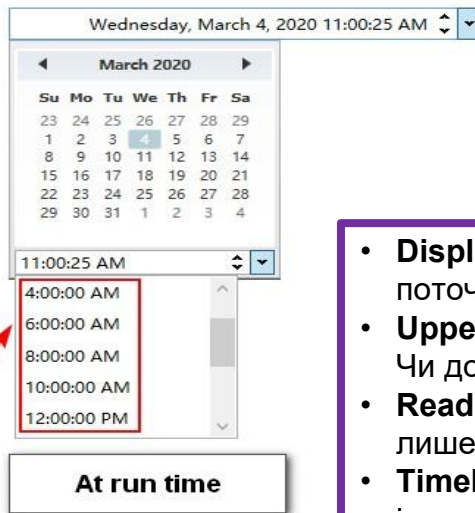
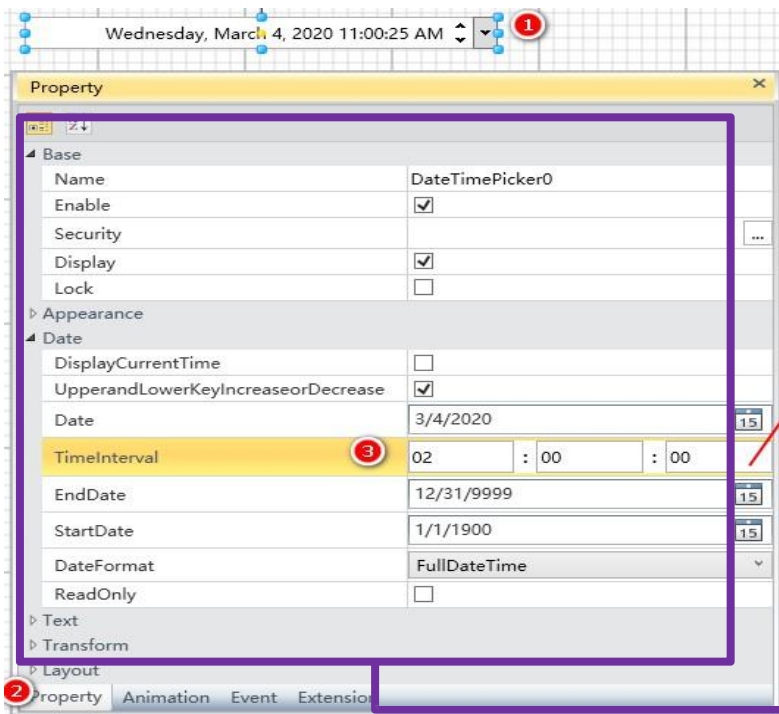
Введіть літери в текстовому полі 0 : текст, який відображається на кнопці, підтримує багатомовність

③ Властивість WrapMode: Wrap



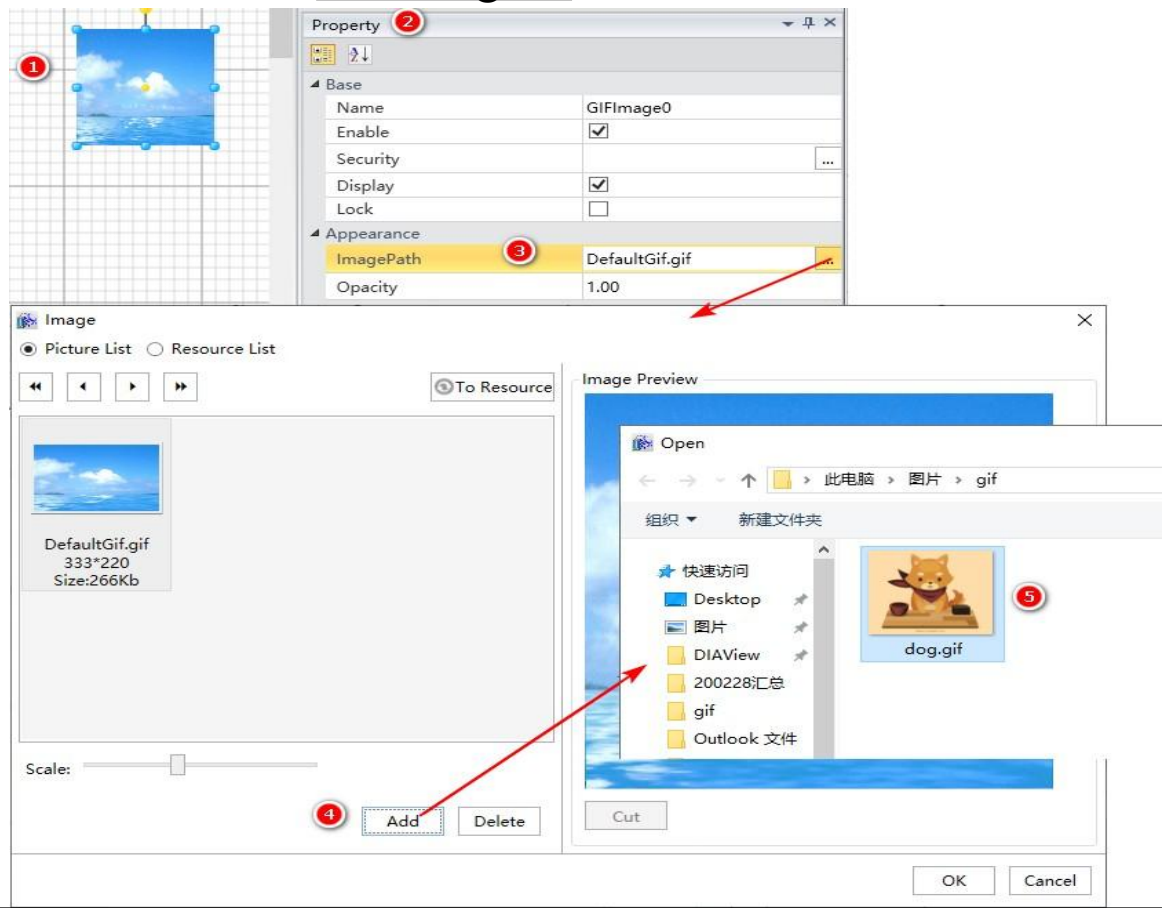
Елементи керування вікном-DateTimePicker

➤ Намалюйте DateTimePicker0, відкрийте вікно його властивостей



- **DisplayCurrentTime** : чи відображати поточний час
- **UpperandLowerKeyIncreaseorDecrease** : Чи доступні клавіші вгору та низький рівень
- **ReadOnly** : значення можна прочитати лише під час виконання
- **TimeInterval** : Встановить часовий інтервал, як показано на малюнку.

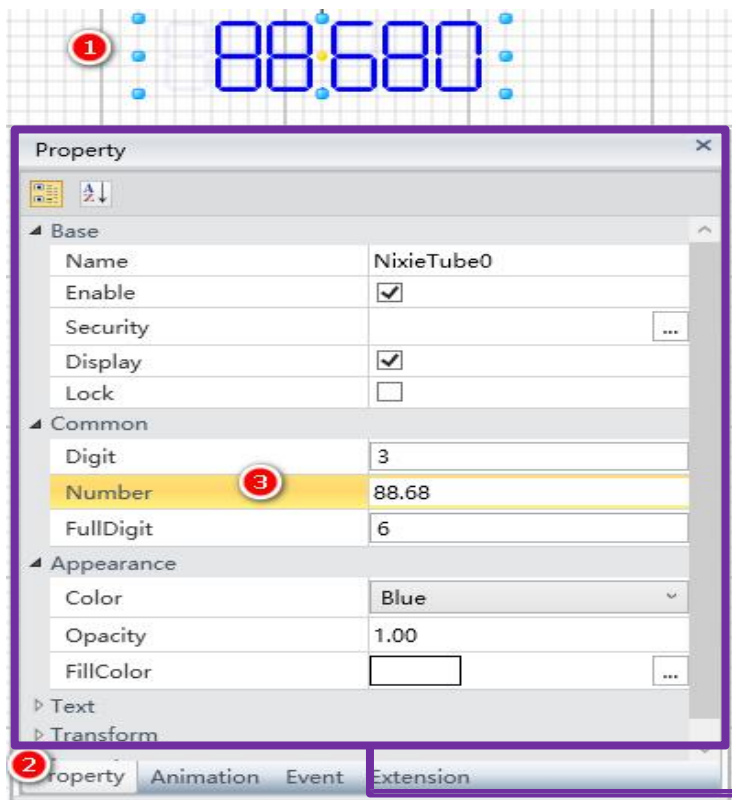
- Намалюйте GIFImage0, відкрийте вікно його



※ Для отримання додаткової інформації зверніться до розділу «7.6.11 GIFImage» у посібнику користувача.

Елементи керування вікном - NixieTube

- Намалюйте NixieTube0, відкрийте вікно його властивостей

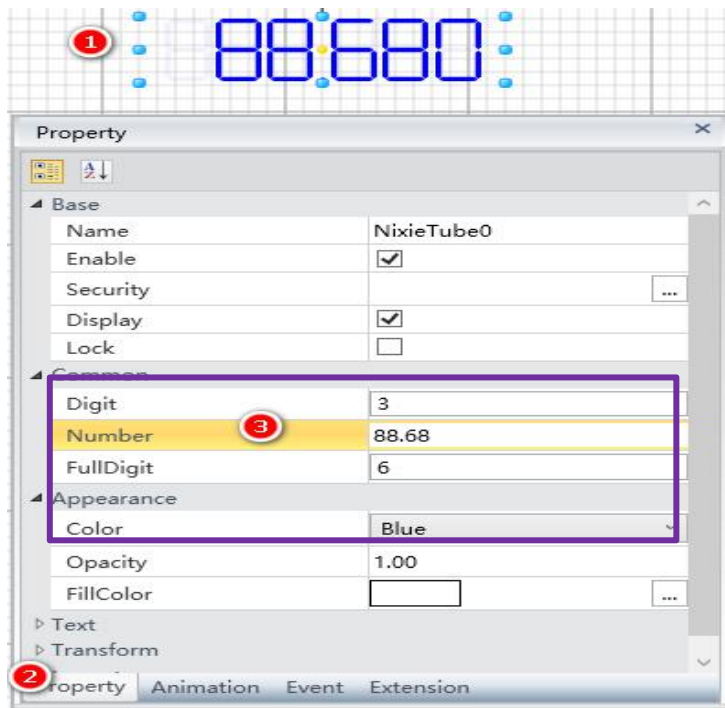


- **Цифра:** кількість цифр у цілій частині
- **FullDigit:** загальна кількість цифр у NixieTube
- **Число:** число для відображення
- **Колір:** установіть колір шрифту NixieTube
- **FillColor:** установіть колір фону NixieTube

Елементи керування вікном - NixieTube

➤ Значення відображення NixieTube

(1) Створіть NixieTube0 у Window0, установіть його властивості



Елементи керування вікном - NixieTube

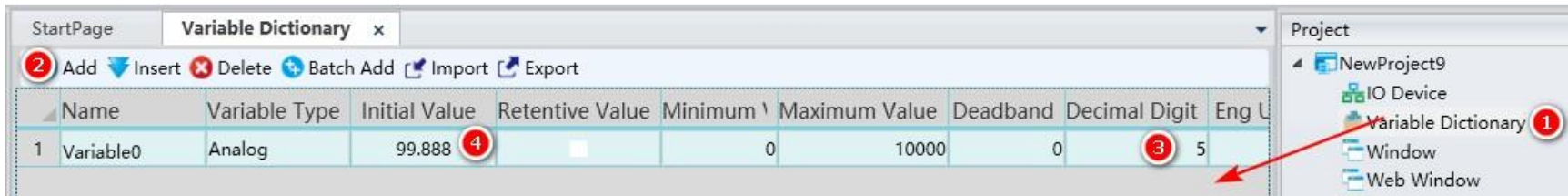
(2) Запустіть проект

A digital display in a NixieTube style, showing the number '88.680' in blue. The digits are composed of horizontal and vertical segments, with a decimal point between the second and third digits.

Елементи керування вікном - NixieTube

➤ Змінна відображення NixieTube

(1) Створіть змінну0, встановіть її початкове значення 99,888



StartPage Variable Dictionary x

2 Add Insert Delete Batch Add Import Export

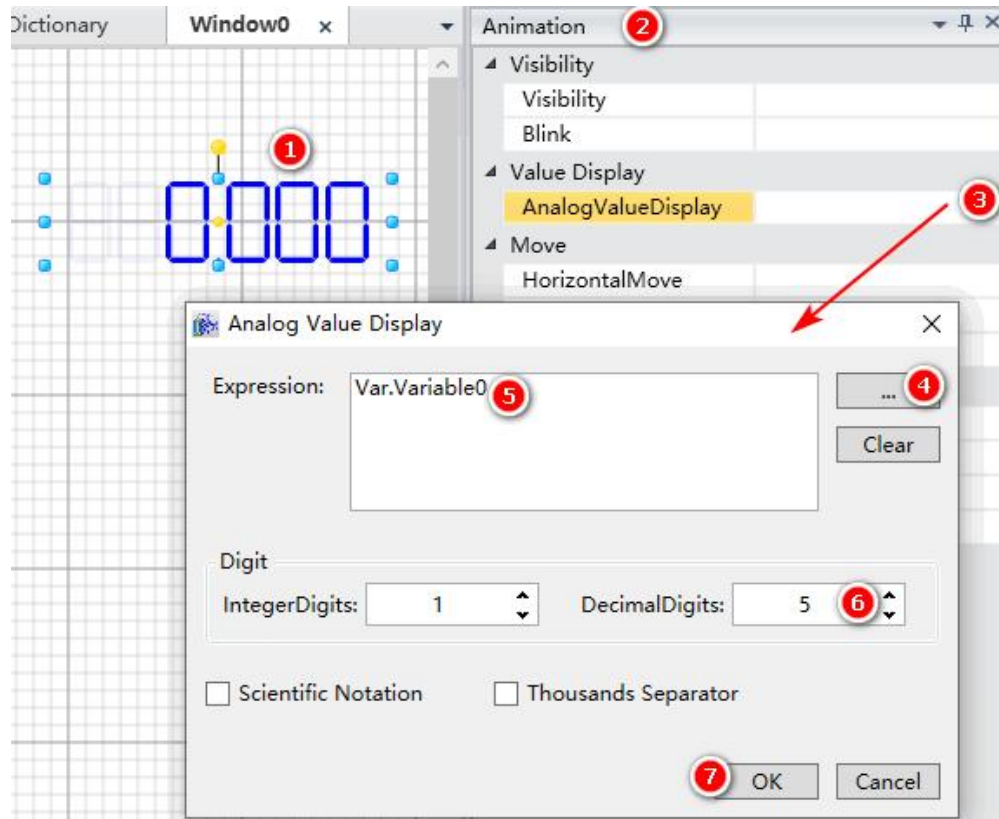
	Name	Variable Type	Initial Value	Retentive Value	Minimum	Maximum Value	Deadband	Decimal Digit	Eng U
1	Variable0	Analog	99.888 4		0	10000	0	3 5	

Project

- NewProject9
 - IO Device
 - Variable Dictionary 1
 - Window
 - Web Window

Елементи керування вікном - NixieTube

(2) Створіть NixieTube0 у вікні Window0, дисплеї аналогових значень пов'язана анімація Variable0



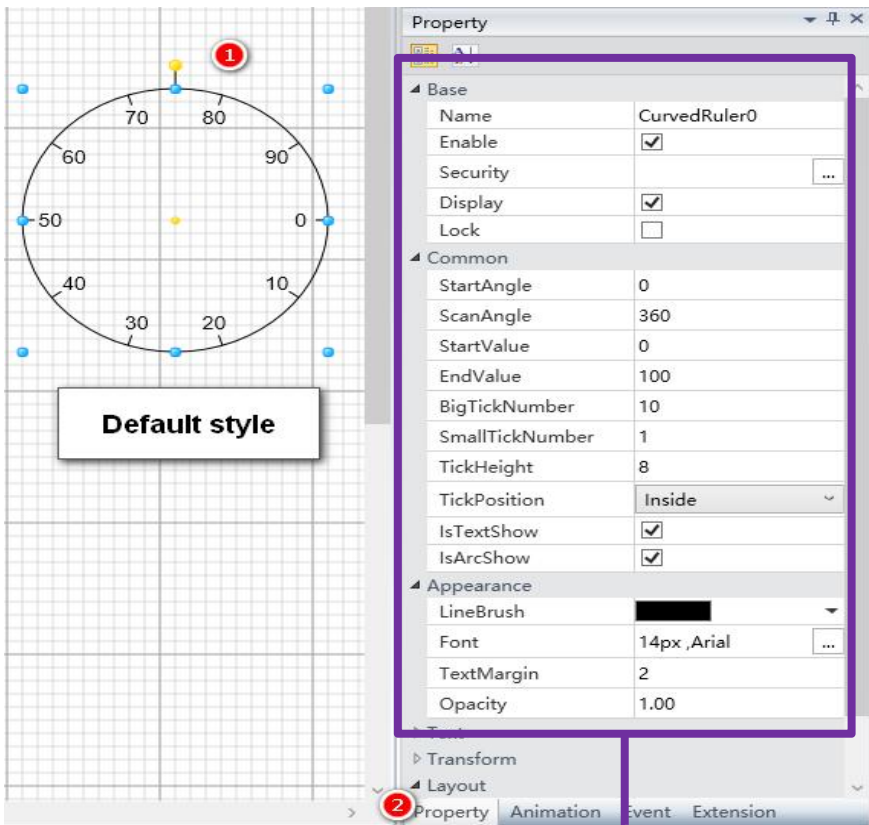
Елементи керування вікном - NixieTube

(3) Запустіть проект

A digital display in a NixieTube style, showing the number '99.888' in blue. The first two digits are slightly faded, while the rest are bright blue.

Елементи керування вікном - вигнута лінійка

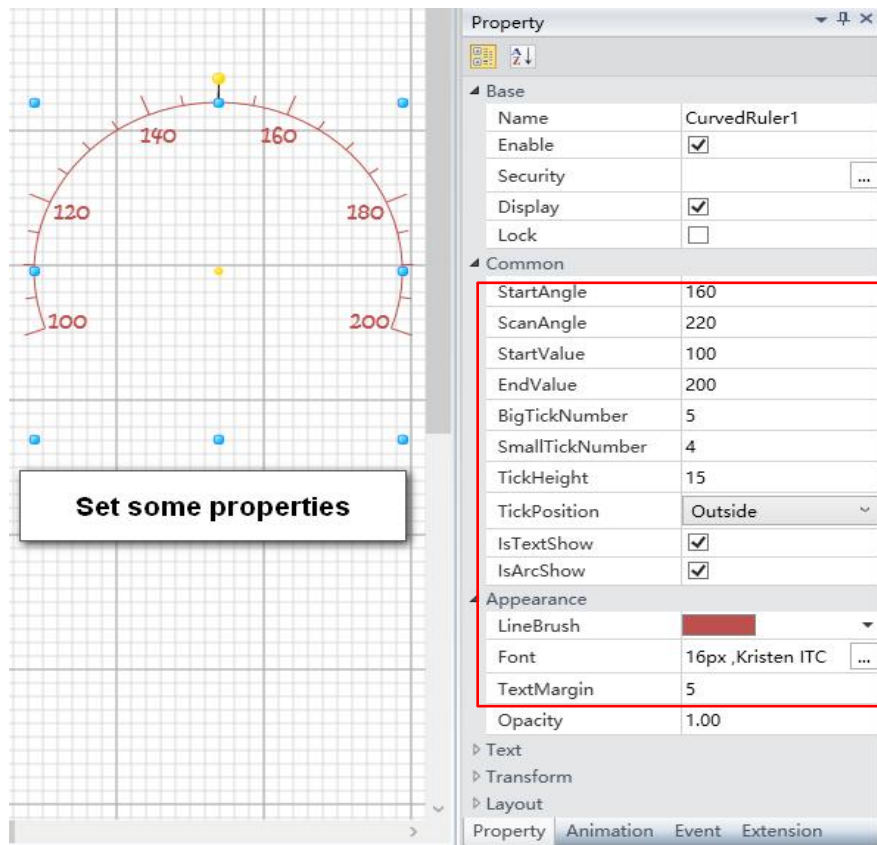
➤ Намалюйте CurvedRuler0, відкрийте вікно його властивостей



- **StartAngle:** установіть початковий кут CurvedRuler (одиниця: градуси).
- **ScanAngle:** установіть кут сканування CurvedRuler (одиниця: ступенів).
- **StartValue:** установіть початкове значення CurvedRuler.
- **EndValue:** установіть кінцеве значення CurvedRuler.
- **BigTickNumber:** Установіть кількість великих позначок CurvedRuler.
- **SmallTickNumber:** Установіть кількість маленьких позначок CurvedRuler.
- **TickHeight:** установіть висоту позначки вигнутої лінійки.
- **TickPosition:** Установіть позицію позначки CurvedRuler.
- **IsTextShow:** установіть, чи відображатиметься текст CurvedRuler
- **IsArcShow:** встановіть, чи відображатиметься лінія кордону CurvedRuler
- **LineBrush:** установіть поточний колір CurvedRuler.
- **Шрифт:** установіть формат шрифту для тексту, що відображається на Вигнута лінійка.
- **TextMargin:** установіть відстань між текстом і лініями рамки

Елементи керування вікном - вигнута лінійка

- Намалюйте CurvedRuler0, установіть його властивості



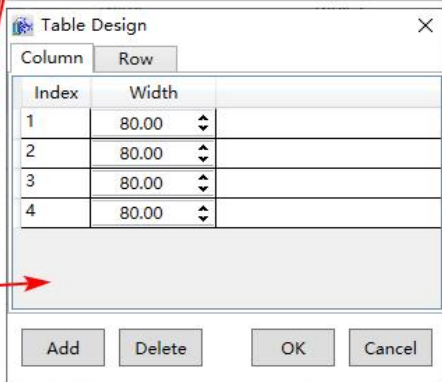
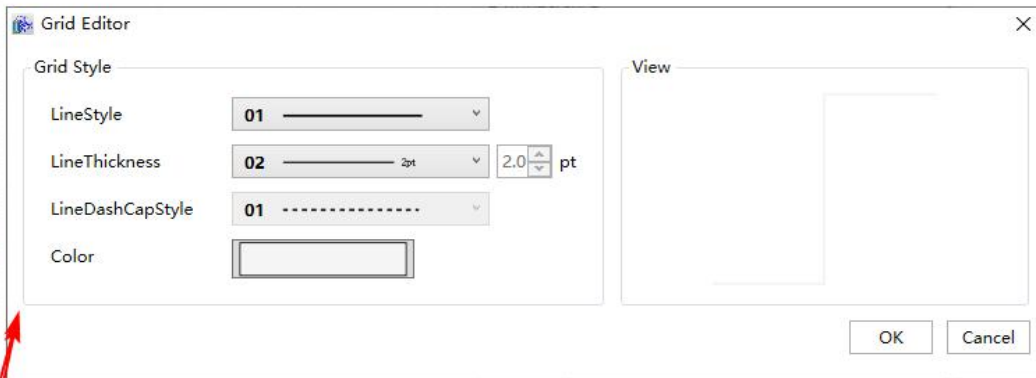
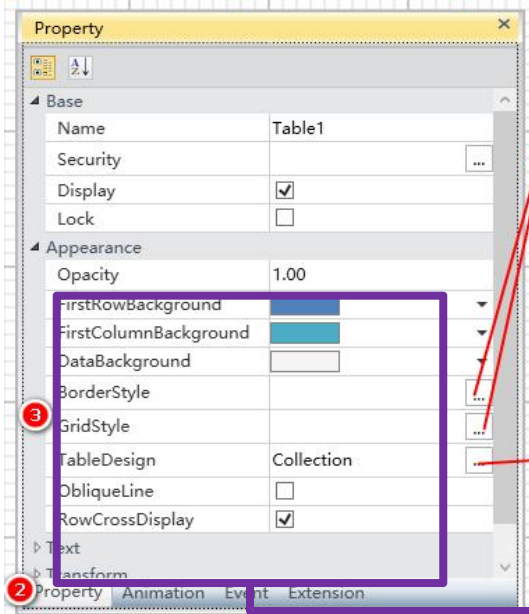
The image shows a screenshot of the Delta software interface. On the left, a curved ruler is drawn on a grid, with tick marks and labels at 100, 120, 140, 160, 180, and 200. A yellow dot is placed on the ruler. A white box with the text "Set some properties" is overlaid on the grid. On the right, the "Property" window is open, showing the properties of the selected object, "CurvedRuler1". The properties are organized into sections: Base, Common, Appearance, Text, Transform, and Layout. The "Common" section is highlighted with a red box, showing the following values:

Property	Value
StartAngle	160
ScanAngle	220
StartValue	100
EndValue	200
BigTickNumber	5
SmallTickNumber	4
TickHeight	15
TickPosition	Outside
IsTextShow	☒
IsArcShow	☒

The "Appearance" section shows the "LineBrush" set to a red color, "Font" set to "16px ,Kristen ITC", and "TextMargin" set to "5". The "Opacity" is set to "1.00". The "Text", "Transform", and "Layout" sections are collapsed. The "Property" window has tabs for "Property", "Animation", "Event", and "Extension".

Елементи керування вікном - Таблиця

➤ Намалюйте Table0, відкрийте вікно його властивостей

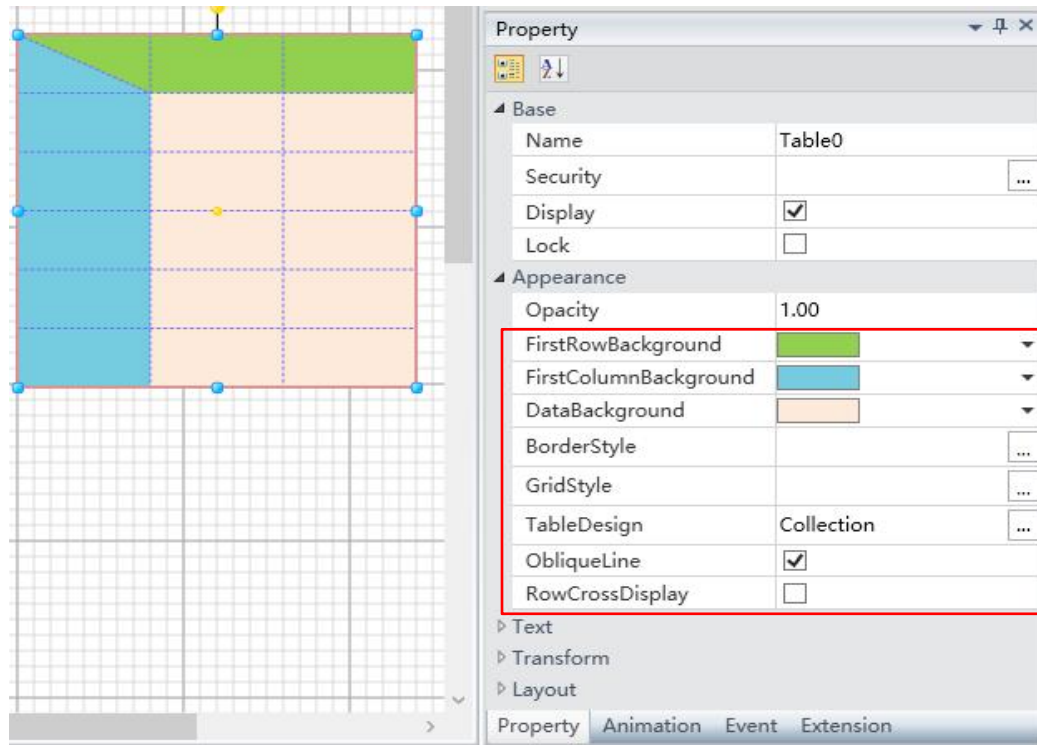


- **FirstRowBackground:** встановити фон першого рядка.
- **FirstColumnBackground:** Установить фон першого стовпця. Це працює, лише якщо позначено властивість *ObliqueLine*.
- **DataBackground:** Установить фон області даних. Це працює, лише якщо властивість *RowCrossDisplay* не позначено.
- **TableDesign:** Установить рядки та стовпці таблиці.
- **ObliqueLine:** установить, чи відображати похилу лінію у верхньому лівому куті.
- **RowCrossDisplay:** увімкніть перехресне відображення рядків.

Елементи керування вікном - Таблиця

➤ Створіть таблицю з 6 рядків і 3 стовпців

(1) Створіть таблицю Table0 у вікні Window0 і встановіть її деякі властивості



The screenshot displays a software interface with a table and its corresponding property window.

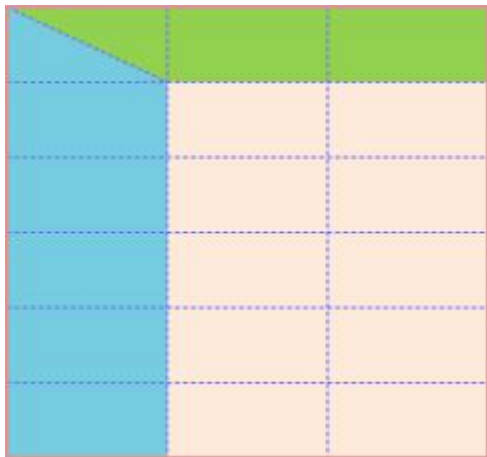
Table Structure:

Property Window (Table0):

- Base:**
 - Name: Table0
 - Security: ...
 - Display: ☒
 - Lock: ☐
- Appearance:**
 - Opacity: 1.00
 - FirstRowBackground: [Green]
 - FirstColumnBackground: [Blue]
 - DataBackground: [Orange]
 - BorderStyle: ...
 - GridStyle: ...
 - TableDesign: Collection
 - ObliqueLine: ☒
 - RowCrossDisplay: ☐
- Text:**
- Transform:**
- Layout:**
- Property:** Animation Event Extension

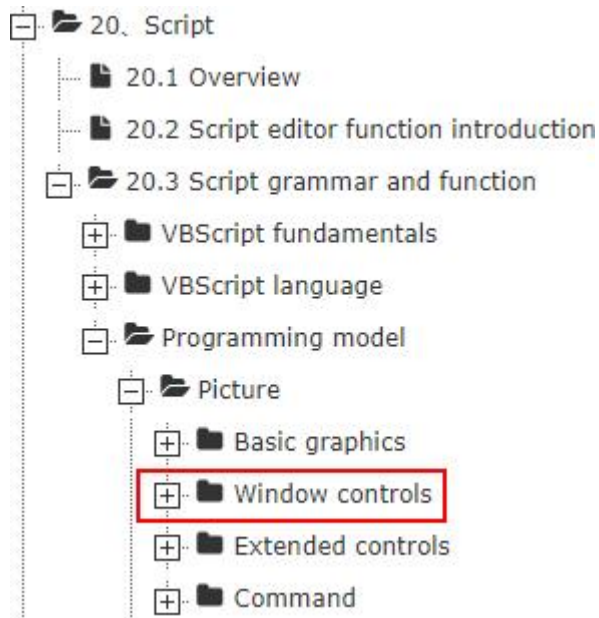
Елементи керування вікном - Таблиця

(2) Запустіть проект



Скрипти елементів керування вікном

Використання сценаріїв віконних елементів керування див. у розділі « 20.3 Сценарій граматика та функції » в посібнику користувача. Як показано на малюнку нижче:

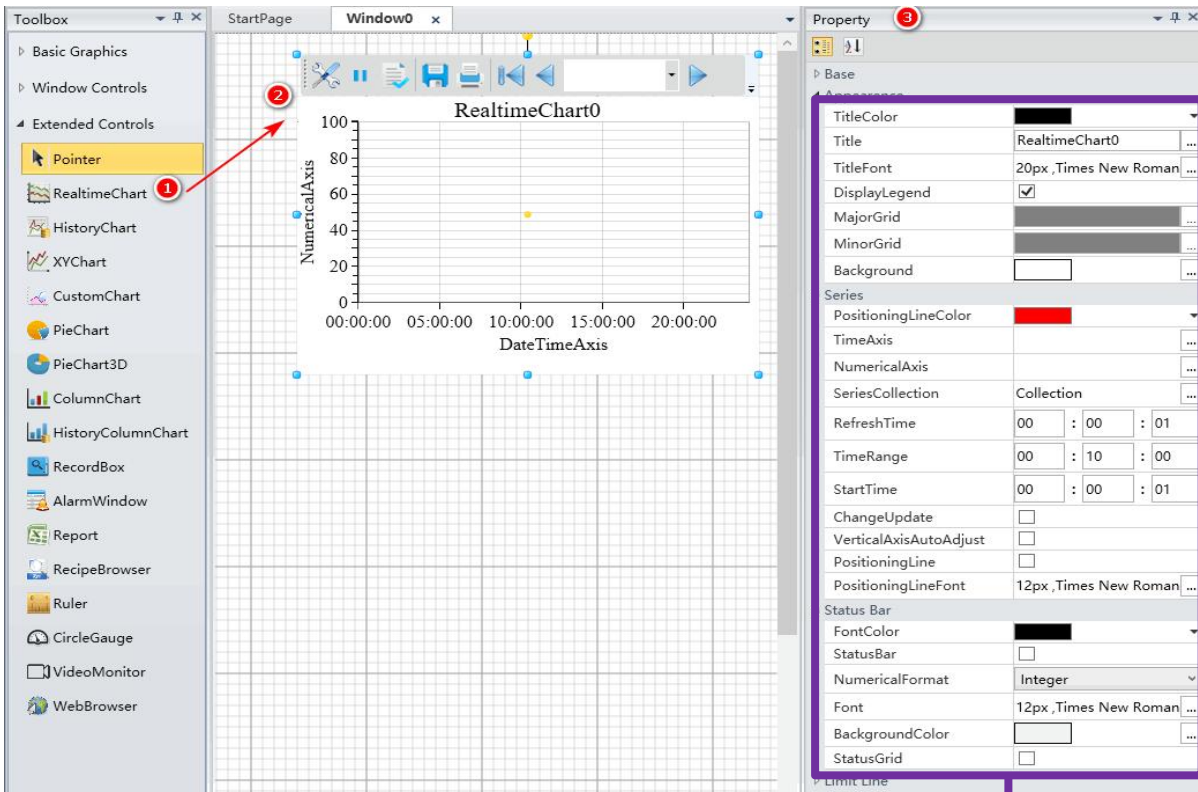


Контур

- Концепції Toolbox
- Як малювати базову графіку
- Як намалювати елементи керування вікном
- Як намалювати Extend Controls

Розширення Controls-RealTimeChart

➤ Створіть RealTimeChart0, відкрийте вікно його властивостей



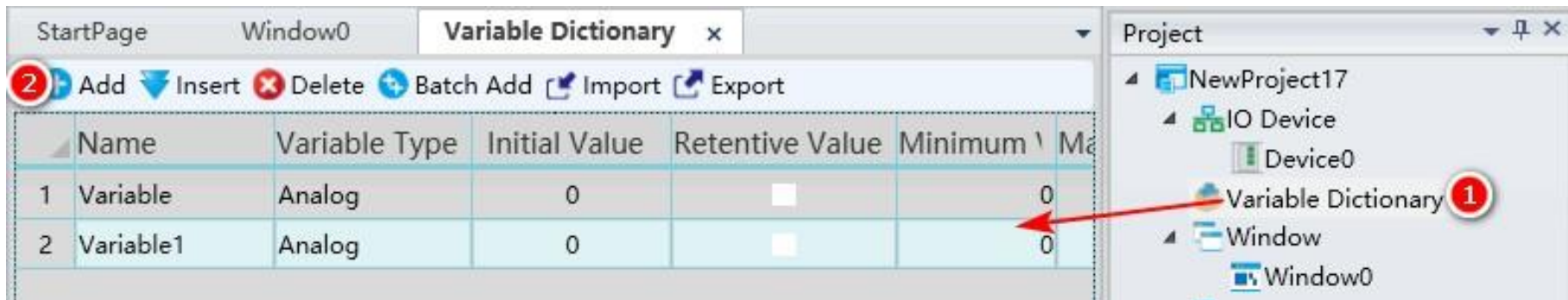
- **TimesAxis** : Встановити властивості TimesAxis
- **NumericalAxis** : встановити властивості NumericalAxis
- **SeriesCollection** : додайте або видаліть серію, установіть властивості серії
- **ChangeUpdate** : встановіть, чи оновлювати дані тільки при зміні значення
- **VerticalAxisAutoChange** : встановіть автоматичне налаштування максимального значення вертикальної осі.
- **PositionLine** : чи відображати позицію лінії
- **PositionLineColor** : Установіть колір позиційної лінії
- **PositionLineFont** : установіть формат шрифту даних на позиційному рядку
- **DisplayLegend**: чи відображати легенду діаграми.
- **MajorGrid**: Установіть стиль основної сітки
- **MinorGrid**: встановити стиль другорядної сітки

※ Для отримання додаткової інформації зверніться до розділу « 7.7.1 RealTimeChart » у посібнику користувача
Властивості HistoryChart · XYChart · CustomChart · ColumnChart · HistoryColumnChart подібні до RealTimeChart, тому пояснення не надаються

Розширення Controls-RealTimeChart

➤ Створіть RealTimeChart0, який оновлює дані в реальному часі:

(1) Створіть 2 змінні: змінна, змінна 1

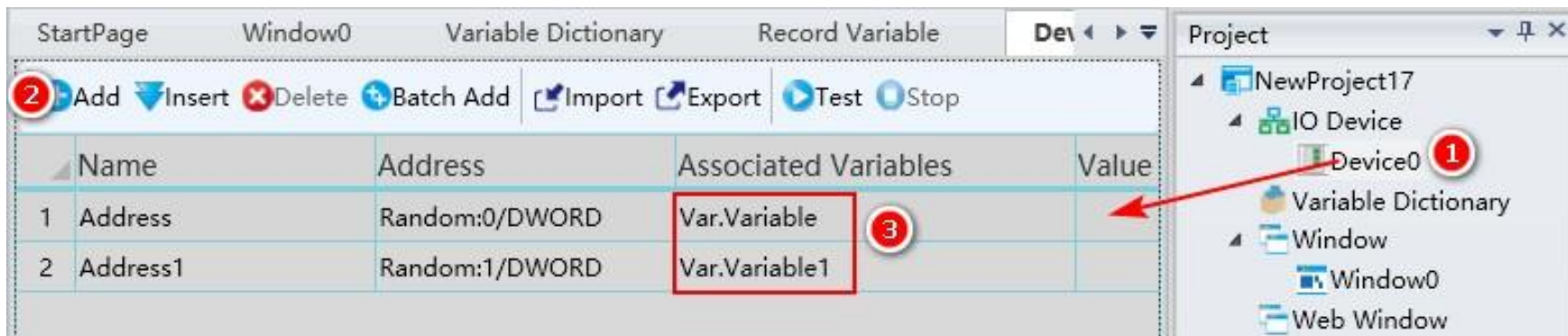


	Name	Variable Type	Initial Value	Retentive Value	Minimum \	Ma
1	Variable	Analog	0	☐	0	
2	Variable1	Analog	0	☐	0	

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

Розширення Controls-RealTimeChart

(2) Створіть пристрій імітації Device0, додайте дві адреси в Device0, які пов'язані з Variable, Variable1 відповідно



The screenshot shows the software interface with the 'Variable Dictionary' tab selected. The 'Project' tree on the right shows a hierarchy: NewProject17 > IO Device > Device0 (marked with a red circle 1). A red arrow points from Device0 to the 'Associated Variables' column of the table below. The table has the following data:

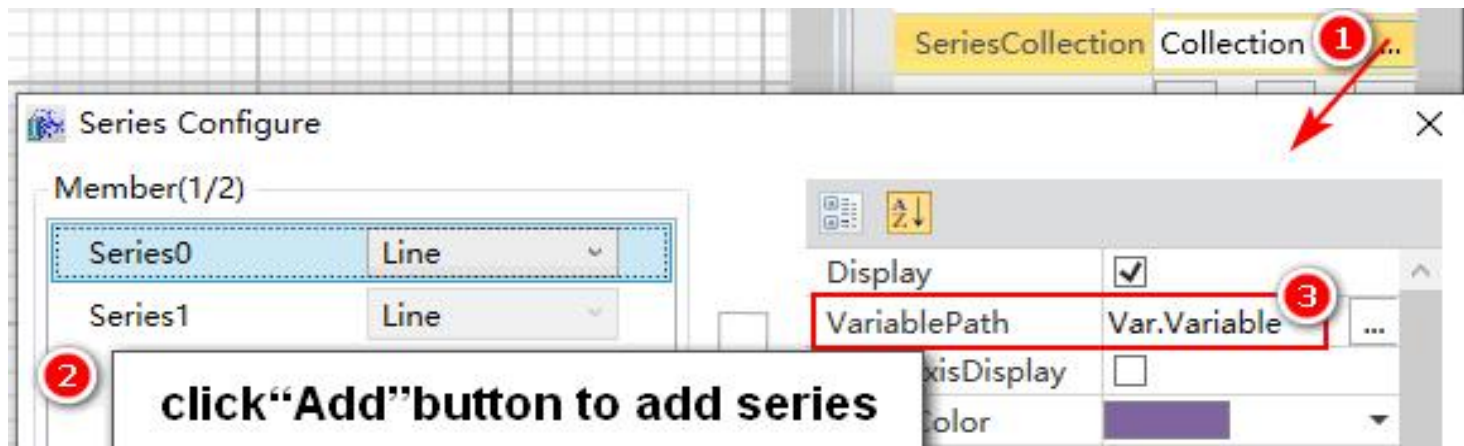
	Name	Address	Associated Variables	Value
1	Address	Random:0/DWORD	Var.Variable (marked with a red circle 3)	
2	Address1	Random:1/DWORD	Var.Variable1	

At the top left of the interface, there are buttons: Add (marked with a red circle 2), Insert, Delete, Batch Add, Import, Export, Test, and Stop.

※ Спосіб створення пристрою моделювання див. у розділі «5.10.1 Симулятор» у посібнику користувача

Розширення Controls-RealTimeChart

(3) Створіть RealtimeChart0 у Window0 , додайте дві криві (Series0 , Series1) у RealtimeChart0 , які пов'язують Variable та Variable1 відповідно



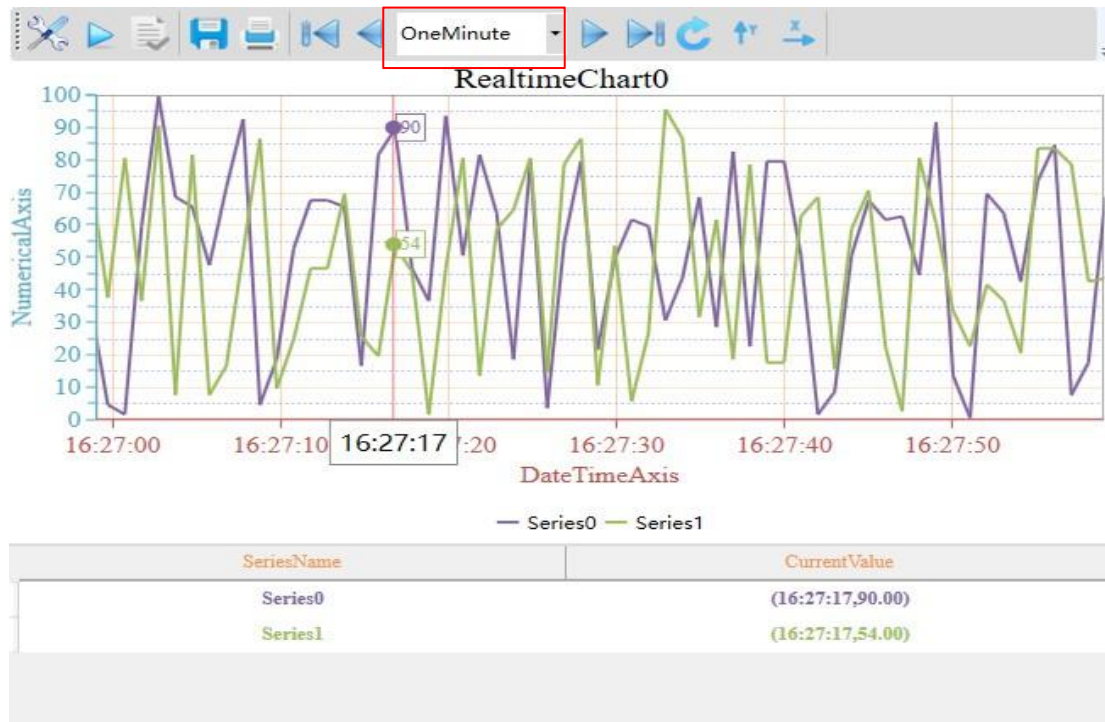
Розширення Controls-RealTimeChart

(4) Встановіть деякі властивості RealtimeChart0

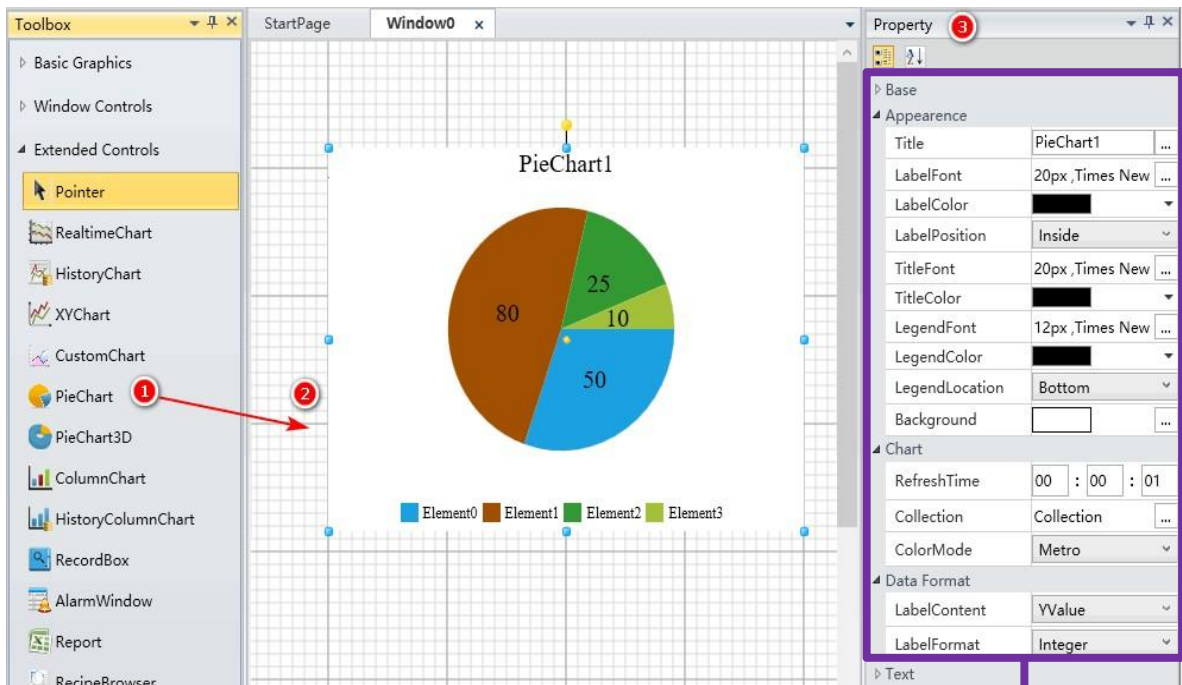
Appearance	
TitleColor	
Title	RealtimeChart0
TitleFont	20px, Times New Roman
DisplayLegend	☒
MajorGrid	
MinorGrid	
Background	
Series	
PositioningLineColor	
TimeAxis	
NumericalAxis	
SeriesCollection	Collection
RefreshTime	00 : 00 : 01
TimeRange	00 : 10 : 00
StartTime	00 : 00 : 01
ChangeUpdate	☐
VerticalAxisAutoAdjust	☐
PositioningLine	☒
PositioningLineFont	12px, Times New Roman
Status Bar	
FontColor	
StatusBar	☒
NumericalFormat	Double2
Font	12px, Times New Roman
BackgroundColor	
StatusGrid	☐

Розширення Controls-RealTimeChart

(5) Запустіть проект



➤ Створіть PieChart0, відкрийте вікно його властивостей



- **Назва** : встановіть назву
- **TitleFont** : установіть стиль шрифту заголовка
- **TitleColor** : установіть колір заголовка
- **LabelFont** : встановити стиль шрифту мітки
- **LabelColor** : встановити колір мітки
- **LabelPosition** : встановити мітку позиція (всередині, назовні, назовні)
- **LegendFont** : встановити шрифт легенди
- **LegendColor** : встановити колір легенди
- **LegendLocation** : встановити положення легенди
- **RefreshTime** : установіть час оновлення
- **ColorMode** : установіть колір режим (Metro, Custom)
- **LabelContent** : установіть вміст мітки (YValue, XValue, Percentage)
- **LabelFormat** : установіть формат мітки (Int, Double1, Double2, Scientific, Perc ent)
- **Колекція** : налаштовує елементи та пов'язані зі змінними секторної діаграми

※ Щоб дізнатися більше, зверніться до розділу « 7.7.5 Секторна діаграма » в посібнику користувача. Властивості Pie3DChart подібні до PieChart, тому пояснення не надаються.

➤ Створіть сектор, який оновлює дані в реальному часі:

(1) Створіть чотири змінні: змінна, змінна1, змінна2, змінна3



StartPage Window0 Variable Dictionary x

2 Add Insert Delete Batch Add Import Export

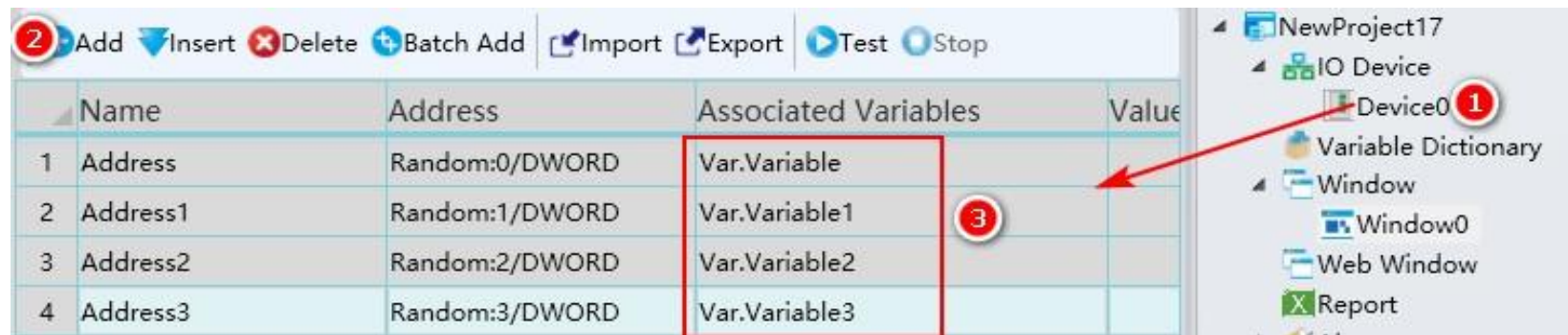
	Name	Variable Type	Initial Value	Retentive Value	Minimum
1	Variable	Analog	0	☐	
2	Variable1	Analog	0	☐	
3	Variable2	Analog	0	☐	
4	Variable3	Analog	0	☐	

Project

- NewProject17
 - IO Device
 - Device0
 - Variable Dictionary 1
 - Window
 - Window0
 - Web Window
 - Report

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

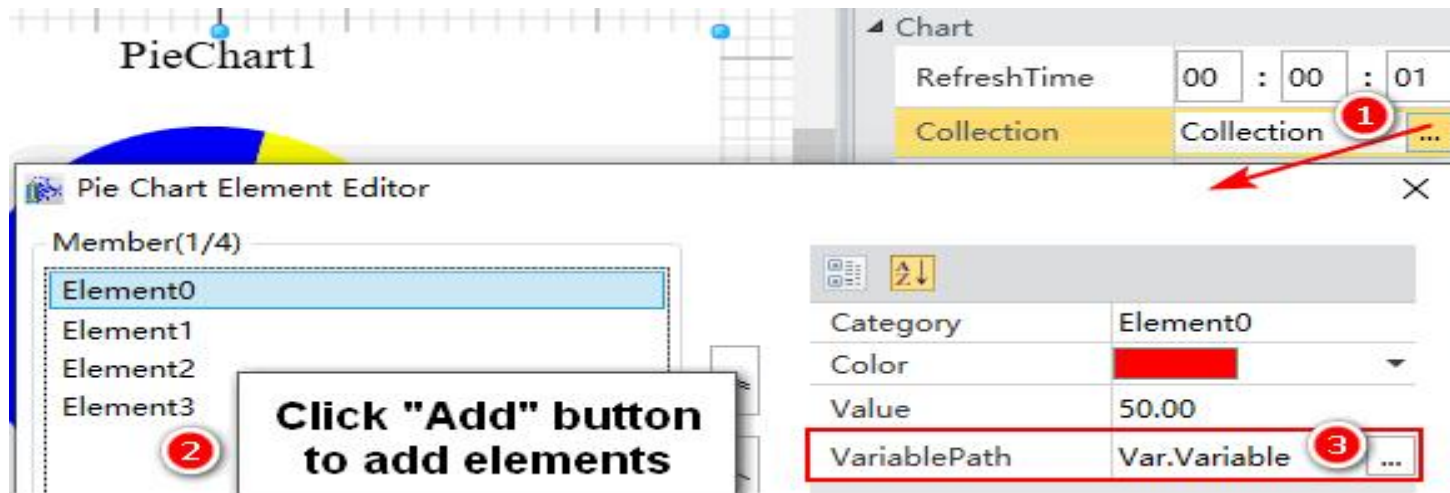
(2) Створіть пристрій імітації Device0, додайте чотири адреси в Device0, які пов'язані з Variable, Variable1, Variable2, Variable3 відповідно



	Name	Address	Associated Variables	Value
1	Address	Random:0/DWORD	Var.Variable	
2	Address1	Random:1/DWORD	Var.Variable1	
3	Address2	Random:2/DWORD	Var.Variable2	
4	Address3	Random:3/DWORD	Var.Variable3	

※ Спосіб створення пристрою моделювання див. у розділі «5.10.1 Симулятор» у посібнику користувача

(3) Створіть PieChart0 у Window0, додайте чотири елементи (Element0, Element1, Element2, Element3) у PieChart0, які пов'язують Variable, Variable1, Variable2, Variable3 відповідно



Pie Chart Element Editor

Member(1/4)

- Element0
- Element1
- Element2
- Element3

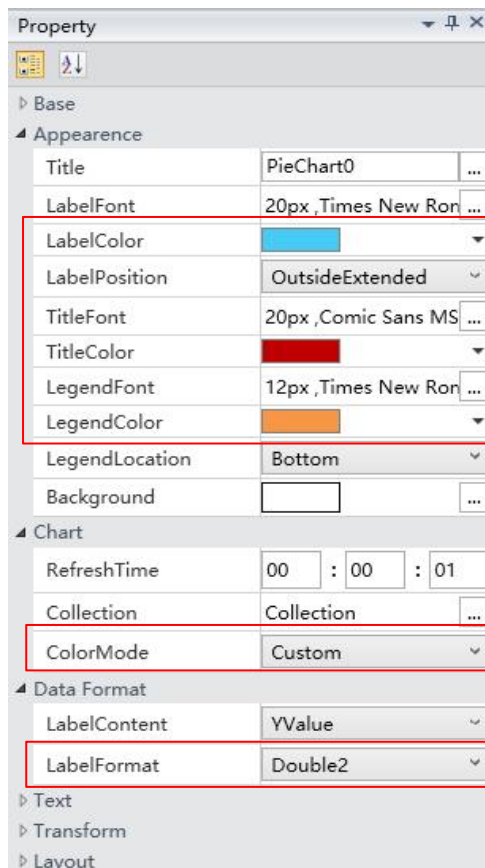
Click "Add" button to add elements

Chart

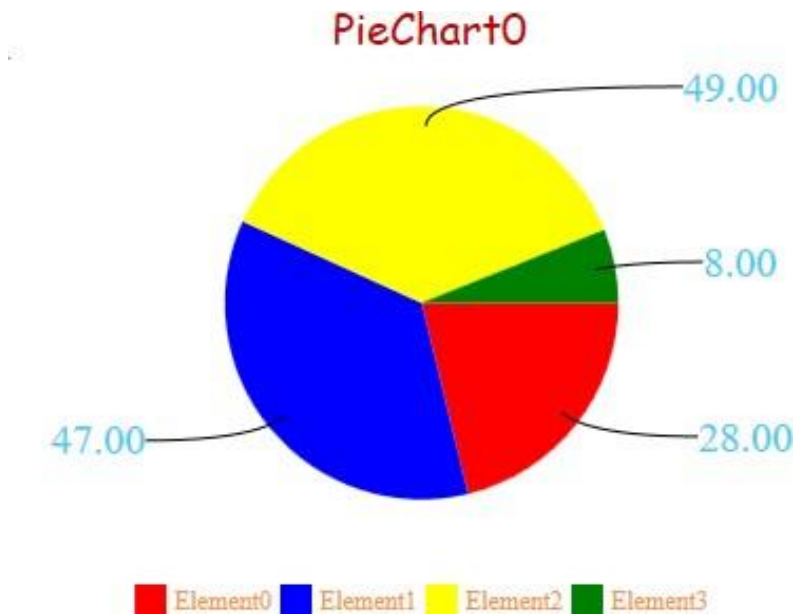
RefreshTime	00 : 00 : 01
Collection	Collection 1 ...

Category	Element0
Color	
Value	50.00
VariablePath	Var.Variable 3 ...

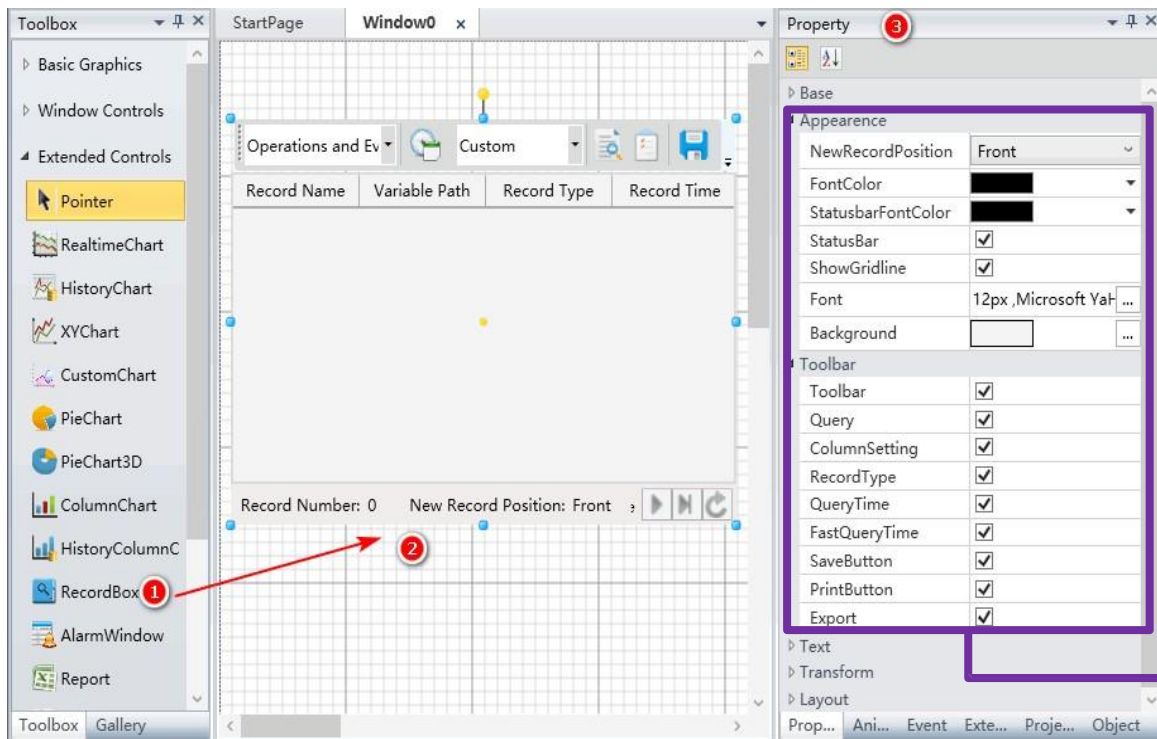
(4) Встановіть деякі властивості



(5) Запустіть проект



➤ Створіть RecordBox0, відкрийте вікно його властивостей

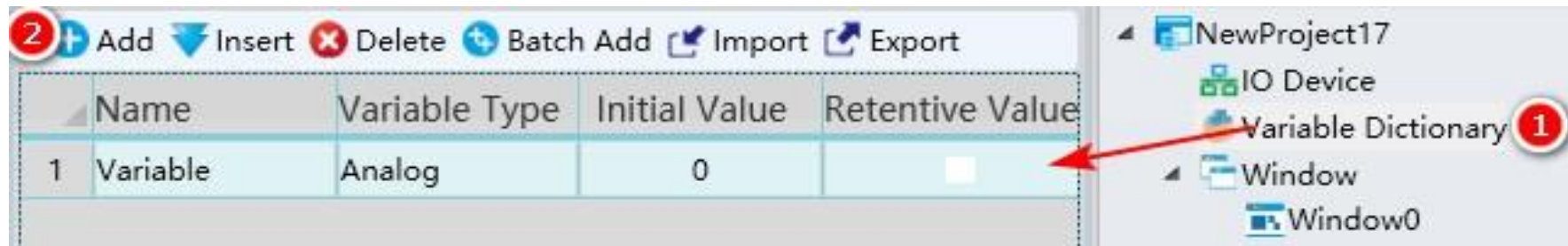


- **Шрифт** : установіть стиль шрифту
- **FontColor** : встановити колір шрифту
- **Рядок стану** : встановіть, чи відображати Рядок стану.
- **StatusbarFontColor** : Установіть колір шрифту рядка стану.
- **ShowGridline** : встановіть, чи відображати сітку.
- **NewRecordPosition** : установіть розташування останнього запису.
- **Фон** : установіть фон керування
- **Панель інструментів** : установіть, чи відображати панель інструментів

Розширення Controls-RecordBox

➤ Змінні операції запису RecordBox:

(1) Створіть змінну: змінна



The screenshot shows the Delta software interface. On the left, a table lists variables. On the right, a tree view shows the project structure. A red arrow points from the 'Variable Dictionary' in the tree view to the table.

	Name	Variable Type	Initial Value	Retentive Value
1	Variable	Analog	0	

Tree View Structure:

- NewProject17
 - IO Device
 - Variable Dictionary (1)
 - Window
 - Window0

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

Розширення Controls-RecordBox

(2) Створіть OperationVariable , пов'язану зі змінною



2 Add Insert Delete Import Export

	Name	Associated Variables	Description
1	OperationVariable	Var.Variable 3	

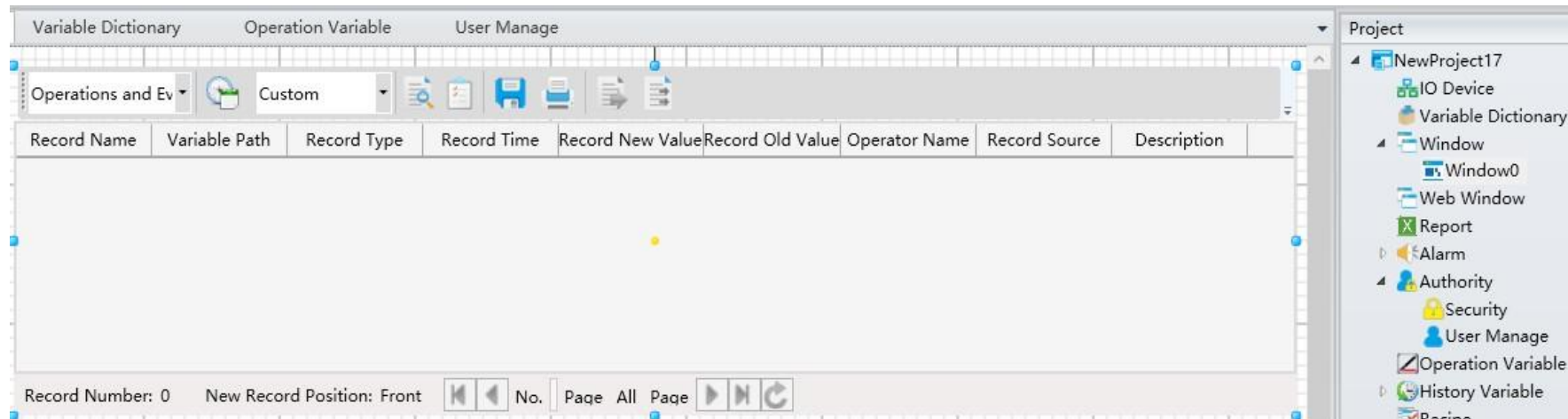
Tree structure on the right:

- NewProject17
 - IO Device
 - Variable Dictionary
 - Window
 - Window0
 - Web Window
 - Report
 - Alarm
 - Authority
 - Operation Variable 1

※Зверніться до розділу «13 Робоча змінна» в посібнику користувача.

Розширення Controls-RecordBox

(3) Створіть RecordBox0 у Window0



Variable Dictionary Operation Variable User Manage

Operations and Ev Custom

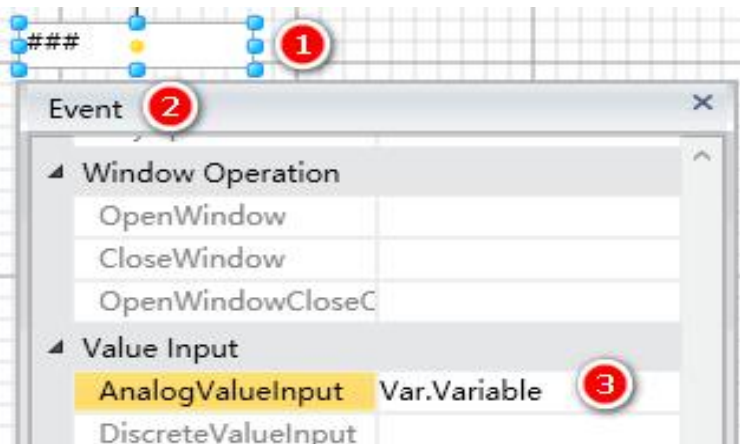
Record Name	Variable Path	Record Type	Record Time	Record New Value	Record Old Value	Operator Name	Record Source	Description
-------------	---------------	-------------	-------------	------------------	------------------	---------------	---------------	-------------

Record Number: 0 New Record Position: Front No. Page All Page [Navigation Icons]

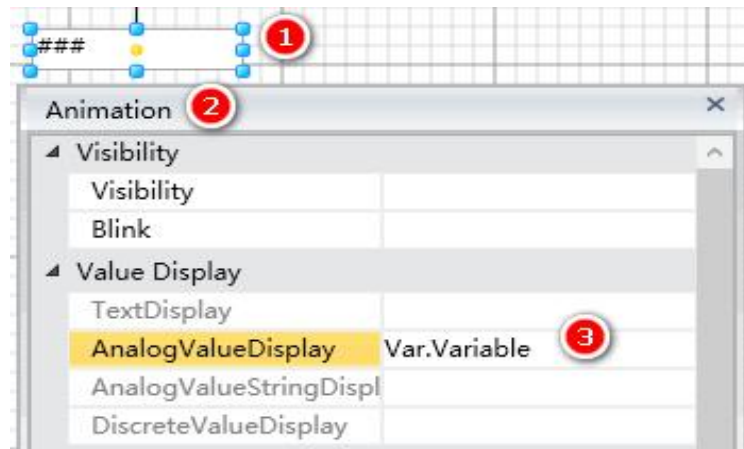
Project

- NewProject17
 - IO Device
 - Variable Dictionary
 - Window
 - Window0
 - Web Window
 - Report
 - Alarm
 - Authority
 - Security
 - User Manage
 - Operation Variable
 - History Variable
 - RecordBox

(4) Створіть Textbox0 у Window0, подію введення аналогового значення та аналогове значення
відображення анімації TextBox0 пов'язано зі змінною



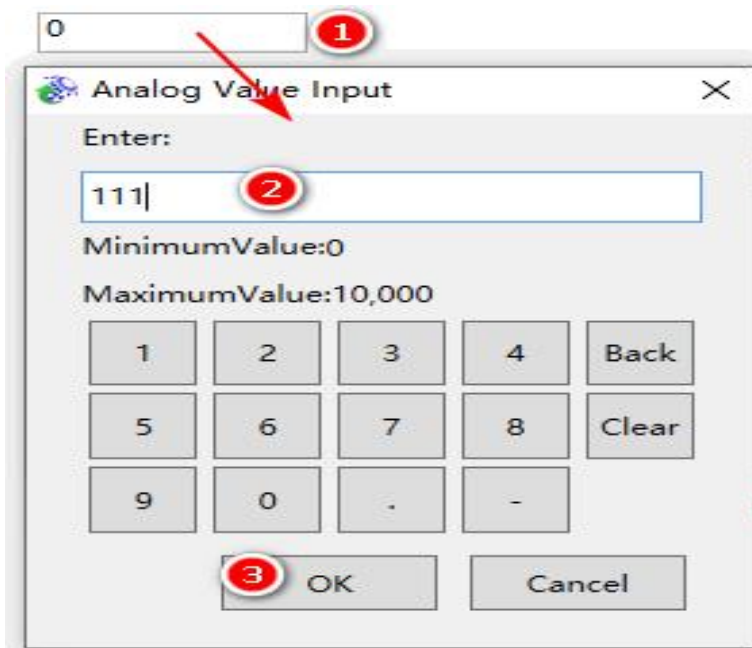
Введення аналогового
значення



Відображення аналогового
значення

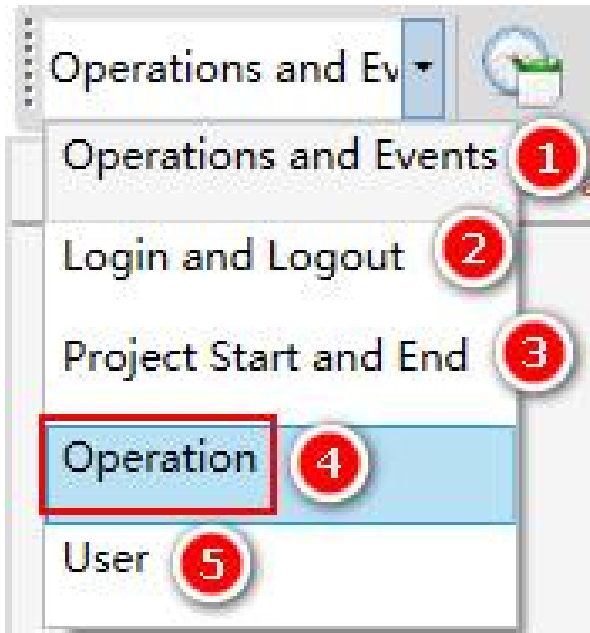
Розширення Controls-RecordBox

(5) Запустіть проект. У Textbox0 введіть 111, потім введіть 222



Розширення Controls-RecordBox

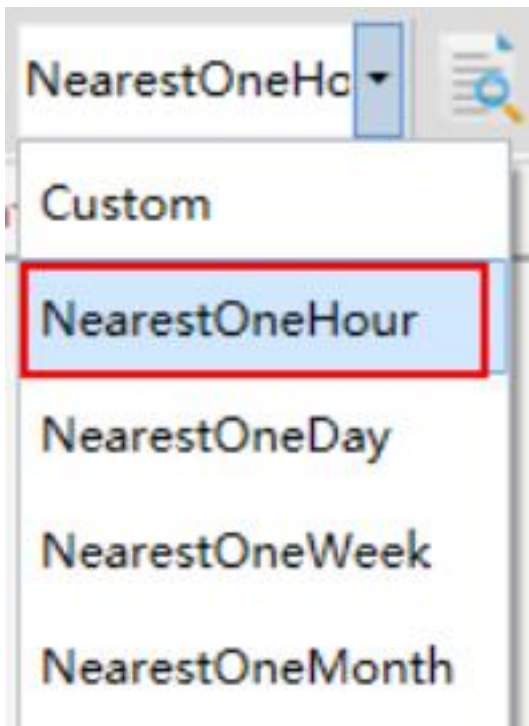
(6) Встановити тип запиту: виберіть Операцію



- ① Записуйте всі операції та події
- ② Лише вхід і вихід користувача
- ③ Тільки початок і кінець проекту
- ④ Тільки операції
- ⑤ Лише зміна параметрів користувача

Розширення Controls-RecordBox

(7) Установити час запиту: виберіть NearestOnehour



Розширення Controls-RecordBox

(8) Натисніть кнопку запиту, щоб отримати запит щодо записів операційних змінних

Operation

NearestOneHc








Record Name	Variable Path	Record Type	Record Time	Record New Value	Record Old Value	Operator Name	Record Source	Description
OperationVariable	Var.Variable		3/30/2020 1:38:01 PM	222	111		CNWJ6IAPC015	
OperationVariable	Var.Variable		3/30/2020 1:35:29 PM	111	0		CNWJ6IAPC015	

Record Number: 2
New Record Position: Front

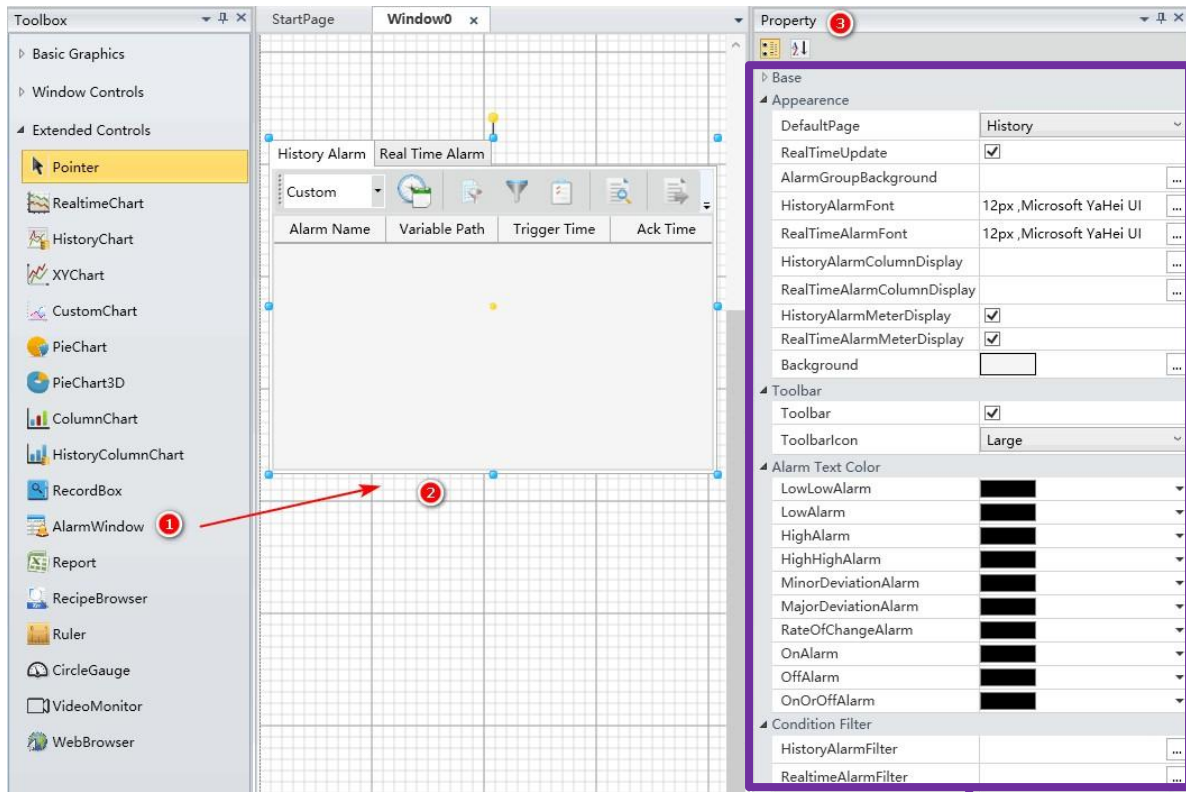


No. 1
Page All 1 Page




Розширити Controls-AlarmWindow

➤ Створіть AlarmWindow0, відкрийте його властивості

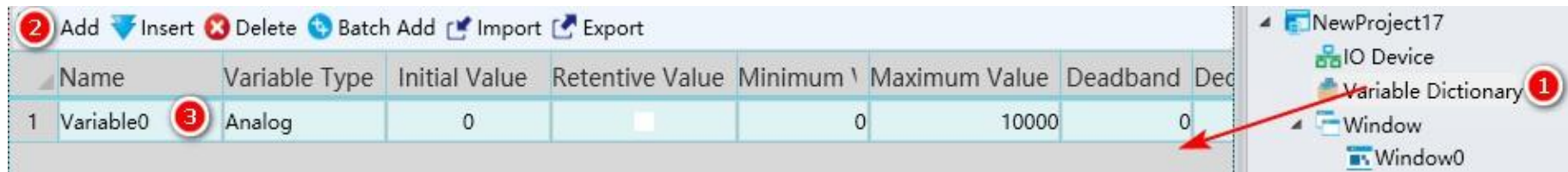


- **DefaultPage** : встановити сторінку за замовчуванням.
- **RealTimeUpdate** : вікно будильника в реальному часі автоматично оновлювати дані
- **AlarmGroupBackground** : установіть колір фону для групи сигналів.
- **HistoryAlarmColumnDisplay** : Встановити відображення стовпця вікна нагадування історії.
- **RealTimeAlarmColumnDisplay** : встановити реальний відображення колонки вікна нагадування часу.
- **HistoryAlarmMeterDisplay** : встановіть, чи відображати заголовки нагадування історії.
- **RealTimeAlarmMeterDisplay** : встановіть, чи відображати заголовок сигналу в реальному часі.
- **Панель інструментів** : встановіть, чи відображати панель інструментів
- **Значок панелі інструментів** : установіть розмір кнопки на панелі інструментів
- **Колір тексту нагадування** : установіть колір тексту нагадування
- **HistoryAlarmFilter** : Встановіть умови фільтра вікна історичних тривог
- **RealtimeAlarmFilter** : Встановити умови фільтра для вікна тривоги в реальному часі

Розширити Controls-AlarmWindow

- AlarmWindow показує нагадування в реальному часі та нагадування історії запитів

(1) Створіть змінну : Variable0



The screenshot shows the Delta software interface. On the left, a table lists variables. On the right, a tree view shows the project structure. A red arrow points from the 'Variable Dictionary' in the tree to the 'Variable0' row in the table.

	Name	Variable Type	Initial Value	Retentive Value	Minimum \	Maximum Value	Deadband	Dec
1	Variable0	Analog	0		0	10000	0	

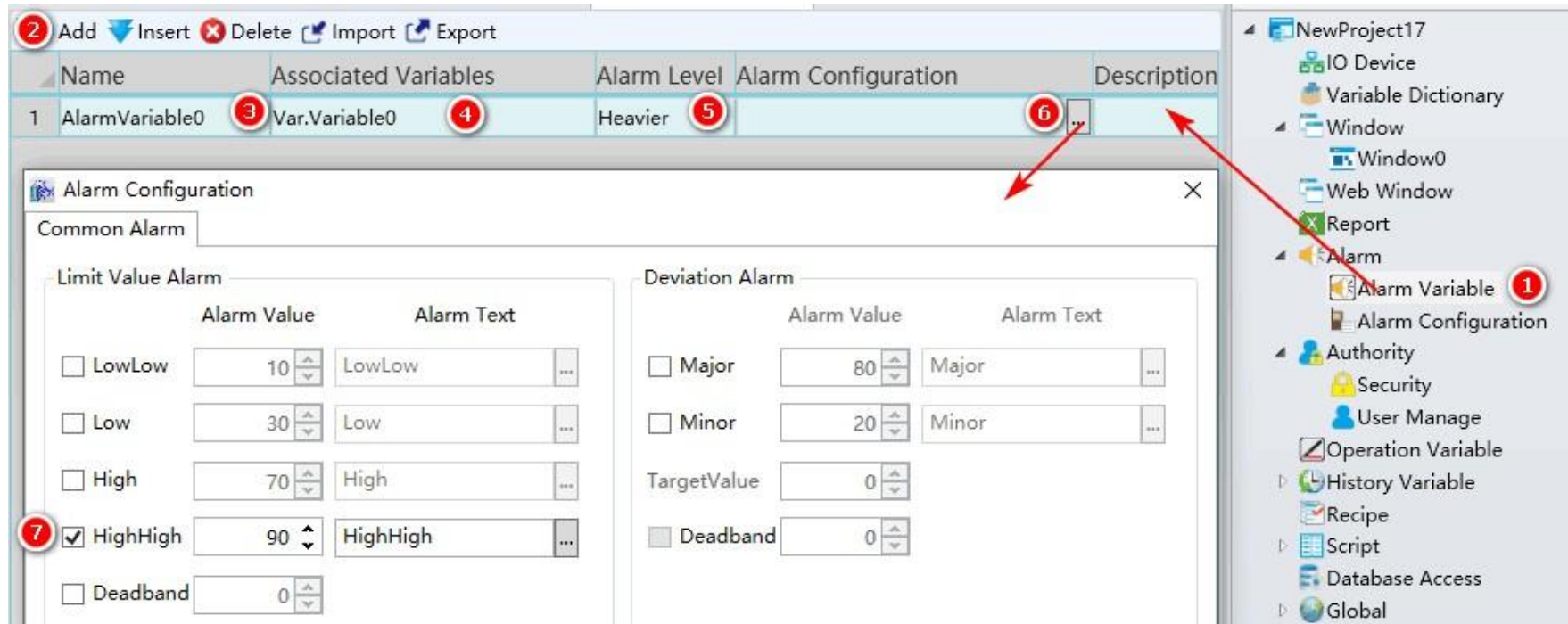
Buttons: Add, Insert, Delete, Batch Add, Import, Export

Tree View: NewProject17, IO Device, Variable Dictionary, Window, Window0

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

Розширити Controls-AlarmWindow

(2) Створіть змінну AlarmVariable0 , пов'язану зі змінною0 , встановіть деякі параметри AlarmVariable0 (сигналізація HighHigh)



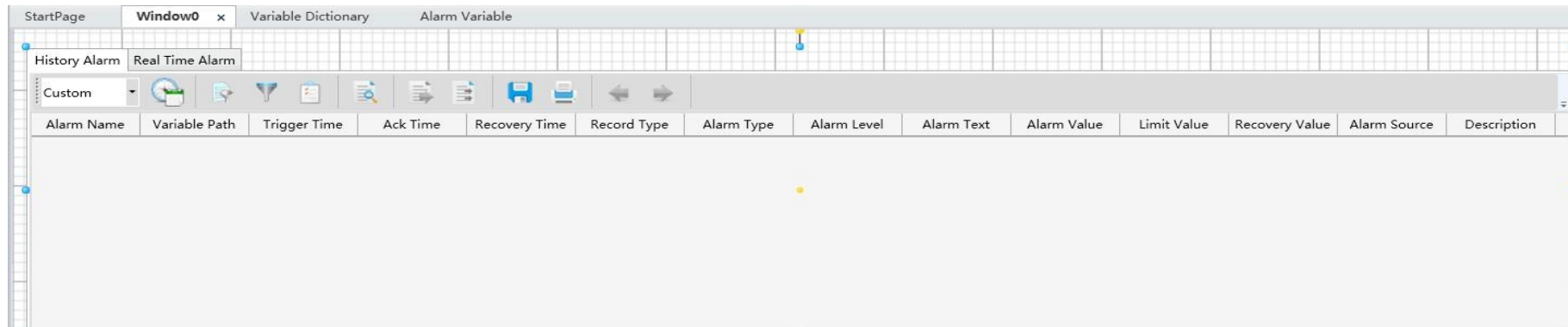
The screenshot shows the Delta Controls-AlarmWindow configuration interface. The main window displays a table of alarm variables. The table has columns: Name, Associated Variables, Alarm Level, Alarm Configuration, and Description. The first row is highlighted, showing 'AlarmVariable0' with 'Var.Variable0' as the associated variable, 'Heavier' as the alarm level, and 'HighHigh' as the alarm configuration. Red circles and arrows highlight the steps to create and configure the alarm variable.

The 'Alarm Configuration' dialog box is open, showing the 'Common Alarm' tab. It has two sections: 'Limit Value Alarm' and 'Deviation Alarm'. In the 'Limit Value Alarm' section, the 'HighHigh' checkbox is checked, and the 'Alarm Value' is set to 90. In the 'Deviation Alarm' section, the 'Major' checkbox is checked, and the 'Alarm Value' is set to 80.

※Зверніться до розділу «11.3 Змінна тривоги» в посібнику користувача.

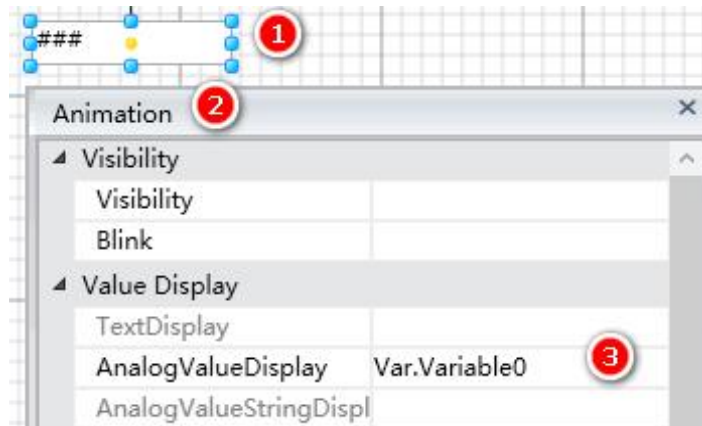
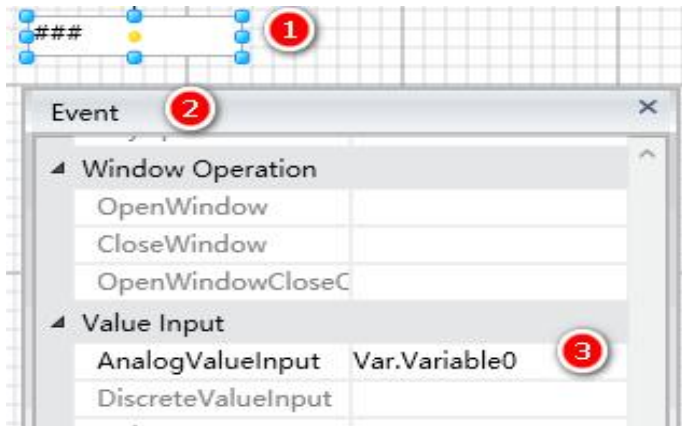
Розширити Controls-AlarmWindow

(3) Створіть AlarmWindow0 у Window0



Розширити Controls-AlarmWindow

(4) Створіть Textbox0 у Window0 , подію введення аналогового значення та аналогове значення відображення анімації TextBox0 пов'язано зі змінною

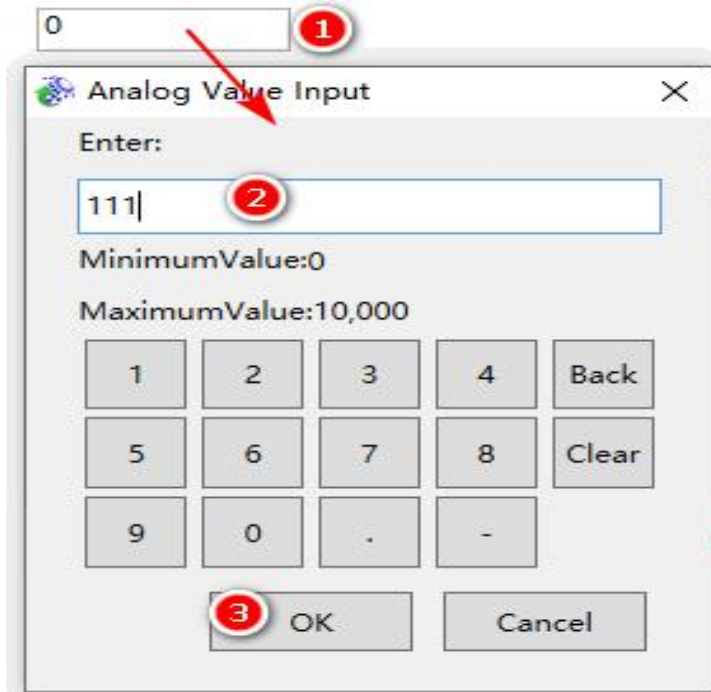


Введення аналогового значення

Відображення аналогового значення

Розширити Controls-AlarmWindow

(5) Запустіть проект. У текстовому полі 0 введіть 111 (більше граничного значення сигналу HighHigh — 90)



0

1

Analog Value Input

Enter:

111

2

MinimumValue:0

MaximumValue:10,000

1	2	3	4	Back
5	6	7	8	Clear
9	0	.	-	

3 OK Cancel

Розширити Controls-AlarmWindow

(6) Вікно сигналу в реальному часі оновлює сигнал у реальному часі під час виконання

History Alarm		Real Time Alarm 1											
2 		Ack Selected      											
Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	Alarm Text	Alarm Value	Limit Value	Current Value	Recovery Value	Alarm
Alarm.AlarmVariable0	Var.Variable0	3/31/2020 9:22:19 AM			Alarm	HighHighAlarm	Heavier	HighHigh	111	90	111		CNW

(7) Вікно нагадування історії запитує нагадування історії запитів під час виконання

History Alarm Real Time Alarm

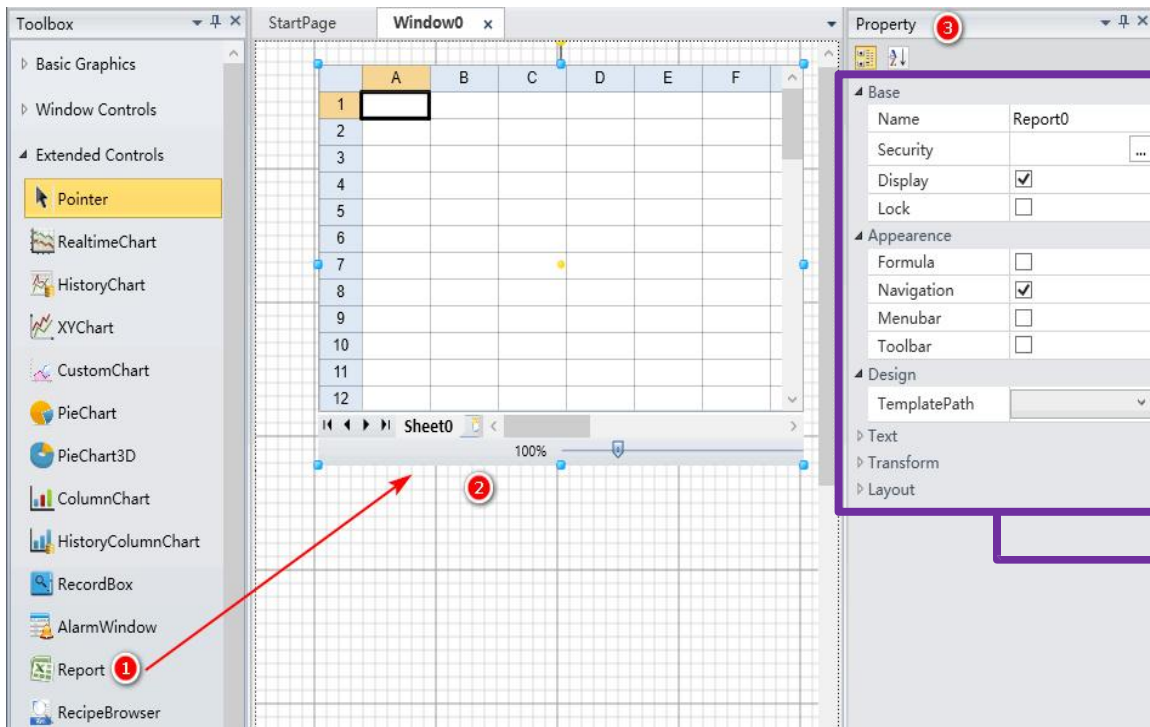
NearestOne

No.1Page

Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	Alarm Text	Alarm Value	Limit Value	Recovery Value	Alarm Source	Description
Alarm.AlarmVariable0	Var.Variable0	3/31/2020 9:22:19 AM			Alarm	HighHighAlarm	Heavier	HighHigh	111	90		CNWJ6IAPC015	

Розширення елементів керування-звіту

➤ Створіть Report0, відкрийте вікно його властивостей

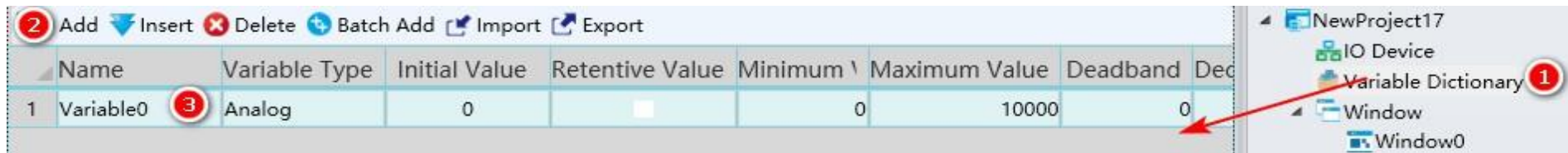


- **Формула** : установіть, чи відображати панель формул
- **Навігація** : укажіть, чи відображати панель навігації
- **Рядок меню** : встановіть, чи відображати рядок меню, позначте його дійсним, лише коли поточний шлях шаблону нульовий
- **Панель інструментів** : встановіть, чи відображати панель інструментів
- **TemplatePath** : встановити поточний шаблон звіту

Розширення елементів керування-звіту

➤ Звітувати про дані історії запитів: (1)

Створіть змінну: Variable0



The screenshot shows the software interface with a toolbar at the top containing buttons: Add (2), Insert, Delete, Batch Add, Import, and Export. Below the toolbar is a table with the following columns: Name, Variable Type, Initial Value, Retentive Value, Minimum Value, Maximum Value, Deadband, and Dec. The table contains one row with the following data:

	Name	Variable Type	Initial Value	Retentive Value	Minimum Value	Maximum Value	Deadband	Dec
1	Variable0 (3)	Analog	0		0	10000	0	

On the right side of the interface, there is a tree view showing the project structure: NewProject17, IO Device, Variable Dictionary (1), Window, and Window0. A red arrow points from the 'Variable Dictionary' entry in the tree view to the 'Variable0' entry in the table.

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

Розширення елементів керування-звіту

(2) Створіть пристрій імітації Device0, додайте адресу в Device0, яка пов'язана зі змінною 0



	Name	Address	Associated Variables	Value	Data Conversion
1	Address	Increase:0/DWORD	Var.Variable0		No Conversion

※ Спосіб створення пристрою моделювання див. у розділі «5.10.1 Симулятор» у посібнику користувача

Розширення елементів керування-звіту

(3) Створіть змінну історії (RecordVariable), яка пов'язана зі змінною 0

2 Add Insert Delete Import Export

Name	Associated Variables	Mode	Timer	Deadband
1 RecordVariable	Var.Variable0	Timing	s1	N/A

3 4 5

Timer Browser

	IsEnable	Name	Timer Unit	Timer Coefficient	Relative Time	Description
1	☒	s1	Second	1	0001-01-01 00:00:00	Triggers every second
2	☐	s10	Second	10	0001-01-01 00:00:00	Triggers every 10 seconds
3	☐	s30	Second	30	0001-01-01 00:00:00	Triggers every 30 seconds
4	☐	m1	Minute	1	0001-01-01 00:00:00	Triggers every minute
5	☐	m30	Minute	30	0001-01-01 00:00:00	Triggers every 30 minutes
6	☐	h1	Hour	1	0001-01-01 00:00:00	Triggers every hour

6

Project Tree:

- NewProject17
 - IO Device
 - Device0
 - Variable Dictionary
 - Window
 - Window0
 - Web Window
 - Report
 - Alarm
 - Alarm Variable
 - Alarm Configuration
 - Authority
 - Security
 - User Manage
 - Operation Variable
 - History Variable
 - Record Variable 1
 - Historical Group

※Зверніться до розділу «14.2 Налаштування змінної запису історії» в посібнику користувача.

Розширення елементів керування-звіту

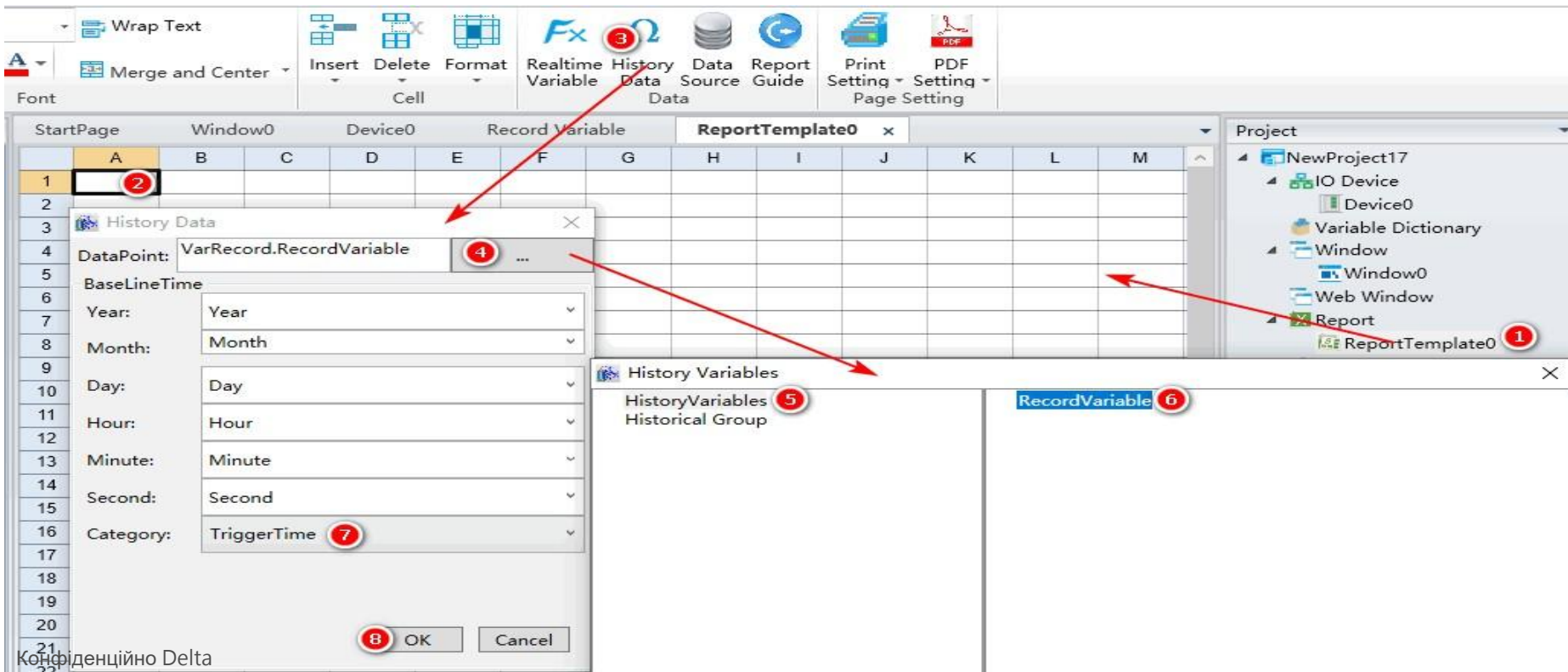
(4) Створіть шаблон звіту та назвіть його ReportTemplate0



- ① Клацніть правою кнопкою миші вузол звіту, щоб відкрити меню
- ② Натисніть Новий звіт, щоб додати шаблон звіту: Report0
- ③ Клацніть правою кнопкою миші вузол Report0, щоб відкрити меню
- ④ Натисніть Перейменувати, щоб перейменувати шаблон звіту
- ⑤ Шаблон звіту перейменовано на ReportTemplate0

Розширення елементів керування-звіту

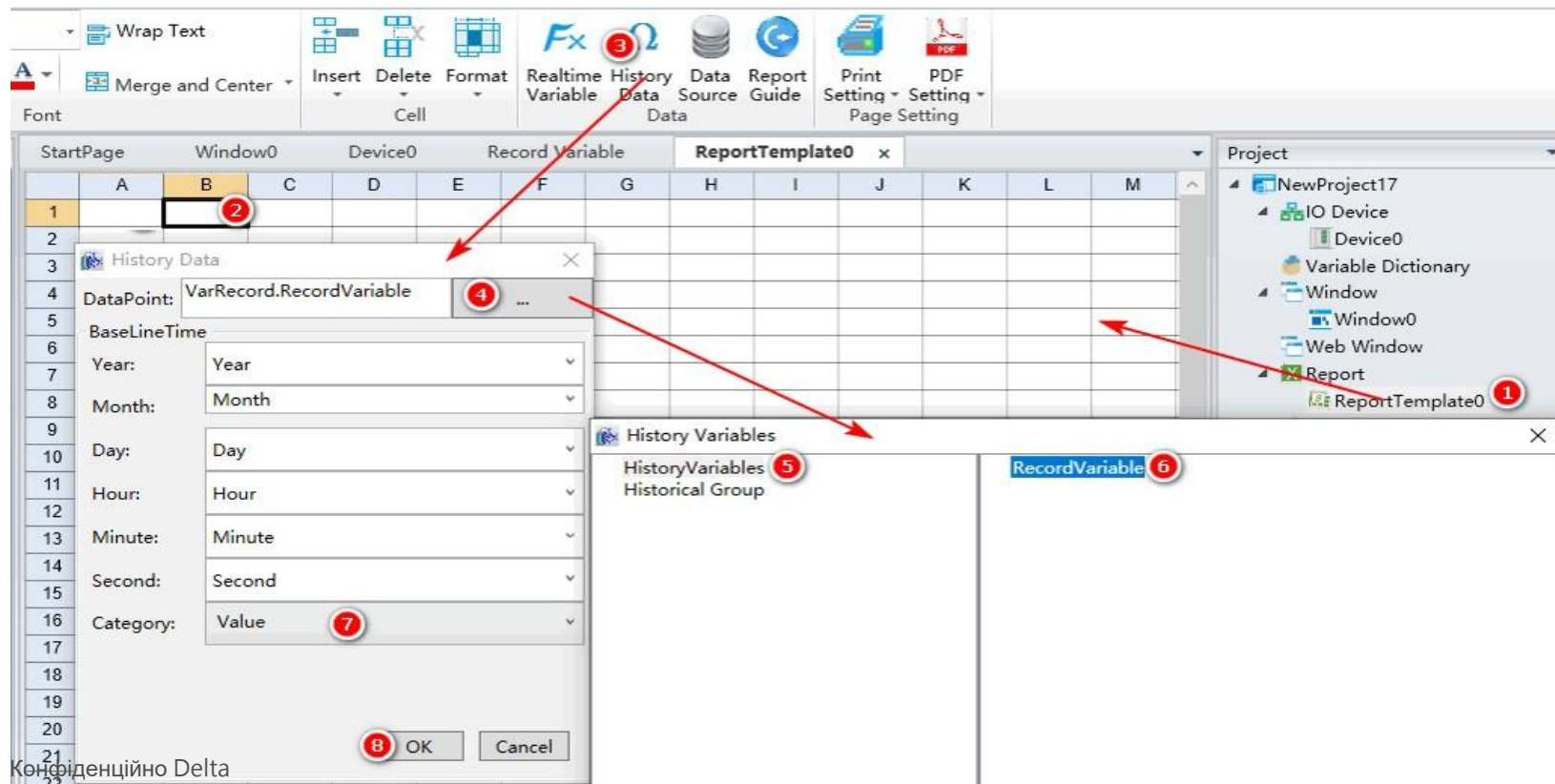
(5) Комірка в першому рядку, стовпці A в ReportTemplate0 пов'язана з часом запуску RecordVariable



The screenshot displays the Delta software interface for configuring a report template. The main window shows a grid with columns labeled A through M and rows numbered 1 through 21. The 'ReportTemplate0' tab is active. The 'History Data' dialog is open, showing the 'DataPoint' field set to 'VarRecord.RecordVariable' (4). The 'BaseLineTime' section shows time units (Year, Month, Day, Hour, Minute, Second) and a 'Category' dropdown set to 'TriggerTime' (7). The 'History Variables' section shows 'HistoryVariables Historical Group' (5) and 'RecordVariable' (6). The 'Project' tree on the right shows the hierarchy: NewProject17 > IO Device > Device0 > Variable Dictionary > Window > Window0 > Web Window > Report > ReportTemplate0 (1). Red arrows indicate the flow from the 'DataPoint' field to the 'RecordVariable' variable and then to the 'History Variables' section.

Розширення елементів керування-звіту

(6) Комірка в першому рядку, стовпці В у ReportTemplate0 пов'язана з значення RecordVariable



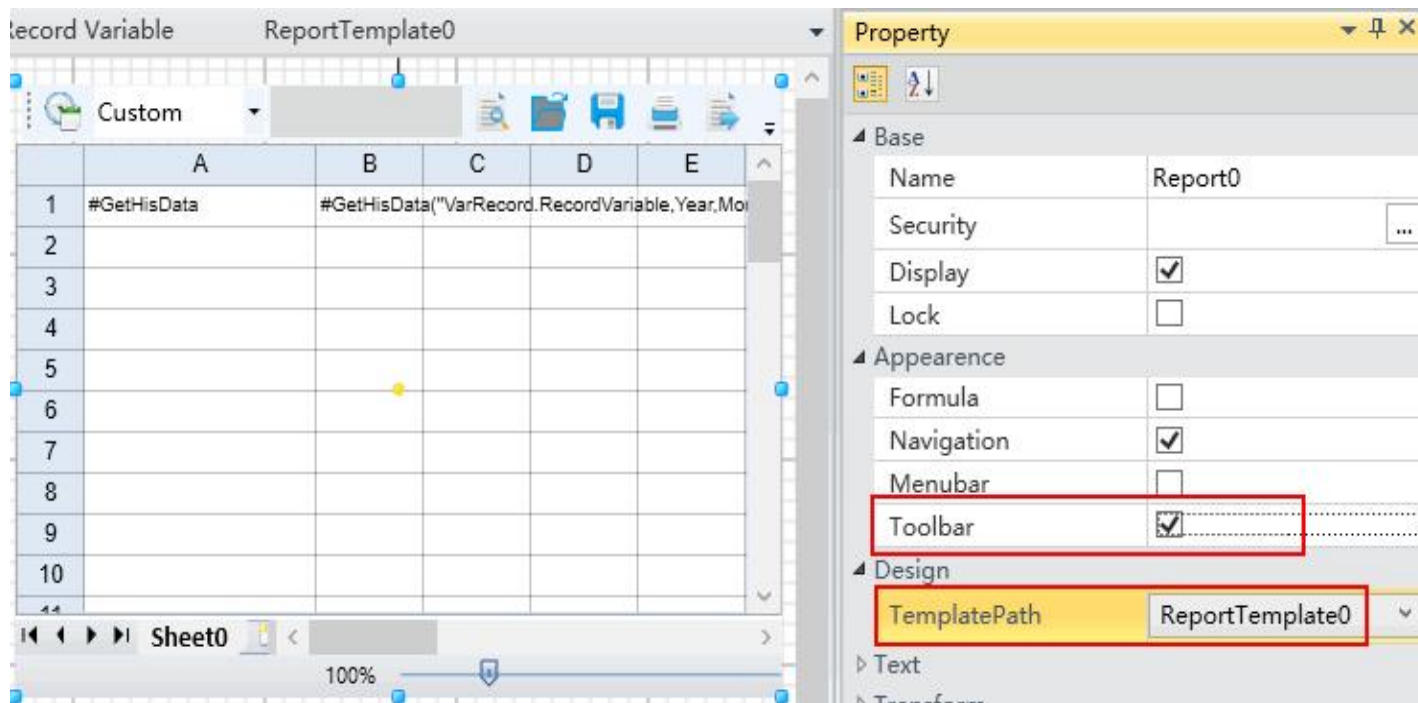
The screenshot displays the Delta software interface for configuring a report template. The main window shows a spreadsheet with columns A through M and rows 1 through 21. The 'ReportTemplate0' tab is active. The 'History Data' dialog box is open, showing the 'DataPoint' field set to 'VarRecord.RecordVariable'. The 'History Variables' dialog box is also open, showing the 'RecordVariable' variable selected. The 'History Data' dialog box has a 'Category' field set to 'Value'. The 'History Variables' dialog box has a 'Historical Group' field set to 'RecordVariable'. The 'History Data' dialog box has an 'OK' button and a 'Cancel' button.

Key elements and steps highlighted:

- 1**: Selecting 'ReportTemplate0' in the Project tree.
- 2**: Selecting cell B1 in the spreadsheet.
- 3**: Clicking the 'History' icon in the toolbar.
- 4**: Clicking the ellipsis button in the 'DataPoint' field of the 'History Data' dialog.
- 5**: Selecting 'RecordVariable' in the 'History Variables' dialog.
- 6**: The 'RecordVariable' variable being assigned to cell B1.
- 7**: Selecting 'Value' as the category in the 'History Data' dialog.

Розширення елементів керування-звіту

(7) Створіть Report0 у Window0 та пов'яжіть ReportTemplate0, перевірте властивість Toolbar

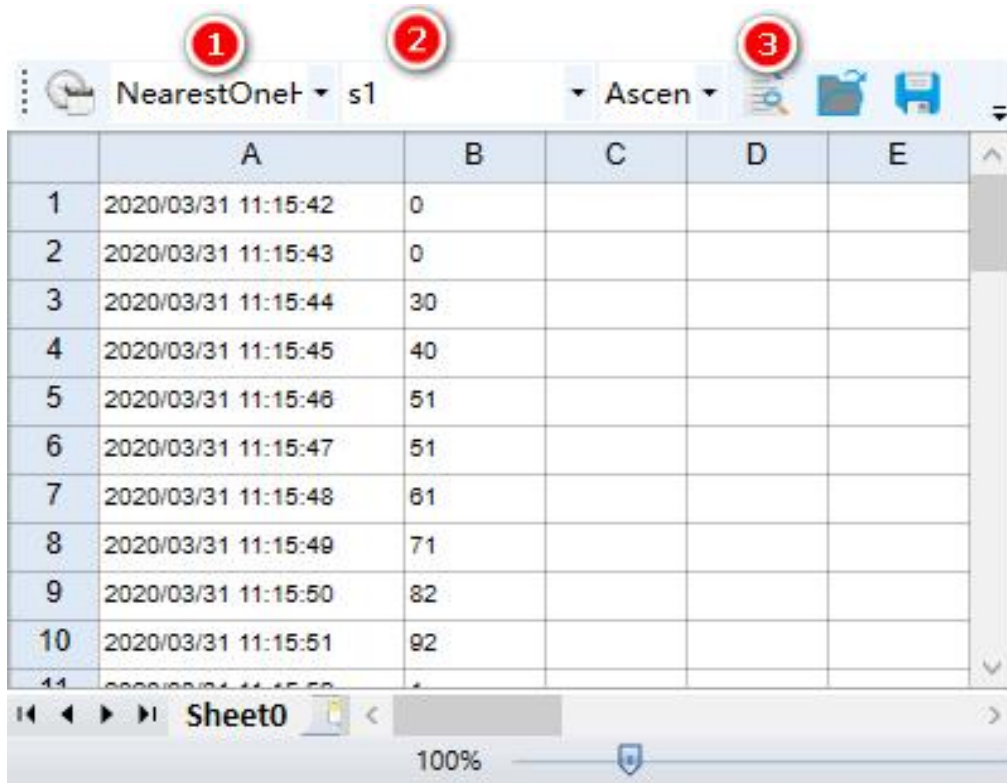


The screenshot displays the 'ReportTemplate0' editor window. The main area is a grid with columns A through E and rows 1 through 10. A 'Custom' toolbar is visible at the top of the grid. The 'Property' panel on the right shows the configuration for 'Report0'.

Property	
Base	
Name	Report0
Security	...
Display	☒
Lock	☐
Appearance	
Formula	☐
Navigation	☒
Menubar	☐
Toolbar	☒
Design	
TemplatePath	ReportTemplate0
Text	
Transform	

Розширення елементів керування-звіту

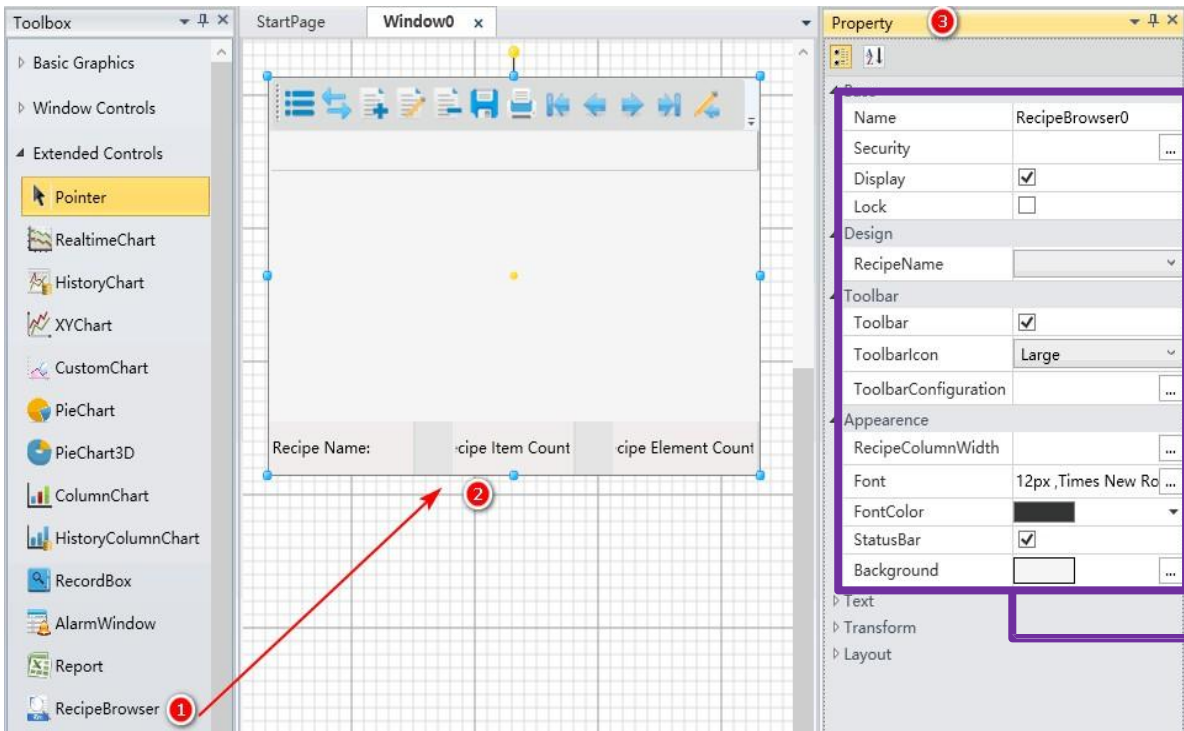
(8) Запустіть Window0 і запитайте дані за останню годину під час виконання



	A	B	C	D	E
1	2020/03/31 11:15:42	0			
2	2020/03/31 11:15:43	0			
3	2020/03/31 11:15:44	30			
4	2020/03/31 11:15:45	40			
5	2020/03/31 11:15:46	51			
6	2020/03/31 11:15:47	51			
7	2020/03/31 11:15:48	61			
8	2020/03/31 11:15:49	71			
9	2020/03/31 11:15:50	82			
10	2020/03/31 11:15:51	92			
11	2020/03/31 11:15:52				

Розширення Controls-RecipeBrowser

➤ Створіть RecipeBrowser0, відкрийте вікно його властивостей

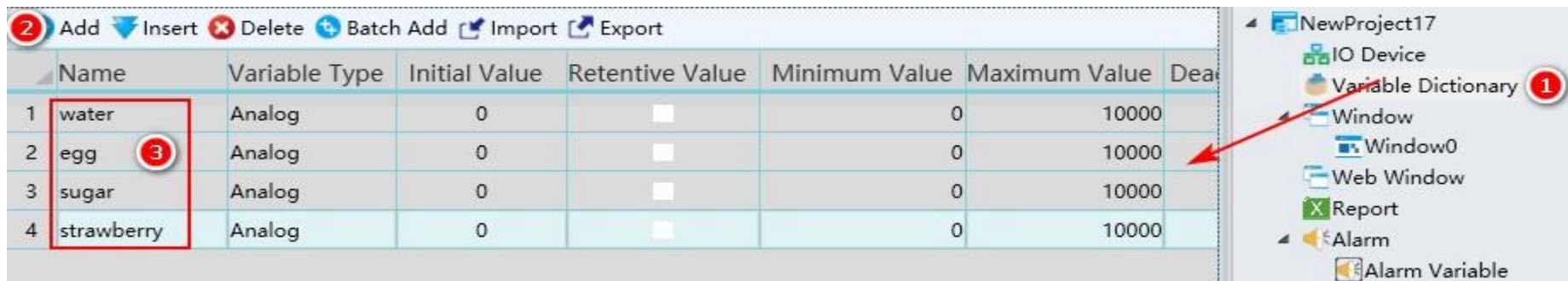


- **RecipeName** : установіть рецепт, прив'язаний до браузера рецептів
- **Панель інструментів** : встановіть, чи відображати панель інструментів
- **Значок панелі інструментів** : установіть розмір кнопки на панелі інструментів
- **RecipeColumnWidth** : установіть рецепт ширина стовпця
- **Шрифт** : установіть шрифт браузера рецептів
- **FontColor** : Установіть колір шрифту браузера рецептів
- **Рядок стану** : встановіть, чи відображати рядок стану

Розширення Controls-RecipeBrowser

Використовуйте браузер рецептів, щоб відобразити рецепт:

(1) Створіть 4 змінні та назвіть їх вода, яйце, цукор, полуниця



The screenshot shows the Controls-RecipeBrowser interface. On the left, a table lists four variables: 'water', 'egg', 'sugar', and 'strawberry'. Each variable is of type 'Analog' and has an initial value of 0, a retentive value checkbox, and a range from 0 to 10000. A red box highlights the first four rows of the table, and a red circle with the number 3 is next to the 'egg' row. On the right, a project tree shows the hierarchy: 'NewProject17' > 'IO Device' > 'Variable Dictionary' (marked with a red circle 1) > 'Window' > 'Window0' > 'Web Window' > 'Report' > 'Alarm' > 'Alarm Variable'. A red arrow points from the 'Variable Dictionary' node to the table.

	Name	Variable Type	Initial Value	Retentive Value	Minimum Value	Maximum Value	Dead
1	water	Analog	0	☐	0	10000	
2	egg	Analog	0	☐	0	10000	
3	sugar	Analog	0	☐	0	10000	
4	strawberry	Analog	0	☐	0	10000	

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

Розширення Controls-RecipeBrowser

(2) Створіть рецепт і назвіть його Cake0



- ① Клацніть правою кнопкою миші вузол «Рецепт», щоб відкрити меню
- ② Натисніть New Recipe, щоб додати рецепт: Recipe0
- ③ Клацніть правою кнопкою миші вузол Recipe0, щоб відкрити меню
- ④ Натисніть Перейменувати, щоб назвати рецепт
- ⑤ Рецепт перейменовано на Cake0

Розширення Controls-RecipeBrowser

(3) Створіть 4 змінні рецепту та назвіть їх Recipe_water, Recipe_egg, Recipe_sugar, Recipe_strawberry, які пов'язані з водою, яйцем, цукром і полуницею відповідно

2 Add Element
Insert Element
Delete Element
5 Add Item
Insert Item
Delete Item
Import
Export

	Name	Recipe Type	Associated Variables	Description	StrawberryCake1	StrawberryCake2
1	Recipe_water	Analog	Var.water		6 100	7 100
2	Recipe_egg	Analog	Var.egg		50	50
3	Recipe_sugar	Analog	Var.sugar		30	20
4	Recipe_strawberry	Analog	Var.strawberry		50	60

① Натисніть вузол Cake0, щоб відкрити інтерфейс рецепта

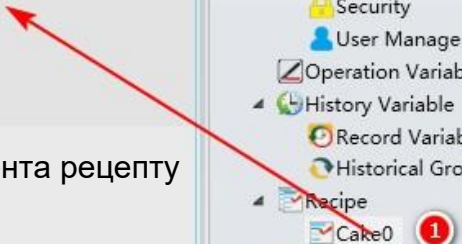
② Додайте 4 елементи рецепту

③ Назвіть елементи рецепту

④ Асоційовані змінні елементів рецепта

⑤ Додайте елементи рецепту

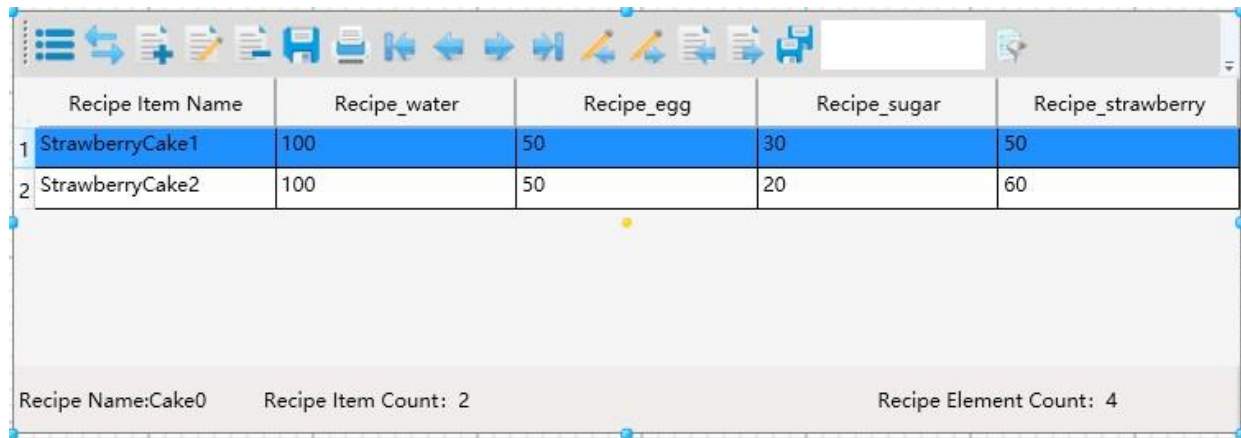
⑥⑦ Назвіть пункти рецепта та встановіть параметри кожного елемента рецепту



NewProject17

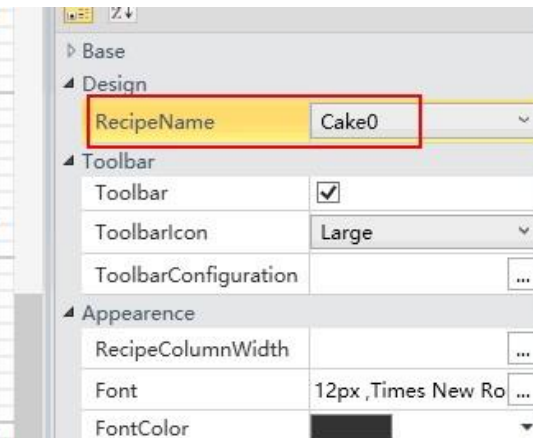
- IO Device
- Variable Dictionary
- Window
 - Window0
 - Web Window
- Report
- Alarm
 - Alarm Variable
 - Alarm Configuration
- Authority
 - Security
 - User Manage
- Operation Variable
- History Variable
 - Record Variable
 - Historical Group
- Recipe
 - Cake0 1

(4) Створіть RecipeBrowser0 і зв'яжіть Cake0

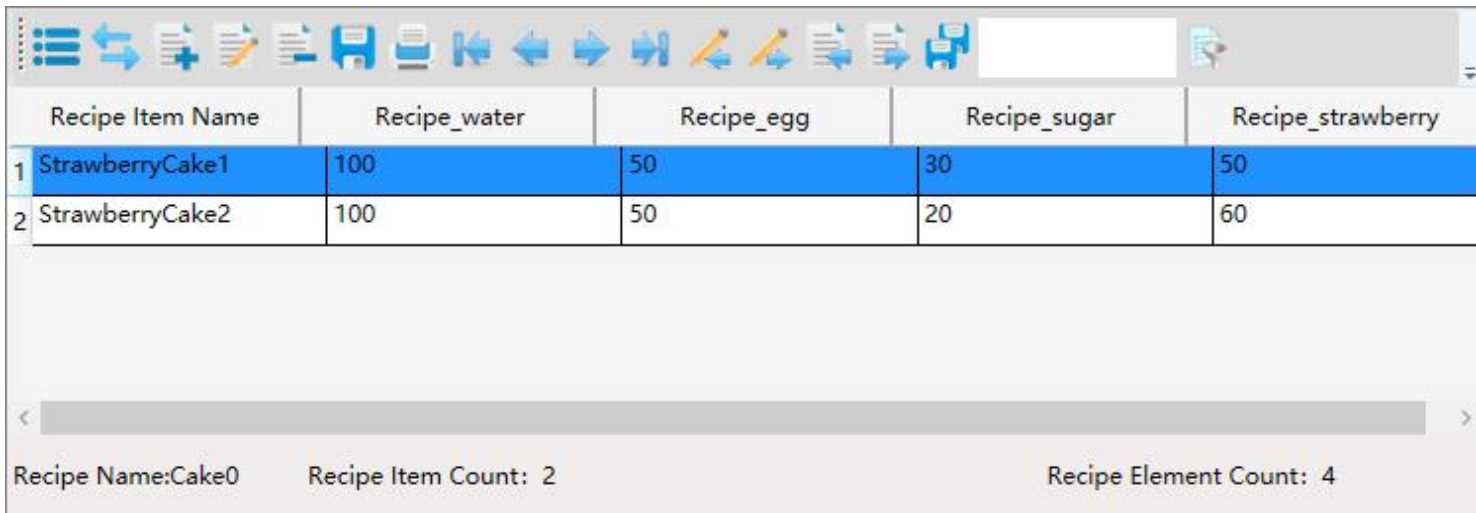


	Recipe Item Name	Recipe_water	Recipe_egg	Recipe_sugar	Recipe_strawberry
1	StrawberryCake1	100	50	30	50
2	StrawberryCake2	100	50	20	60

Recipe Name: Cake0 Recipe Item Count: 2 Recipe Element Count: 4



(5) Запустіть Window0



The screenshot shows the Controls-RecipeBrowser interface. At the top is a toolbar with various icons for navigation and editing. Below the toolbar is a table with the following data:

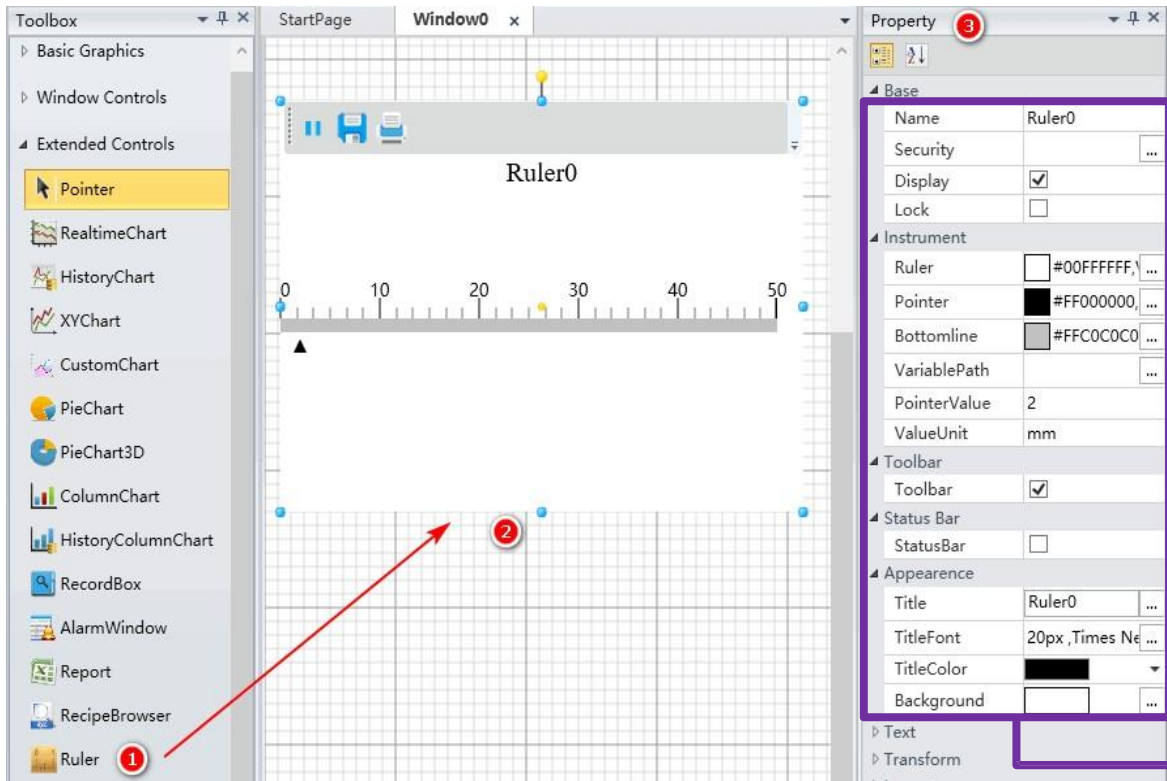
	Recipe Item Name	Recipe_water	Recipe_egg	Recipe_sugar	Recipe_strawberry
1	StrawberryCake1	100	50	30	50
2	StrawberryCake2	100	50	20	60

Below the table is a horizontal scrollbar. At the bottom of the interface, the following information is displayed:

Recipe Name: Cake0 Recipe Item Count: 2 Recipe Element Count: 4

Розширення елементів управління-лінійка

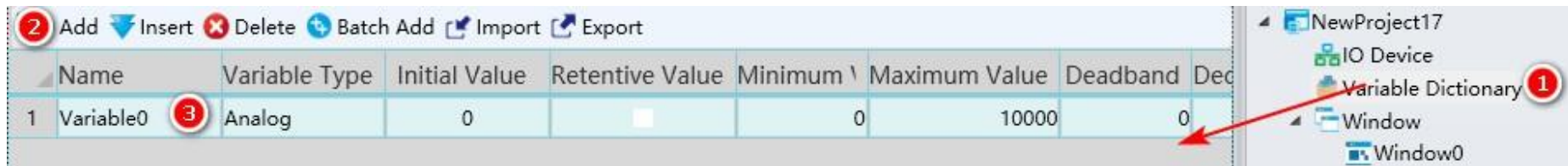
➤ Створіть Ruler0, відкрийте вікно його властивостей



- **Лінійка** : установіть параметри лінійки
- **Покажчик** : установіть стиль покажчика
- **Нижня лінія** : установіть стиль нижньої лінії
- **VariablePath** : установіть шлях змінної, пов'язаний із лінійкою
- **PointerValue** : Установити значення за замовчуванням для покажчик
- **ValueUnit** : установіть одиницю значення
- **Панель інструментів** : встановіть, чи відображати панель інструментів
- **Рядок стану** : встановіть, чи відображати рядок стану
- **Заголовок** : установіть вміст заголовка
- **TitleFont** : установіть шрифт для заголовка
- **TitleColor** : Установіть колір заголовка
- **Фон** : установіть колір фону лінійки

➤ Лінійка відображає дані в реальному часі:

(1) Створіть змінну : Variable0

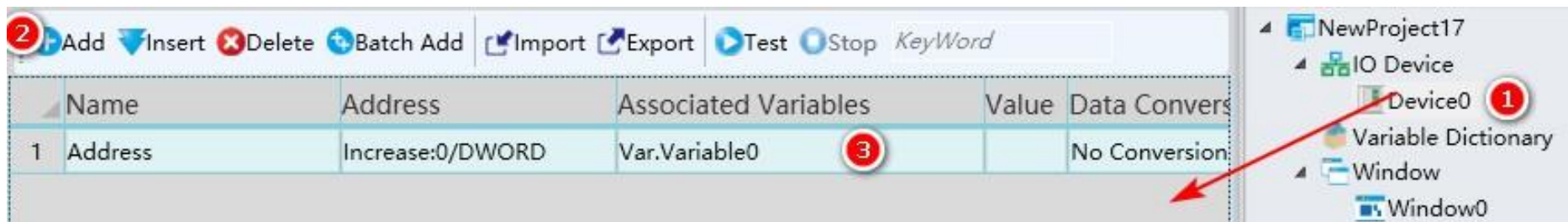


The screenshot shows the Delta software interface. On the right, the 'Variable Dictionary' is expanded, showing a tree structure with 'NewProject17' at the top, followed by 'IO Device', 'Variable Dictionary' (marked with a red circle 1), 'Window', and 'Window0'. A red arrow points from the 'Variable Dictionary' to the table below. The table has columns: Name, Variable Type, Initial Value, Retentive Value, Minimum Value, Maximum Value, Deadband, and Dec. The first row shows 'Variable0' (marked with a red circle 3) as an 'Analog' variable with an 'Initial Value' of 0, 'Minimum Value' of 0, and 'Maximum Value' of 10000. The table also has a toolbar at the top with buttons: Add (marked with a red circle 2), Insert, Delete, Batch Add, Import, and Export.

	Name	Variable Type	Initial Value	Retentive Value	Minimum Value	Maximum Value	Deadband	Dec
1	Variable0	Analog	0		0	10000	0	

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

(2) Створіть пристрій імітації Device0, додайте адресу в Device0, яка пов'язана зі змінною 0



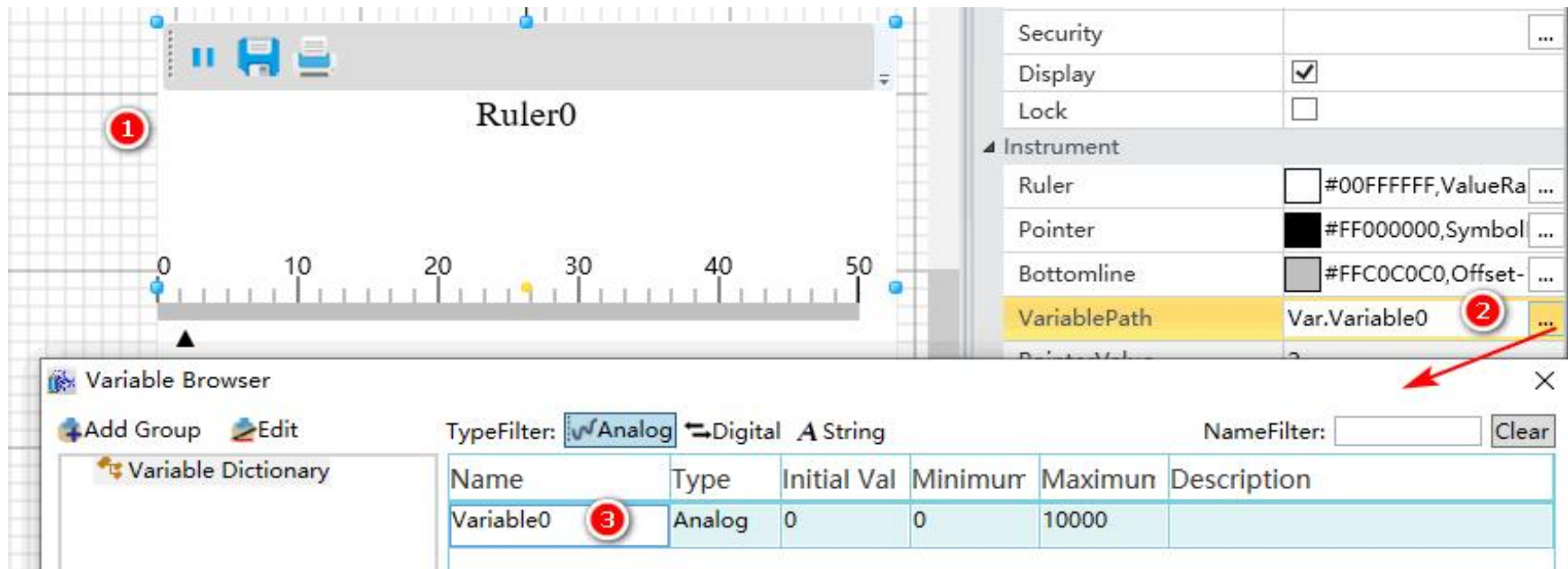
	Name	Address	Associated Variables	Value	Data Conversion
1	Address	Increase:0/DWORD	Var.Variable0 (3)		No Conversion

Project Tree:

- NewProject17
 - IO Device
 - Device0 (1)
 - Variable Dictionary
 - Window
 - Window0

※ Спосіб створення пристрою моделювання див. у розділі «5.10.1 Симулятор» у посібнику користувача

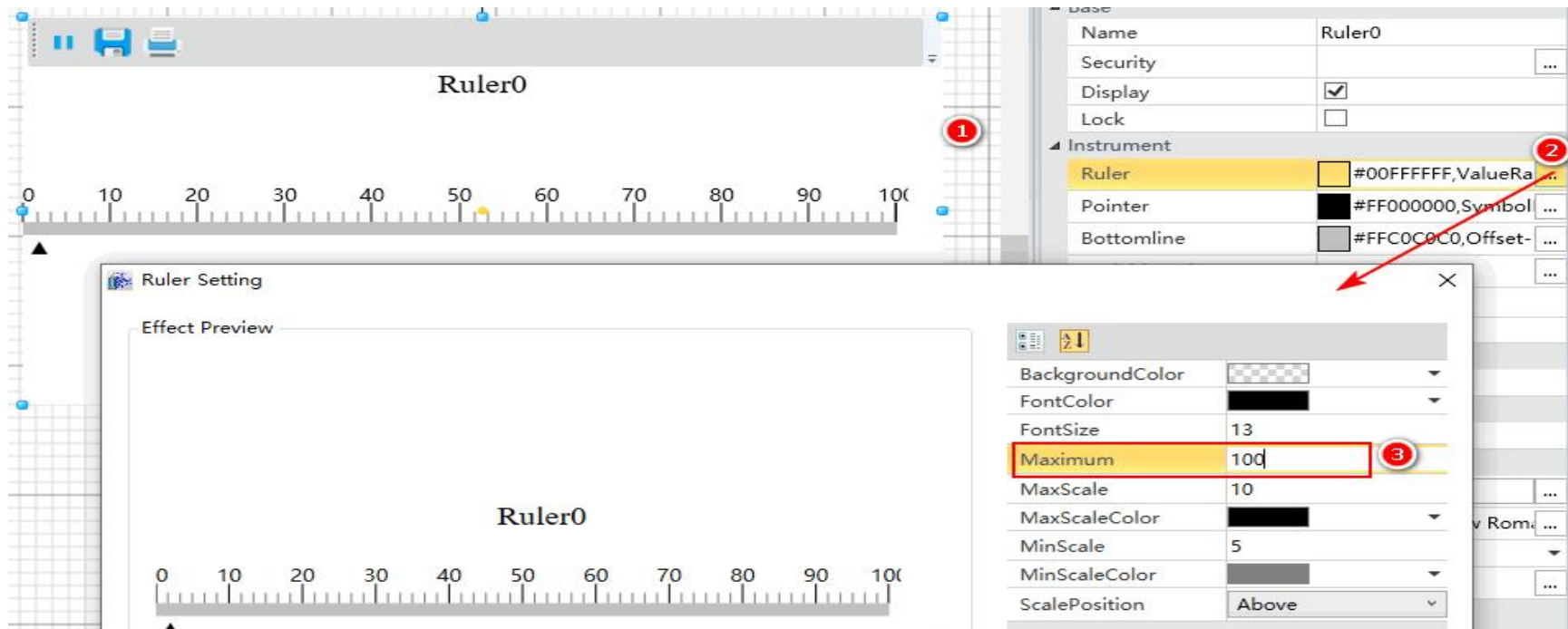
(3) Створіть лінійку 0 у вікні 0, пов'язану зі змінною 0



The screenshot displays the Delta HMI software interface. At the top, a ruler widget labeled "Ruler0" is shown on a grid. A red circle with the number "1" is placed near the ruler's start. To the right, a properties panel lists various settings. The "Instrument" section is expanded, and the "VariablePath" property is highlighted in yellow, showing the value "Var.Variable0". A red circle with the number "2" is placed next to this value, and a red arrow points to it. Below the ruler, a "Variable Browser" window is open. It has tabs for "Add Group", "Edit", and "Variable Dictionary". The "TypeFilter" is set to "Analog", and the "NameFilter" is empty. A table lists variables, with "Variable0" selected. A red circle with the number "3" is placed next to "Variable0" in the table.

Name	Type	Initial Val	Minimur	Maximun	Description
Variable0	Analog	0	0	10000	

(4) Встановіть для лінійки 0 максимум 100



The screenshot displays the Delta software interface for configuring a ruler. The main window shows a ruler labeled "Ruler0" with a red circle "1" at the end. A "Ruler Setting" dialog is open, showing an "Effect Preview" of the ruler and a "Properties" panel. In the "Properties" panel, the "Maximum" value is set to 100, highlighted with a red box and a red circle "3". A red arrow points from circle "2" in the "Instrument" list to the "Ruler Setting" dialog.

Base	
Name	Ruler0
Security	...
Display	☒
Lock	☐
Instrument	
Ruler	#00FFFFFF, ValueRa...
Pointer	#FF000000, Symbol...
Bottomline	#FFC0C0C0, Offset-

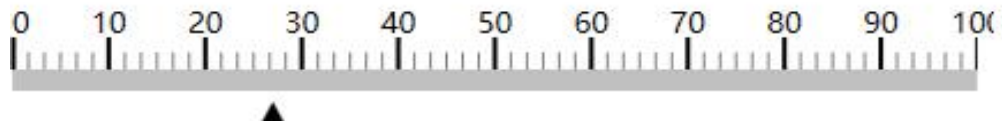
Ruler Setting	
Effect Preview	
Ruler0	
0 10 20 30 40 50 60 70 80 90 100	

Properties	
BackgroundColor	...
FontColor	...
FontSize	13
Maximum	100
MaxScale	10
MaxScaleColor	...
MinScale	5
MinScaleColor	...
ScalePosition	Above

(5) Запустіть проект

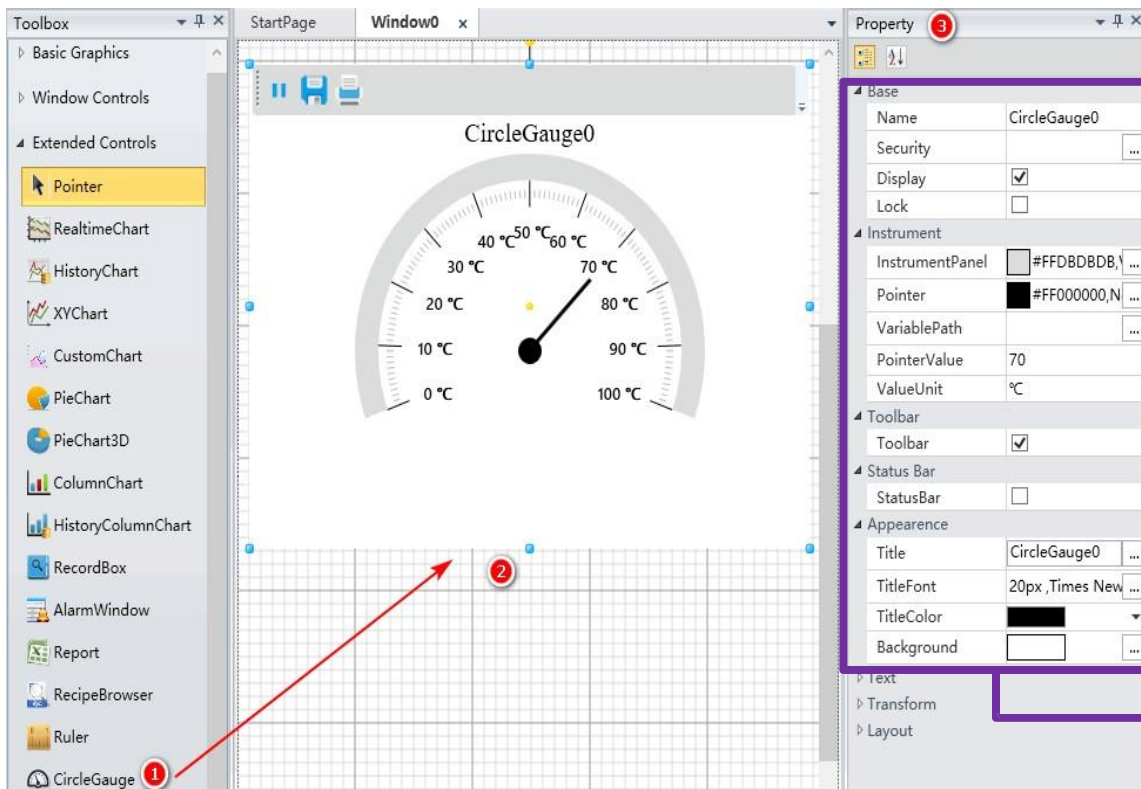


Ruler0



Розширення Controls-CircleGauge

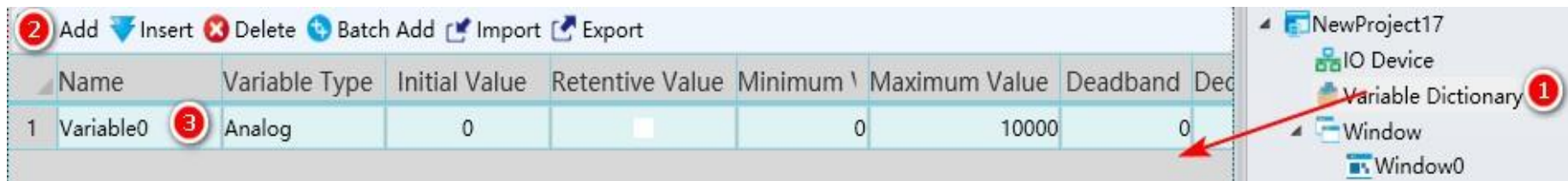
➤ Створіть CircleGauge0, відкрийте вікно його властивостей



- **InstrumentPanel** : Встановіть параметри панелі приладів
- **Показчик** : установіть стиль показчика
- **VariablePath** : установіть шлях до змінної, пов'язаний із CircleGauge
- **PointerValue** : Установити значення вказівника за умовчанням
- **ValueUnit** : установіть одиницю значення
- **Панель інструментів** : встановіть, чи відображати панель інструментів
- **Рядок стану** : встановіть, чи відображати рядок стану
- **Заголовок** : установіть вміст заголовка
- **TitleFont** : установіть шрифт для заголовка
- **TitleColor** : Установіть колір заголовка
- **Фон** : установіть колір фону CircleGauge

- CircleGauge відображає дані в реальному часі:

(1) Створіть змінну : Variable0



The screenshot shows the Delta Controls software interface. On the right, the 'Variable Dictionary' is expanded, showing a tree structure with 'NewProject17' at the top, followed by 'IO Device', 'Variable Dictionary' (marked with a red circle 1), 'Window', and 'Window0'. A red arrow points from the 'Variable Dictionary' to the table below. The table has columns: Name, Variable Type, Initial Value, Retentive Value, Minimum Value, Maximum Value, Deadband, and Dec. The first row is highlighted with a red circle 3 and contains the following data:

	Name	Variable Type	Initial Value	Retentive Value	Minimum Value	Maximum Value	Deadband	Dec
1	Variable0	Analog	0		0	10000	0	

At the top of the table, there are buttons: Add (marked with a red circle 2), Insert, Delete, Batch Add, Import, and Export.

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

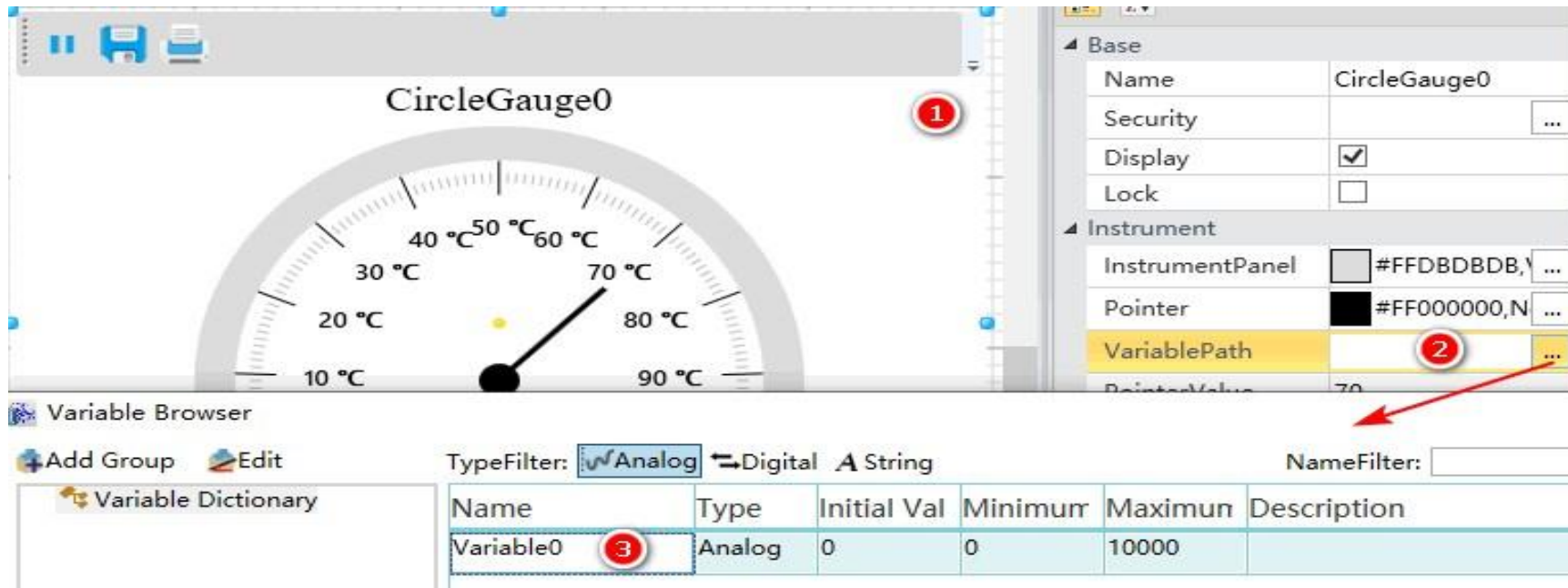
(2) Створіть пристрій імітації Device0, додайте адресу в Device0, яка пов'язана зі змінною 0



	Name	Address	Associated Variables	Value	Data Conversion
1	Address	Increase:0/DWORD	Var.Variable0		No Conversion

※ Спосіб створення пристрою моделювання див. у розділі «5.10.1 Симулятор» у посібнику користувача

(3) Створіть CircleGauge0 у Window0, який пов'язує Variable0



The screenshot shows the Delta HMI software interface. On the left, a circular gauge labeled 'CircleGauge0' is displayed, showing a temperature scale from 10 °C to 90 °C. The needle points to approximately 70 °C. On the right, the configuration panel for 'CircleGauge0' is visible. The 'Base' section shows 'Name' as 'CircleGauge0', 'Security' as empty, 'Display' as checked, and 'Lock' as unchecked. The 'Instrument' section shows 'InstrumentPanel' as '#FFDBDBDB', 'Pointer' as '#FF000000', and 'VariablePath' as 'Variable0'. A red arrow points from the 'VariablePath' property to the 'Variable0' entry in the 'Variable Dictionary' table below.

Variable Browser

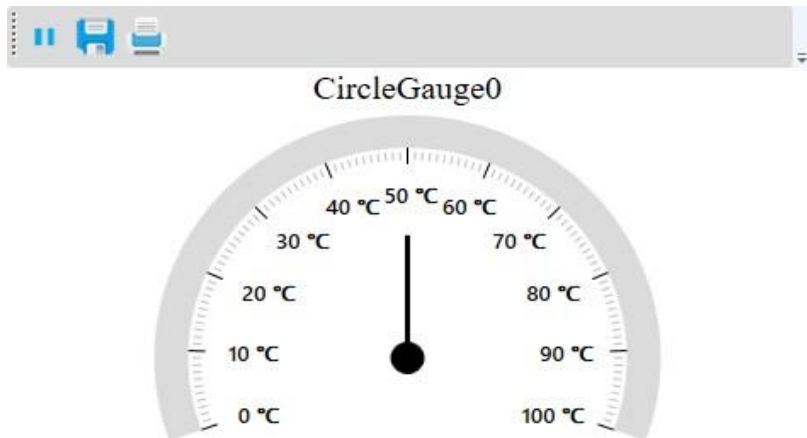
Add Group Edit

Variable Dictionary

TypeFilter: ☒ Analog ☐ Digital ☐ String NameFilter:

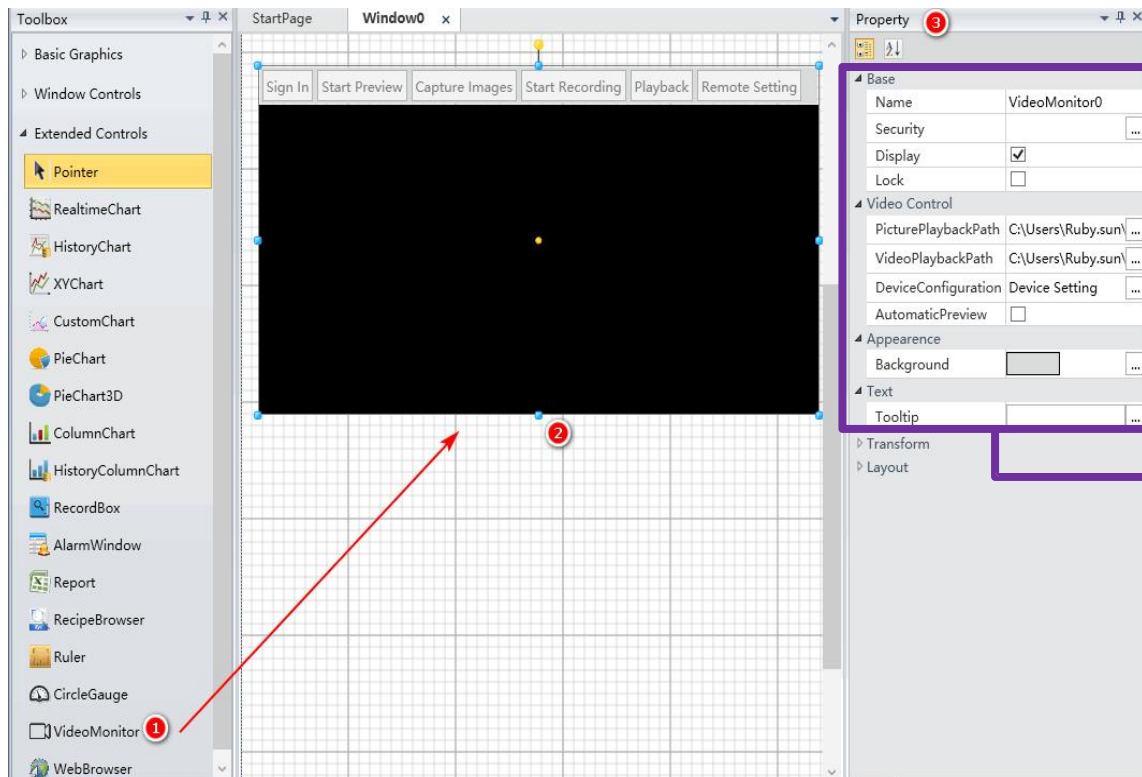
Name	Type	Initial Val	Minimum	Maximum	Description
Variable0	Analog	0	0	10000	

(4) Запустіть проект



Розширення Controls-VideoMintor

➤ Створіть VideoMintor0, відкрийте вікно його властивостей



- **PicturePlaybackPath** : отримати або встановити шлях відтворення зображення.
- **VideoPlaybackPath** : отримати або встановити шлях відтворення відео.
- **Конфігурація пристрою** : Device IP, порт Device, і ім'я користувача, пароль.
- **AutomaticPreview** : встановить автоматичний попередній перегляд відео
- **Фон** : установить колір фону VideoMintor



Розширення Controls-VideoMintor

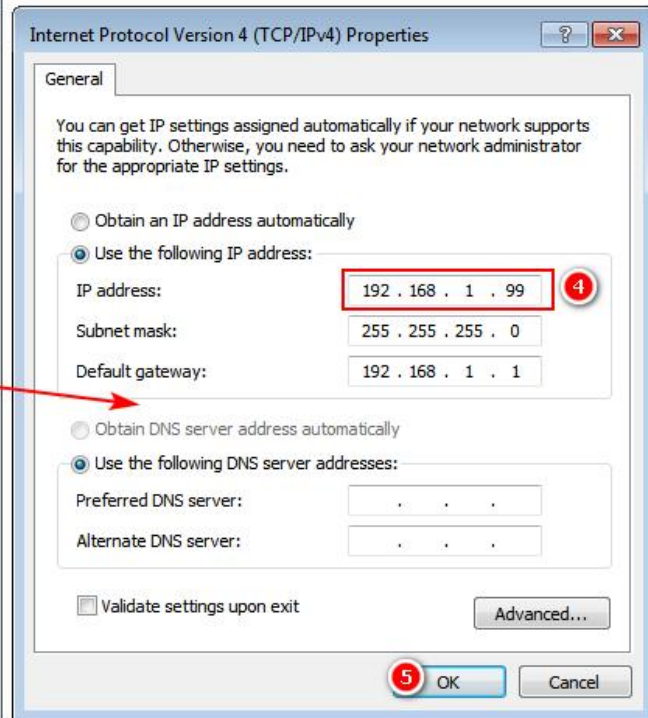
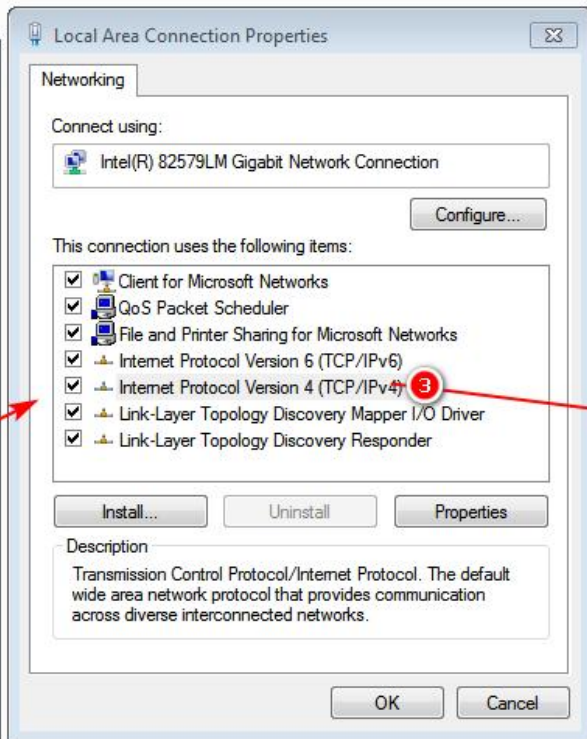
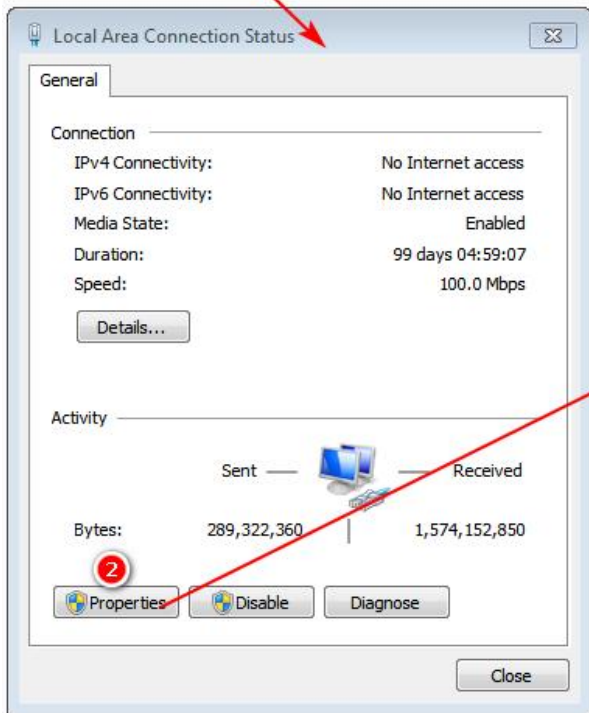
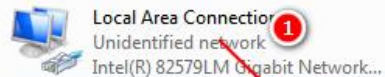
➤ Запишіть відео та відтворіть його:

(1) Інформація для камери (віддаленого пристрою) така

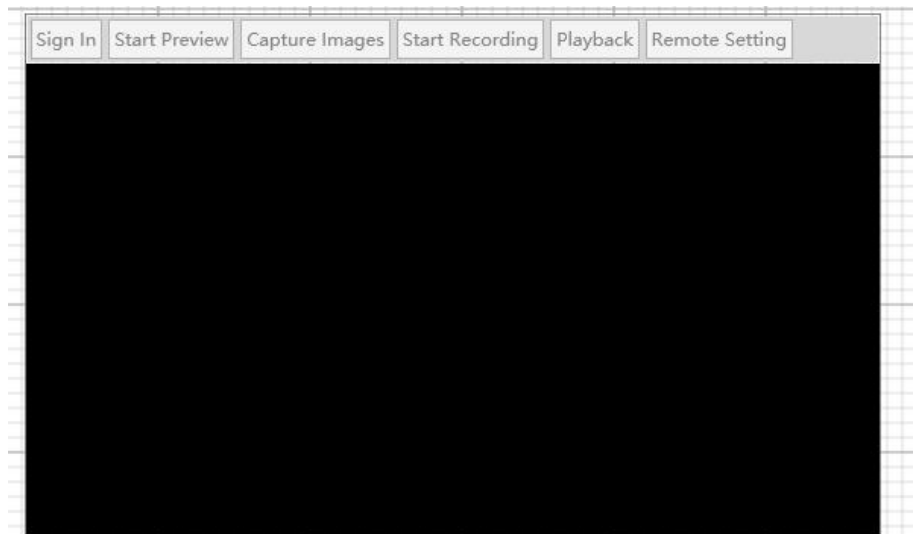
Device Parameters

Device IP:	192.168.1.32	Device Port:	8000	Logging Device
User Name:	admin	User Password:	•••••	

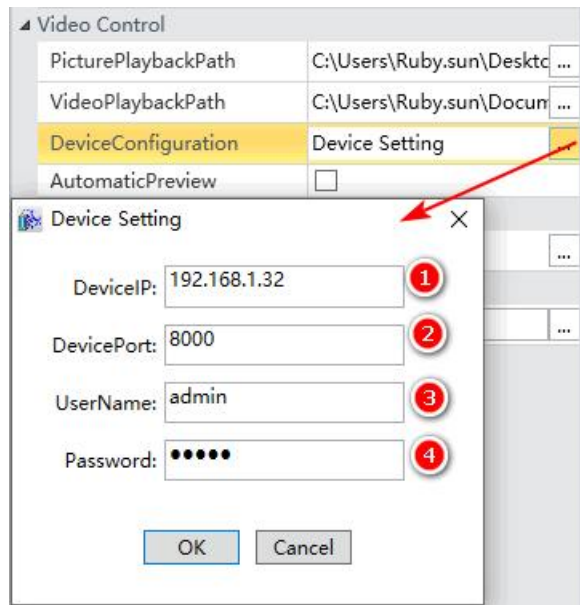
(2) Комп'ютер, підключений до камери через Ethernet



(3) Створіть VideoMintor0 у Window0

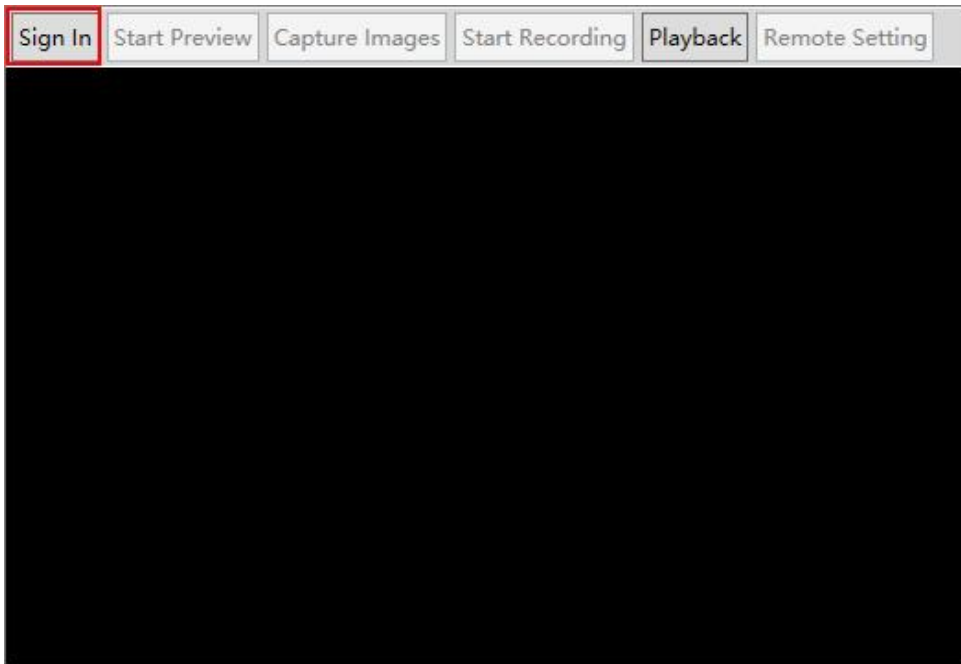


(4) Введіть параметри пристрою у вікні DeviceConfiguration

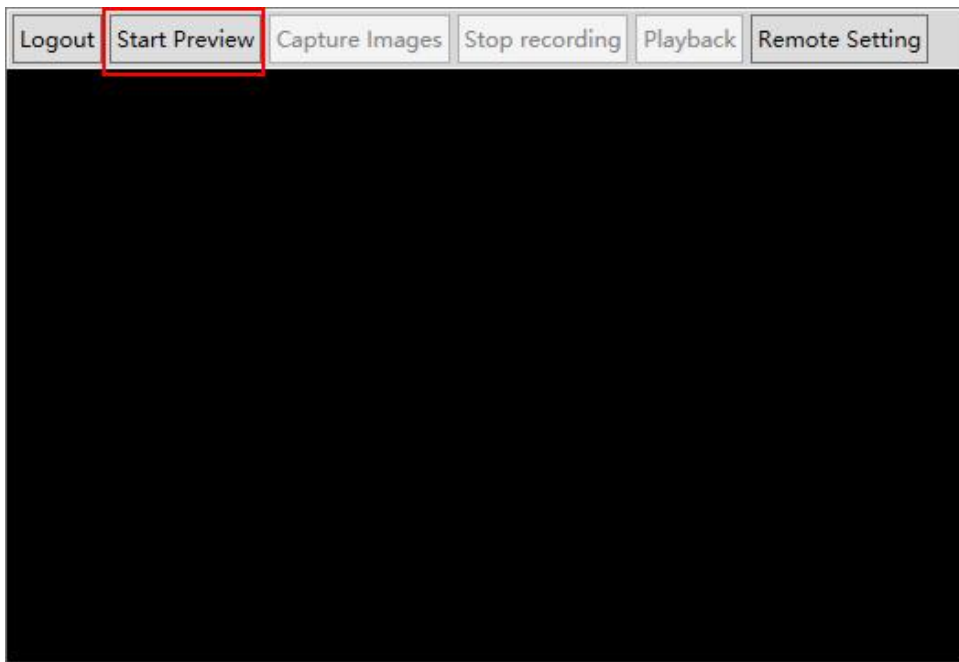


- ① IP віддаленого пристрою
- ② Порт віддаленого пристрою
- ③ Ім'я користувача віддаленого пристрою
- ④ Пароль користувача віддаленого пристрою

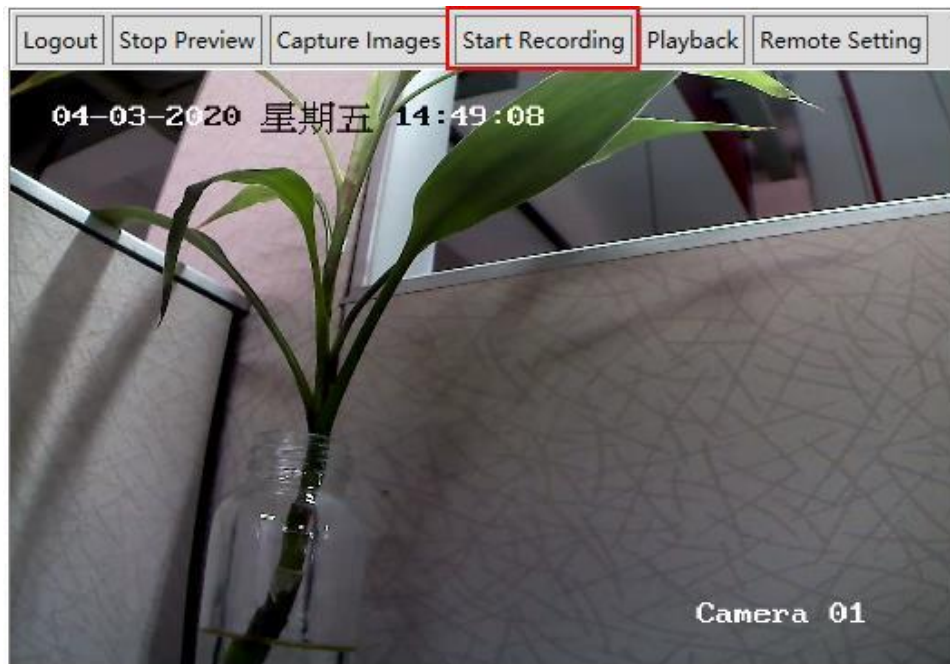
(5) Запустіть проект, увійдіть успішно



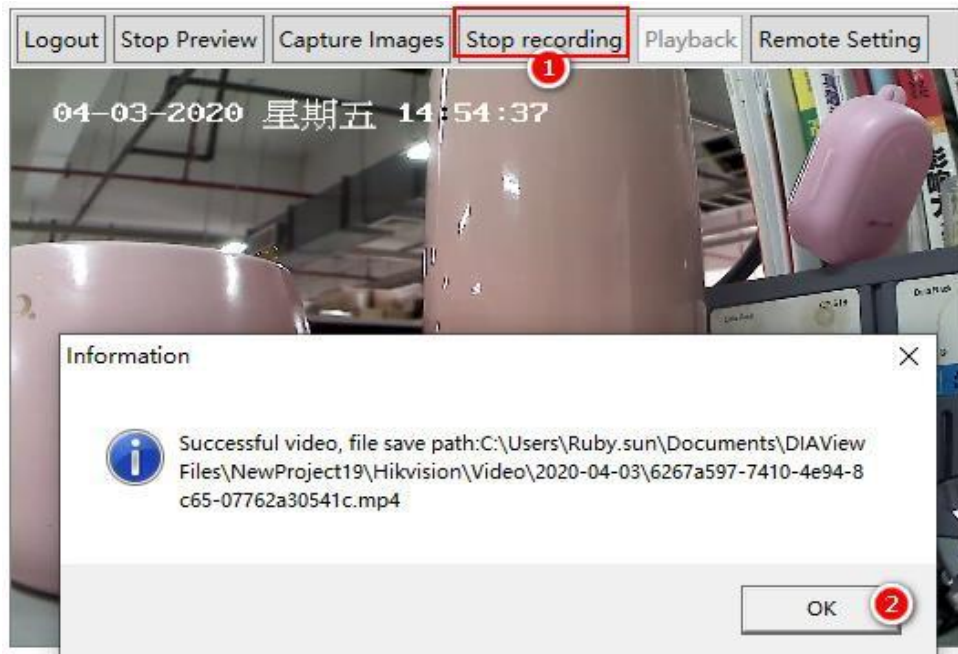
(6) Почати попередній перегляд



(7) Розпочніть запис



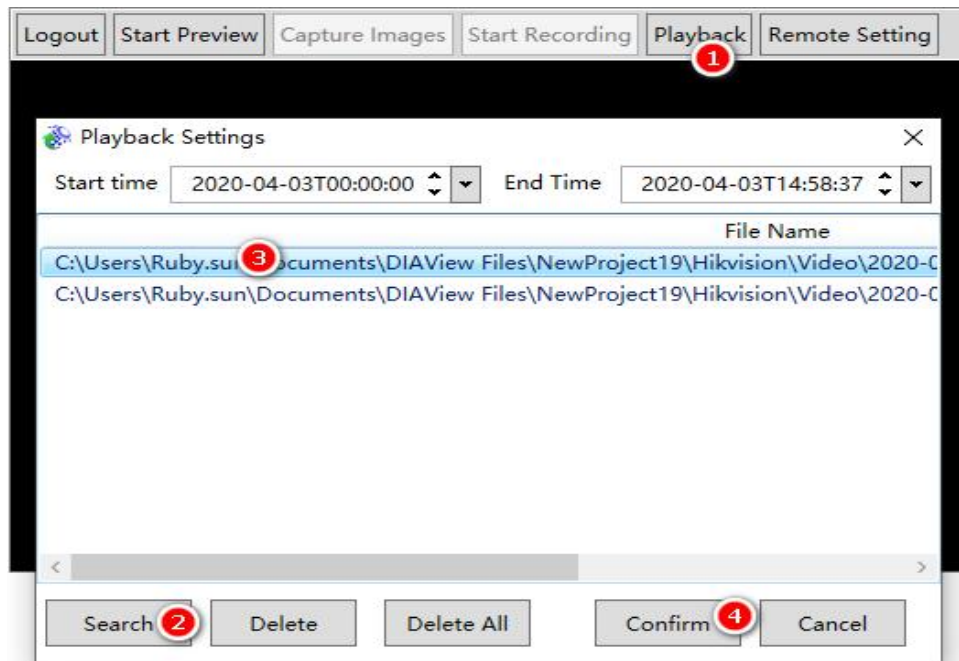
(8) Зупиніть запис, щоб завершити запис і зберегти відео



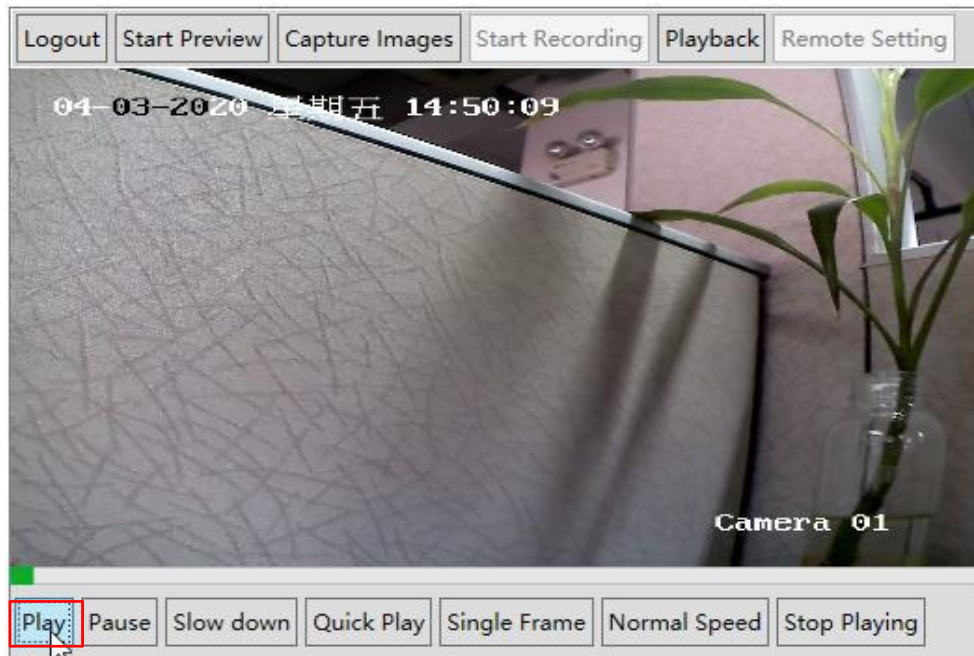
(9) Зупинити попередній перегляд



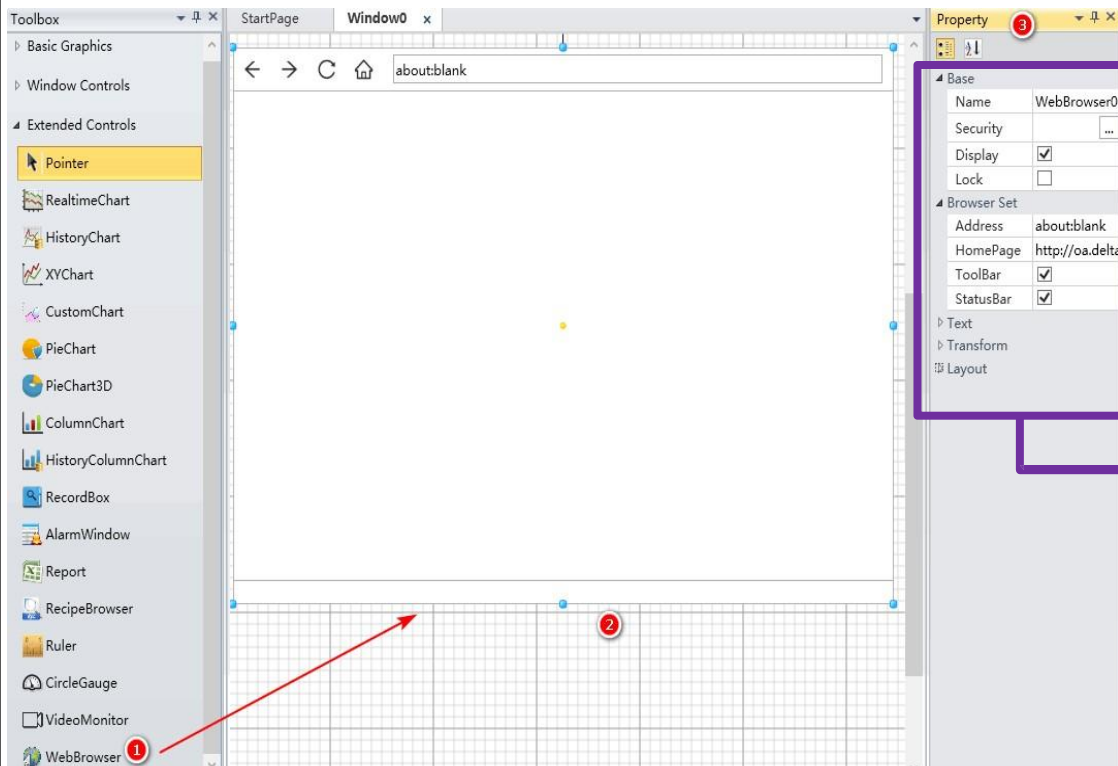
(10) Відтворення відео



(11) Розпочати гру



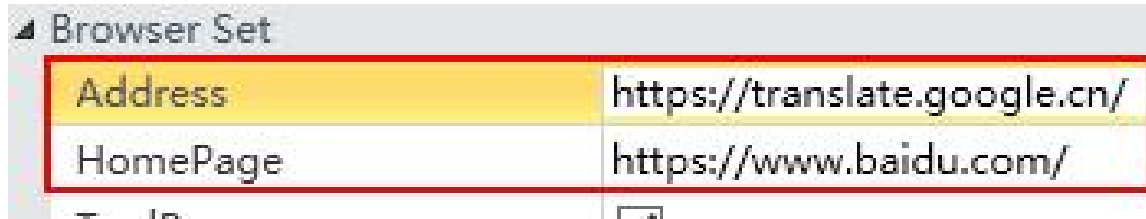
- Створіть WebBrowser0, відкрийте вікно його властивостей



- **Адреса:** установіть поточну адресу веб-браузера
- **Домашня сторінка:** установіть домашню адресу Інтернету браузер
- **Панель інструментів:** установіть, чи відображати панель інструментів
- **Рядок стану:** встановіть, чи відображати рядок стану

- Використання властивостей Address і Homepage :

(1) Створіть WebBrowser0 у Window0 , установіть його адресу та домашню сторінку, як показано на малюнку нижче



Browser Set	
Address	https://translate.google.cn/
HomePage	https://www.baidu.com/

Адреса: <https://translate.google.cn/>

Домашня сторінка: [https:// www.baidu.com/](https://www.baidu.com/)

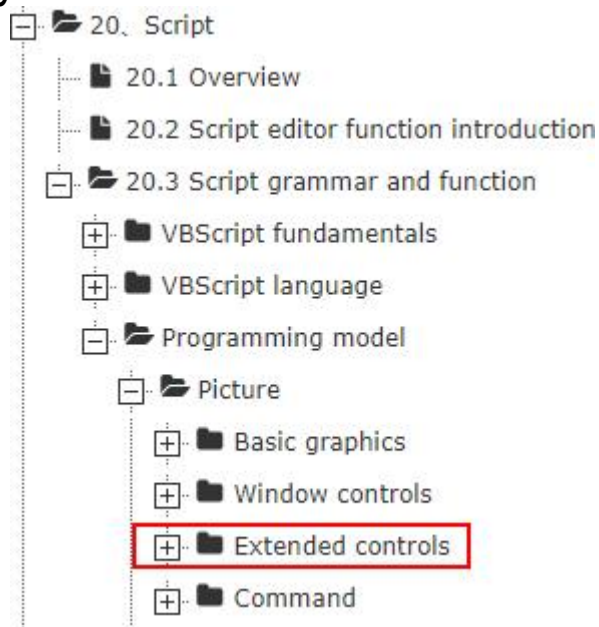
(2) Запустіть проект, початкова сторінка виглядає наступним чином



(3) Натисніть кнопку Домашня сторінка , і з'явиться наступне зображення



Використання сценаріїв для розширених елементів керування див. у розділі « 20.3 Граматика та функції сценаріїв » у посібнику користувача. Як показано на малюнку нижче:



Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta,
відвідайте www.deltaww.com.



DIAView

Галерея

2019/04/12
ЯН'ЯН



- Поняття галереї огляд і класифікація
- Галерея графіки
- Користувачька функція галереї
- Скрипт галереї

У цьому розділі ви дізнаєтеся....

- ... Детальніше про огляд галереї

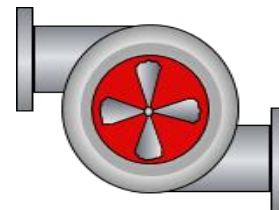
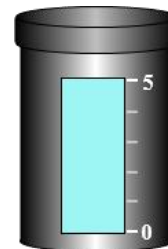
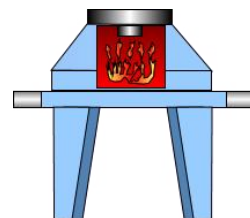
- ... Детальніше про використання галереї

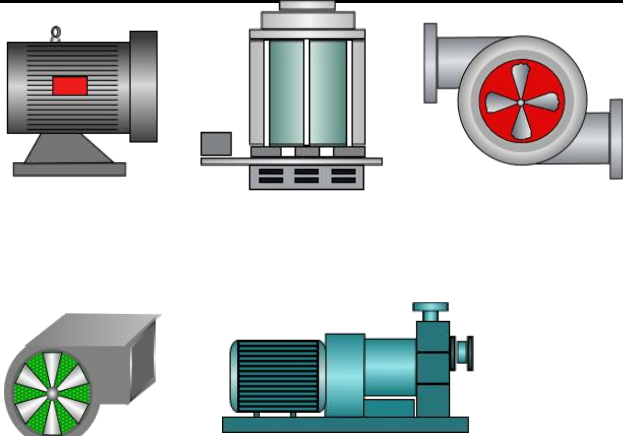
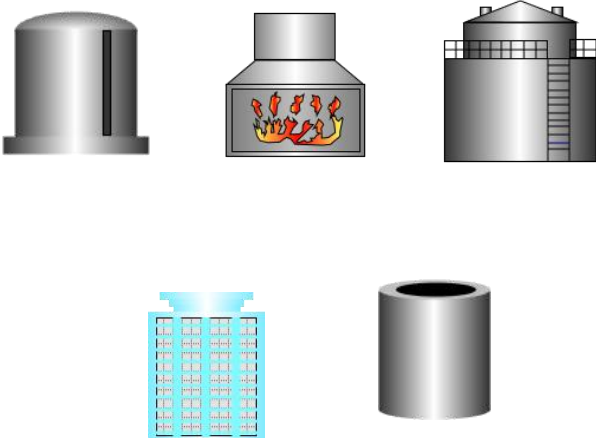
- ... Детальніше про індивідуальне створення галереї

- Класифікація огляду галереї
- Використання галереї
- Користувацька функція галереї
- Сценарій галереї

➤ Що таке Галерея?

- Галерея містить колекцію **графічних моделей, які зазвичай використовуються, і графічних колекцій, визначених користувачем**, які постачаються з програмним забезпеченням конфігурації DIAView, наприклад індикаторів, кнопкових перемикачів, контейнерів клапанів, вентиляторів мотопомпи тощо.
- Користувачі можуть створювати власні часто використовувані графічні моделі та додавати їх до графічної бібліотеки. Більшість графічних анімацій галереї мають підключення до змінних, що дозволяє швидше та зручніше створювати проекти.



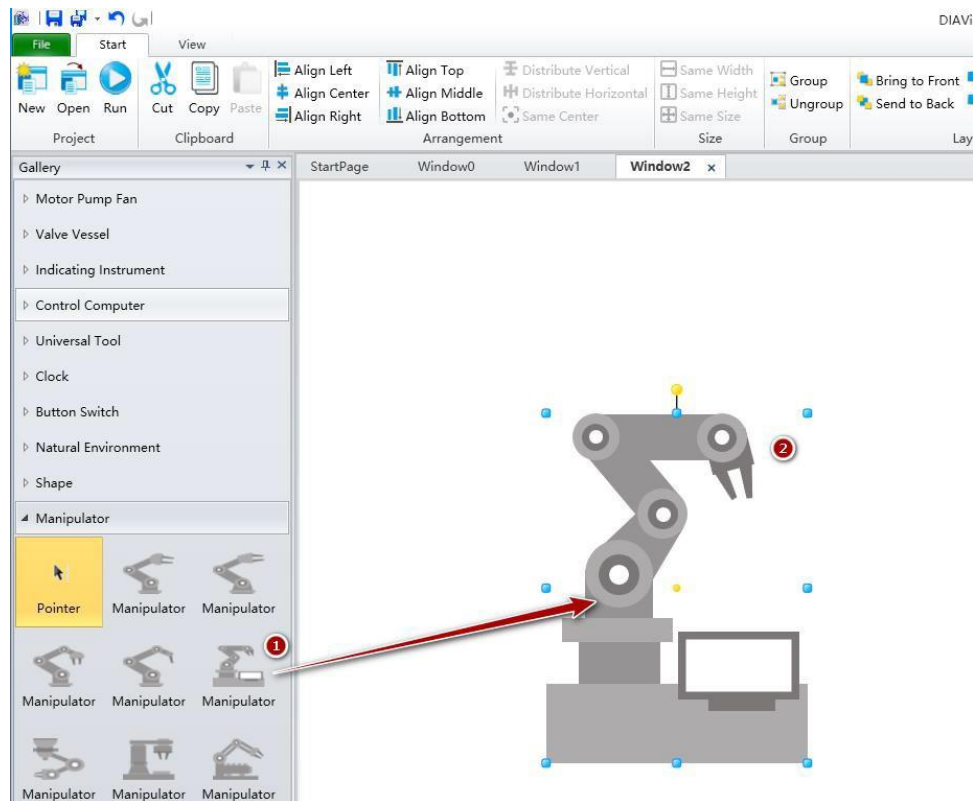
Двигун/Насос/Вентилятор	Клапан/посудина
Вентилятор мотопомпи містить різні типи двигунів, вентиляторів, різні форми лопатей вентилятора, насоси, фільтри, повітряні клапани та іншу графіку.	Контейнер клапана складається з різних типів резервуарів і раковин, клапанів, труб з різними напрямками з'єднання, вигинів, а також нагрівачів і мішалок.
	

- Класифікація огляду галереї
- Використання галереї
- Користувацька функція галереї
- Сценарій галереї

Використання галереї

➤ Приклади використання галереї:

- ① Натисніть на потрібну комбінацію графіка в галереї
- ② Перетягніть графіку до потрібного положення або клацніть положення монтажна область, де ви бажаєте поєднати графіку, а також налаштувати розмір та інші основні властивості об'єднаної графіки

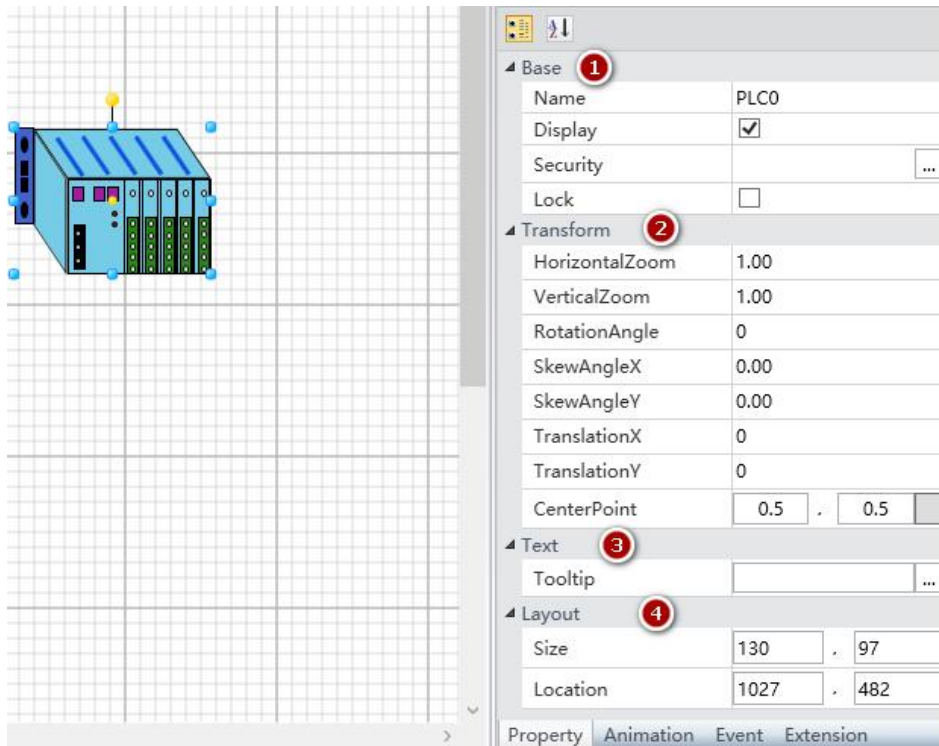


Використання галереї

➤ Приклад налаштування властивостей графіки:

Виберіть графіку в галереї

- ① Основний: містить ім'я, яке можна змінити, чи відображається воно, чи заблоковано від зміни, а також налаштування зони безпеки.
- ② Перетворення: включаючи графічне перетворення, масштабування, викривлення, зсув, кут повороту та центральну точку.
- ③ Текст: Ви можете заповнити мітку підказки
- ④ Макет: тут можна переглянути координати та розмір графіки на полотні або налаштувати координати та розмір графіки.

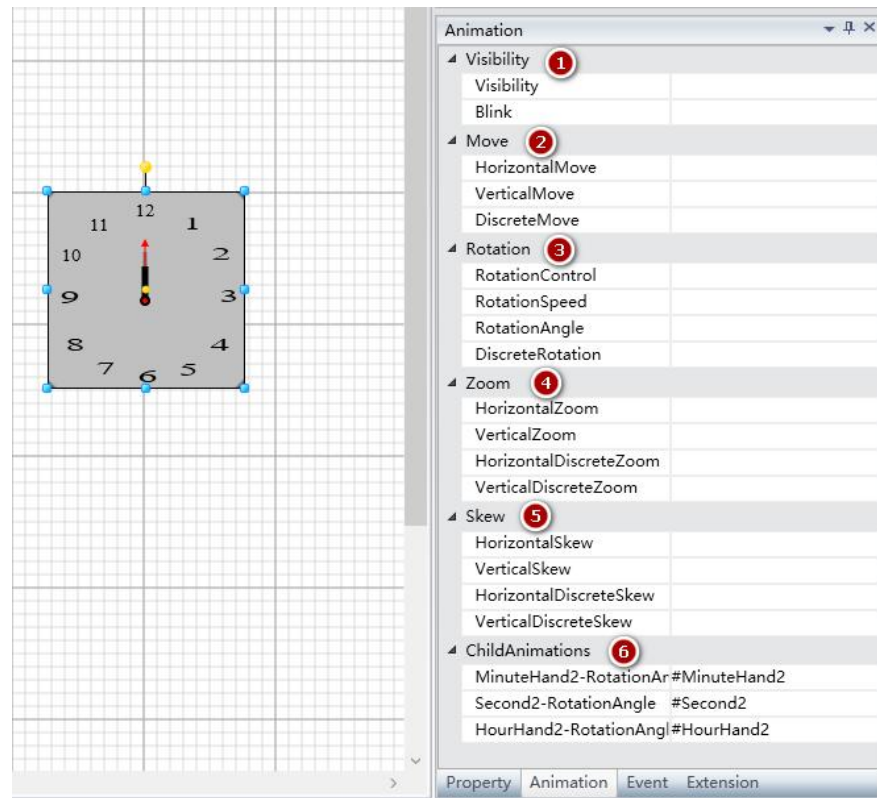


Використання галереї

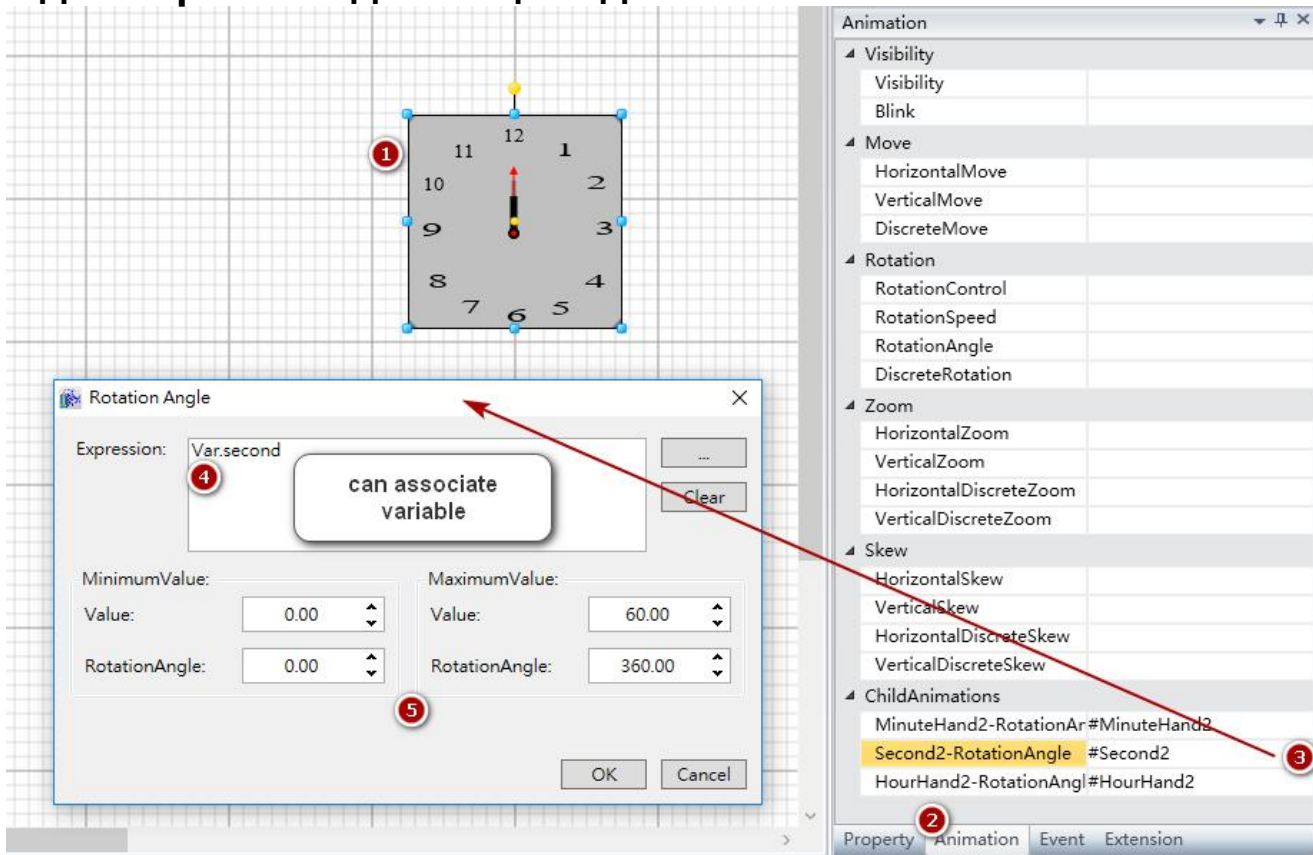
➤ Приклади графічної анімації:

Виберіть графіку в галереї

- ① **Видимість:** ви можете вибрати, чи буде це видимо або блимає.
- ② **Переміщення:** Ви можете вибрати напрямок руху графіка.
- ③ **Обертання:** включає початок і зупинку обертання, швидкість, кут і дискретне обертання
- ④ **Масштаб:** включає горизонтальне, вертикальне та дискретне масштабування.
- ⑤ **Нахил:** виберіть напрямок нахилу, включаючи горизонтальний, вертикальний і дискретний нахил.
- ⑥ **Субграфічна анімація:** субграфічна анімація, що відповідає графікам у різних галереях, відрізняється, наприклад, анімація обертання вказівника, що відповідає годинам, хвилинам і секундам годинника на малюнку.



➤ Приклад створення піданімації годинника:



The image illustrates the process of creating an animation for a clock face. A clock face is shown on a grid. A dialog box titled "Rotation Angle" is open, showing the expression "Var.second" (labeled 4) and a range from 0.00 to 60.00 (labeled 5). A red arrow points from the "Second2-RotationAngle" entry in the "ChildAnimations" list (labeled 3) to the dialog box. The "Animation" panel on the right shows a tree of animation properties, including Visibility, Move, Rotation, Zoom, Skew, and ChildAnimations. The "ChildAnimations" list includes "MinuteHand2-RotationAr #MinuteHand2", "Second2-RotationAngle #Second2" (highlighted), and "HourHand2-RotationAngl #HourHand2". The "Animation" panel also has tabs for Property, Animation, Event, and Extension (labeled 2).

Rotation Angle

Expression: Var.second (4)

can associate variable

Clear

MinimumValue: Value: 0.00

MaximumValue: Value: 60.00

RotationAngle: 0.00

RotationAngle: 360.00 (5)

OK Cancel

Animation

- Visibility
 - Visibility
 - Blink
- Move
 - HorizontalMove
 - VerticalMove
 - DiscreteMove
- Rotation
 - RotationControl
 - RotationSpeed
 - RotationAngle
 - DiscreteRotation
- Zoom
 - HorizontalZoom
 - VerticalZoom
 - HorizontalDiscreteZoom
 - VerticalDiscreteZoom
- Skew
 - HorizontalSkew
 - VerticalSkew
 - HorizontalDiscreteSkew
 - VerticalDiscreteSkew
- ChildAnimations
 - MinuteHand2-RotationAr #MinuteHand2
 - Second2-RotationAngle #Second2 (3)
 - HourHand2-RotationAngl #HourHand2

Property Animation (2) Event Extension

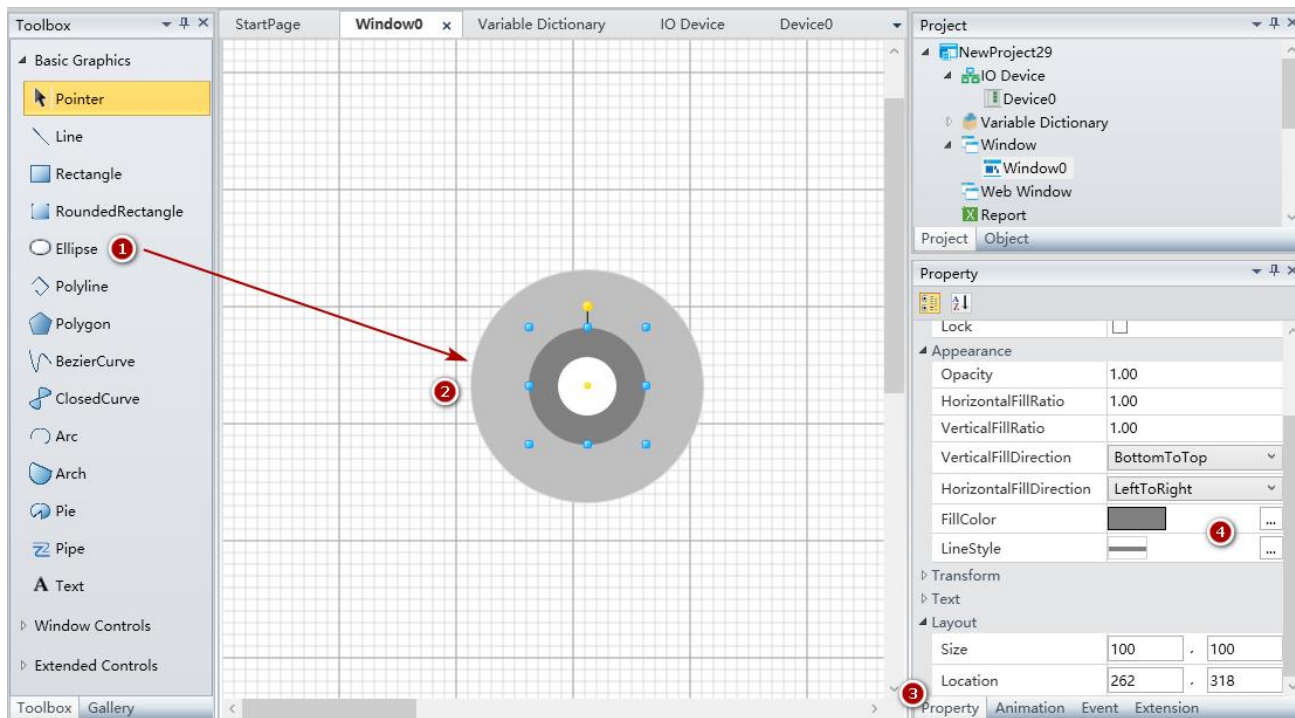
- Класифікація огляду галереї
- Використання галереї
- Користувацька функція галереї
- Сценарій галереї

Користувачка функція галереї

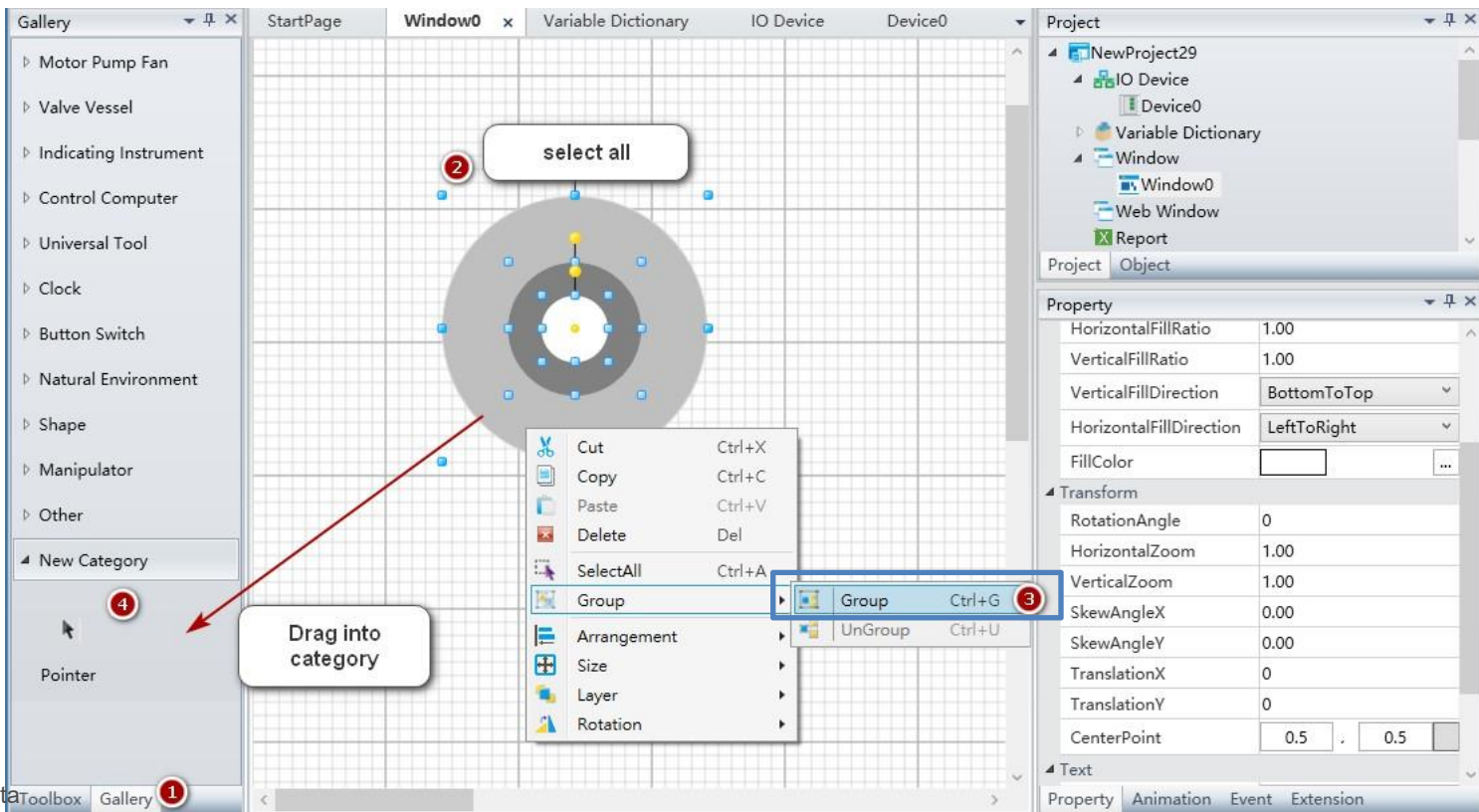
➤ Що таке спеціальна функція Галереї?

Користувачка функція галереї є характерною функцією галереї, і користувачі можуть додавати галерею графіки відповідно до власних звичок.

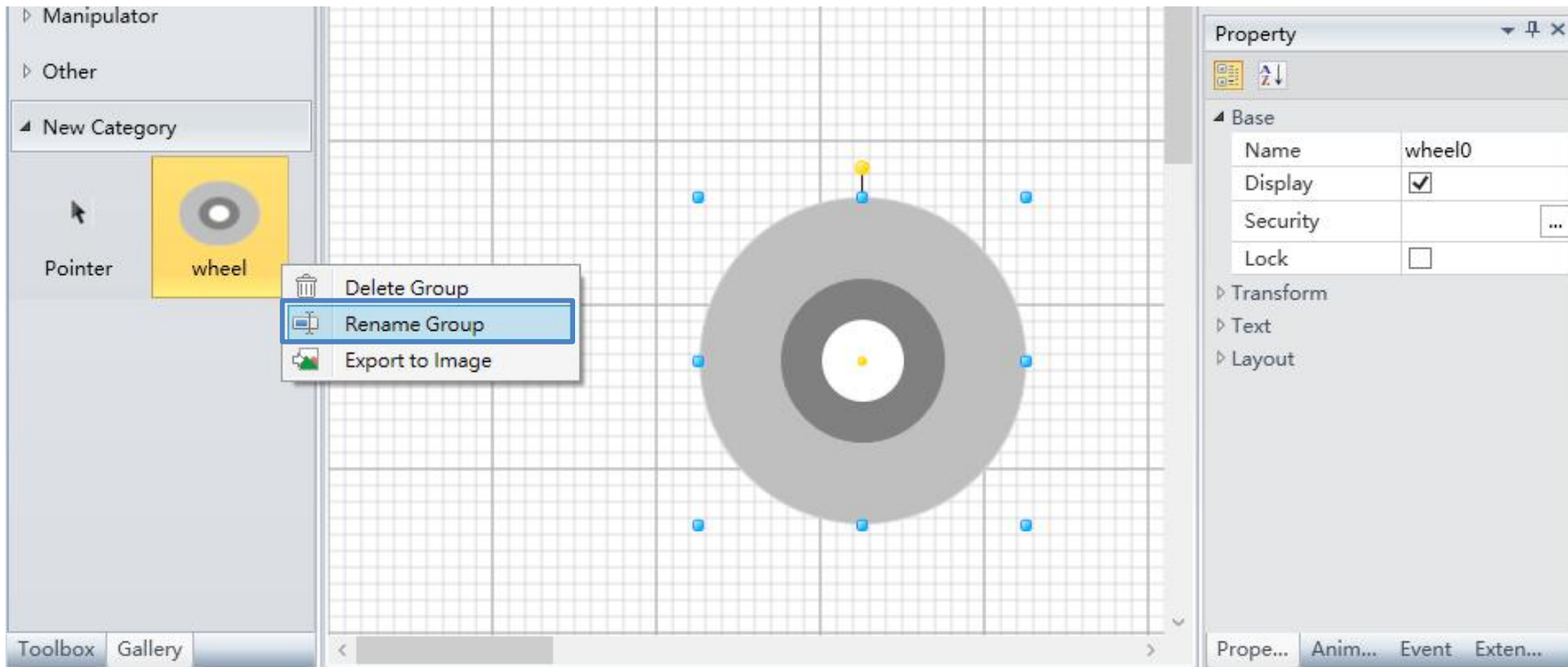
(1) Намалюйте графіку та змініть властивості.



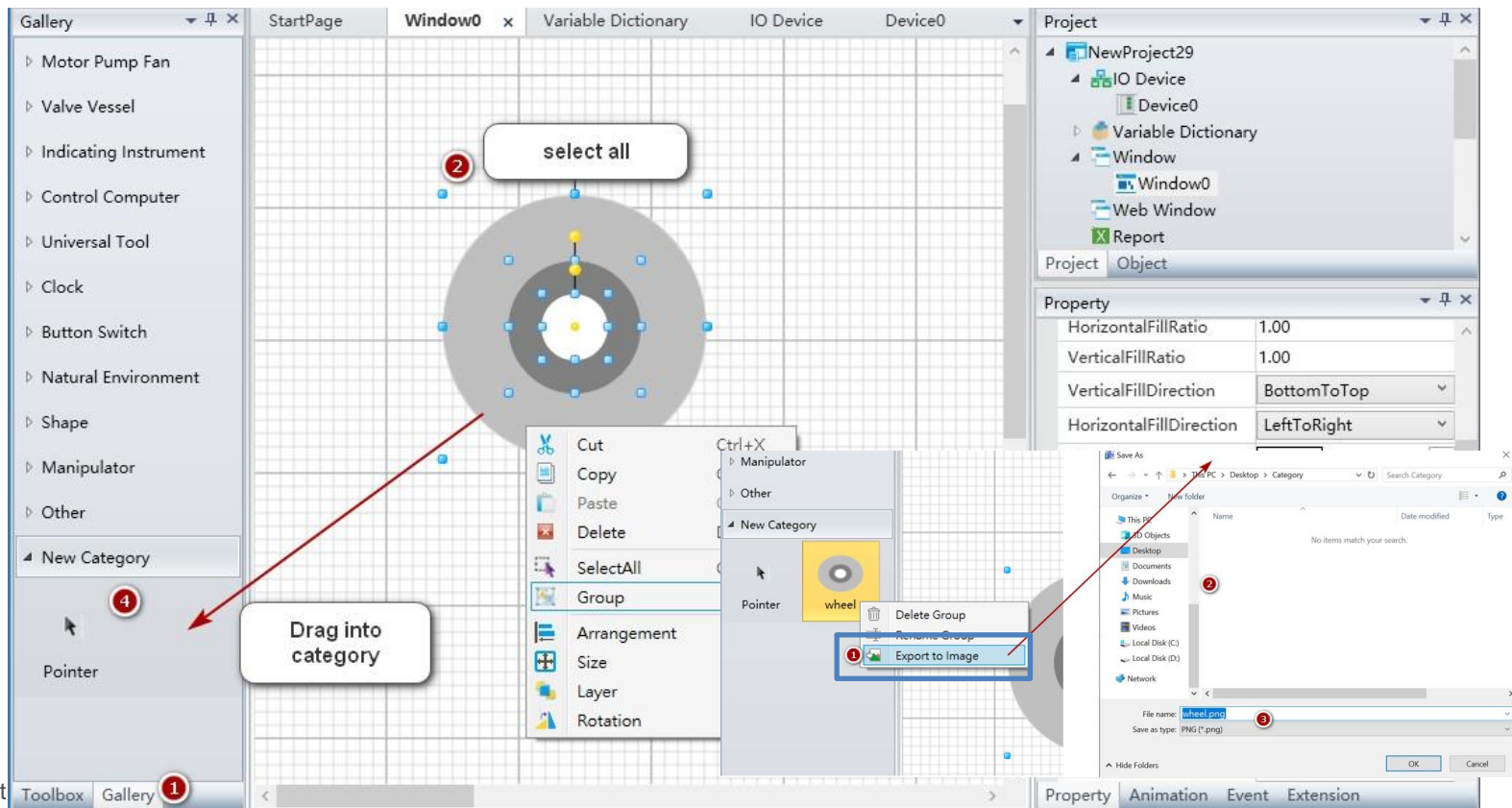
(2) Об'єднайте графіку та перетягніть її в галерею.



(3) Перейменуйте об'єднану графіку



(4) Експорт до зображення



The screenshot illustrates the process of exporting a diagram to an image. The interface includes a Gallery on the left, a main workspace, and a Project/Property panel on the right.

Steps:

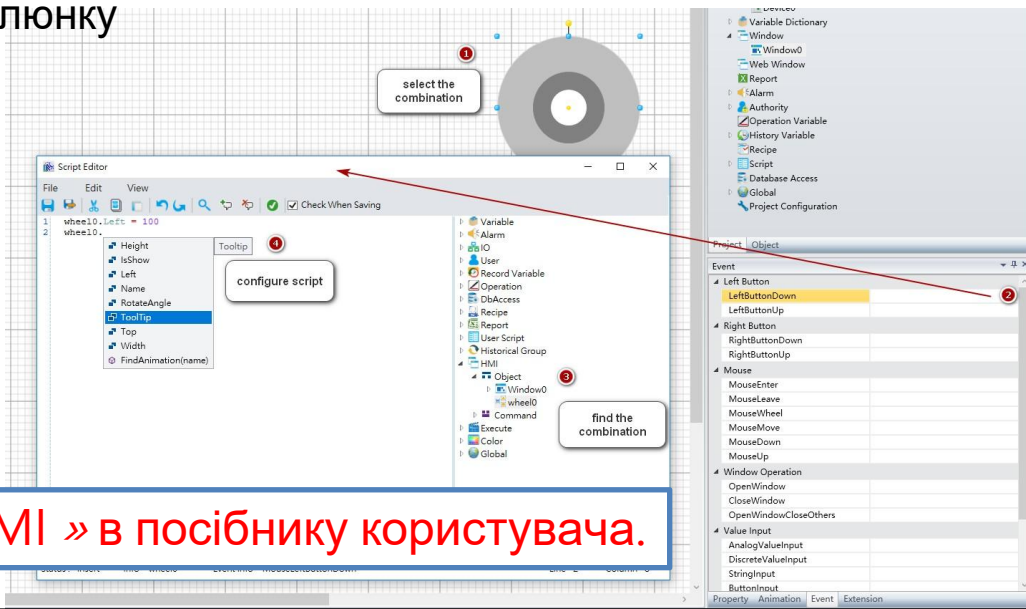
- Click **Export to Image** in the context menu.
- Click **wheel.png** in the **Save As** dialog.
- Click **OK** in the **Save As** dialog.
- Drag the **Pointer** object from the **New Category** in the Gallery to the workspace.

Property Panel:

Property	Value
HorizontalFillRatio	1.00
VerticalFillRatio	1.00
VerticalFillDirection	BottomToTop
HorizontalFillDirection	LeftToRight

- Класифікація огляду галереї
- Використання галереї
- Користувацька функція галереї
- Сценарій галереї

- Графіка галереї — це лише векторна графіка, анімацію властивостей можна налаштувати безпосередньо, за нормального використання не потрібно додавати події сценарію.
 - Графіка бібліотеки така сама, як і сценарії базових графічних методів. Ви можете безпосередньо звернутися до основної графіки.
 - Якщо вам потрібно додати сценарій змінення атрибутів у проект, виконайте кроки, як показано на малюнку



✗ Див. розділ « 20.3 Група НМІ » в посібнику користувача.

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

сигналізація

朱明杰 杨阳

20.03.2020



- Огляд сигналізації
- Використання вікна тривоги
- Інструмент керування сигналізацією
- Параметри конфігурації сигналізації
- Функція клацання правою кнопкою миші групи тривог

призначення

У цьому розділі ви дізнаєтеся...

- ... Основні дії та керування сигналізацією

- ... Функція тривожної групи

- Огляд сигналізації
- Використання вікна тривоги
- Інструмент керування сигналізацією
- Параметри конфігурації сигналізації
- Функція клацання правою кнопкою миші групи тривог

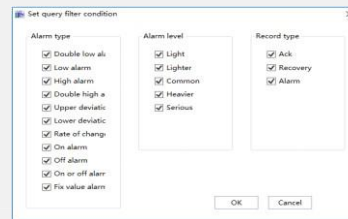
➤ Функція DIAView Alarm

- Будильник – це нагадування, коли пов'язані з системою змінні дані перевищують значення попередження, щоб досягнення раннього попередження про небезпеку у виробничому процесі та запобігання виходу з ладу обладнання.
- Сигналізація відіграє незамінну роль у безпеці системи управління промисловою автоматикою системи.



Функція фільтрації вікна тривоги:

Якщо користувач бажає переглянути запис нагадування з найвищим рівнем нагадування, вам потрібно лише позначити опцію серйозний на сторінці конфігурації фільтра та зняти позначки з інших пунктів. .



Сторінка конфігурації фільтрації вікна тривоги

➤ Надійний інструмент керування сигналізацією:

- Вікно тривоги записує назву тривоги, змінний шлях, час ініціювання, час відповіді, час відновлення, тип запису, тип тривоги та дані тривоги про рівень тривоги, а також може здійснювати фільтрування, групування запиту даних, відображення налаштувань, збереження, друк і експортувати дані у звіт або Excel тощо.

History Alarm Real Time Alarm

Ack Select All

Alarm name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level
AlarmVariable4	Var.Variable4	8/9/2017 9:39:53 AM		8/9/2017 9:41:44 AM	Recovery	LowAlarm	Serious
2 AlarmVariable2	Var.Variable2	8/9/2017 9:41:34 AM			Alarm	DoubleHighAlarm	Common
3 AlarmVariable11	Var.Variable11	8/9/2017 9:39:17 AM		8/9/2017 9:40:38 AM	Recovery	DisAlarm	Lighter
AlarmVariable	Var.Variable	8/9/2017 9:39:27 AM	8/9/2017 9:40:38 AM		Ack	UpperDeviationAlarm	Light
5 AlarmVariable1	Var.Variable1	8/9/2017 9:39:41 AM			Alarm	LowDeviationAlarm	Lighter
6 AlarmVariable10	Var.Variable10	8/9/2017 9:39:18 AM			Alarm	OffAlarm	Light

Вікна
сигналізації

Set query filter condition

Alarm Type

- ☒ DoubleLowAlarm
- ☒ LowAlarm
- ☒ HighAlarm
- ☒ DoubleHighAlarm
- ☒ UpperDeviationAlarm
- ☒ LowDeviationAlarm
- ☒ RateOfChangeAlarm
- ☒ OnAlarm
- ☒ OffAlarm
- ☒ OnOrOffAlarm
- ☒ Fix value alarm

Alarm Level

- ☒ Light
- ☒ Lighter
- ☒ Common
- ☒ Heavier
- ☒ Serious

Record Type

- ☒ Ack
- ☒ Recovery
- ☒ Alarm

Alarm Group Filter

- ☒ Alarm
 - ☒ AlarmGroup0
 - ☒ AlarmGroup1
 - ☒ AlarmGroup2
 - ☒ AlarmGroup3

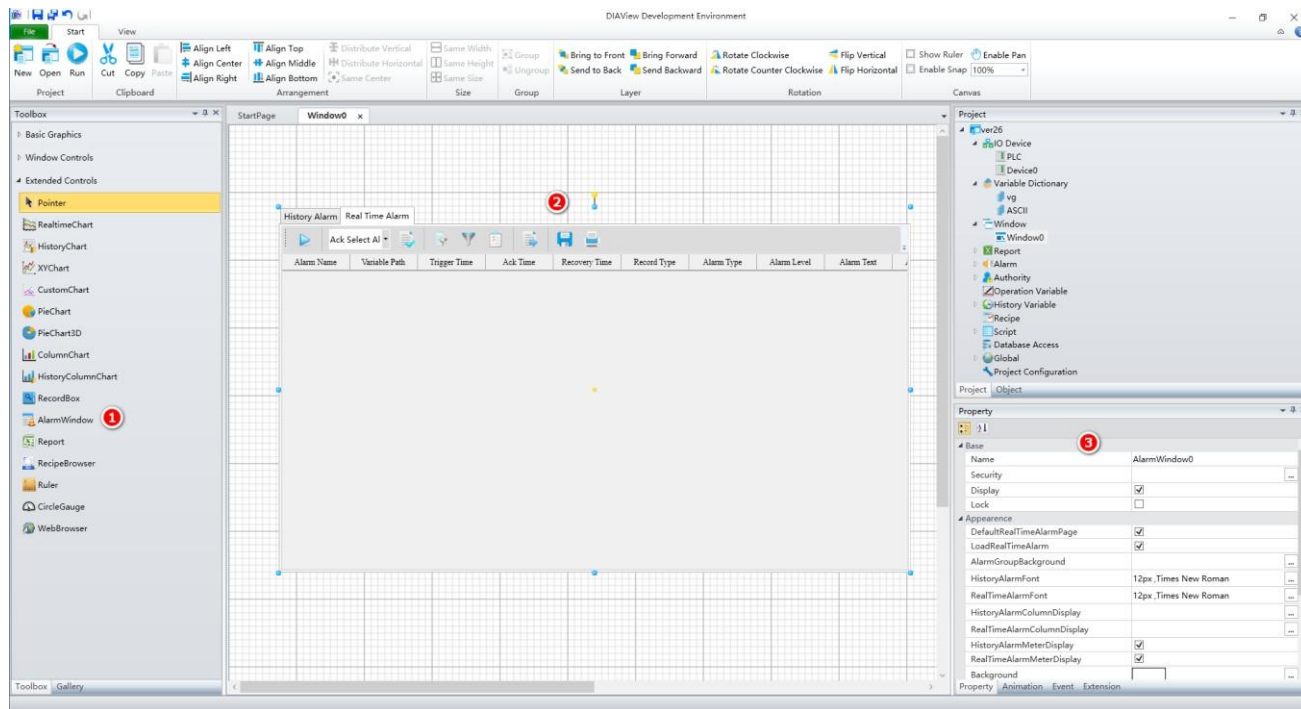
Фільтрація запитів

Групи сигналізації фільтрація

- Огляд сигналізації
- Використання вікна тривоги
- Інструмент керування сигналізацією
- Параметри конфігурації сигналізації
- Функція клацання правою кнопкою миші групи тривог

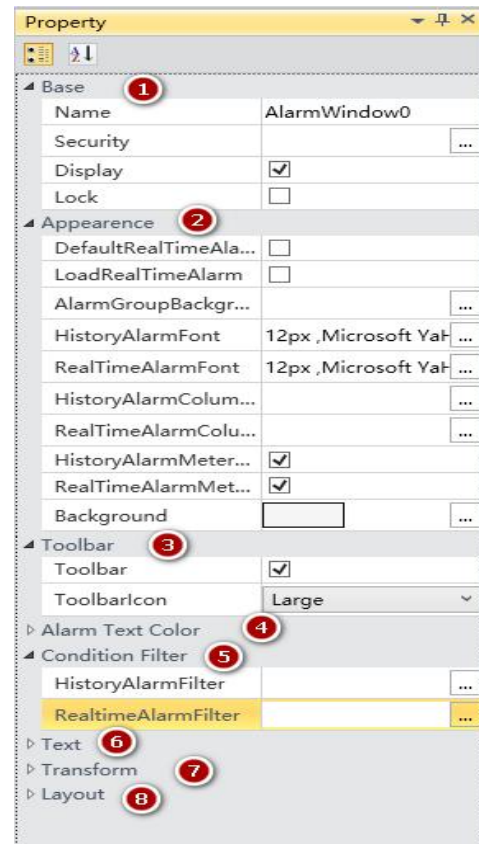
➤ Створіть вікно тривоги

- ① Натисніть інструмент, щоб налаштувати вікно будильника
- ② Перетягніть до потрібного положення на полотні
- ③ Налаштувати будильник властивості вікна



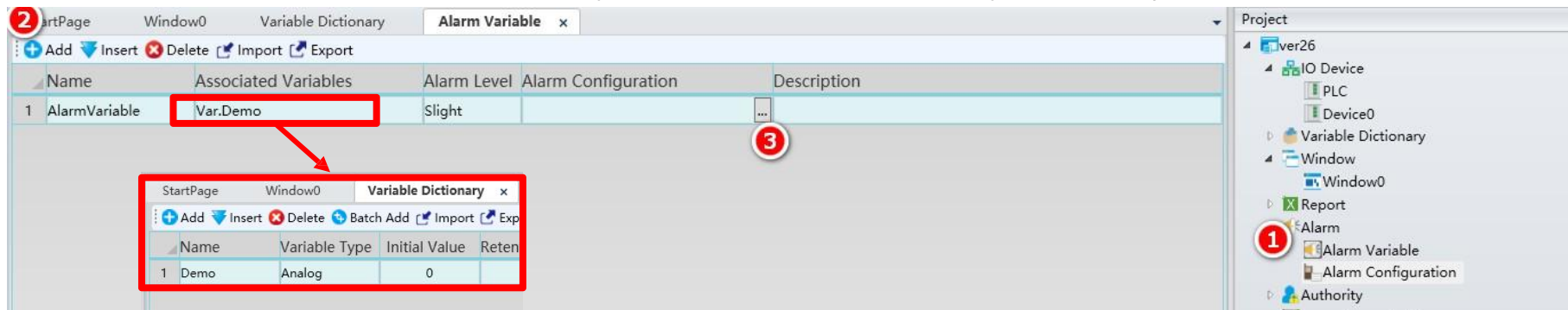
➤ Властивості сигналізації

- ① **Ім'я:** основні властивості, такі ж, як і в інших елементів управління
- ② **Зовнішній вигляд:** Зовнішній вигляд вікна нагадування включає налаштування сторінки нагадування в режимі реального часу для відображення за замовчуванням, завантаження даних нагадування в реальному часі, налаштування фоновому кольору групи нагадувань, формат шрифту даних нагадування в реальному часі та історії, фільтрування та фільтрування даних у реальному часі та історії групи нагадувань, чи відображати стовпці нагадувань у реальному часі та хронологію та колір фону нагадувань
- ③ **Панель інструментів:** панель інструментів визначає, чи потрібно відображати панель інструментів, і розмір панелі інструментів
- ④ **Налаштування кольору будильника:** необхідний колір можна встановити відповідно до різних види сигналізації
- ⑤ **Фільтр умов:** дані тривоги в режимі реального часу та історичні дані відповідно до типу
- ⑥ **Текст:** вікно тривоги запитує текстову інформацію
- ⑦ **Трансформація:** містить центральну точку елемента керування вікном тривоги
- ⑧ **Макет:** Розмір і положення вікна тривоги на полотні можна налаштувати за допомогою числового значення



➤ Створіть змінну будильника

- ① Виберіть кнопку « **Alarm Variable** ».
- ② Створення змінних сигналізації: натисніть кнопку « **Додати** » у вікні, щоб додати змінні сигналізації; система дасть попередньо встановлену назву
- ③ **Асоційована змінна**: натисніть кнопку в комірці зберігання змінних і виберіть систему змінна для зв'язування (тільки змінні типу аналогового значення та цифрового значення можна використовувати для встановлення будильників)



The screenshot shows the Delta software interface. The main window is titled 'Alarm Variable' and contains a table with the following data:

Name	Associated Variables	Alarm Level	Alarm Configuration	Description
1 AlarmVariable	Var.Demo	Slight		

The 'Associated Variables' column is highlighted with a red box, and a red arrow points to a smaller window titled 'Variable Dictionary'. This window contains a table with the following data:

Name	Variable Type	Initial Value	Reten
1 Demo	Analog	0	

The 'Variable Dictionary' window is also highlighted with a red box. The Project tree on the right shows the hierarchy: Project > ver26 > IO Device > PLC > Device0 > Variable Dictionary > Window > Window0 > Report > Alarm > Alarm Variable (highlighted with a red circle 1) > Alarm Configuration > Authority.

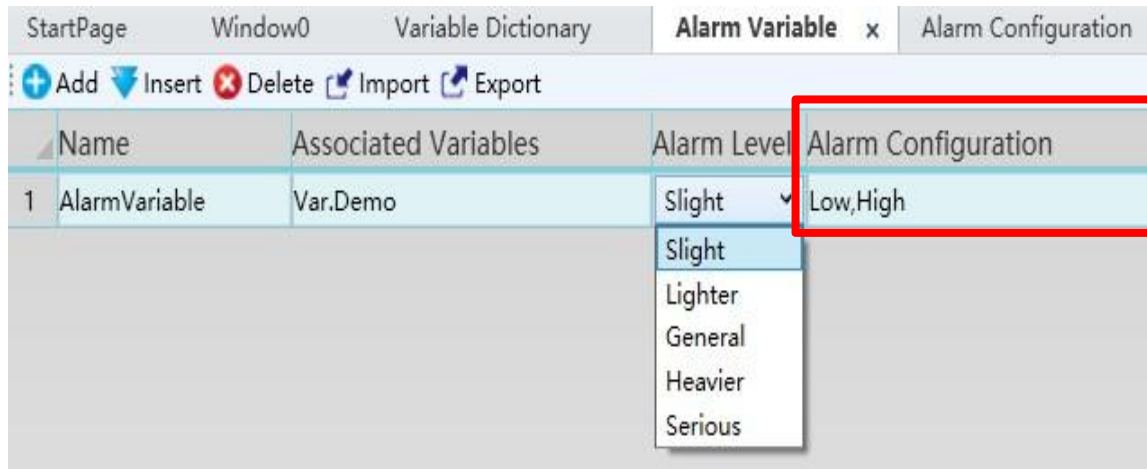
Використання вікна тривоги

➤ Рівень тривоги:

- Натисніть кнопку, і з'явиться меню для вибору рівня.

➤ Конфігурація сигналізації:

- Натисніть кнопку в комірці збереження конфігурації сигналізації, і з'явиться вікно конфігурації сигналізації; система виконуватиме відповідні конфігурації відповідно до типу змінної



Використання вікна тривоги

➤ Конфігурація сигналізації:

① Загальний сигнал тривоги:

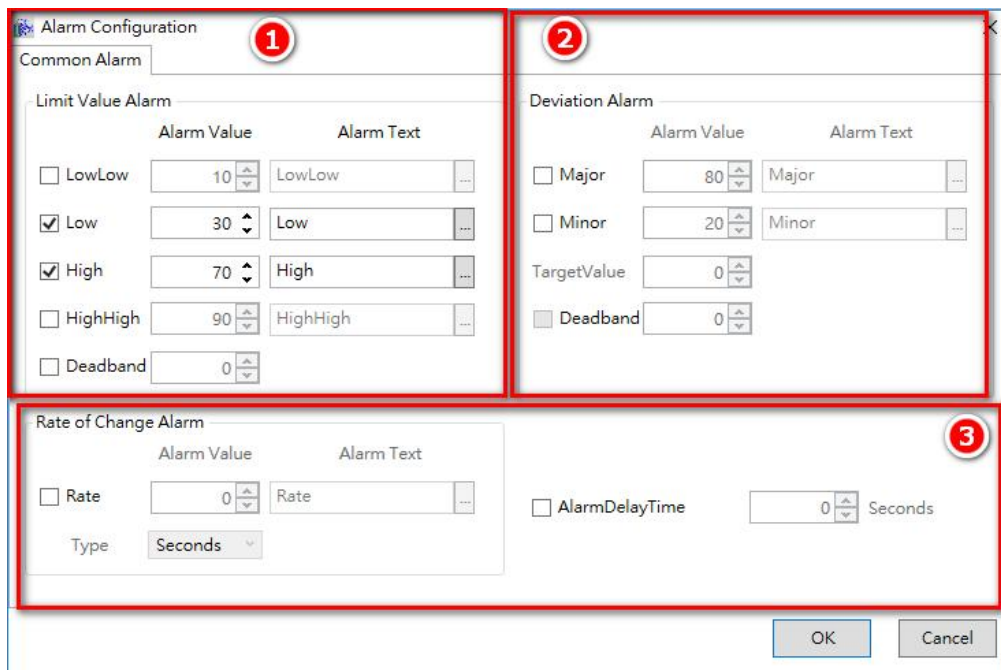
Сигналізація генерується, коли значення аналогового значення перевищує встановлене значення порогу тривоги.

② Сигналізація відхилення:

Це сигнал тривоги, який виникає, коли відношення аналогового значення до коливання цільового значення відхилення перевищує встановлений діапазон зміни.

③ Сигналізація швидкості зміни:

Це аварійний сигнал, який виникає, коли швидкість зміни аналогового значення перевищує значення, встановлене для періоду, що означає, що аварійний сигнал, який виникає, коли значення змінної змінюється занадто швидко.



The image shows the 'Alarm Configuration' dialog box with three sections highlighted by red boxes and numbered 1, 2, and 3.

Section 1: Common Alarm

Limit Value Alarm

	Alarm Value	Alarm Text
☐ LowLow	10	LowLow
☒ Low	30	Low
☒ High	70	High
☐ HighHigh	90	HighHigh
☐ Deadband	0	

Section 2: Deviation Alarm

	Alarm Value	Alarm Text
☐ Major	80	Major
☐ Minor	20	Minor

TargetValue: 0

☐ Deadband: 0

Section 3: Rate of Change Alarm

	Alarm Value	Alarm Text
☐ Rate	0	Rate

Type: Seconds

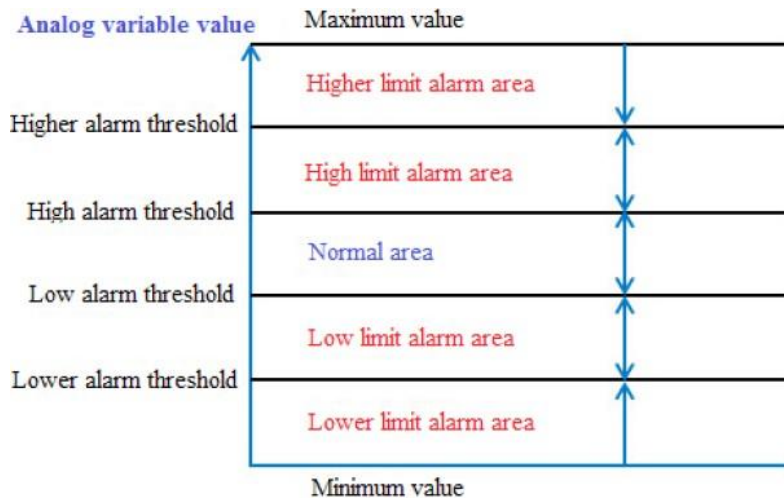
☐ AlarmDelayTime: 0 Seconds

OK Cancel

Метод розрахунку швидкості зміни
наведено в посібнику користувача 11.3

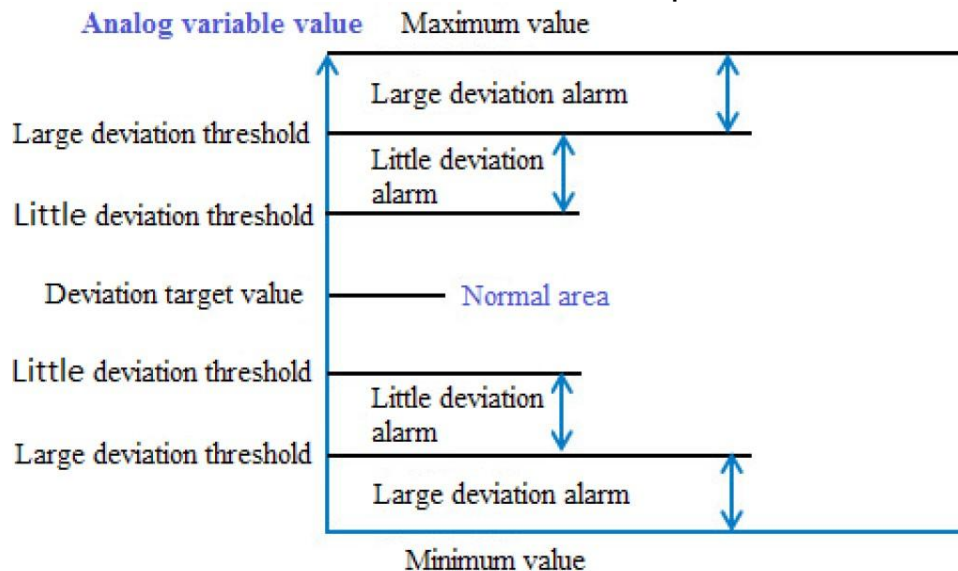
➤ Сигналізація перевищення:

- Існує 4 пороги тривоги для сигналізації про перевищення ліміту: **надзвичайно низький, низький, високий, надзвичайно високий**.
- Порогове значення тривоги має бути встановлено між максимальним і мінімальним значенням змінної.
- Поріг тривоги та принципи, як показано на малюнку нижче:



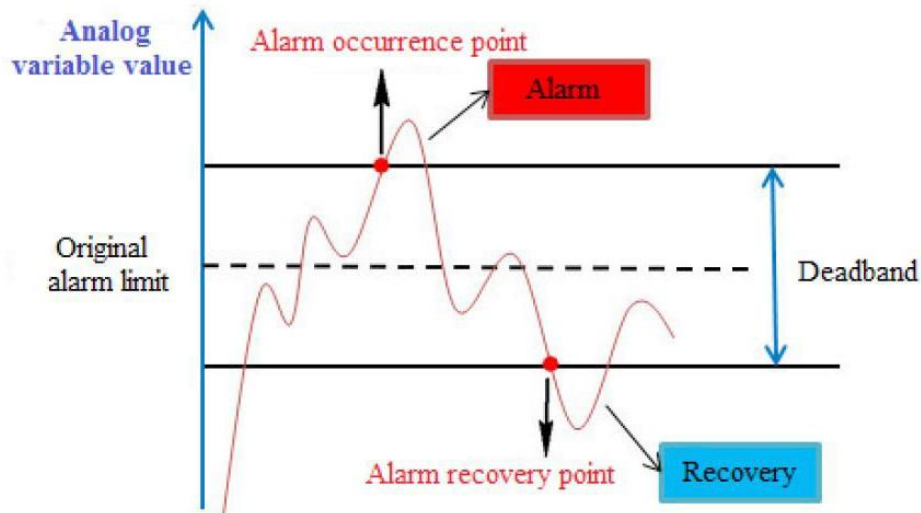
➤ Сигналізація відхилення:

- Сигналізація відхилення поділяється на два типи: **Велике відхилення** та **Мале відхилення**.
- Коли значення коливань перевищує діапазон великого/малого відхилення, спрацьовують сигнали тривоги великого відхилення та сигнали тривоги малого відхилення відповідно. Крім того, для кожної змінної з'явиться лише один сигнал тривоги відхилення.



➤ Мертва зона:

- Ефект зони нечутливості полягає в тому, щоб запобігти створенню нереалістичних сигналів тривоги, коли значення змінної часто змінюється між початковим максимальним і мінімальним порогом тривоги шляхом додавання порогового значення для максимальної та мінімальної межі сигналу тривоги, змінюючи вихідний поріг тривоги з один рядок до порогової смуги тривоги.
- Принципи мертвої зони сигналізації показані на малюнку нижче:



➤ Приклад конфігурації сигналізації:

- ① Ми встановлюємо значення, яке контролюється конфігурацією тривоги, між 30 і 70.
- ② Таким чином, коли значення дорівнює 25, він показує низький рівень тривоги, а коли значення піднімається до 80, він відновлюється зі статусу низького сигналу. Але це понад 70, тож перейдіть до високого статусу тривоги.
- ③ Нарешті, коли значення повертається до 50, всі тривоги зникають.

①

25.00

History Alarm

Real Time Alarm

...		Ack Select All						
Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	
Alarm AlarmVariable	Var Demo	4/23/2019 9:42:11 AM			Alarm	Low Alarm	Slight	

②

80.00

History Alarm

Real Time Alarm

...		Ack Select All						
Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	
Alarm AlarmVariable	Var Demo	4/23/2019 9:46:18 AM			Alarm	High Alarm	Slight	
Alarm AlarmVariable	Var Demo	4/23/2019 9:42:11 AM		4/23/2019 9:46:18 AM	Recovery	Low Alarm	Slight	

③

50.00

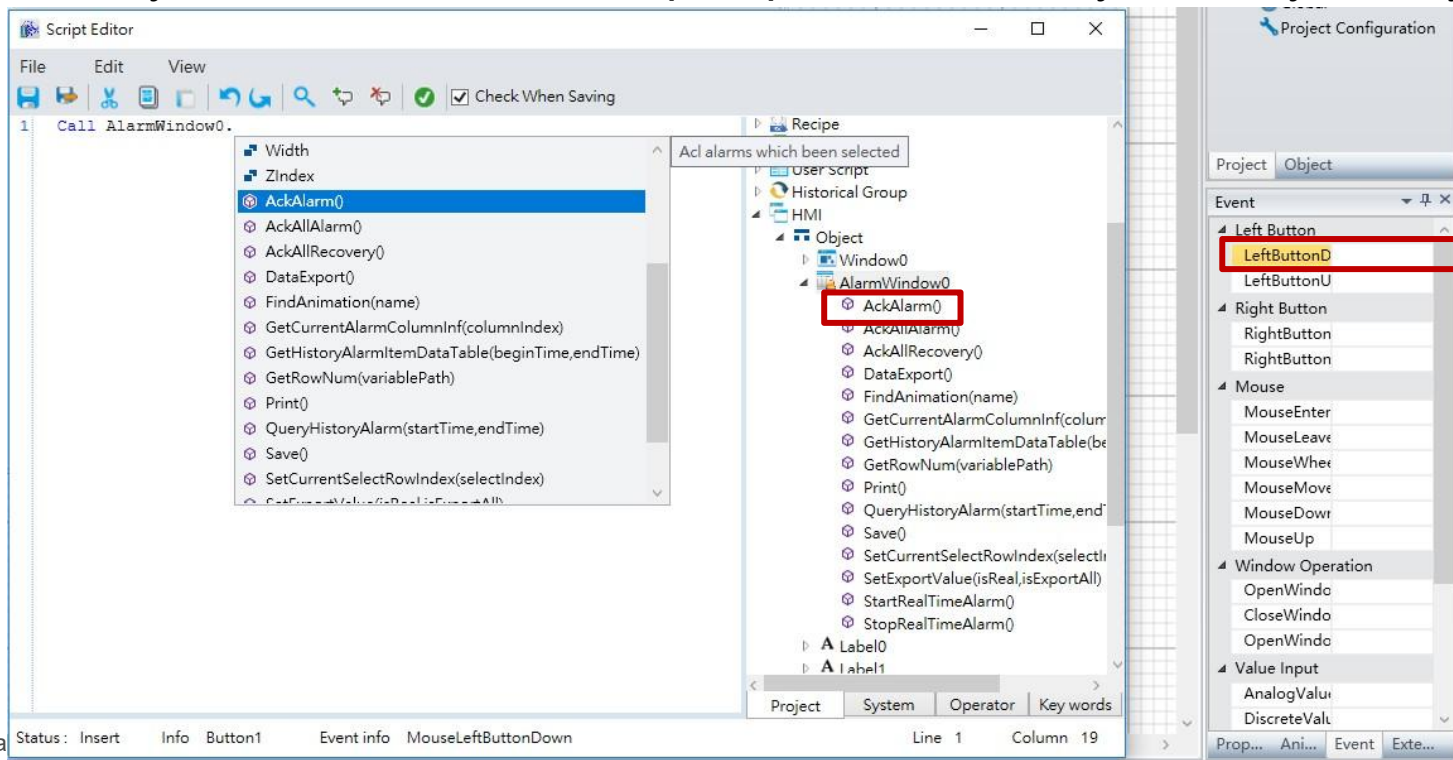
History Alarm

Real Time Alarm

...		Ack Select All						
Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	
Alarm AlarmVariable	Var Demo	4/23/2019 9:46:18 AM		4/23/2019 9:47:21 AM	Recovery	High Alarm	Slight	
Alarm AlarmVariable	Var Demo	4/23/2019 9:42:11 AM		4/23/2019 9:46:18 AM	Recovery	Low Alarm	Slight	

➤ Приклад сценарію Ack an alarm button :

① Налаштуйте відповідний сценарій тривоги, натиснувши ліву кнопку



➤ Приклад сценарію Ack an alarm button :

② Викликати тривогу під час роботи

History Alarm Real Time Alarm									
⏸ Ack Selected 📄 ⚡ 🔍 🔊 📅 🔄 💾 🖨									
Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	Alarm Text	A
Alarm.Low	Var.Variable	7/2/2020 11:17:5	1/1/0001 12:00:0	1/1/0001 12:00:0	Alarm	Low Alarm	Slight	Low	
Alarm.high	Var.Variable	7/2/2020 11:17:5	1/1/0001 12:00:0	7/2/2020 11:17:5	Recovery	High Alarm	Heavier	High	
Alarm.Low	Var.Variable	7/2/2020 11:17:4	1/1/0001 12:00:0	7/2/2020 11:17:5	Recovery	Low Alarm	Slight	Low	
Alarm.high	Var.Variable	7/2/2020 11:17:4	1/1/0001 12:00:0	7/2/2020 11:17:4	Recovery	High Alarm	Heavier	High	
Alarm.high1	Var.Variable1	7/2/2020 11:17:4	1/1/0001 12:00:0	1/1/0001 12:00:0	Alarm	LowLowAlarm	Heavier	LowLow	
Alarm.Low	Var.Variable	7/2/2020 11:17:4	1/1/0001 12:00:0	7/2/2020 11:17:4	Recovery	Low Alarm	Slight	Low	
Alarm.high	Var.Variable	7/2/2020 11:17:3	1/1/0001 12:00:0	7/2/2020 11:17:4	Recovery	High Alarm	Heavier	High	
Alarm.high	Var.Variable	7/2/2020 11:17:3	1/1/0001 12:00:0	7/2/2020 11:17:3	Recovery	High Alarm	Heavier	High	
Alarm.Low	Var.Variable	7/2/2020 11:17:3	1/1/0001 12:00:0	7/2/2020 11:17:3	Recovery	Low Alarm	Slight	Low	
Alarm.Low	Var.Variable	7/2/2020 11:17:2		7/2/2020 11:17:3	Recovery	Low Alarm	Slight	Low	

➤ Приклад сценарію Ack an alarm button :

③ Проект у виконанні

① Виберіть елемент будильника

② Натисніть кнопку налаштованої відповіді

History Alarm		Real Time Alarm						
		Ack Selected						
Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	Alarm Text
Alarm.Low	Var.Variable	7/2/2020 11:18:0	1/1/0001 12:00:0	7/2/2020 11:18:0	Recovery	Low Alarm	Slight	Low
Alarm.Low	Var.Variable	7/2/2020 11:18:0	1/1/0001 12:00:0	7/2/2020 11:18:0	Recovery	Low Alarm	Slight	Low
Alarm.high	Var.Variable	7/2/2020 11:17:5	1/1/0001 12:00:0	7/2/2020 11:17:5	Recovery	High Alarm	Heavier	High
Alarm.Low	Var.Variable	7/2/2020 11:17:5	1/1/0001 12:00:0	7/2/2020 11:17:5	Recovery	Low Alarm	Slight	Low
Alarm.high	Var.Variable	7/2/2020 11:17:5	1/1/0001 12:00:0	7/2/2020 11:17:5	Recovery	High Alarm	Heavier	High
Alarm.Low	Var.Variable	7/2/2020 11:17:5	1/1/0001 12:00:0	7/2/2020 11:17:5	Recovery	Low Alarm	Slight	Low
Alarm.Low	Var.Variable	7/2/2020 11:17:5	1/1/0001 12:00:0	7/2/2020 11:17:5	Recovery	Low Alarm	Slight	Low
Alarm.high	Var.Variable	7/2/2020 11:17:5	1/1/0001 12:00:0	7/2/2020 11:17:5	Recovery	High Alarm	Heavier	High
Alarm.Low	Var.Variable	7/2/2020 11:17:4	1/1/0001 12:00:0	7/2/2020 11:17:5	Recovery	Low Alarm	Slight	Low
Alarm.high	Var.Variable	7/2/2020 11:17:4	1/1/0001 12:00:0	7/2/2020 11:17:4	Recovery	High Alarm	Heavier	High
Alarm.high1	Var.Variable1	7/2/2020 11:17:4	1/1/0001 12:00:0	1/1/0001 12:00:0	Alarm	LowLowAlarm	Heavier	LowLow
Alarm.Low	Var.Variable	7/2/2020 11:17:4	1/1/0001 12:00:0	7/2/2020 11:17:4	Recovery	Low Alarm	Slight	Low



➤ Приклад сценарію Ack an alarm button :

④ Колір АСК відповіді на тривогу

History Alarm

Real Time Alarm

- Огляд сигналізації
- Використання вікна тривоги
- **Інструмент керування сигналізацією**
- Параметри конфігурації сигналізації
- Функція клацання правою кнопкою миші групи тривог

➤ Робота інструменту керування сигналізацією:

- Редагувати такі дії, як додавання, вставлення, видалення, імпорт та експорт нагадувань
- Фільтрування змінних за групою змінних, типом змінної, назвою змінної
- Налаштування будильника відповідно до типу змінної

StartPage Alarm Variable x

+ Add - Insert x Delete Import Export

Вікно редагування нагадувань

Name	Associated Variables	Alarm Level	Alarm Condition	Description
1 AlarmVariable	Var.Variable	Slight	LowLow	
2 AlarmVariable1	Var.Variable1	Lighter	Rate	
3 AlarmVariable2	Var.Variable2	General	Major	
4 AlarmVariable3	Var.Variable3	Heavier	OFF	
5 AlarmVariable4	Var.Variable4	Serious	ON	
6 AlarmVariable5	Var.Variable5	Serious	FromOnToOff,FromOffToOn	

+ Add Alt+A
- Insert Alt+I
x Delete Del
Save Ctrl+S
Copy Ctrl+C
Cut Ctrl+X
Paste Ctrl+V
Import Ctrl+I
Export Ctrl+E

Alarm Configuration

Common Alarm

Вікно налаштувань

Limit Value Alarm

	Alarm Value	Alarm Text
☐ LowLow	10	LowLow
☐ Low	30	Low
☐ High	70	High
☐ HighHigh	90	HighHigh
☐ Deadband	0	

Rate of Change Alarm

	Alarm Value	Alarm Text
☐ Rate	0	Rate
Type	Second	

Deviation Alarm

	Alarm Value	Alarm Text
☒ Major	80	Major
☐ Minor	20	Minor
TargetValue	0	
☐ Deadband	0	

☐ AlarmDelayTime 0 Second

OK Cancel

➤ Робота інструменту керування сигналізацією:

- Редагувати такі дії, як додавання, вставлення, видалення, імпорт та експорт нагадувань
- Фільтрування змінних за групою змінних, типом змінної, назвою змінної
- Налаштування будильника відповідно до типу змінної

Variable Browser

Add Group Edit

Variable Dictionary

TypeFilter: ☒ Analog ☒ Digital A String

Name	Type	Initial Val	Minimum	Maximum	Description
Variable	Analog	0	0	10000	
Variable1	Analog	0	0	10000	
Variable2	Analog	0	0	10000	
Variable3	Digital	False	N/A	N/A	
Variable4	Digital	False	N/A	N/A	
Variable5	Digital	False	N/A	N/A	

Варіативний

Alarm Configuration

Вікно налаштування

☐ OFF ☐ On ☒ Displacement

Alarm Text

FromOnToOff FromOnToOff ...

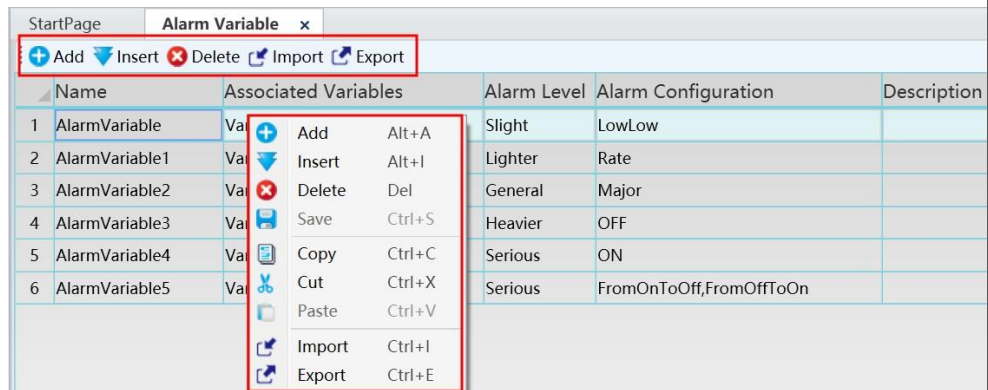
FromOffToOn FromOffToOn ...

OK Cancel

➤ Основні операції та атрибути змінних сигналізації

- Основні операції зі змінними

додати	Додайте змінну нагадування, щойно додана змінна відображається в списку змінних в області редагування змінних
Вставка	Вставте змінну, щойно вставлена змінна відображається над вставленою змінною
Видалити	Видалити змінну
Імпорт	Імпортуйте файли конфігурації зовнішніх змінних у підвищення ефективності змінної конфігурації
Експорт	Інформацію про змінну конфігурацію можна експортувати як файл Excel, який підтримує формати xls іxlsx
Копія	Копіювати змінні
Вирізати	Змінна зсуву
Вставити	Вставити змінні



The screenshot shows the 'Alarm Variable' window with a table of variables and a context menu open over the 'Associated Variables' column.

Name	Associated Variables	Alarm Level	Alarm Configuration	Description
1 AlarmVariable	Va	Slight	LowLow	
2 AlarmVariable1	Va	Lighter	Rate	
3 AlarmVariable2	Va	General	Major	
4 AlarmVariable3	Va	Heavier	OFF	
5 AlarmVariable4	Va	Serious	ON	
6 AlarmVariable5	Va	Serious	FromOnToOff,FromOffToOn	

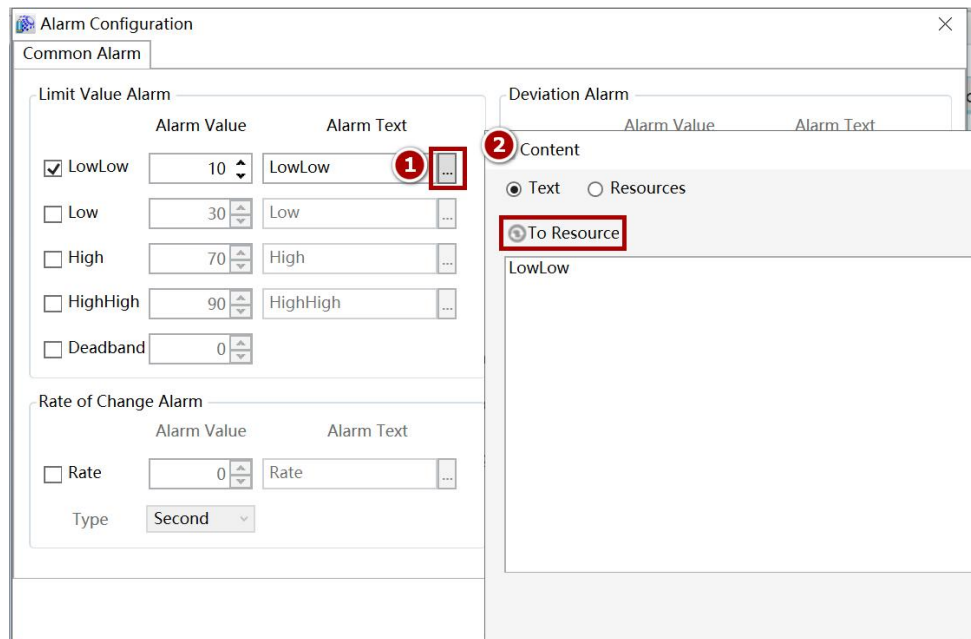
The context menu shows the following options:

- Add (Alt+A)
- Insert (Alt+I)
- Delete (Del)
- Save (Ctrl+S)
- Copy (Ctrl+C)
- Cut (Ctrl+X)
- Paste (Ctrl+V)
- Import (Ctrl+I)
- Export (Ctrl+E)

✘Зверніться до розділу «11.3 Змінні сигналізації» в посібнику користувача.

➤ Багатомовна функція сигналізації налаштування

- ① Натисніть кнопку розширення праворуч від вибраних налаштувань будильника. (Налаштування будильника перемикача може стосуватися цього процесу)
- ② У вікні «Вміст» натисніть кнопку «До ресурсу», щоб вибрати та встановити багатомовний вміст у вікні та перетворити текст нагадування на багатомовні ресурси.



Alarm Configuration

Common Alarm

Limit Value Alarm

	Alarm Value	Alarm Text
☒ LowLow	10	LowLow
☐ Low	30	Low
☐ High	70	High
☐ HighHigh	90	HighHigh
☐ Deadband	0	

Rate of Change Alarm

	Alarm Value	Alarm Text
☐ Rate	0	Rate

Type: Second

Deviation Alarm

Content

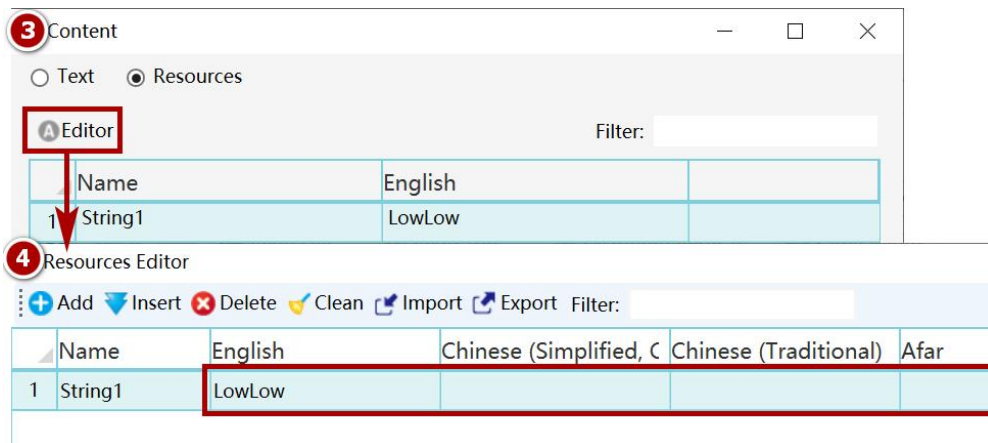
☒ Text ☐ Resources

☒ To Resource

LowLow

➤ Багатомовна функція сигналізації налаштування

- ③ Натисніть «Редактор», щоб увійти в «Ресурс
- Вікно редактора
- ④ У вікні «Редактор ресурсів» ви можете редагувати доданий багатомовний вміст безпосередньо на панелі багатомовного ресурсу. Після редагування виберіть набір багатомовних ресурсів у вікні «Вміст» і натисніть «ОК», щоб завершити багатомовне налаштування



- Огляд сигналізації
- Використання вікна тривоги
- Інструмент керування сигналізацією
- **Параметри конфігурації будильника**
- Функція клацання правою кнопкою миші групи тривог

Параметри конфігурації будильника

➤ Тип конфігурації сигналізації:

1. Електронна пошта

2. Звук

3. CMC

- ① Позначте кнопку «Надіслати електронну пошту», щоб запустити функцію надсилання електронних листів.
- ② Заповніть електронну адресу користувача-одержувача.
- ③ Встановіть відповідну інформацію для надсилання електронних листів.
- ④ Натисніть кнопку «Новий», щоб додати інформацію про користувача для отримання інформації про тривогу.

※Зверніться до розділу «11.5 Параметри конфігурації сигналізації» в посібнику користувача.

➤ Тип конфігурації сигналізації:

1. Електронна пошта

2.Звук

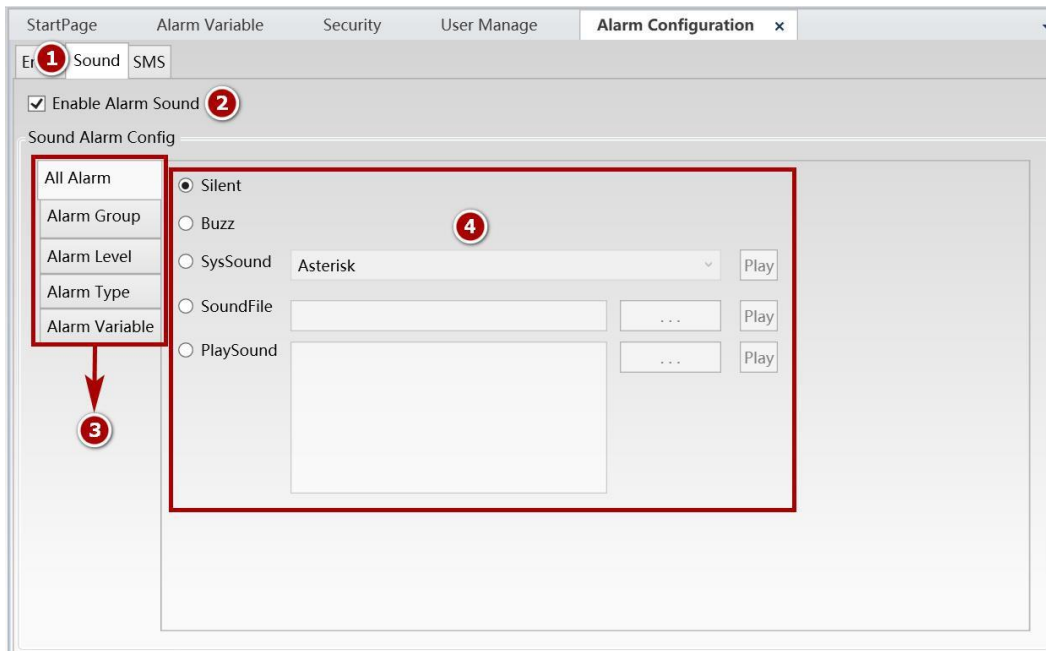
3.СМС

① Виберіть вкладку «Звук», щоб увійти в інтерфейс налаштування звуку.

② Позначте кнопку «Увімкнути звуковий сигнал», щоб активувати функцію звукового сигналу.

③ Виберіть потрібний вміст будильника.

④ Встановіть різний вміст конфігурації звукової сигналізації.





Параметри конфігурації будильника

➤ Тип конфігурації сигналізації:

1. Електронна пошта

2. Звук

3. CMC

① Виберіть вкладку «SMS», щоб увійти в інтерфейс налаштування SMS мобільного телефону.

② Перевірте кнопку «Надіслати повідомлення», щоб розпочати надсилання SMS.

③ Введіть номер SMS-центру.

④ Натисніть на кнопку «Конфігурація», щоб налаштувати інформацію про відправлення SMS.

⑤ Натисніть кнопку «Тестувати», щоб перевірити поточний стан підключення.

⑥ Натисніть кнопку «Новий», щоб додати інформацію про користувача для отримання інформації про тривогу.

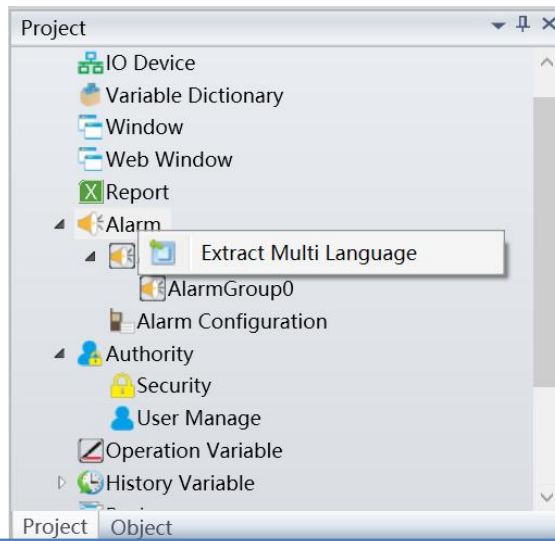
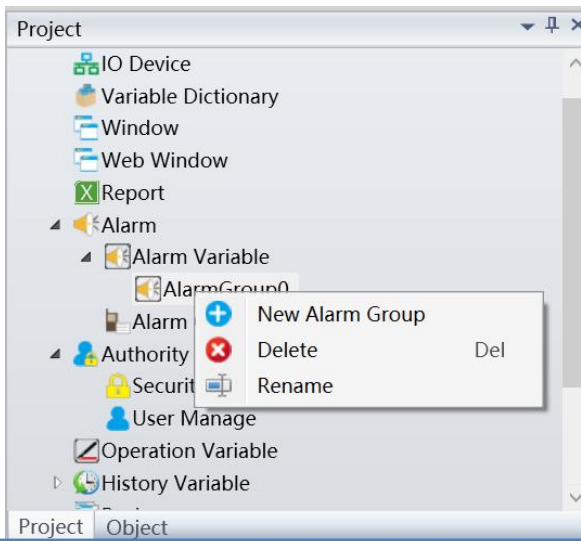
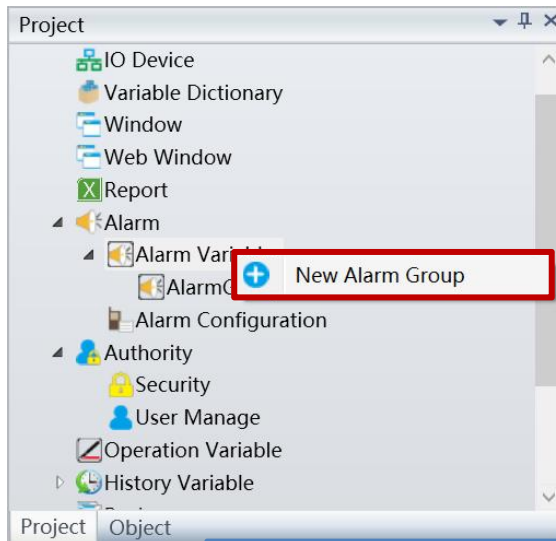
※Зверніться до розділу «11.5 Параметри конфігурації сигналізації» в посібнику користувача.

- Огляд сигналізації
- Використання вікна тривоги
- Інструмент керування сигналізацією
- Параметри конфігурації сигналізації
- Функція клацання правою кнопкою миші групи тривог

Функція клацання правою кнопкою миші групи тривог

➤ Функція клацання правою кнопкою миші групи тривог

- Група тривог може здійснювати класифіковане керування тривогами та класифікувати пов'язані тривоги в одній групі тривог, що зручно для системи, щоб керувати групами тривог і керувати ними єдиним чином.
- Змінні сигналізації в конфігураційному програмному забезпеченні DIAView зазвичай належать до групи сигналів. Спочатку встановлюється група сигналів, а потім у ній створюються змінні сигналів.



Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

RealTimeChart/HistoryChart

рубін

2020/05/01



- Поняття діаграми
- RealtimeChart
 - Огляд
 - Створіть RealtimeChart
 - Властивості
 - Кнопки панелі інструментів
 - Сценарії
- HistoryChart
 - Огляд
 - Створіть історію
 - Властивості
 - Кнопки панелі інструментів
 - Сценарії

призначення

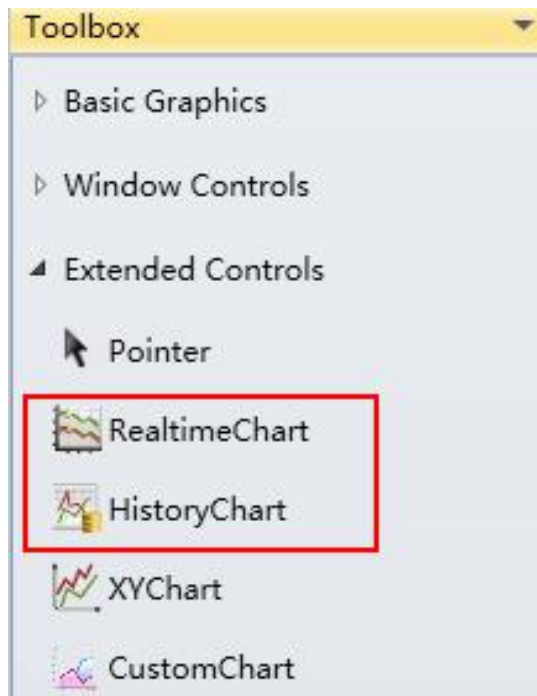
У цьому розділі ви дізнаєтеся....

- ... докладніше про діаграму реального часу
- ... докладніше про діаграму історії

Контур

- Поняття діаграми
- RealtimeChart
 - Огляд
 - Створіть RealtimeChart
 - Властивості
 - Кнопки панелі інструментів
 - Сценарії
- HistoryChart
 - Огляд
 - Створіть історію
 - Властивості
 - Кнопки панелі інструментів
 - Сценарії

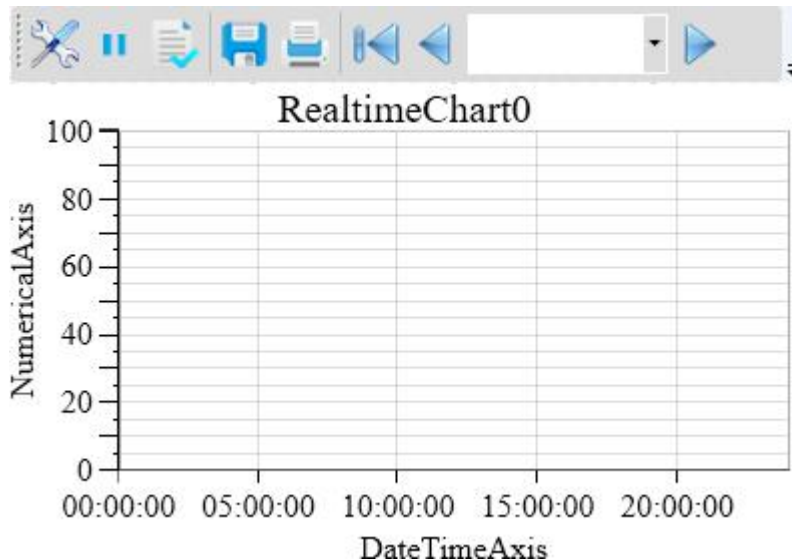
Крива, яка відображає тенденцію зміни даних з часом, і широко використовується в промисловості. Наприклад, він відображає тенденцію зміни температури навколишнього середовища, температури води, продуктивності та інших змін з часом, що зручно для користувачів, щоб ефективно та швидко збирати, аналізувати та обробляти різні промислові дані, тим самим підвищуючи ефективність виробництва. Програмне забезпечення DIAView містить різні елементи керування діаграмами. Тут ми представляємо графік реального часу та графік історії.



- Поняття діаграми
- RealtimeChart
 - Огляд
 - Створіть RealtimeChart
 - Властивості
 - Кнопки панелі інструментів
 - Сценарії
- HistoryChart
 - Огляд
 - Створіть історію
 - Властивості
 - Кнопки панелі інструментів
 - Сценарії

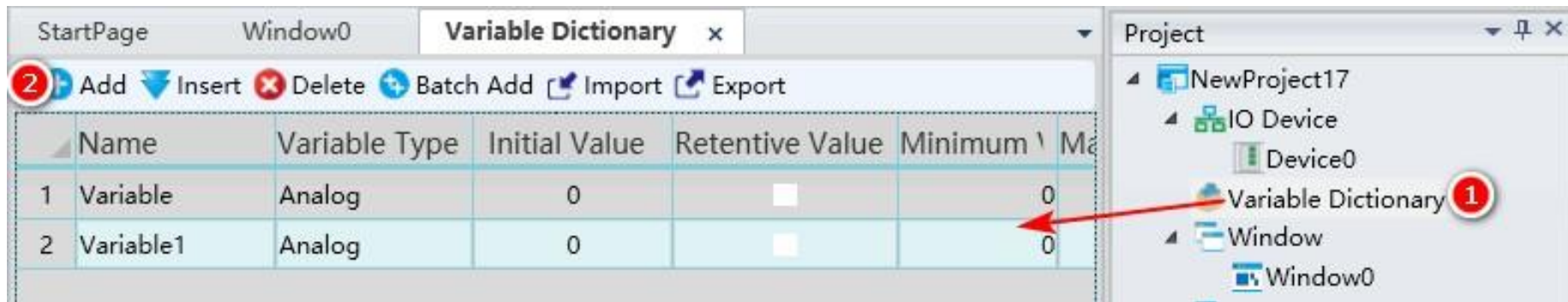
➤ Огляд

RealtimeChart — це графік значень змінних або виразів за період часу, який змінюється з часом. Зазвичай використовується в енергетиці, медицині та інших галузях промисловості. Щоб полегшити запит користувача, тренд у реальному часі може не лише намалювати криву даних у реальному часі, але й надати спеціальні типи конфігурації, такі як режим тренду, час тренду, крива, вісь часу, вісь значень тощо. Малювання та діаграми методи відповідають різноманітним потребам.



➤ Створіть RealtimeChart, який оновлює дані в реальному часі

(1) Створіть 2 змінні: Variable, Variable1



The screenshot displays the Delta Studio software interface. The main window is titled 'Variable Dictionary' and contains a table with the following data:

	Name	Variable Type	Initial Value	Retentive Value	Minimum \	Ma
1	Variable	Analog	0	☐	0	
2	Variable1	Analog	0	☐	0	

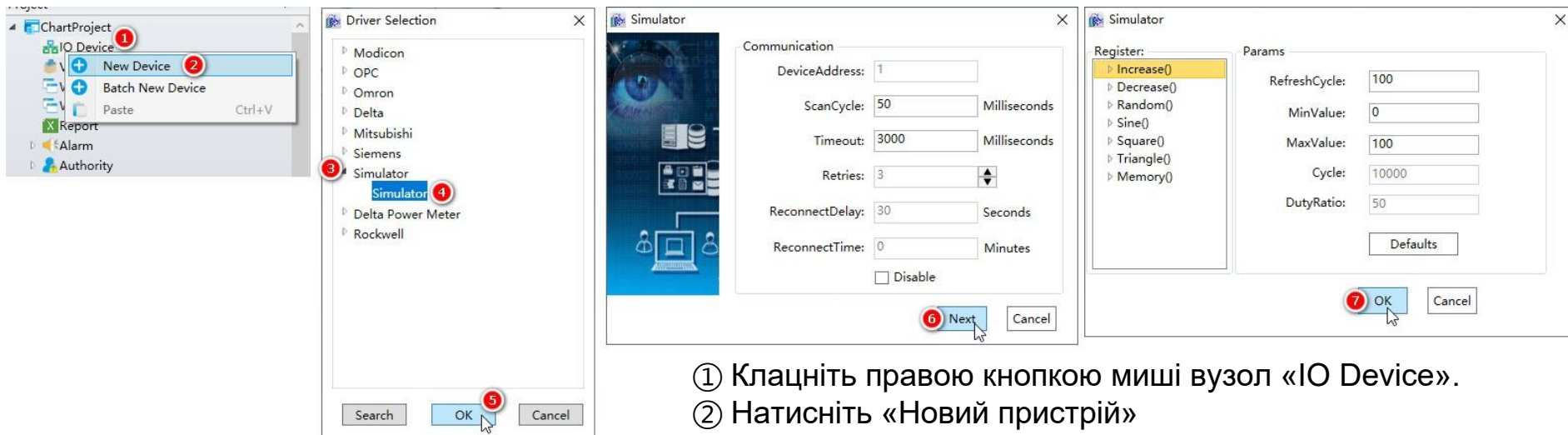
Below the table, there are buttons for 'Add', 'Insert', 'Delete', 'Batch Add', 'Import', and 'Export'. A red circle with the number '2' is next to the 'Add' button. To the right, the 'Project' window shows a tree view with the following structure:

- NewProject17
 - IO Device
 - Device0
 - Variable Dictionary (1)
 - Window
 - Window0

A red arrow points from the 'Variable Dictionary' entry in the Project window to the table in the main window.

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

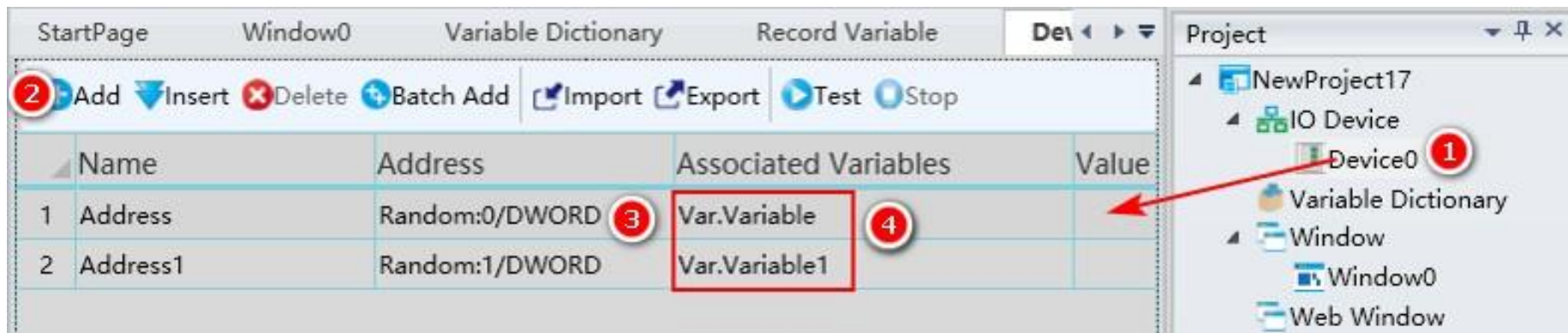
(2) Створіть імітований пристрій : Device0



- ① Клацніть правою кнопкою миші вузол «IO Device».
- ② Натисніть «Новий пристрій»
- ③④ Двічі клацніть «Симулятор»

※Див. розділ «5.10.1 Симулятор» у посібнику користувача.

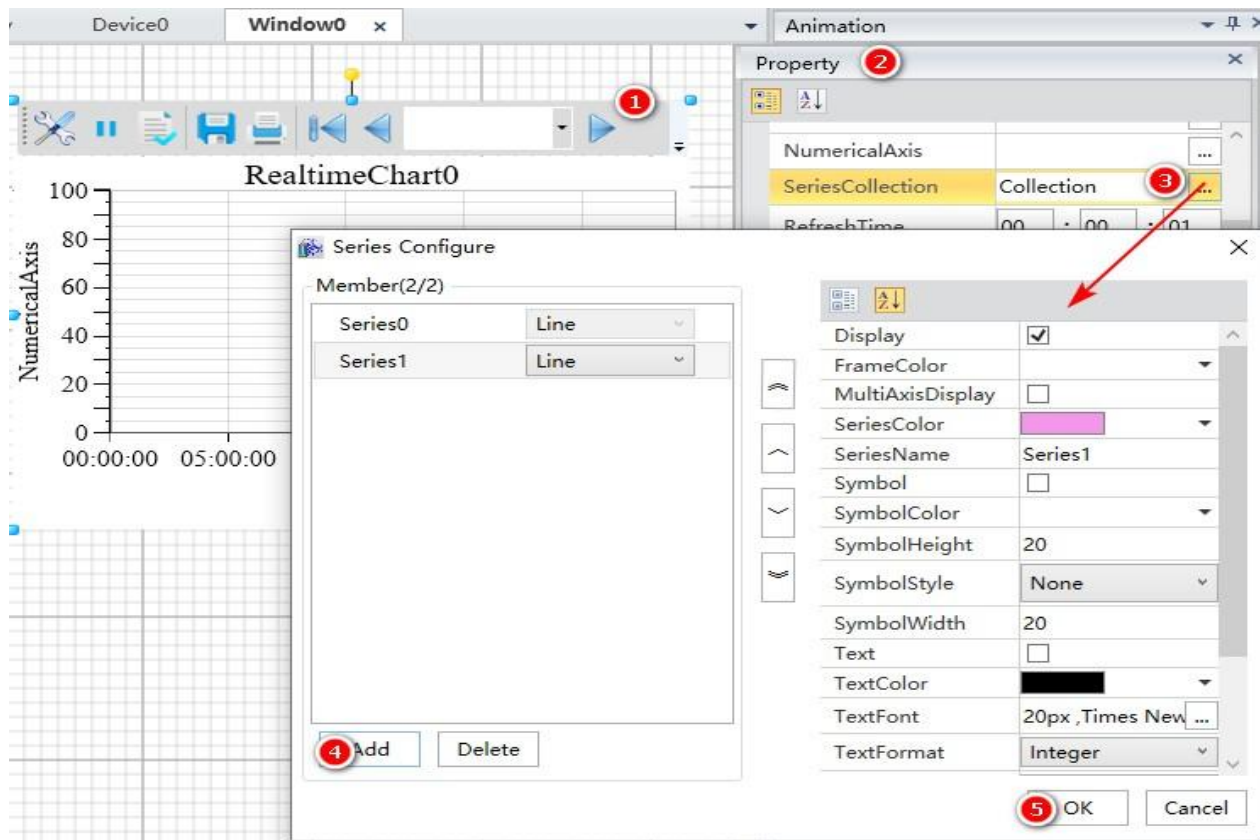
(3) Створіть дві адреси моделювання в Device0, пов'язані зі змінною, змінною1 відповідно



The screenshot displays the Delta software interface. The 'Variable Dictionary' tab is active, showing a table with two entries. The 'Associated Variables' column for both entries is highlighted with a red box. The 'Project' tree on the right shows the hierarchy: NewProject17 > IO Device > Device0 (circled with a red 1). A red arrow points from Device0 to the 'Value' column of the table.

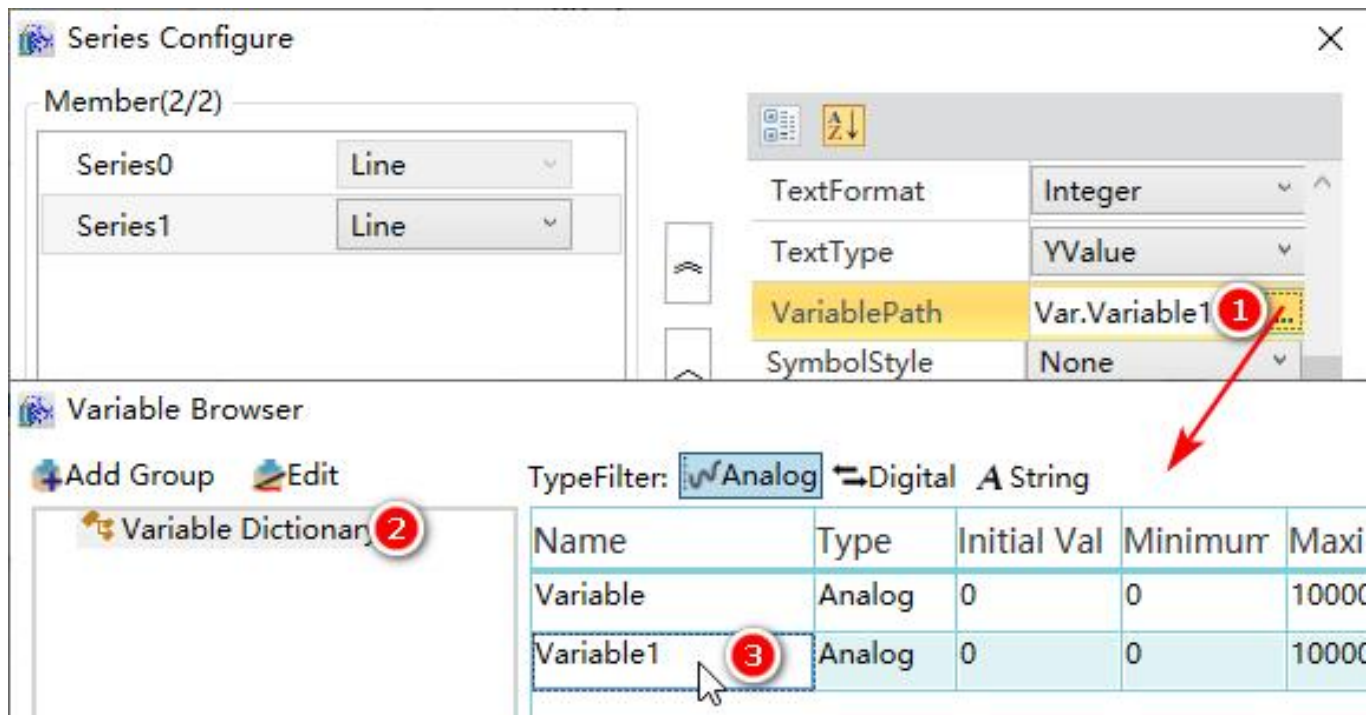
	Name	Address	Associated Variables	Value
1	Address	Random:0/DWORD	Var.Variable	
2	Address1	Random:1/DWORD	Var.Variable1	

(4) Створіть RealtimeChart0 у Window0 та додайте 2 криві у RealtimeChart0



Графік у реальному часі

(5) Криві асоційовані змінні: асоційована змінна Series0 ,
асоційована змінна Series1 , асоційована змінна1



The screenshot displays two windows from the Delta software interface. The top window, titled "Series Configure", shows a list of series members on the left and configuration options on the right. The bottom window, titled "Variable Browser", shows a list of variables and their properties.

Series Configure Window:

- Member(2/2):
 - Series0: Line
 - Series1: Line
- Configuration options:
 - TextFormat: Integer
 - TextType: YValue
 - VariablePath: Var.Variable1 (highlighted with a red circle 1 and an arrow pointing to the Variable Browser)
 - SymbolStyle: None

Variable Browser Window:

- Buttons: Add Group, Edit
- Variable Dictionary (highlighted with a red circle 2)
- TypeFilter: Analog (selected), Digital, String
- Table:

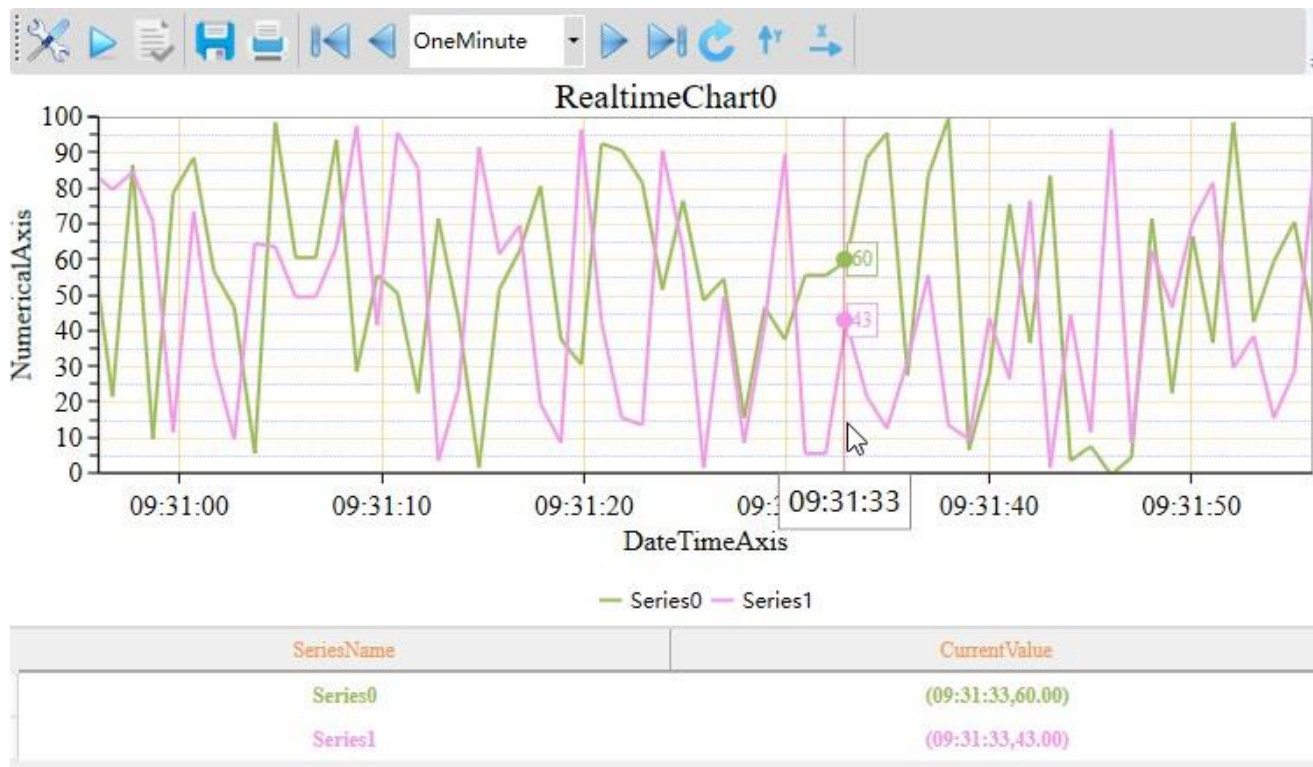
Name	Type	Initial Val	Minimum	Maximum
Variable	Analog	0	0	10000
Variable1	Analog	0	0	10000

The "Variable1" row in the table is highlighted with a red circle 3 and a mouse cursor.

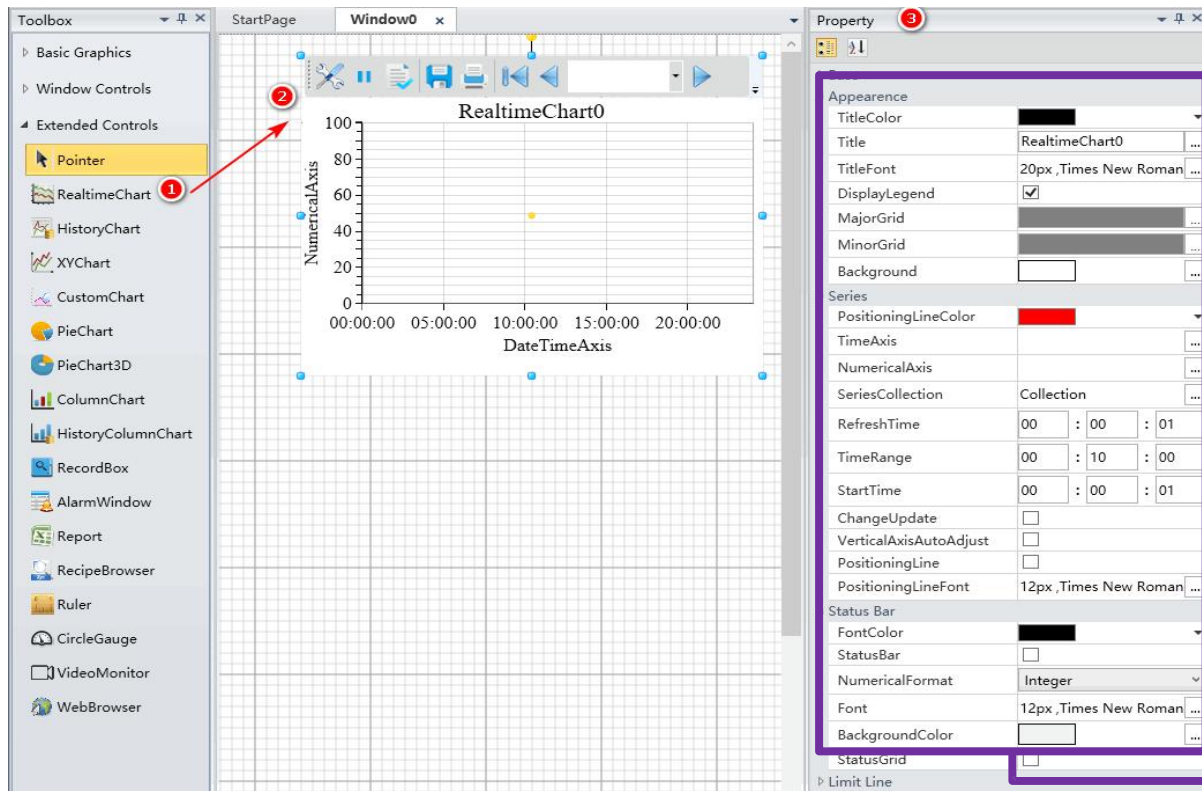
(6) Встановіть деякі властивості

Appearance	
TitleColor	
Title	RealtimeChart0 ...
TitleFont	20px ,Times New Roman ...
DisplayLegend	☒
MajorGrid	 ...
MinorGrid	 ...
Background	 ...
Series	
PositioningLineColor	
TimeAxis	...
NumericalAxis	...
SeriesCollection	Collection ...
RefreshTime	00 : 00 : 01
TimeRange	00 : 10 : 00
StartTime	00 : 00 : 01
ChangeUpdate	☐
VerticalAxisAutoAdjust	☐
PositioningLine	☒
PositioningLineFont	12px ,Times New Roman ...
Status Bar	
FontColor	
StatusBar	☒
NumericalFormat	Double2 ▾
Font	12px ,Times New Roman ...
BackgroundColor	 ...
StatusGrid	☐

(7) Запустіть Window0 , RealtimeChart0 оновить дані наступним чином

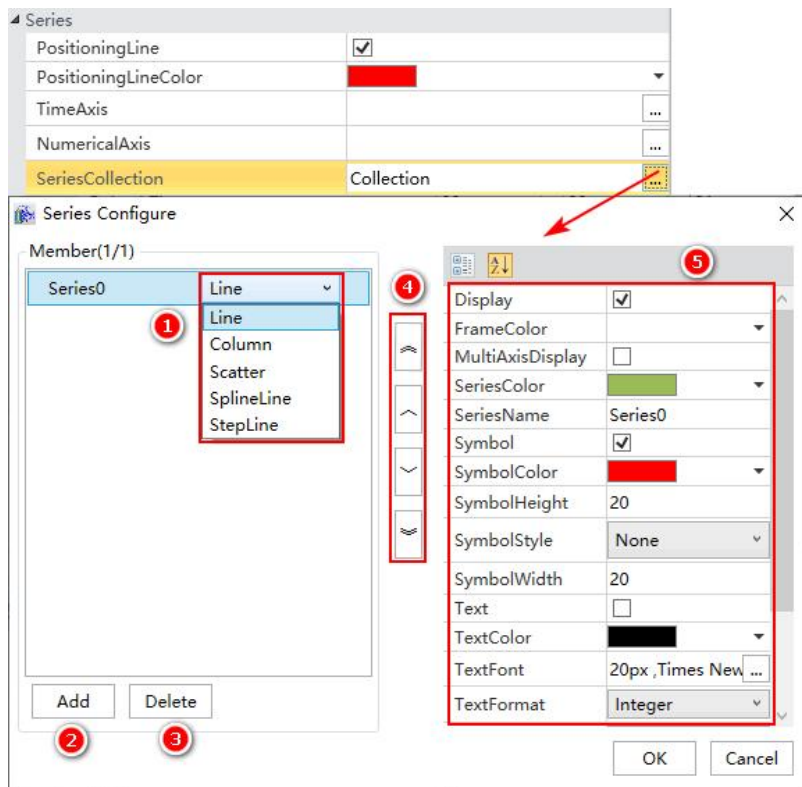


➤ Вікно власності



- **TimesAxis** : Встановити властивості TimesAxis
- **NumericalAxis** : встановити властивості NumericalAxis
- **SeriesCollection** : додайте або видаліть серію, установіть властивості серії
- **ChangeUpdate** : встановіть, чи оновлювати дані тільки при зміні значення
- **VerticalAxisAutoChange** : встановіть автоматичне налаштування максимального значення вертикальної осі.
- **PositionLine** : чи відображати позицію лінії
- **PositionLineColor** : Установіть колір позиційної лінії
- **PositionLineFont** : установіть формат шрифту даних на позиційному рядку
- **DisplayLegend** : чи відображати легенду діаграми.
- **MajorGrid** : Установіть стиль основної сітки
- **MinorGrid** : встановити стиль другорядної сітки

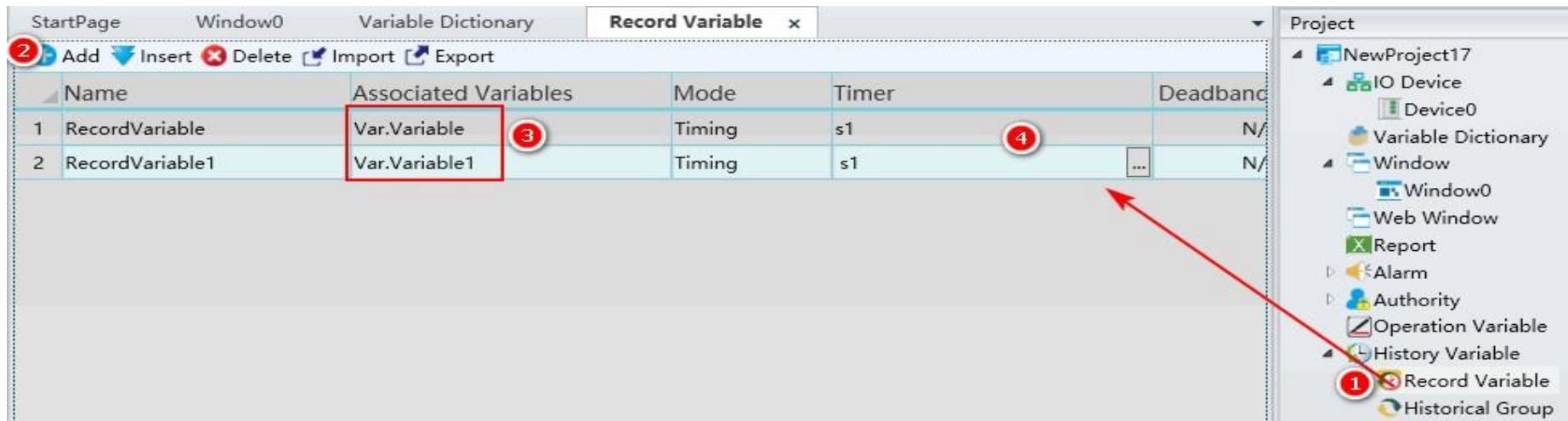
➤ Властивість SeriesCollection



- ① SeriesType : установіть тип серії
- ② Додайте серію в RealtimeChart
- ③ Видаліть серію з RealtimeChart
- ④ Змініть порядок відображення серій
- ⑤ Встановіть властивості поточної вибраної серії

➤ Приклад властивості TimeRange

(1) Змінні, пов'язані з кривими в колекції серій RealtimeChart, також пов'язані з пристроєм моделювання та історичними змінними

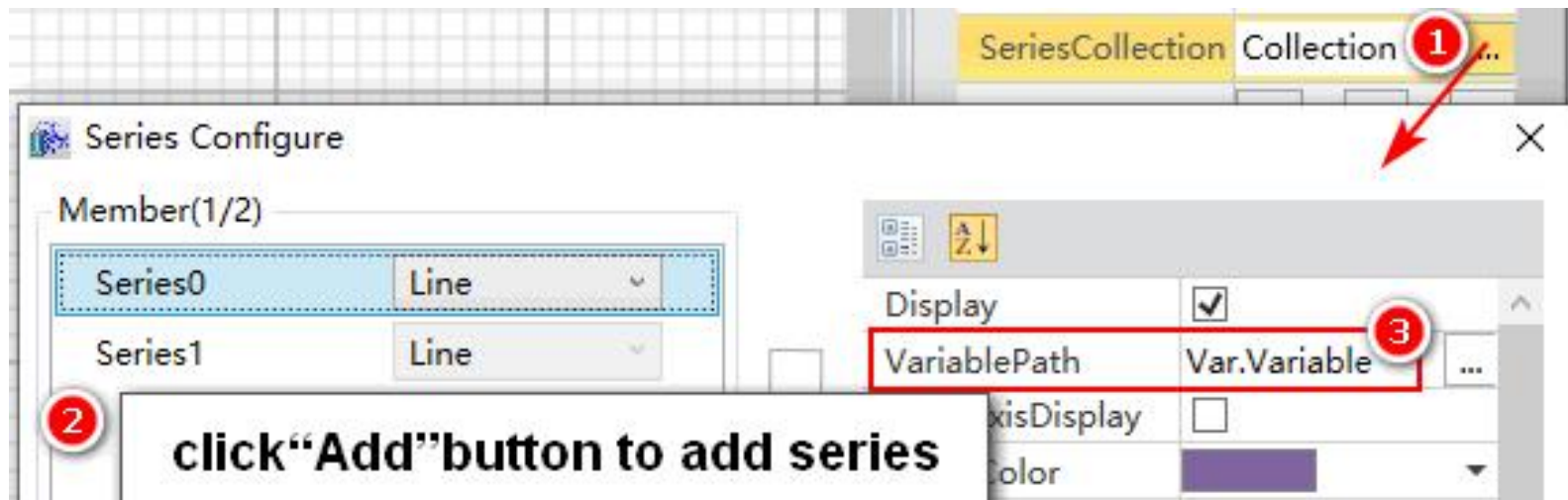


	Name	Associated Variables	Mode	Timer	Deadband
1	RecordVariable	Var.Variable	Timing	s1	N/
2	RecordVariable1	Var.Variable1	Timing	s1	N/

Project tree structure:

- NewProject17
 - IO Device
 - Device0
 - Variable Dictionary
 - Window
 - Window0
 - Web Window
 - Report
 - Alarm
 - Authority
 - Operation Variable
 - History Variable
 - Record Variable
 - Historical Group

(2) Серія0, пов'язана змінна , Серія1, пов'язана змінна1



(3) Встановіть для властивості
TimeRange значення 10 хв

RefreshTime	00	:	00	:	01
TimeRange	00	:	10	:	00
StartTime	00	:	00	:	01

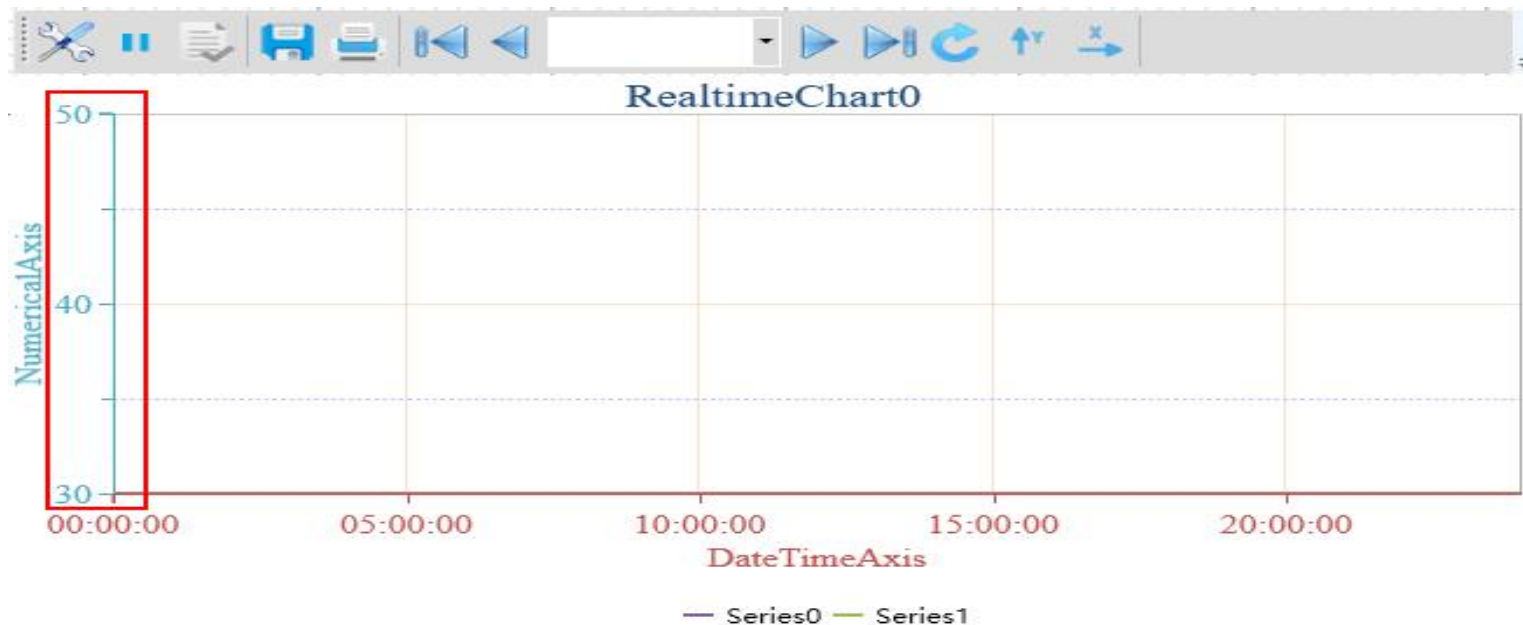
(4) Запустіть проект



- ① Встановіть діапазон сторінок на 1 хв
- ②③ Використовуйте « Перший », « Вперед », « Назад », « Останній » для перегляду історичних даних (протягом 10 хвилин).
У цій конфігурації можна переглядати до 10 сторінок даних

➤ Приклад властивості VerticalAxisAutoAdjust

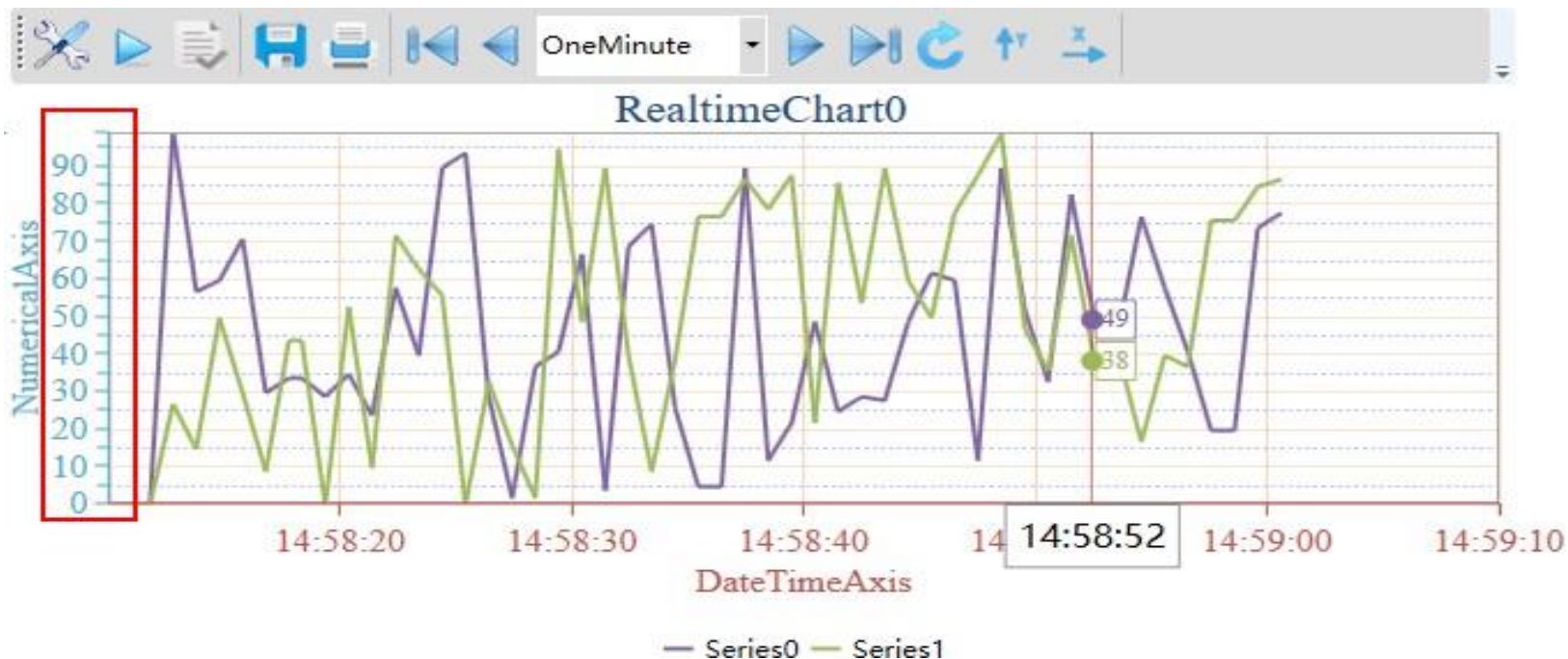
(1) Створіть RealtimeChart0 і встановіть діапазон числової осі від 30 до 50, Series0, Series1, пов'язані зі змінною0, Variable1 відповідно (пов'язано з Simulation Device) в діапазоні від 0 до 100



(2) Перевірте властивість VerticalAxisAutoAdjust
RealtimeChart0

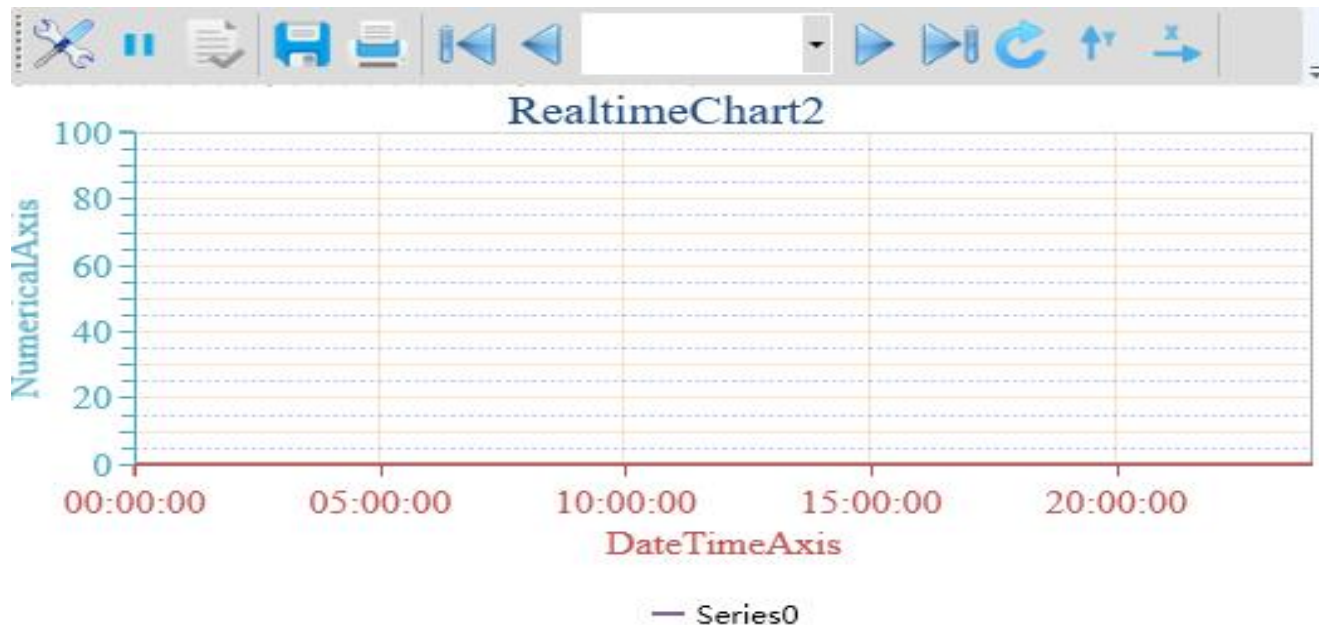


(3) Запустіть проект, діапазон числової осі коригується зі значенням точок даних



➤ Приклад властивості ChangeUpdate

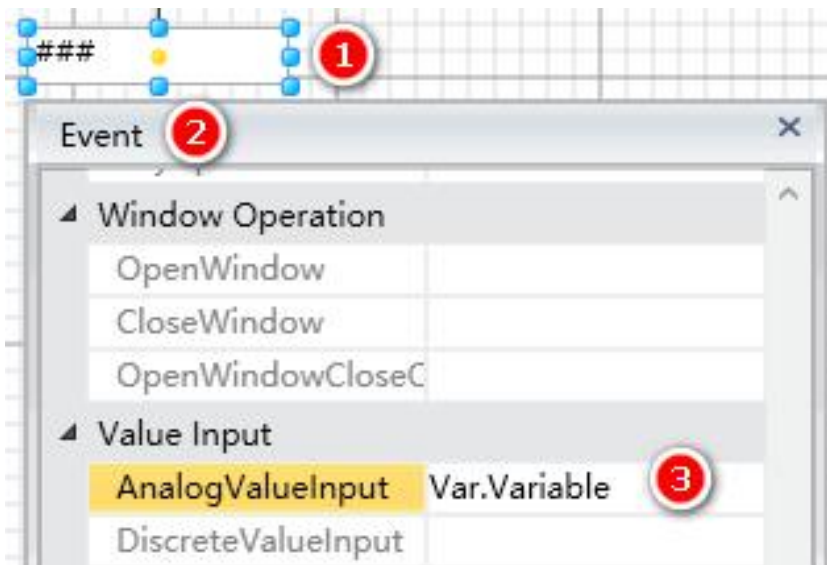
- (1) Створіть RealtimeChart1, RealtimeChart2 у Window0 і всі додайте Series0, який пов'язаний зі змінною4 (не пов'язаною з жодним пристроєм) у словнику змінних



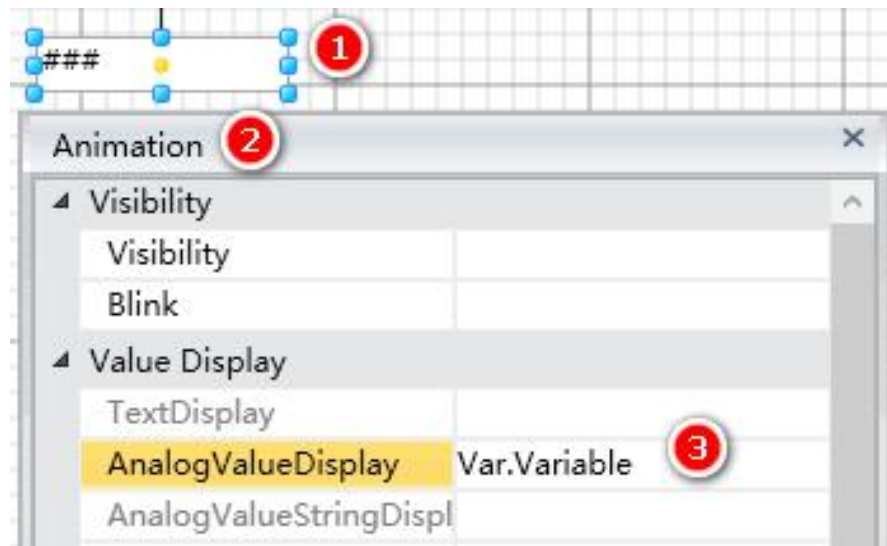
(2) Зніміть прапорець біля властивості ChangeUpdate
RealtimeChart1, перевірте
ChangeUpdate RealtimeChart2

ChangeUpdate	☒
VerticalAxisAutoAdjust	☐
PositioningLineFont	12px , Times New Roman ...

(3) Створіть TextBox0 у Window0. Налаштуйте подію та анімацію TextBox0



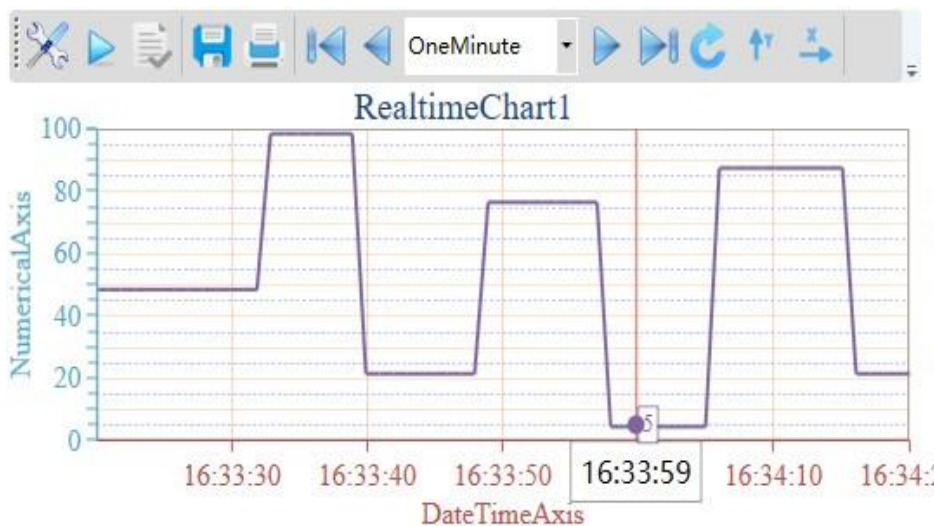
Подія введення аналогового значення



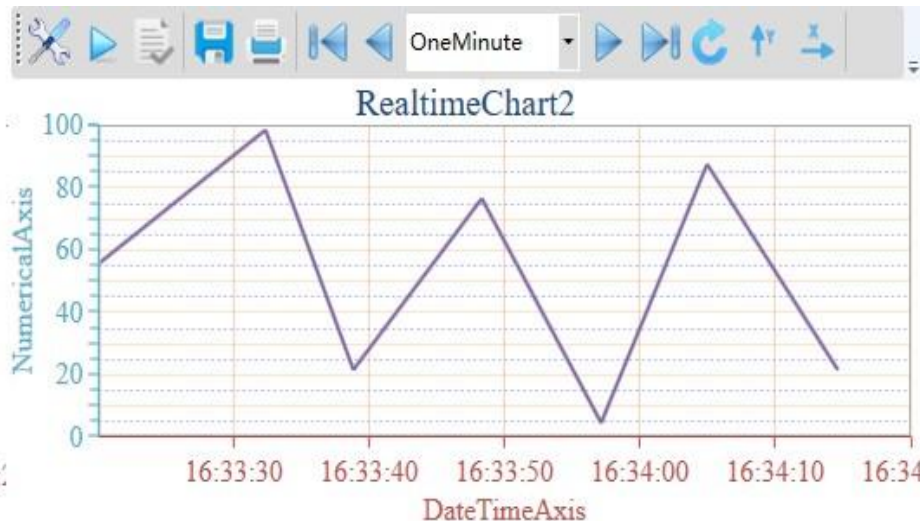
Анімація відображення аналогового значення

Графік у реальному часі

(4) Під час виконання вводьте дані в TextBox0 кожні кілька секунд, ефект оновлення RealtimeChart1 і RealtimeChart2, як показано нижче



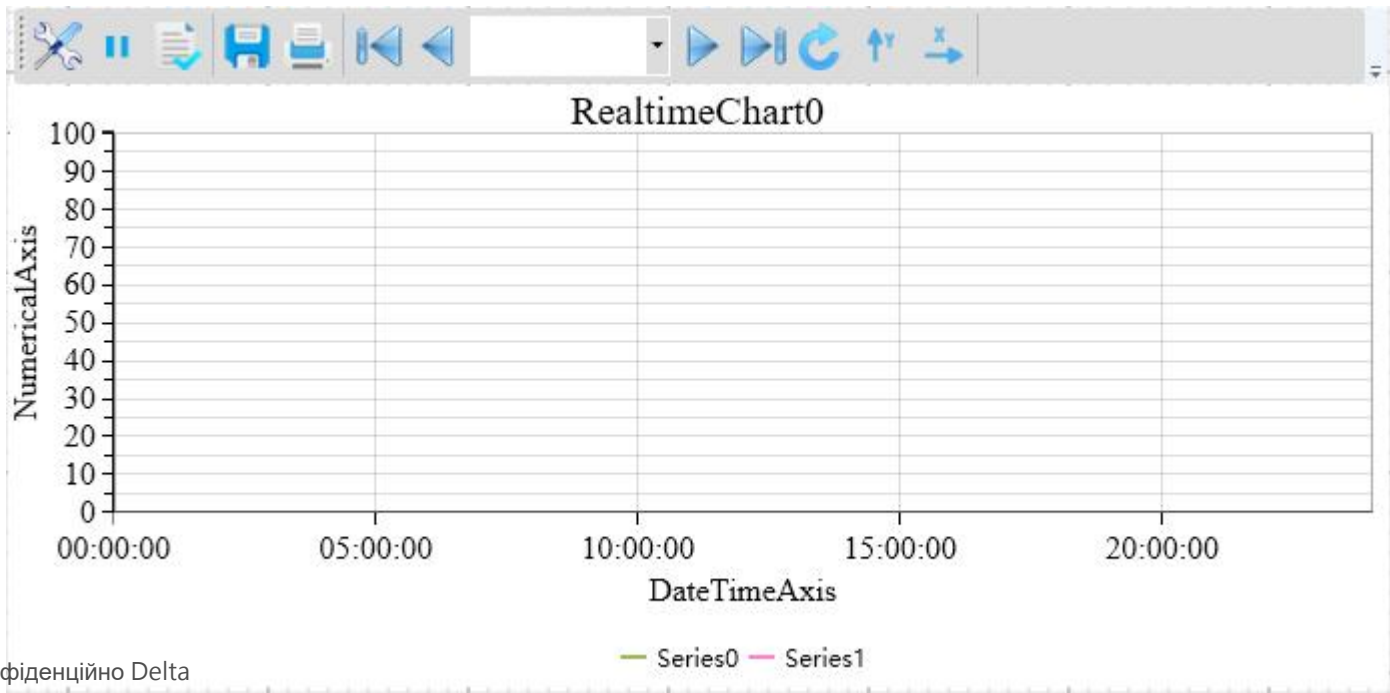
Властивість ChangeUpdate не позначено



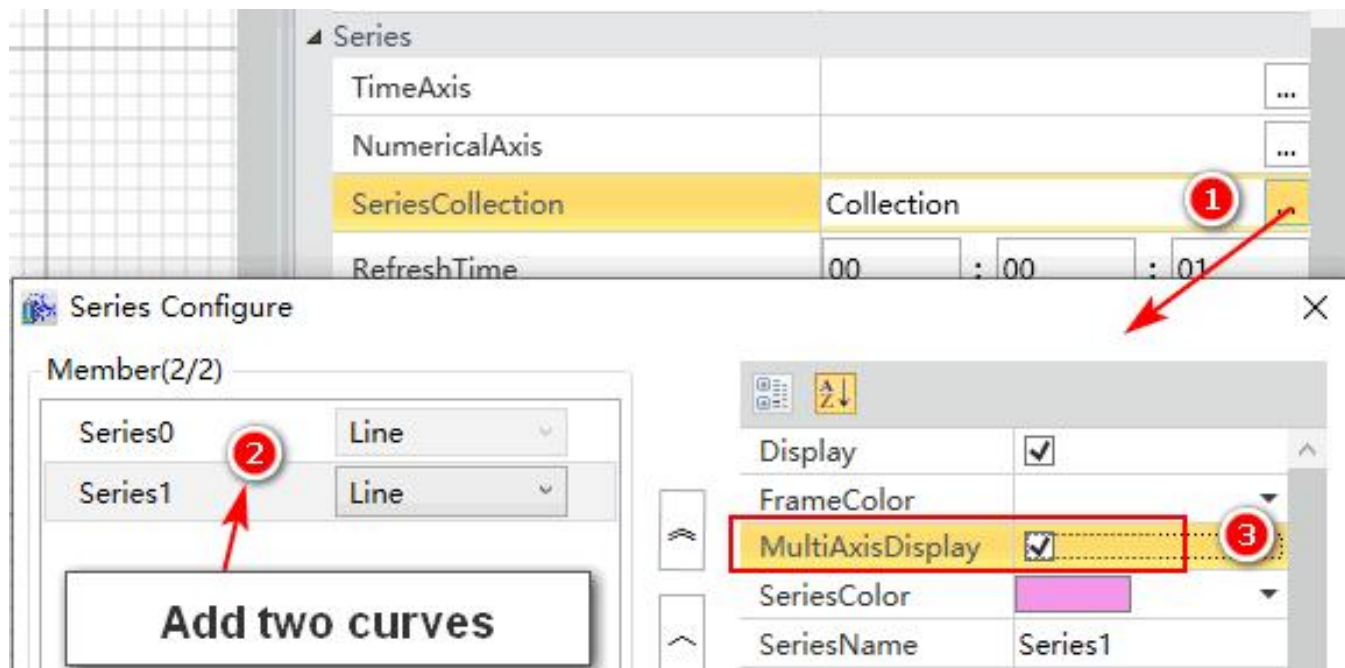
Властивість ChangeUpdate перевірено

➤ Приклад властивості MultiAxisDisplay

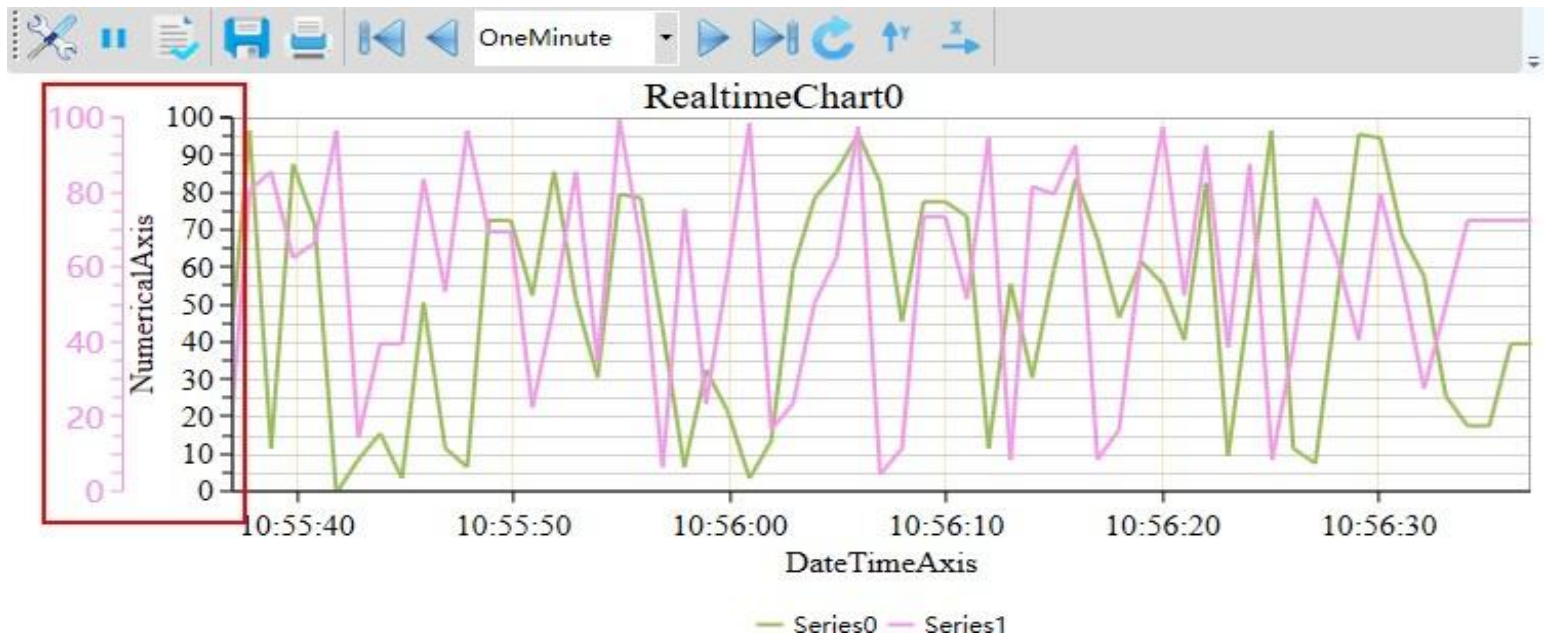
(1) Створіть RealtimeChart0 і додайте дві криві, Series0, Series1, пов'язані з Variable0, Variable1 відповідно (пов'язані з Simulation Device).



(2) Перевірте властивість MultiAxisDisplay Series1 у RealTimeChart0

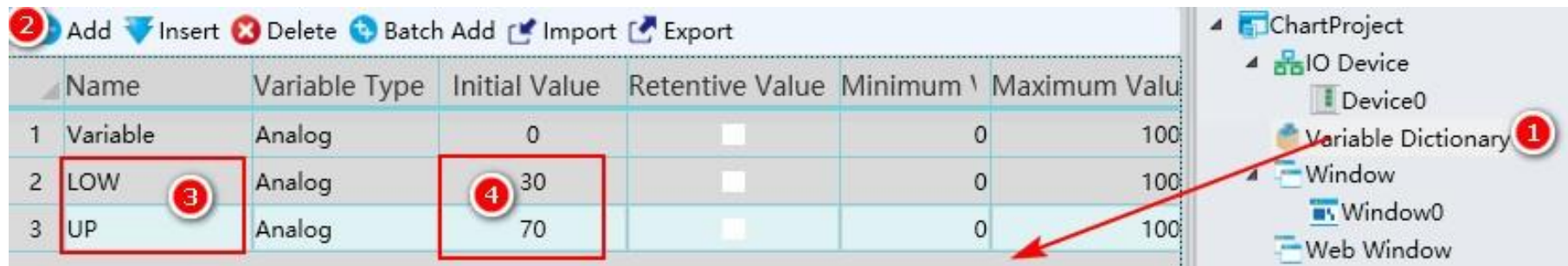


(3) Запустіть проект, дві числові осі відображатимуться на RealtimeChart0



➤ Приклад властивості Limit Line

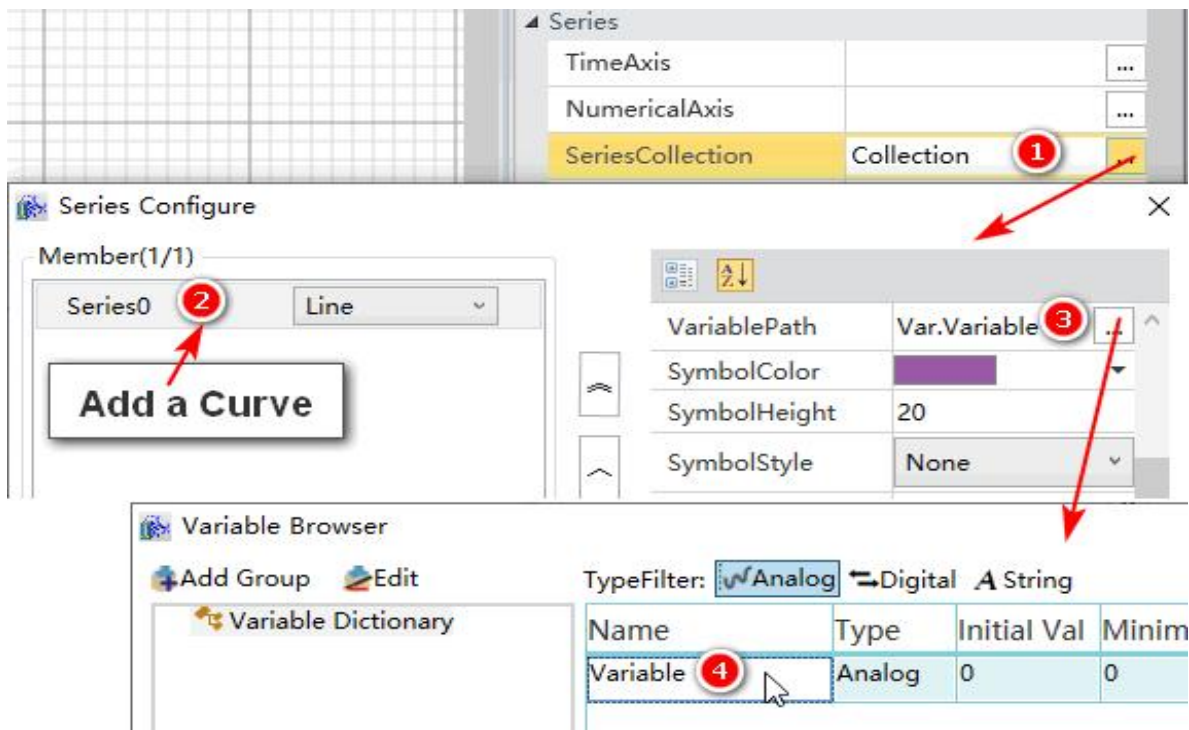
(1) Створіть три змінні: Variable, UP, LOW (змінна, пов'язана з пристроєм імітації).



The screenshot displays the Delta software interface. On the left, a table lists variables with columns: Name, Variable Type, Initial Value, Retentive Value, Minimum Value, and Maximum Value. The table contains three rows: 'Variable' (Analog, Initial Value 0), 'LOW' (Analog, Initial Value 30), and 'UP' (Analog, Initial Value 70). The 'LOW' and 'UP' rows are highlighted with red boxes, and the 'Initial Value' column is also highlighted. On the right, a tree view shows the project structure: ChartProject, IO Device, Device0, Variable Dictionary (highlighted with a red circle and arrow), Window, Window0, and Web Window.

	Name	Variable Type	Initial Value	Retentive Value	Minimum Value	Maximum Value
1	Variable	Analog	0		0	100
2	LOW	Analog	30		0	100
3	UP	Analog	70		0	100

(2) Створіть RealtimeChart0 і додайте Series0 (SeriesType — Line),
Series0 пов'язаний із змінною



The screenshot illustrates the configuration process for a real-time chart. It features three main windows: a chart area at the top, a 'Series Configure' dialog in the middle, and a 'Variable Browser' at the bottom.

- Series Collection:** In the top chart area, the 'SeriesCollection' is highlighted, with a red circle '1' and an arrow pointing to the 'Add' button.
- Series Configure:** This dialog shows 'Member(1/1)' with 'Series0' selected and 'Line' as the 'SeriesType'. A red circle '2' and an arrow point to the 'Add a Curve' button.
- Variable Path:** In the 'Series Configure' dialog, the 'VariablePath' field is set to 'Var.Variable', with a red circle '3' and an arrow pointing to the selection button.
- Variable Browser:** This window shows a list of variables. The 'TypeFilter' is set to 'Analog'. A table lists the variables:

Name	Type	Initial Val	Minim
Variable	Analog	0	0

A red circle '4' and an arrow point to the 'Variable' entry in the table.

(3) Встановіть властивість лінії обмеження для RealtimeChart0

Limit Line

Thickness	1.00	①
LimitLineMaximum	Var.UP	④
LimitLineMinimum	Var.LOW	④
UpperAndLowerDisplay	☒	②
UpperLimitLineColor		③
LowerLimitLineColor		③

- ① Встановіть товщину лінії обмеження
- ② Перевірено UpperAndLowerDisplay власність
- ③ Встановіть колір лінії обмеження
- ④⑤ Встановіть відповідну змінну граничної лінії

Variable Browser

Add Group Edit

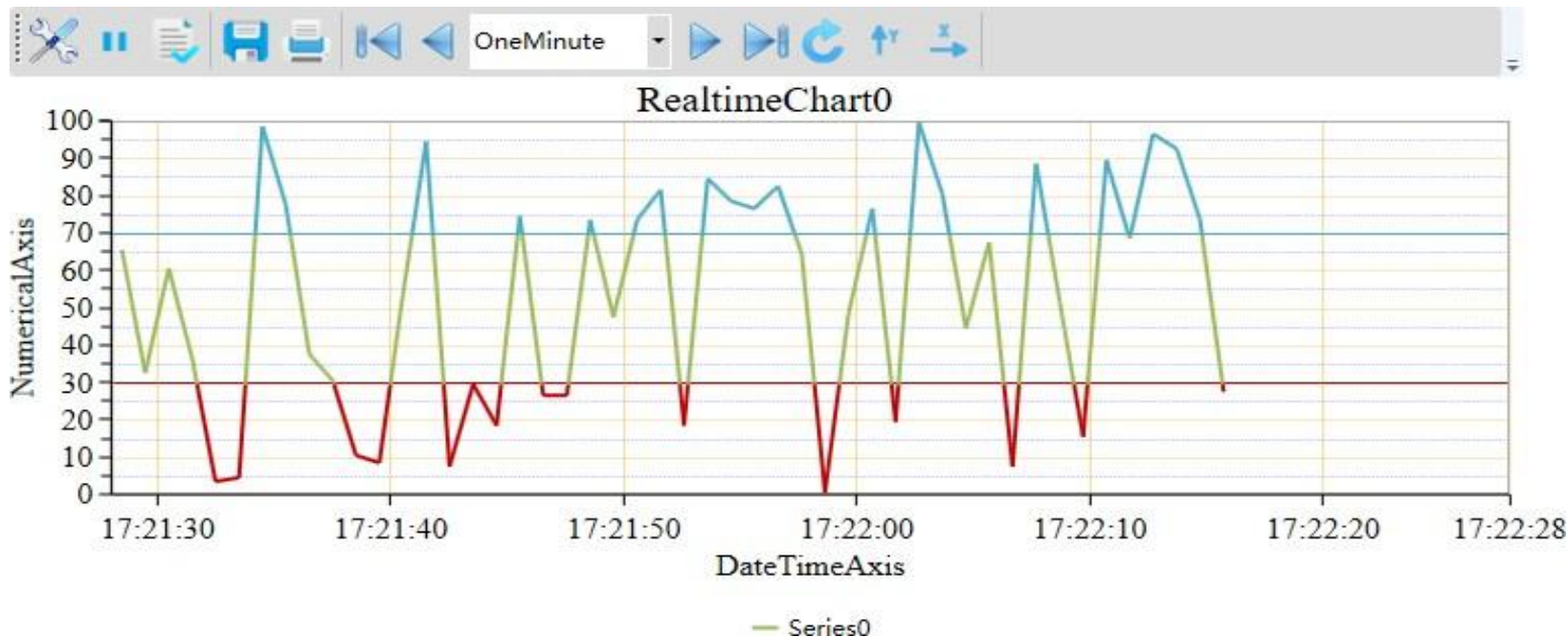
Variable Dictionary

TypeFilter: ☒ Analog ☐ Digital ☐ String

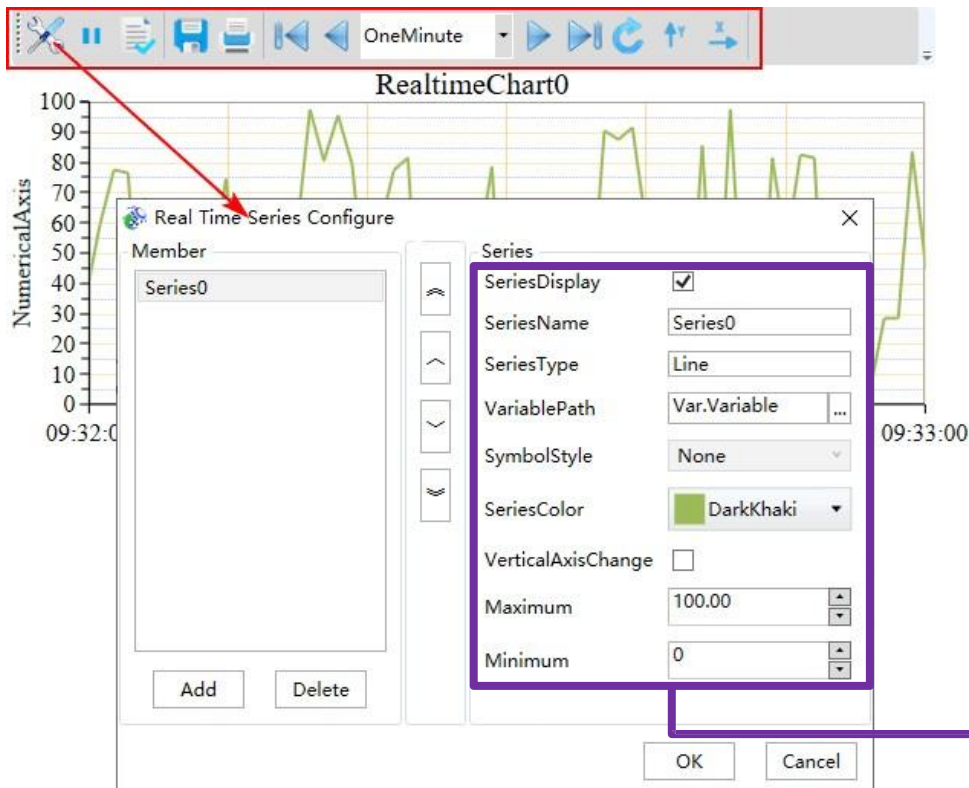
Name	Type	Initial Val	Minimum	Maximum	De
Variable	Analog	0	0	10000	
LOW	Analog	30	0	10000	
UP	Analog	70	0	10000	

⑤

(4) Запустіть Window0 , обмежте відображення лінії



➤ Використання панелі інструментів RealtimeChart



Налаштування рядів у реальному часі (Змінити властивості рядів під час виконання) :

- **SeriesDisplay** : чи відображати серію
- **SeriesName** : змінити назву серії
- **SeriesType** : відображення типу серії
- **VariablePath** : установити змінний шлях серії
- **SymbolStyle** : Встановити стиль символу серії
- **SeriesColor** : установити колір серії
- **VerticalAxisChange** : установити, чи регулювати діапазон вертикальної осі відповідно до значення точки даних. (Перевірте властивість «VerticalAxisAutoChange» під час розробки, ця властивість дійсна)
- **Максимум** : установити максимум вертикальної осі
- **Мінімум** : установити мінімум вертикальної осі

➤ Скрипти RealtimeChart

Для використання сценаріїв RealtimeChart, будь ласка, зверніться до розділу « 20.3 Граматика та функції скриптів » → “ Модель програмування ” → « Зображення » → “ Розширені елементи керування ” → « Об’єкт HMIRealTimeChart » у посібнику користувача

Methods list

	Name	Description
🔗	AddNewLimitLine	Add a new limit line
🔗	ArrayToString	Image array translate into string
🔗	DeletevalueAll	Delete all data points
🔗	Export	Export data
🔗	ExportRealData	Export excel
🔗	FindAnimation	Look for animation to modify the associated variables of the animation
🔗	HiddenSeries	Hidden curve
🔗	OpenShowFileDialog	Get saved path selected by dialog box
🔗	Print	Print
🔗	RemoveAllLimitLine	Remove all the limit line
🔗	Save	Save
🔗	Save	Save as pictures
🔗	SetSeriesVariablePath	Curve correlation variable substitution
🔗	SetTimeRangeWithRefresh	Set the time interval and refresh interval
🔗	StartTimer	Start timer
🔗	StopTimer	Stop timer
🔗	VisiableSeries	Display curves

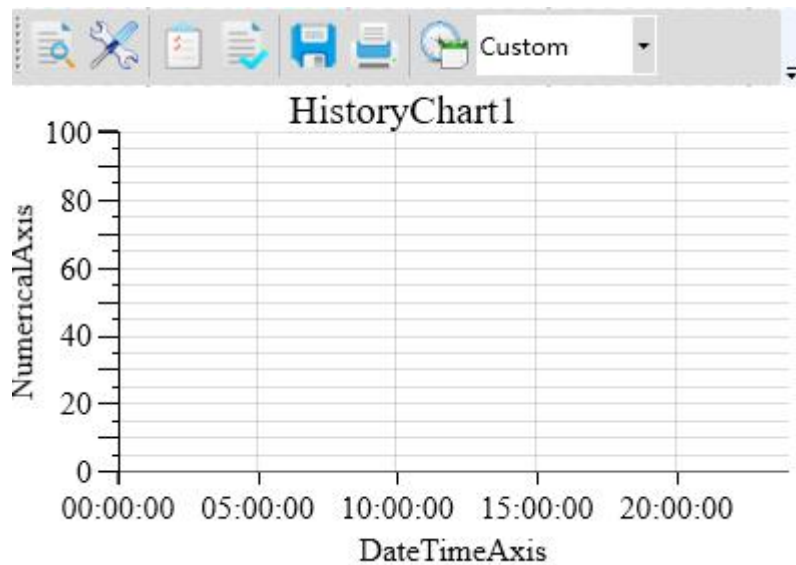
Property list

	Name	Description
🔗	Height	Height
🔗	IsShow	Whether display specified object
🔗	IsShowBtnCrossLine	Whether display the positioning line button
🔗	IsShowBtnLeftMost	Whether display the most left button
🔗	IsShowBtnLeftShift	Whether display the left shift button
🔗	IsShowBtnReSet	Whether display the reset button
🔗	IsShowBtnRightMost	Whether display the most right button
🔗	IsShowBtnRigthShift	Whether display the right shift button
🔗	IsShowRealBtnConfig	Whether display the curve configuration button
🔗	IsShowRealBtnPrint	Whether display the print button
🔗	IsShowRealBtnSave	Whether display the save button
🔗	IsShowRealBtnStop	Whether display the began or pause button
🔗	IsShowStopTime	Whether display the pause time button
🔗	Left	The left coordinate
🔗	Name	Name
🔗	NumberInterval	Get or set the maximum scale of numerical axis
🔗	NumberSmallInterval	Get or set the minimum scale of numerical axis
🔗	NumericalAxisMaximum	Get or set the maximum scale of numerical axis
🔗	NumericalAxisMinimum	Get or set the minimum scale of numerical axis
🔗	ToolTip	Tooltip text
🔗	Top	The top coordinate
🔗	Width	Width
🔗	ZIndex	Layer index

- Поняття діаграми
- RealtimeChart
 - Огляд
 - Створіть RealtimeChart
 - Властивості
 - Кнопки панелі інструментів
 - Сценарії
- HistoryChart
 - Огляд
 - Створіть історію
 - Властивості
 - Кнопки панелі інструментів
 - Сценарії

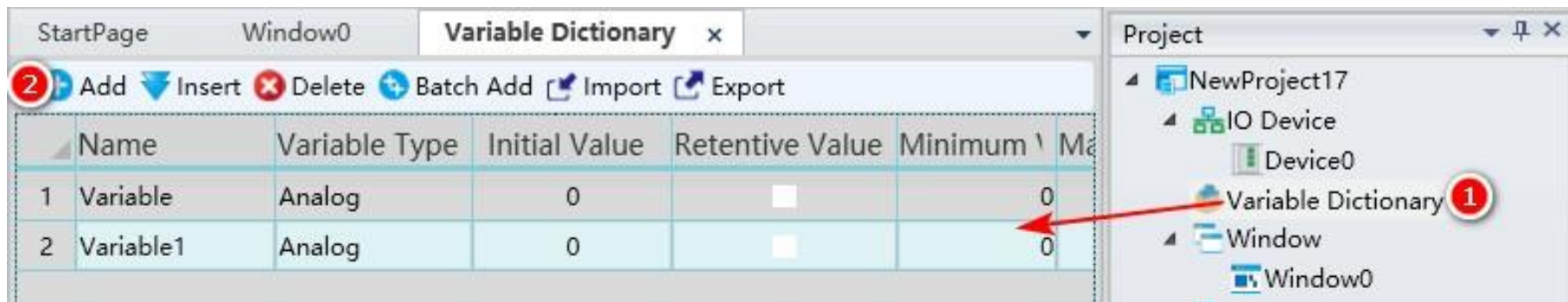
➤ Огляд

HistoryChart — це графік, що відображає історичний процес зміни значення історичної змінної. Історична діаграма в основному використовується для аналізу аварій та аналізу ефективності роботи, щоб забезпечити відповідну основу для оптимізації системи. Крива історії містить основні атрибути, такі як набір кривих, вісь координат, панель інструментів, рядок стану тощо. Історична крива пов'язана зі змінною запису історії конфігурації, і можна прочитати зміну даних змінної в різні періоди часу.



➤ Створіть HistoryChart, який запитує історичні дані

(1) Створіть 2 змінні : змінна , змінна 1



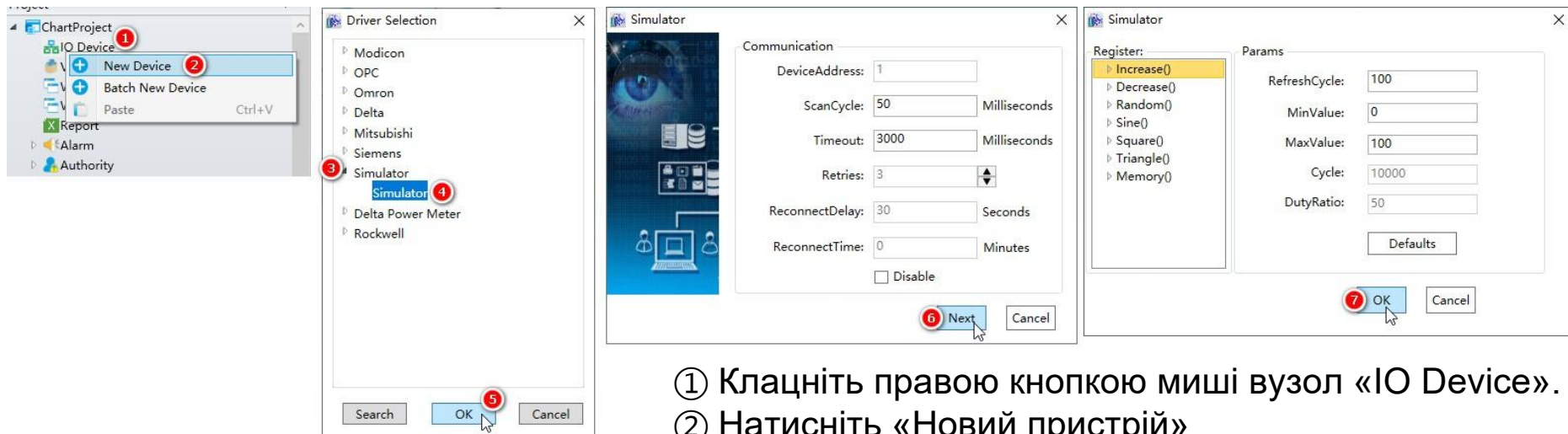
The screenshot displays the Delta Studio software interface. The main window is titled 'Variable Dictionary' and contains a table with the following data:

	Name	Variable Type	Initial Value	Retentive Value	Minimum \	Ma
1	Variable	Analog	0	☐	0	
2	Variable1	Analog	0	☐	0	

Below the table are buttons for 'Add', 'Insert', 'Delete', 'Batch Add', 'Import', and 'Export'. The 'Add' button is highlighted with a red circle and the number '2'. To the right, the 'Project' pane shows a tree structure for 'NewProject17' with nodes for 'IO Device', 'Device0', 'Variable Dictionary' (highlighted with a red circle and the number '1'), 'Window', and 'Window0'. A red arrow points from the 'Variable Dictionary' node in the Project pane to the 'Variable Dictionary' window.

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

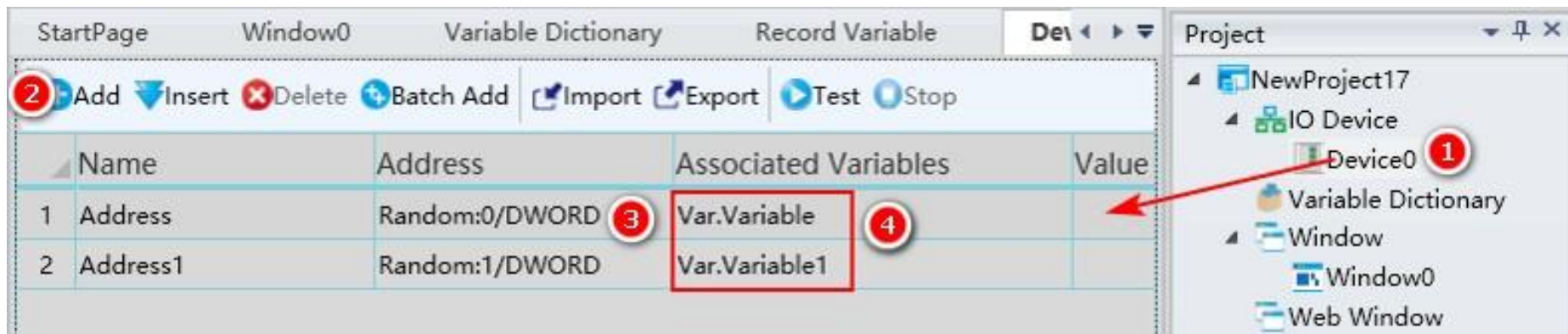
(2) Створіть імітований пристрій : Device0



- ① Клацніть правою кнопкою миші вузол «IO Device».
- ② Натисніть «Новий пристрій»
- ③④ Двічі клацніть «Симулятор»

※Див. розділ «5.10.1 Симулятор» у посібнику користувача.

(3) Створіть дві адреси моделювання в Device0 , пов'язані зі змінною , змінною1 відповідно

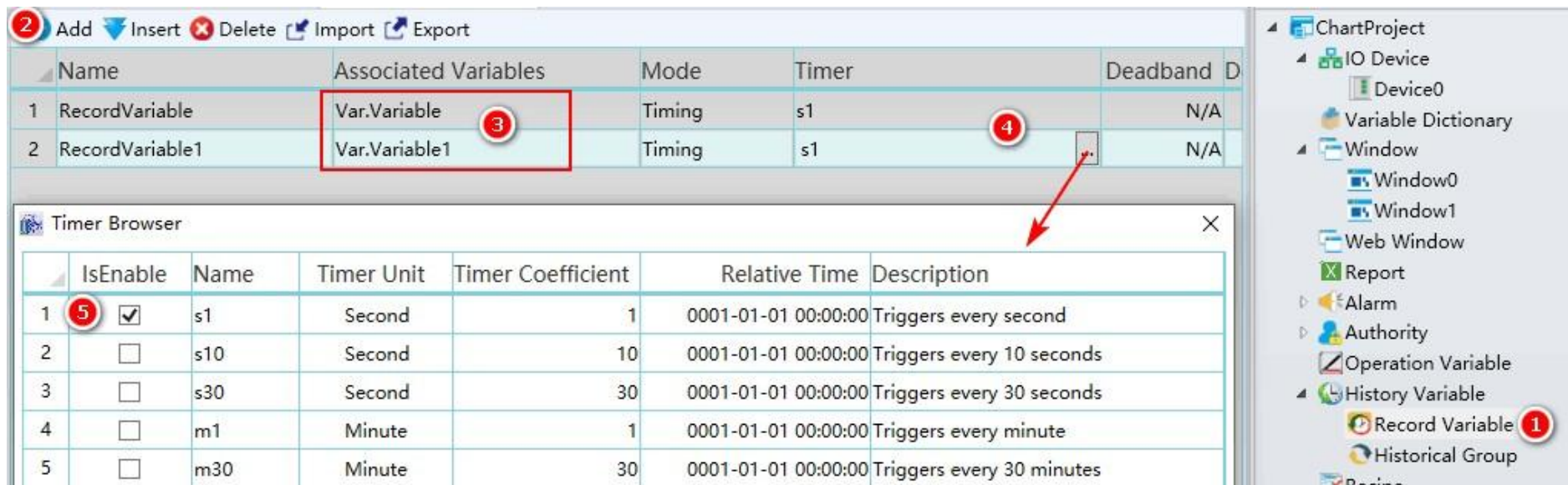


The screenshot displays the Delta IDE interface. The 'Variable Dictionary' tab is active, showing a table with the following data:

	Name	Address	Associated Variables	Value
1	Address	Random:0/DWORD	Var.Variable	
2	Address1	Random:1/DWORD	Var.Variable1	

The 'Project' pane on the right shows a tree structure with 'Device0' highlighted. A red arrow points from the 'Device0' node to the 'Associated Variables' column of the table. Red circles with numbers 1, 2, 3, and 4 highlight the 'Add' button, the 'Device0' node, the 'Address' row, and the 'Var.Variable' cell respectively.

(4) Створіть дві історичні змінні в Record Variable, які пов'язані зі змінною, Variable1 відповідно



The screenshot shows the Delta software interface. At the top, there is a menu bar with options: Add, Insert, Delete, Import, and Export. Below the menu bar is a table with columns: Name, Associated Variables, Mode, Timer, and Deadband. The table contains two rows:

	Name	Associated Variables	Mode	Timer	Deadband
1	RecordVariable	Var.Variable	Timing	s1	N/A
2	RecordVariable1	Var.Variable1	Timing	s1	N/A

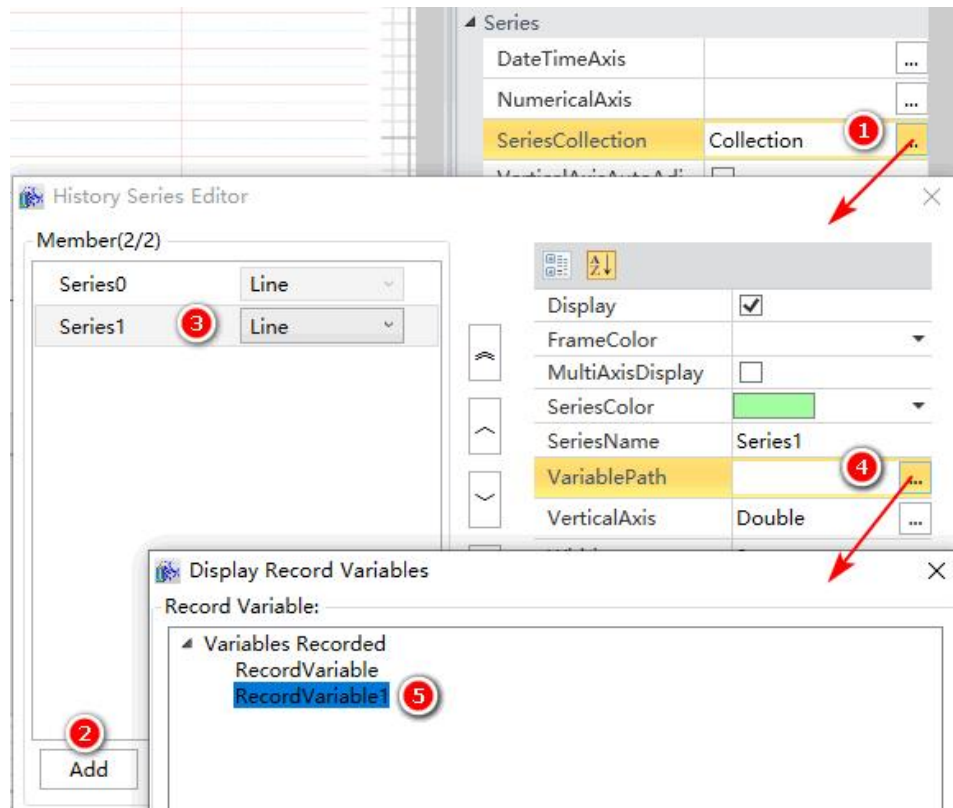
Below the table is a dialog box titled "Timer Browser". It contains a table with columns: IsEnable, Name, Timer Unit, Timer Coefficient, Relative Time, and Description. The table contains five rows:

	IsEnable	Name	Timer Unit	Timer Coefficient	Relative Time	Description
1	☒	s1	Second	1	0001-01-01 00:00:00	Triggers every second
2	☐	s10	Second	10	0001-01-01 00:00:00	Triggers every 10 seconds
3	☐	s30	Second	30	0001-01-01 00:00:00	Triggers every 30 seconds
4	☐	m1	Minute	1	0001-01-01 00:00:00	Triggers every minute
5	☐	m30	Minute	30	0001-01-01 00:00:00	Triggers every 30 minutes

On the right side of the interface, there is a tree view showing the project structure. The "History Variable" folder is expanded, showing "Record Variable" and "Historical Group".

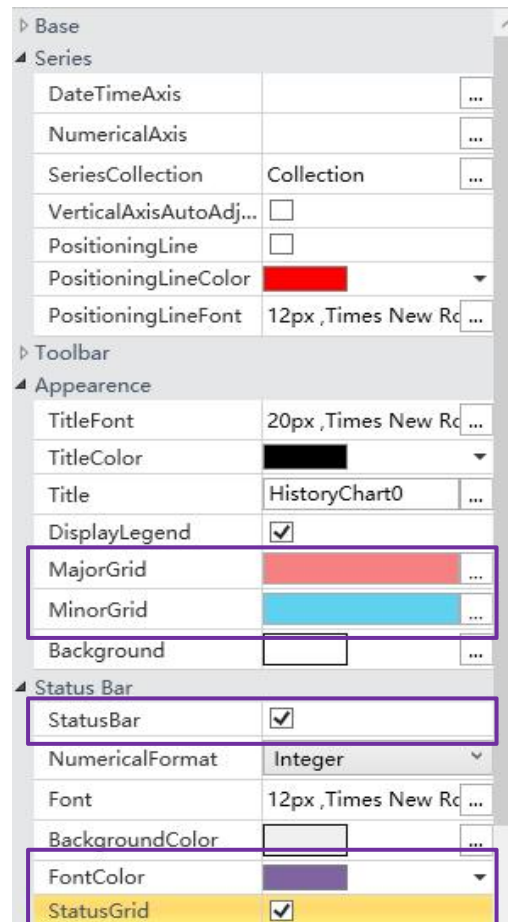
※Зверніться до розділу «14.2 Налаштування змінної запису історії» в посібнику користувача.

(5) Створіть HistoryChart0 у Window0 і додайте 2 криві в HistoryChart0, Series0 пов'язану RecordVariable, Series1 пов'язану RecordVariable1

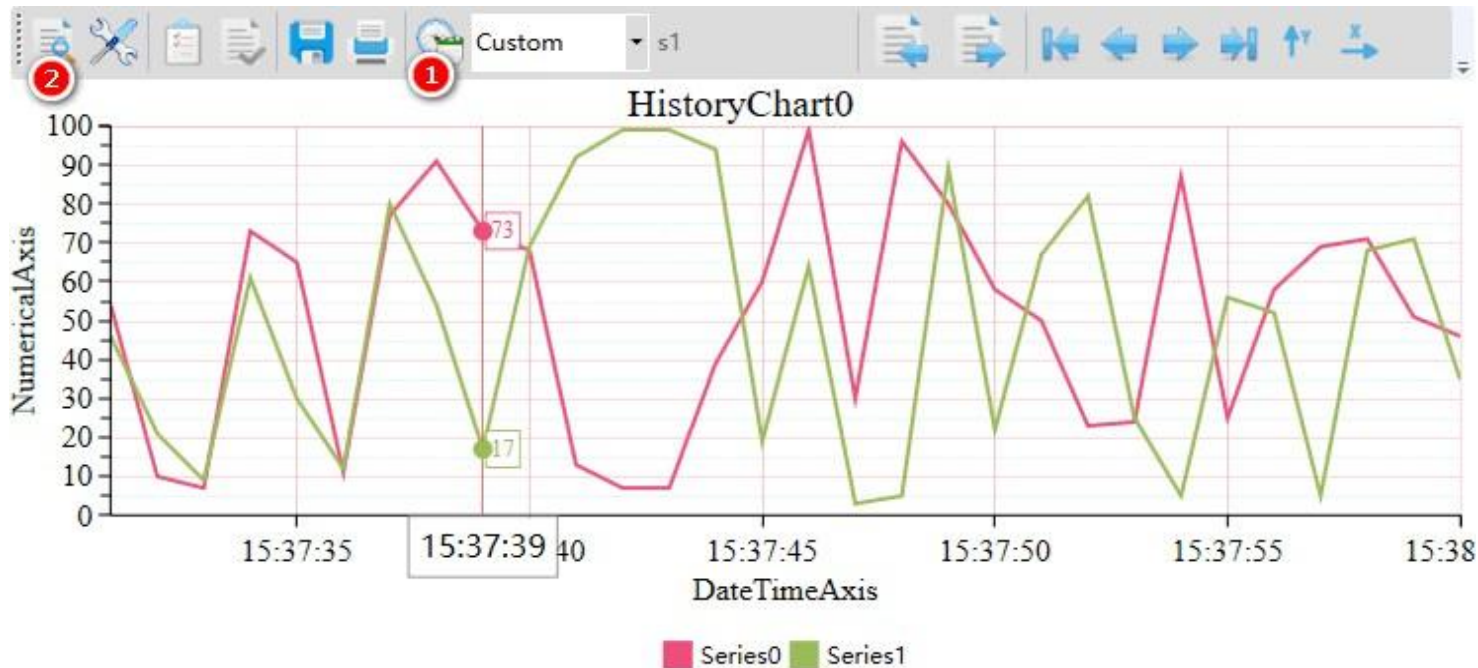


(6) Установіть деякі властивості HistoryChart0

Діаграма історії



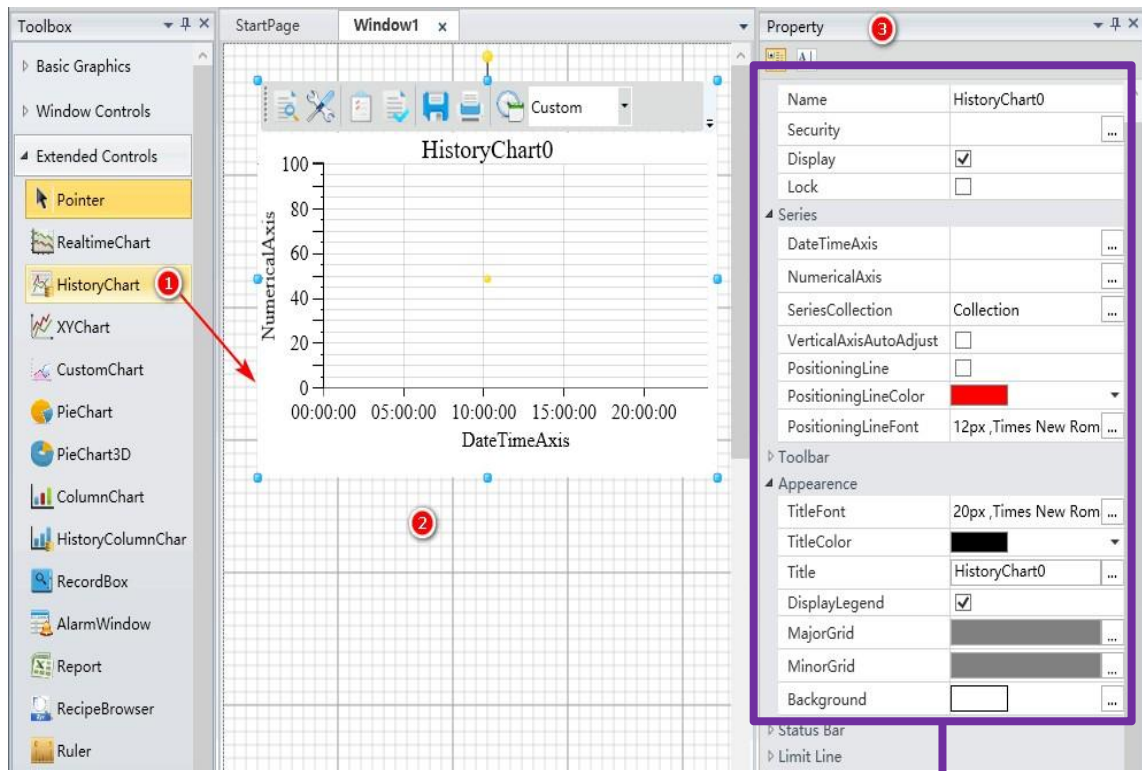
(7) Запустіть Window0, запитайте історичні дані



- ① Установити початок запиту час і час закінчення
- ② Запит

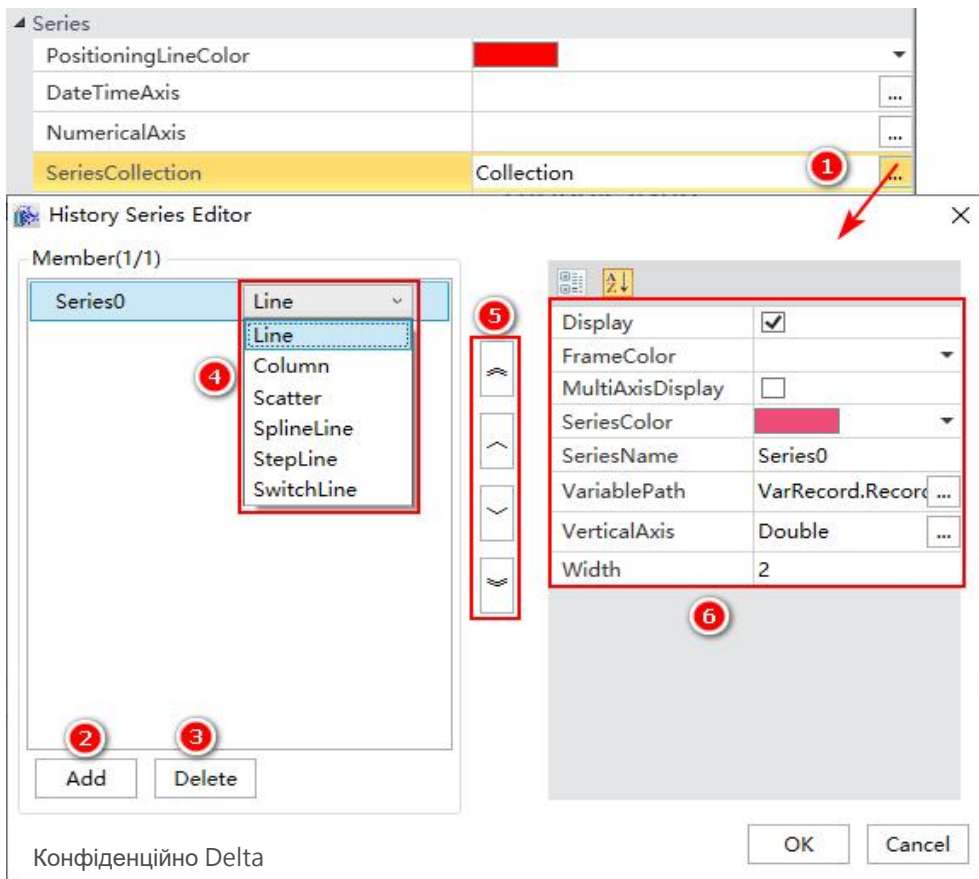
SeriesName	Minimum	Maximum	Average	Sum	Count	CurrentValue
Series0	7	99	51	1522	30	(15:37:39,73)
Series1	3	99	48	1451	30	(15:37:39,17)

➤ Вікно власності



- **DateTimeAxis** : встановити властивості DateTimeAxis
- **NumericalAxis** : встановити властивості Числова вісь
- **SeriesCollection** : додайте або видаліть серію, установіть властивості серії
- **VerticalAxisAutoChange** : установіть, чи регулювати діапазон вертикальної осі відповідно до значення точки даних
- **PositionLine** : чи відображати позиційну лінію
- **PositionLineColor** : Установіть колір позиційної лінії
- **PositionLineFont** : установіть стиль шрифту тесту на позиційному рядку
- **TitleFont** : установіть стиль шрифту заголовка
- **TitleColor** : встановити колір заголовка
- **Заголовок** : установіть вміст заголовка
- **DisplayLegend** : укажіть, чи відображати легенду
- **MajorGrid** : установіть властивість основної сітки
- **MinorGrid** : установіть властивість другорядної сітки

➤ Властивість



Діаграма історії

- ② Додайте серію в RealtimeChart
- ③ Вилучіть ряд із RealtimeChart
- ④ SeriesType : установіть тип серії
- ⑤ Змініть порядок відображення серій
- ⑥ Встановіть властивості поточної вибраної серії

➤ Приклад властивості VerticalAxisAutoAdjust

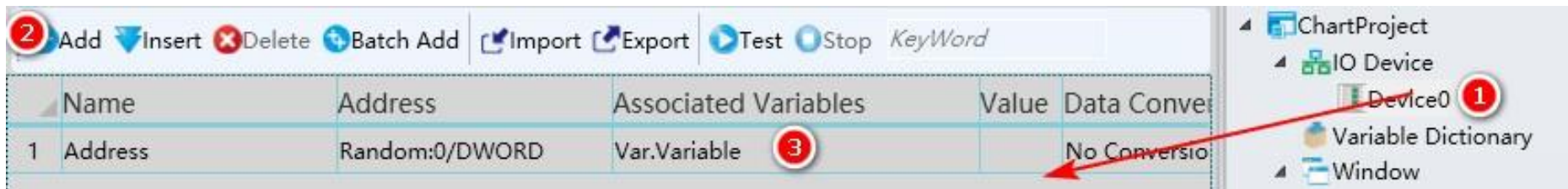
(1) Створіть 1 змінну : змінну



Name	Variable Type	Initial Value	Retentive Value	Minimum Value	Maximum Value
1 Variable	Analog	0		0	10000

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

(2) Створіть адресу (діапазон від 0 до 100) у пристрої моделювання (Device0), який пов'язано зі змінною

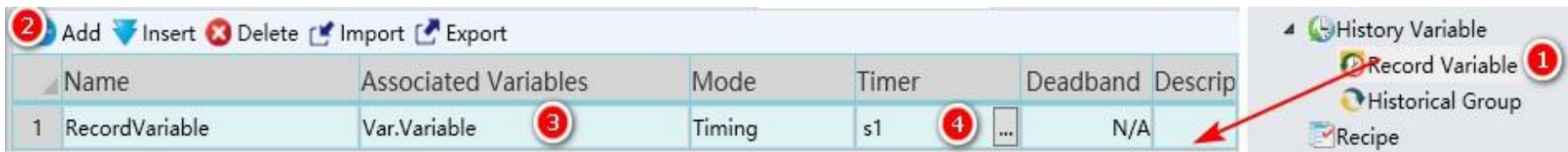


	Name	Address	Associated Variables	Value	Data Conve
1	Address	Random:0/DWORD	Var.Variable		No Conversio

ChartProject
└ IO Device
 └ Device0
Variable Dictionary
Window

※Див. розділ «5.10.1 Симулятор» у посібнику користувача.

(3) Створіть історичну змінну в Record Variable, пов'язану зі змінною



The screenshot displays the 'Record Variable' configuration interface. At the top, there is a toolbar with buttons: Add (2), Insert, Delete, Import, and Export. Below this is a table with the following columns: Name, Associated Variables, Mode, Timer, Deadband, and Descrip. The table contains one row with the following data: Name: RecordVariable, Associated Variables: Var.Variable (3), Mode: Timing, Timer: s1 (4), Deadband: N/A, and Descrip: (empty). To the right of the table is a tree view titled 'History Variable' containing the following items: Record Variable (1), Historical Group, and Recipe. A red arrow points from the 'Record Variable' item in the tree to the 'RecordVariable' row in the table.

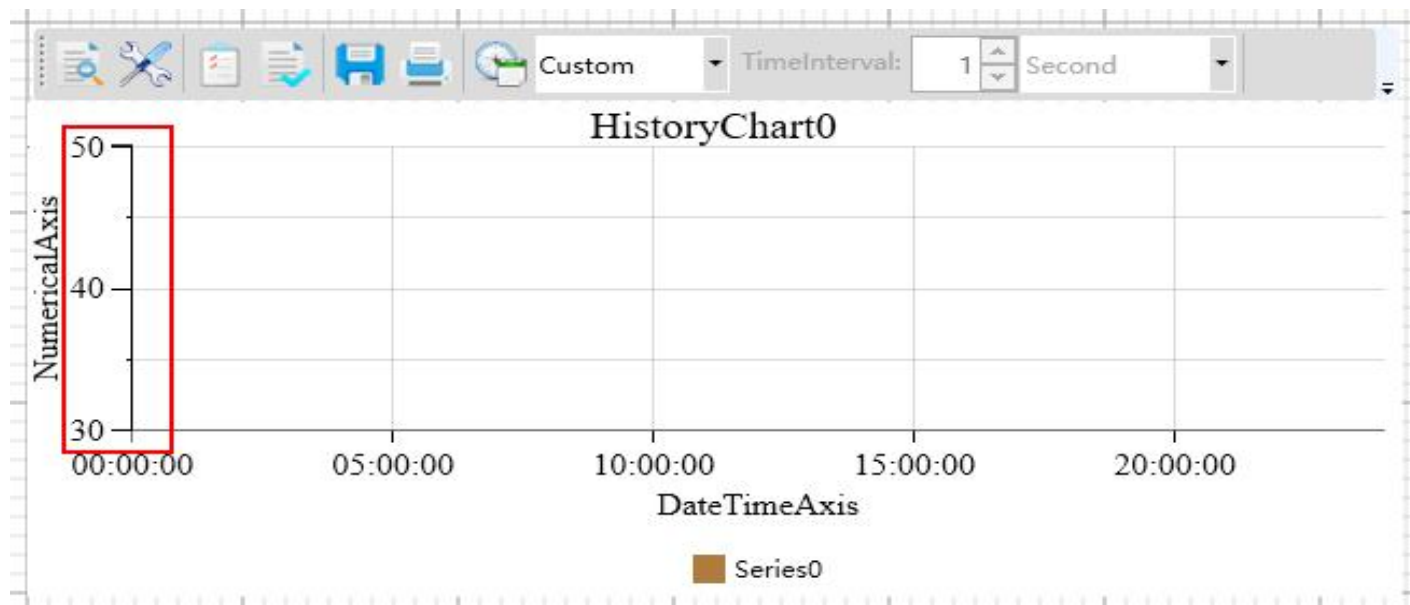
	Name	Associated Variables	Mode	Timer	Deadband	Descrip
1	RecordVariable	Var.Variable (3)	Timing	s1 (4)	N/A	

History Variable tree:

- History Variable
 - Record Variable (1)
 - Historical Group
 - Recipe

※Зверніться до розділу «14.2 Налаштування змінної запису історії» в посібнику користувача.

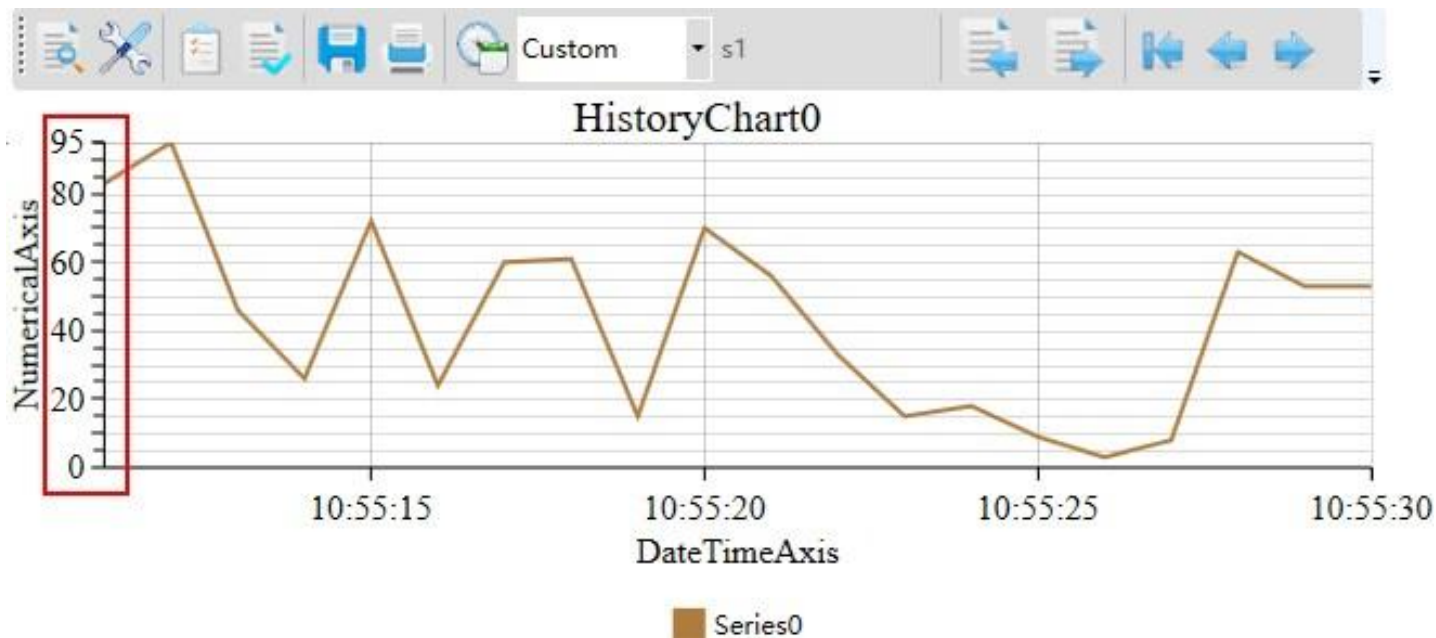
(4) Створіть HistoryChart0 і встановіть діапазон числової осі від 30 до 50, додайте Series0 в HistoryChart0, який пов'язано з історичною змінною (RecordVariable) у Record Variable



(5) Перевірте властивість VerticalAxisAutoAdjust HistoryChart0

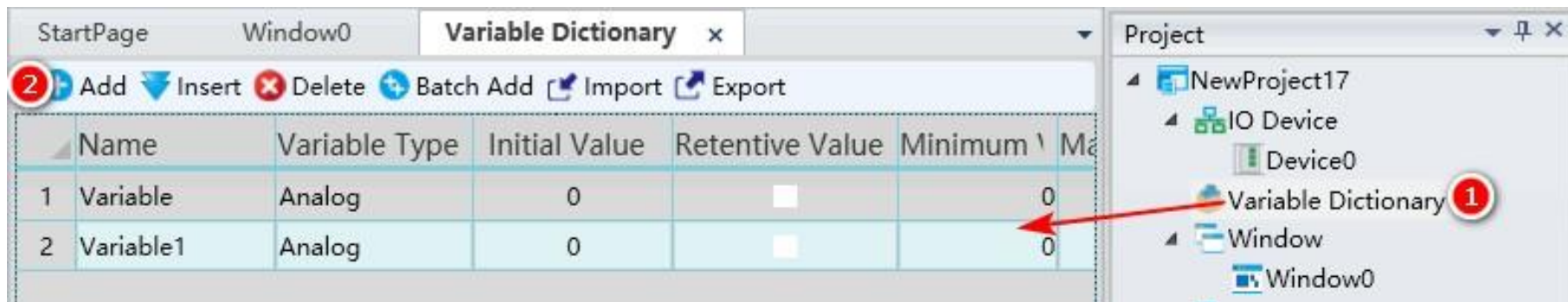
VerticalAxisAutoAdjust	☒
PositioningLine	☐

(6) Під час виконання діапазон числової осі регулюється відповідно до значення значення даних історії.



➤ Приклад властивості MultiAxisDisplay

(1) Створіть 2 змінні : змінна , змінна 1



StartPage Window0 Variable Dictionary x

2 Add Insert Delete Batch Add Import Export

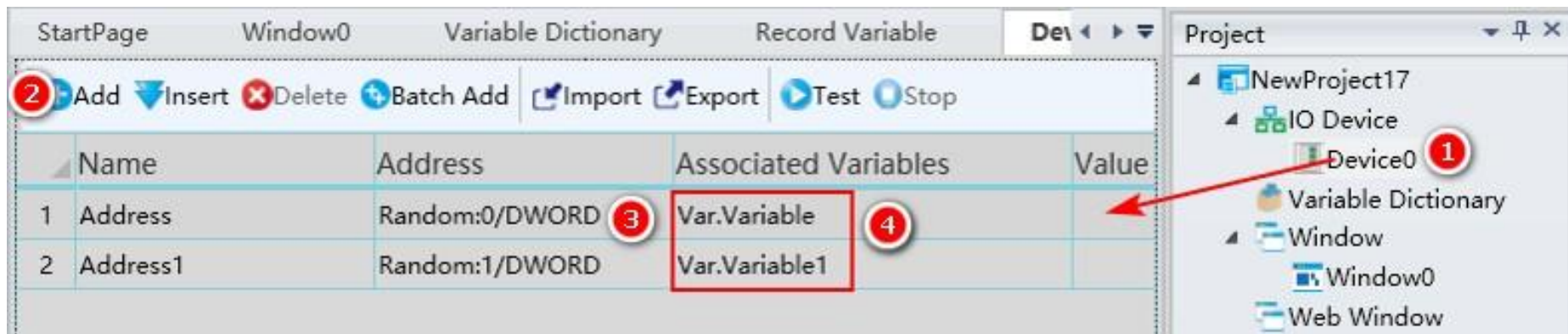
	Name	Variable Type	Initial Value	Retentive Value	Minimum \	Ma
1	Variable	Analog	0	☐	0	
2	Variable1	Analog	0	☐	0	

Project

- NewProject17
 - IO Device
 - Device0
 - Variable Dictionary 1
 - Window
 - Window0

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

(2) Створіть дві адреси моделювання в Device0, пов'язані зі змінною, змінною1 відповідно



StartPage Window0 Variable Dictionary Record Variable Dev

2 Add Insert Delete Batch Add Import Export Test Stop

	Name	Address	Associated Variables	Value
1	Address	Random:0/DWORD	Var.Variable	
2	Address1	Random:1/DWORD	Var.Variable1	

Project

- NewProject17
 - IO Device
 - Device0
 - Variable Dictionary
 - Window
 - Window0
 - Web Window

(3) Створіть дві історичні змінні (RecordVariable0 , RecordVariable1) у Record Змінна , пов'язана зі змінною , змінною1 відповідно

2 Add Insert Delete Import Export

Name	Associated Variables	Mode	Timer	Deadband
1 RecordVariable	Var.Variable 3	Timing	s1 4	N/A
2 RecordVariable1	Var.Variable1	Timing	s1	N/A

Timer Browser

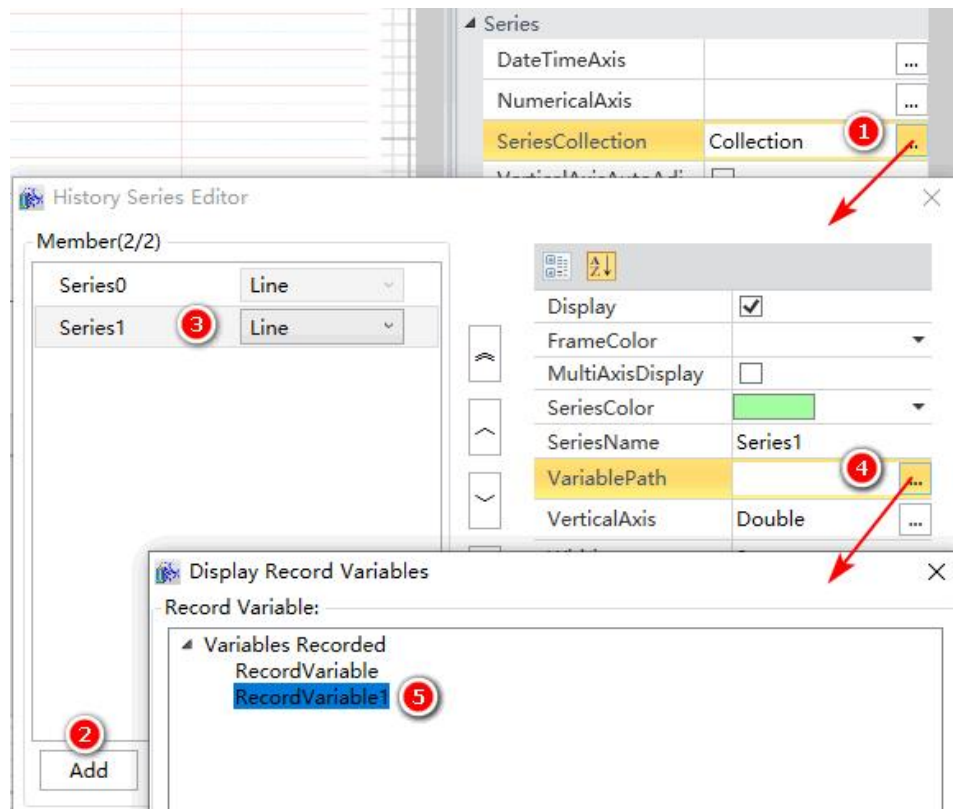
IsEnable	Name	Timer Unit	Timer Coefficient	Relative Time	Description
1 5 ☒	s1	Second	1	0001-01-01 00:00:00	Triggers every second
2 ☐	s10	Second	10	0001-01-01 00:00:00	Triggers every 10 seconds
3 ☐	s30	Second	30	0001-01-01 00:00:00	Triggers every 30 seconds
4 ☐	m1	Minute	1	0001-01-01 00:00:00	Triggers every minute
5 ☐	m30	Minute	30	0001-01-01 00:00:00	Triggers every 30 minutes

ChartProject

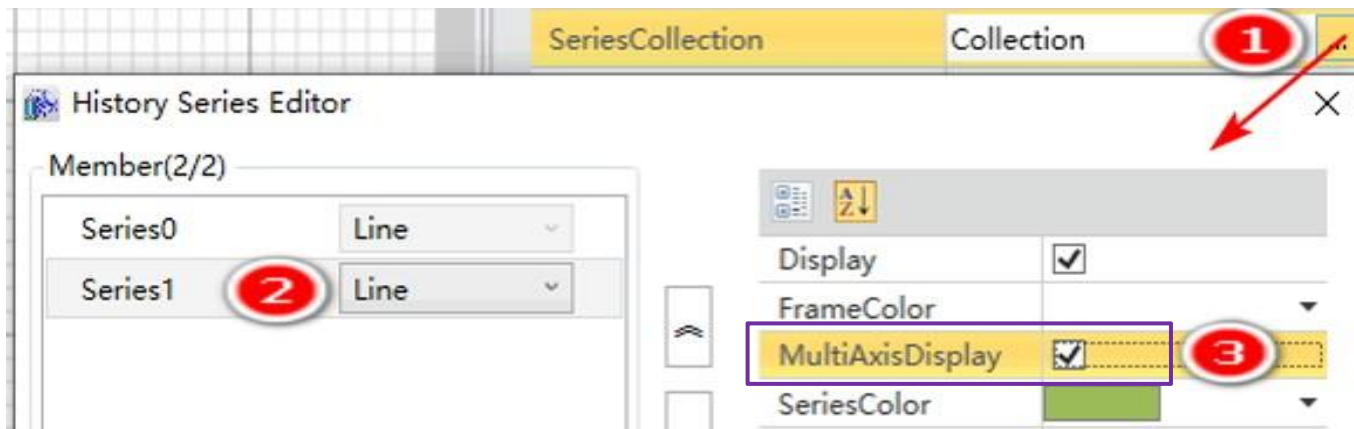
- IO Device
 - Device0
- Variable Dictionary
- Window
 - Window0
 - Window1
- Web Window
- Report
- Alarm
- Authority
- Operation Variable
- History Variable
 - Record Variable 1
 - Historical Group

※Зверніться до розділу «14.2 Налаштування змінної запису історії» в посібнику користувача.

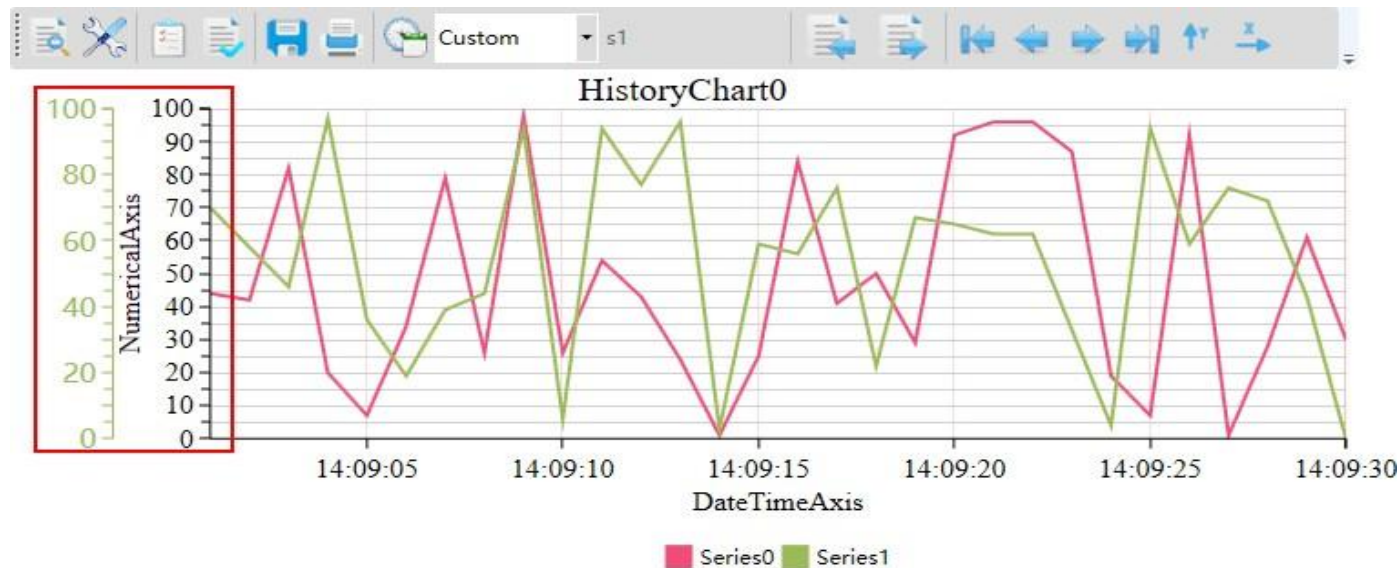
(4) Створіть HistoryChart0 у Window0 і додайте 2 криві в HistoryChart0, Series0 пов'язану RecordVariable, Series1 пов'язану RecordVariable1



(5) Перевірте властивість MultiAxisDisplay Series1 у HistoryChart0

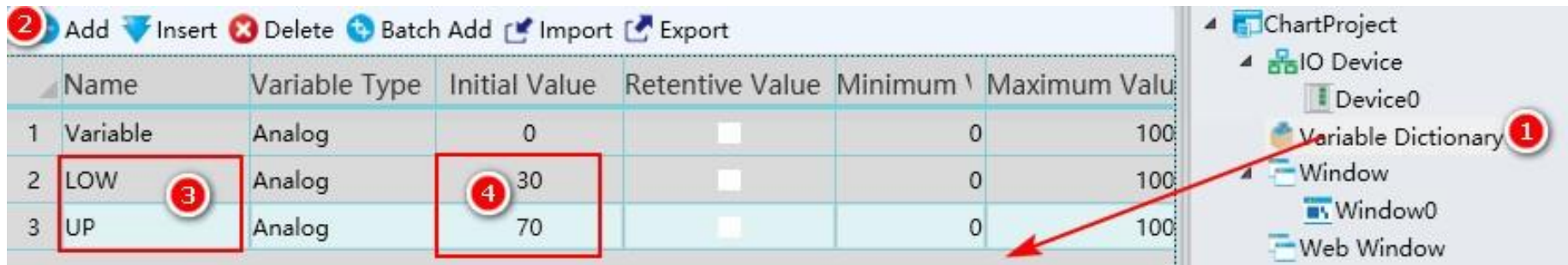


(6) Запустіть проект, дві числові осі відображатимуться на HistoryChart0



➤ Приклад властивості Limit Line

(1) Створіть три змінні: Variable, LOW, UP

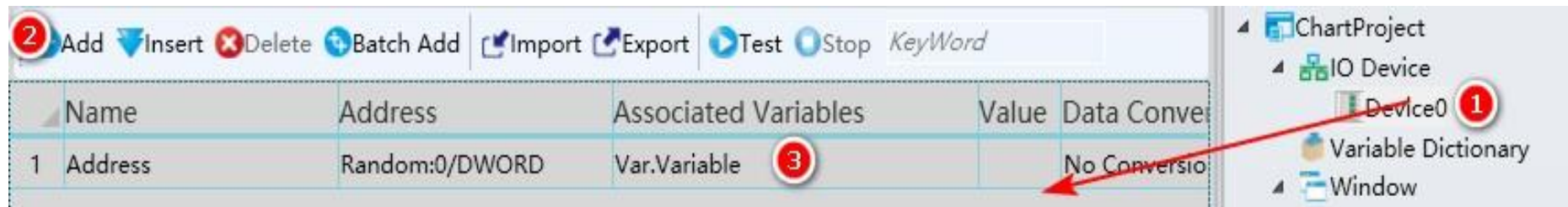


The screenshot shows the Delta software interface. On the right, the 'Variable Dictionary' is highlighted with a red circle and the number 1. On the left, a table lists three variables: 'Variable', 'LOW', and 'UP'. The 'LOW' and 'UP' rows are highlighted with red boxes and the number 3. The 'Initial Value' column for 'LOW' and 'UP' is highlighted with red boxes and the number 4. The 'Initial Value' for 'LOW' is 30 and for 'UP' is 70. The 'Minimum' and 'Maximum' values for all three variables are 0 and 100 respectively. The 'Retentive Value' column has checkboxes that are all unchecked.

	Name	Variable Type	Initial Value	Retentive Value	Minimum	Maximum
1	Variable	Analog	0	☐	0	100
2	LOW	Analog	30	☐	0	100
3	UP	Analog	70	☐	0	100

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

(2) Створіть адресу (діапазон від 0 до 100) у пристрої моделювання (Device0), який пов'язано зі змінною

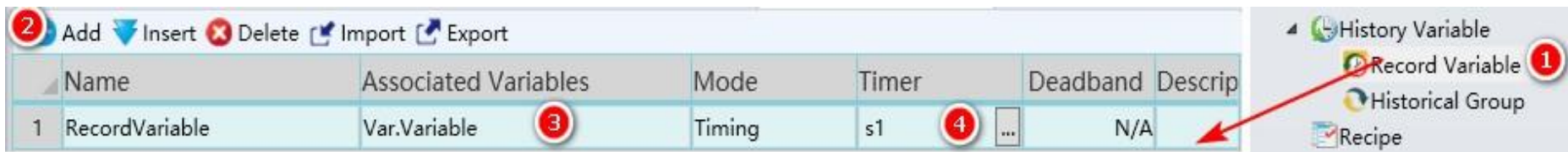


Name	Address	Associated Variables	Value	Data Conversion
1 Address	Random:0/DWORD	Var.Variable (3)		No Conversion

ChartProject
IO Device
Device0 (1)
Variable Dictionary
Window

※Див. розділ «5.10.1 Симулятор» у посібнику користувача.

(3) Створіть історичну змінну в Record Variable, пов'язану зі змінною



The screenshot displays the 'Record Variable' configuration interface. At the top, there is a toolbar with buttons: Add (2), Insert, Delete, Import, and Export. Below this is a table with the following columns: Name, Associated Variables, Mode, Timer, Deadband, and Descrip. The table contains one row with the following data: Name: RecordVariable, Associated Variables: Var.Variable (3), Mode: Timing, Timer: s1 (4), Deadband: N/A, and Descrip: (empty). To the right of the table is a tree view titled 'History Variable' containing the following items: Record Variable (1), Historical Group, and Recipe. A red arrow points from the 'Record Variable' item in the tree to the 'RecordVariable' row in the table.

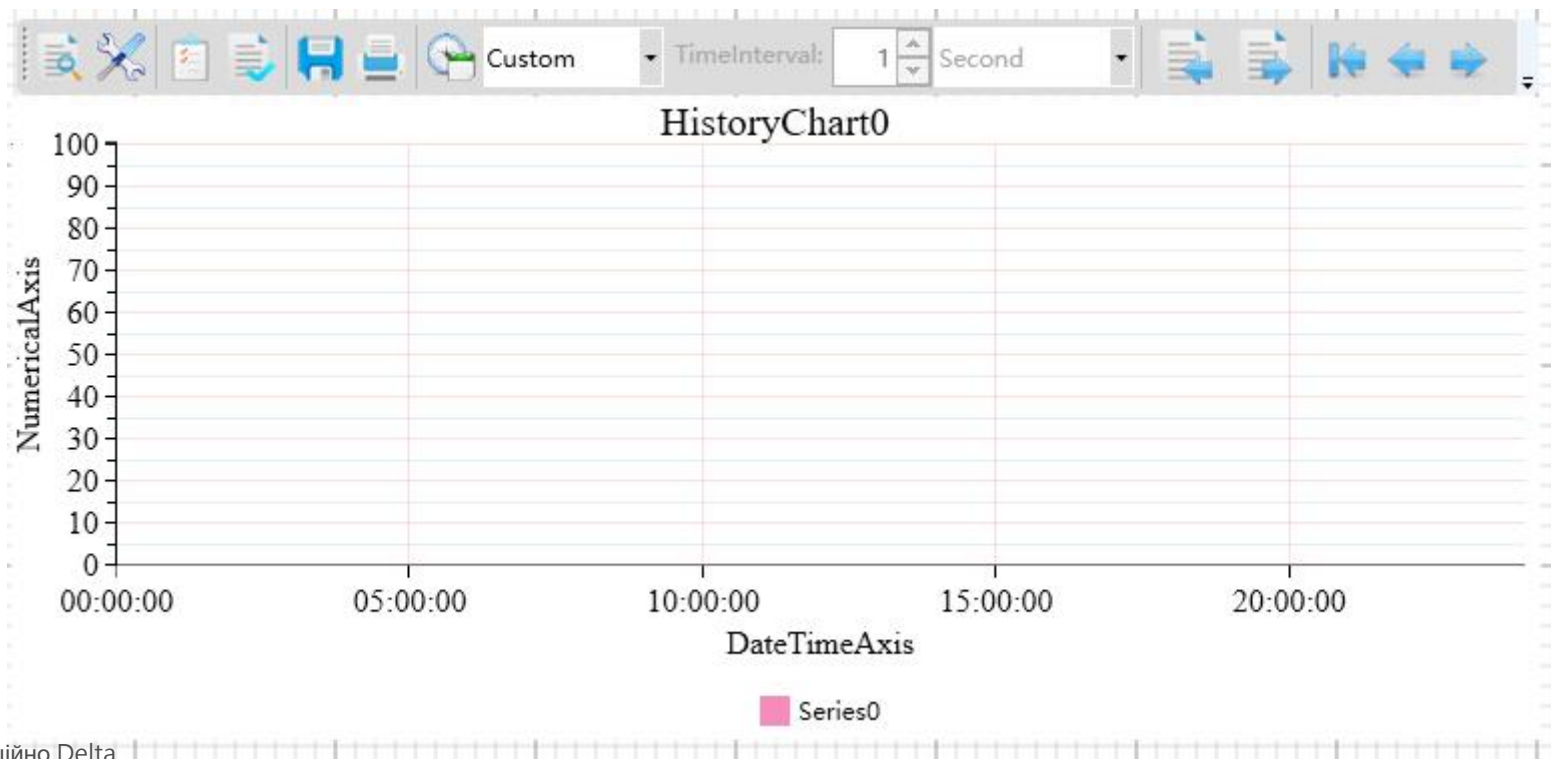
	Name	Associated Variables	Mode	Timer	Deadband	Descrip
1	RecordVariable	Var.Variable (3)	Timing	s1 (4)	N/A	

History Variable tree:

- History Variable
 - Record Variable (1)
 - Historical Group
 - Recipe

※Зверніться до розділу «14.2 Налаштування змінної запису історії» в посібнику користувача.

(4) Створіть HistoryChart0 і додайте Series0 (SeriesType — це Line), Series0 пов'язаний зі змінною hisrotrical (RecordVariable)



(5) Встановіть властивість граничної лінії для HistoryChart0

Limit Line

Thickness	1.00	①
LimitLineMaximum	Var.UP	④
LimitLineMinimum	Var.LOW	⑤
UpperAndLowerDisplay	☒	②
UpperLimitLineColor		③
LowerLimitLineColor		

- ① Встановіть товщину лінії обмеження
- ② Перевірено властивість UpperAndLowerDisplay
- ③ Встановіть колір лінії обмеження
- ④⑤ Встановіть відповідну змінну граничної лінії

Variable Browser

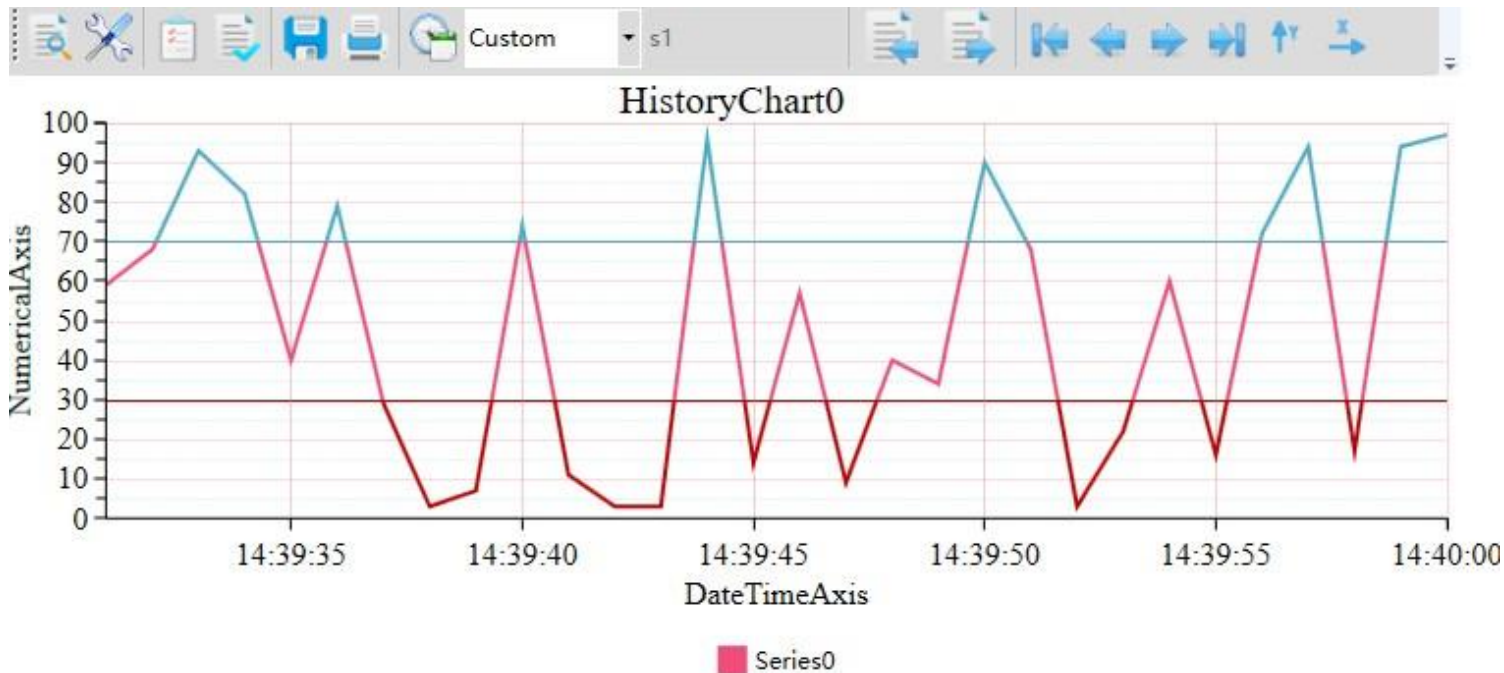
Add Group Edit

Variable Dictionary

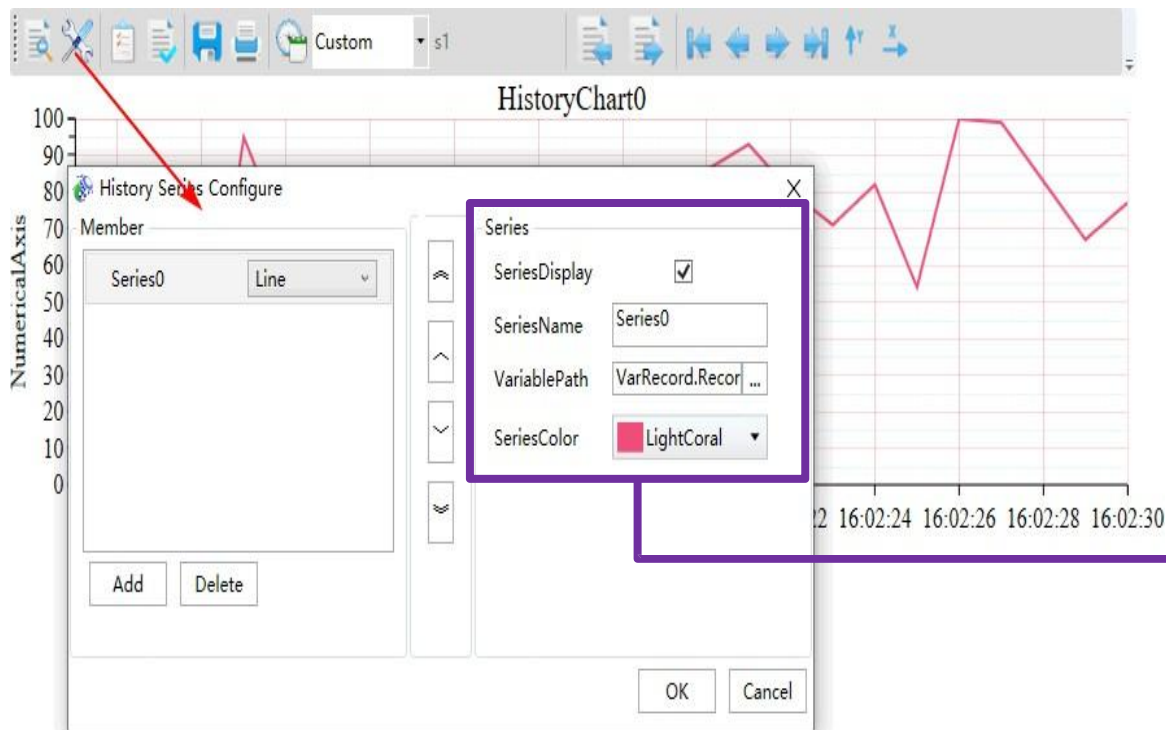
TypeFilter: ☒ Analog ☐ Digital ☐ String

Name	Type	Initial Val	Minimum	Maximum	De
Variable	Analog	0	0	10000	
LOW	Analog	30	0	10000	
UP	Analog	70	0	10000	

(6) Запустіть Window0 , обмежте відображення лінії

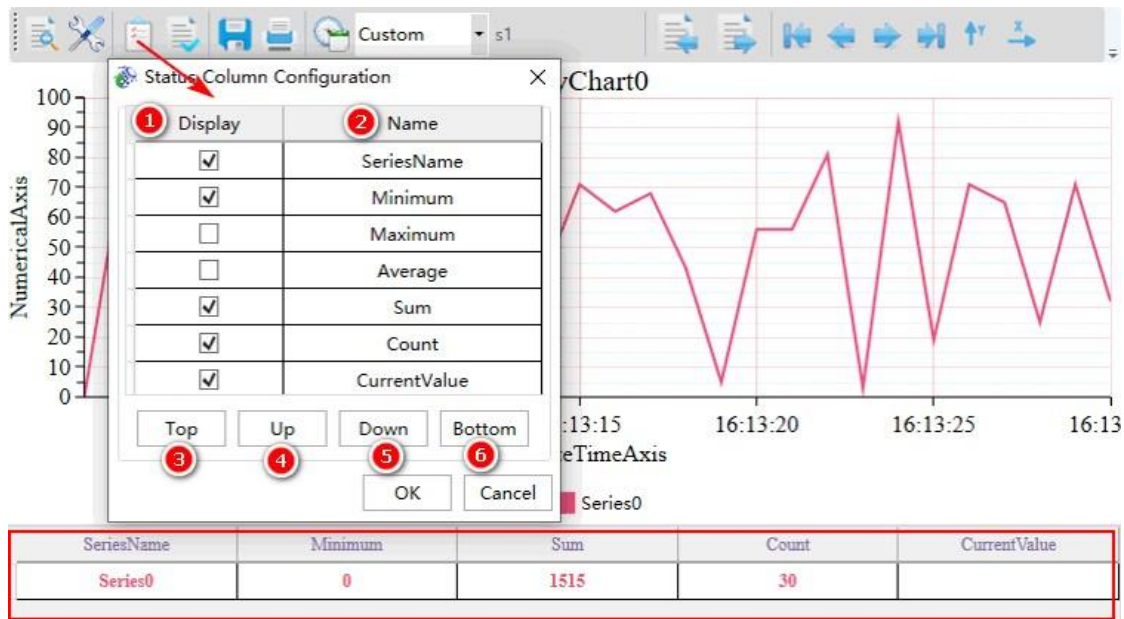


➤ Використання кнопок панелі інструментів



- **History Series Configure** (змінити властивості серій під час виконання) : **SeriesDisplay**: чи відображати серію
- **Назва серії**: змініть назву серії
- **VariablePath**: Установіть змінний шлях серії
- **SeriesColor** : установіть колір серії

➤ Використання кнопок панелі інструментів

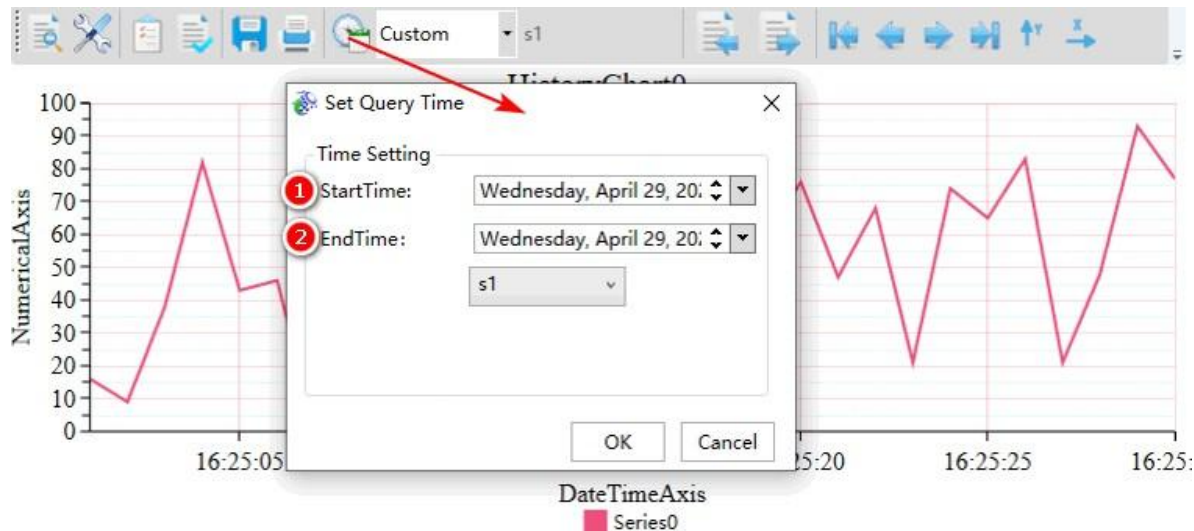


Діаграма історії

конфігурації стовпця стану :

- ① **Відображення** : позначка означає відображення, зняття позначки означає приховування цього стовпця
- ② **Назва** : назва стовпця в статусі
- ③ **Верх** : Перейти вгору
- ④ **Вгору** : рух вперед
- ⑤ **Вниз** : рух назад
- ⑥ **Знизу** : переміщення вниз

➤ Використання кнопок панелі інструментів



Вікно встановлення часу запиту:

- ① **StartTime** : установіть час початку запиту
 - ② **EndTime** : установіть час завершення запиту
- Правило: установіть правило запиту

➤ Скрипти HistoryChart

Щоб дізнатися про використання сценаріїв HistoryChart, зверніться до розділу « 20.3 Граматика та функції скриптів » → “ Модель програмування ” → « Зображення » → “ Розширені елементи керування ” → « Об’єкт HMIHistoryChart » у посібнику користувача

Methods list

Name	Description
AddNewLimitLine	Add new limit line
ArrayToString	Image array translate into string
Export	Export data
Export	Export excel
FindAnimation	Look for animation to modify the associated variables of the animation
FirstAction	The most left
GetDataTable	Store data to data table
HiddenSeries	Hidden curve
Import	Import excel
LastAction	The most right
NextAction	Right shift
OpenShowFileDialog	Get the path of save dialog select
PreAction	Left shift
Print	Print
QueryEndTime	Set the end time of the query data
QueryHistoryDate	Query of the historical data
QueryHistoryDate	Query of the historical data(with parameters)
QueryInverval	Set the query interval
QueryStartTime	Set the start time of the query data
QueryUnit	Set the query unit
RemoveAllLimitLine	Remove all the limit line
Save	Save
Save	Save as pictures
SetSeriesVariablePath	Curve correlation variable substitution
SetVarRecordRuleName	Set the name of query rule
VisibleSeries	Display curve

Property list

Name	Description
Height	Height
IsShow	Whether display the specified object
IsShowBtnConfig	Whether display the curve configuration button
IsShowBtnCrossLine	Whether display the positioning line
IsShowBtnMenu	Whether display the import and export button
IsShowBtnPrint	Whether display the print button
IsShowBtnSave	Whether display the save button
IsShowBtnUpdate	Whether display the query button
IsShowCbTime	Whether display the quick time setting button
IsShowSetTime	Whether display the time setting button
IsShowStatusBar	Whether display the status column
IsShowToolBar	Whether display the tools column
Left	The left coordinate
Name	Name
NumberAxisMaximum	Get or set the maximum value of numerical axis
NumberAxisMinimum	Get or set the minimum value of numerical axis
NumberInterval	Get or set the maximum scale of numerical axis
NumberSmallInterval	Get or set the minimum scale of numerical axis
ToolTip	Tooltip text
Top	The top coordinate
Width	Width
ZIndex	Layer index

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta,
відвідайте www.deltaww.com.



DIAView

3Bit

朱明杰 杨阳

20.03.2020



Контур

- Огляд звіту
- Звіт про використання
- Шаблон звіту
- Створення звіту
- Звіт про налаштування конфігурації

призначення

У цьому розділі ви дізнаєтеся....

- ... Основні операції та керування звітами

- ... Функції, що підтримуються звітами

Контур

- Огляд звіту
- Звіт про використання
- Шаблон звіту
- Створення звіту
- Звіт про налаштування конфігурації

➤ Функція звіту DIAView

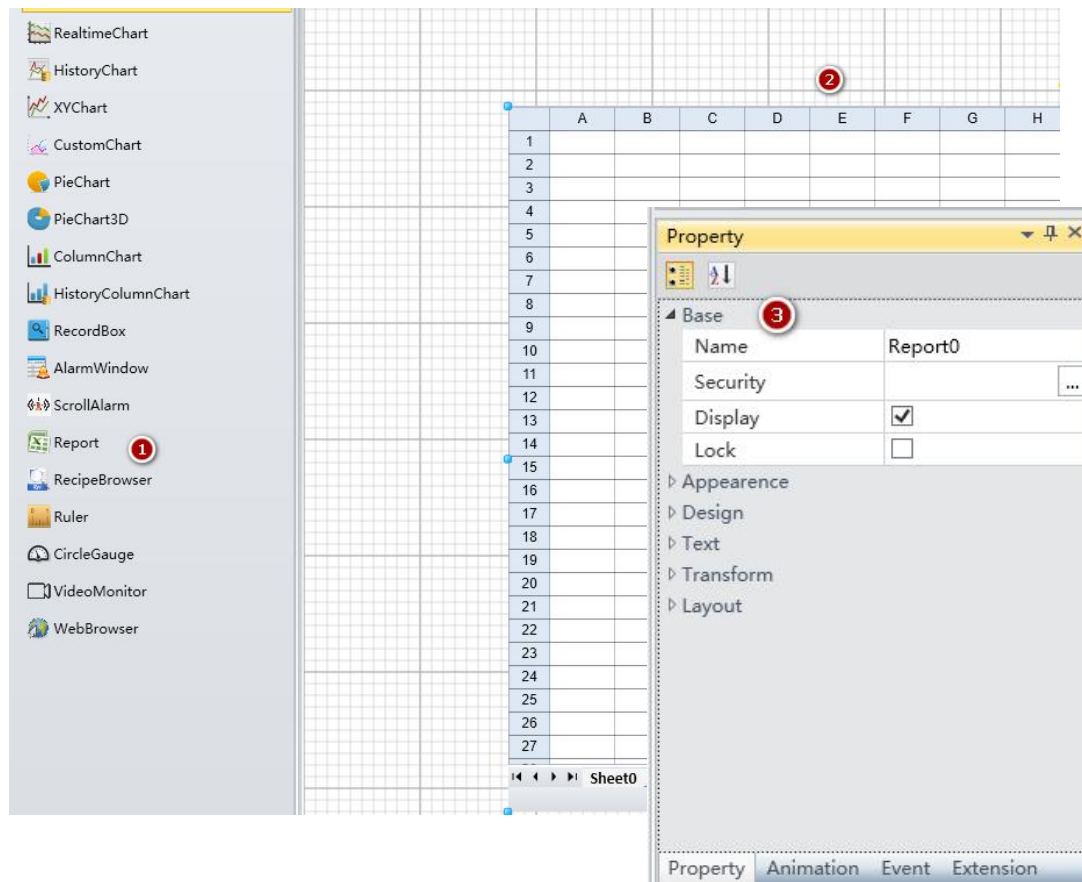
- Забезпечує інструмент звітності в режимі роботи, подібному до Excel, для створення загальних виробничих звітів (**щоденних звітів, місячних звітів, річних звітів тощо**), а також може гнучко створювати звіти в різних складних форматах для виконання запиту в режимі реального часу , історичні та статистичні дані, відображення, друк і вихід.
- Звіт може не тільки відображати дані історичної бази даних у реальному часі, але також може виконувати дані доступ і запити до типових реляційних баз даних, таких як **MySQL, SQL Server, Oracle і Access.**



- Огляд звіту
- **Звіт про використання**
- Шаблон звіту
- Створення звіту
- Звіт про налаштування конфігурації

➤ Приклад створення звіту

- ① Натисніть звіт на панелі інструментів
- ② Перетягніть у потрібне положення на полотні
- ③ Налаштувати властивості звіту



➤ Властивості звіту:

① база:

Основні властивості, такі ж, як і в інших графічних елементів керування

② Зовнішній вигляд:

Зовнішній вигляд може вибрати, чи відображати формулу, навігацію, рядок меню та панель інструментів.

③ Дизайн:

Розробити та встановити шаблон звіту для звіту

④ Текст:

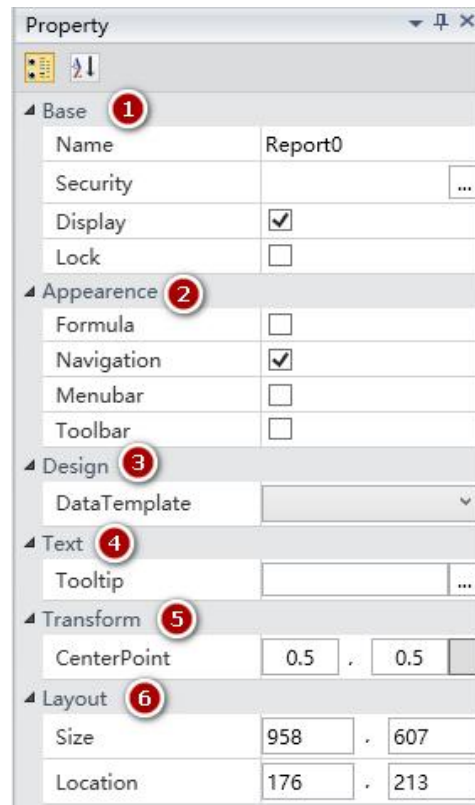
Повідомити оперативну текстову інформацію

⑤ Трансформувати:

Містить центральну точку звіту

⑥ Макет:

Розмір і положення звіту на полотні можна регулювати числовими значеннями



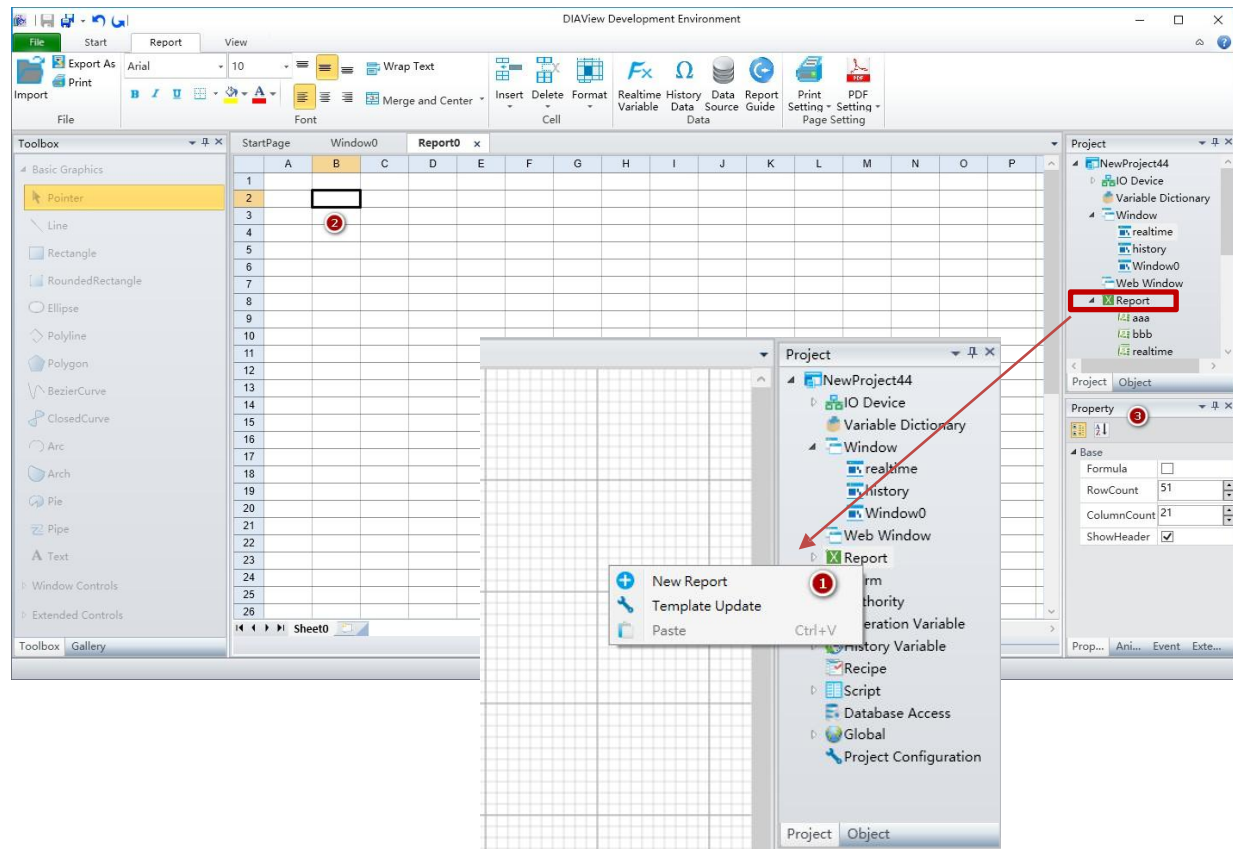
Property	
Base ①	
Name	Report0
Security	...
Display	☒
Lock	☐
Appearance ②	
Formula	☐
Navigation	☒
Menubar	☐
Toolbar	☐
Design ③	
DataTemplate	...
Text ④	
Tooltip	...
Transform ⑤	
CenterPoint	0.5 . 0.5
Layout ⑥	
Size	958 . 607
Location	176 . 213

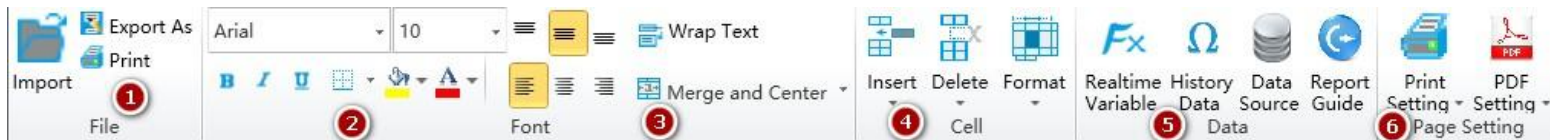
- Огляд звіту
- Звіт про використання
- **Шаблон звіту**
- Створення звіту
- Звіт про налаштування конфігурації

➤ Створіть шаблон звіту:

• Крок 1

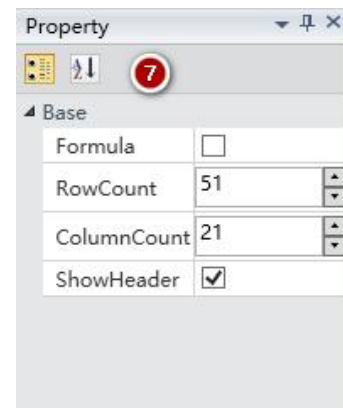
- ① Клацніть звіт правою кнопкою миші
вузол проекту
- ② Створіть новий шаблон звіту
- ③ Налаштувати шаблон звіту властивості





- **Крок 2**

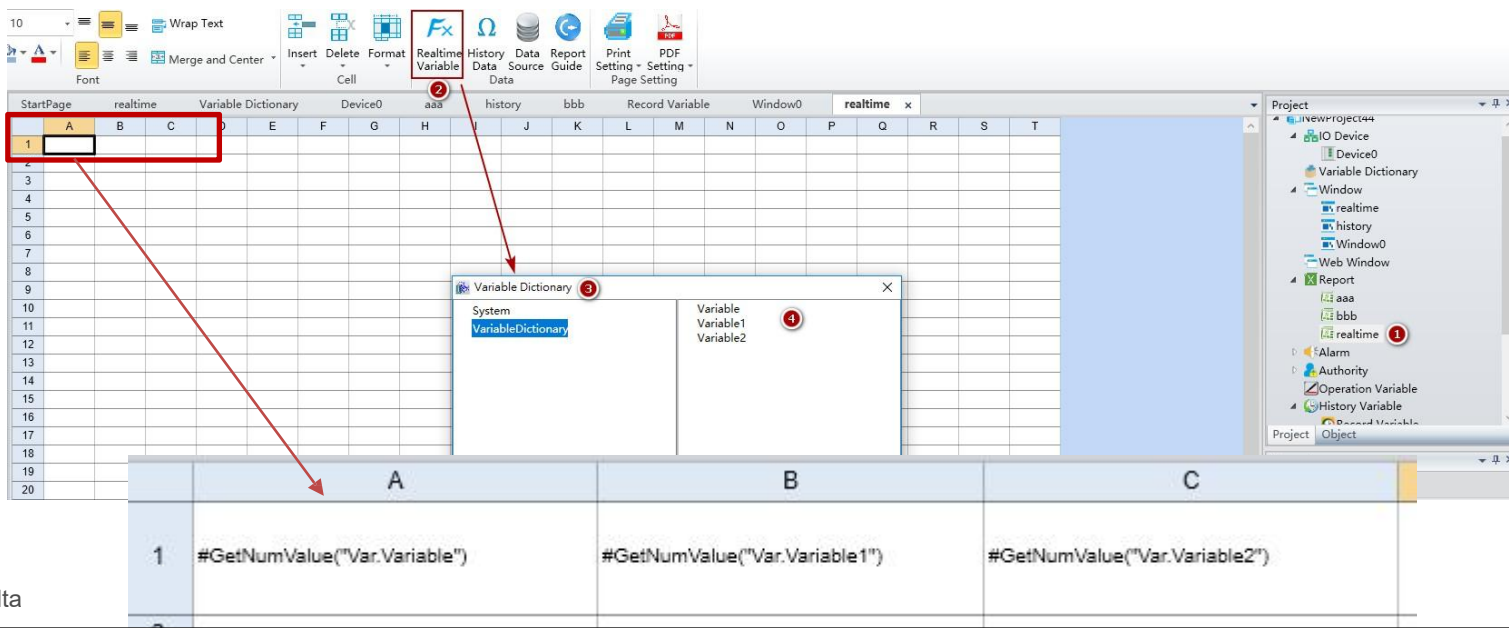
- ① Функція імпорту та експорту друку
- ② Встановити шрифт тексту в таблиці, можна змінити розмір і колір заливки тощо.
- ③ Встановіть формат таблиці, ви можете встановити центр вирівнювання та розрив рядка та об'єднання
- ④ Формат клітинок, операції, подібні до EXCEL
- ⑤ Встановіть дані, ви можете прив'язувати змінні в реальному часі або історичні дані, ви також можете додавати джерела даних і майстри звітів
- ⑥ Налаштування друку та PDF
- ⑦ Основні властивості шаблонів звітів, у тому числі чи потрібно відображати рядок формул, кількість рядків і стовпців, а також заголовки рядків і стовпців



- Огляд звіту
- Звіт про використання
- Шаблон звіту
- Створення звіту
- Звіт про налаштування конфігурації

➤ Створіть звіт у реальному часі

- ① Створіть новий шаблон звіту
- ② Виберіть рядок меню шаблону звіту, щоб зв'язати змінні реального часу
- ③ Виберіть змінну, яку потрібно прив'язати

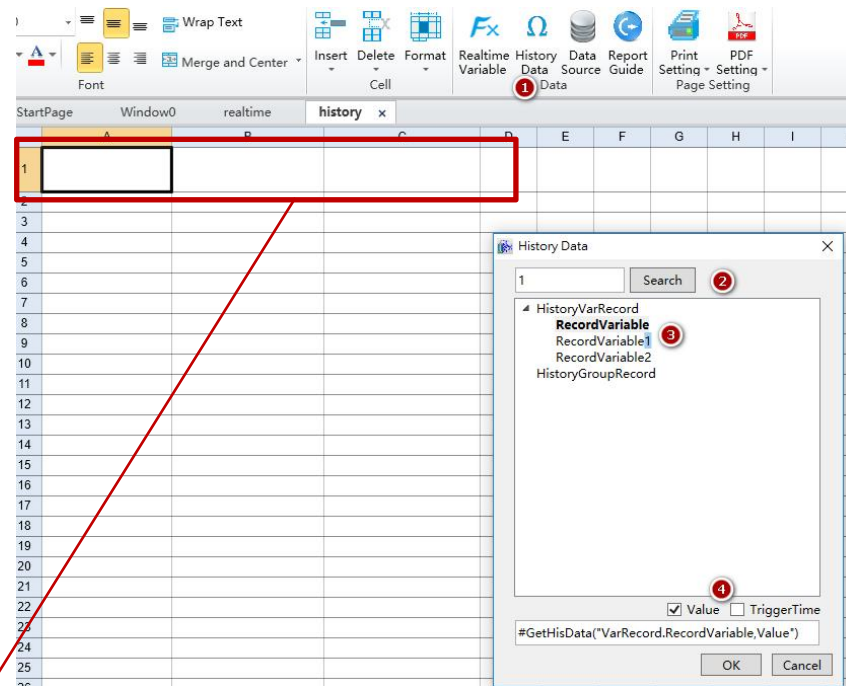


The screenshot illustrates the steps for creating a real-time report in the Delta software. The interface shows a spreadsheet-like editor with columns A, B, and C. A red box highlights the 'Realtime Variable' menu item in the top toolbar. A red arrow points from this menu item to the 'Variable Dictionary' dialog box, which is open in the center. The dialog box shows a list of variables, with 'Variable1' selected. Another red arrow points from the 'Variable1' entry in the dialog box to the formula in cell A1 of the spreadsheet, which is '#GetNumValue("Var.Variable")'. The 'Project' tree view on the right side of the interface shows the hierarchy of the project, including 'Device0', 'Variable Dictionary', 'Window', 'history', 'Window0', 'Web Window', 'Report', 'aaa', 'bbb', 'realtime', 'Alarm', 'Authority', 'Operation Variable', and 'History Variable'.

	A	B	C
1	#GetNumValue("Var.Variable")	#GetNumValue("Var.Variable1")	#GetNumValue("Var.Variable2")

➤ Створіть звіт історії

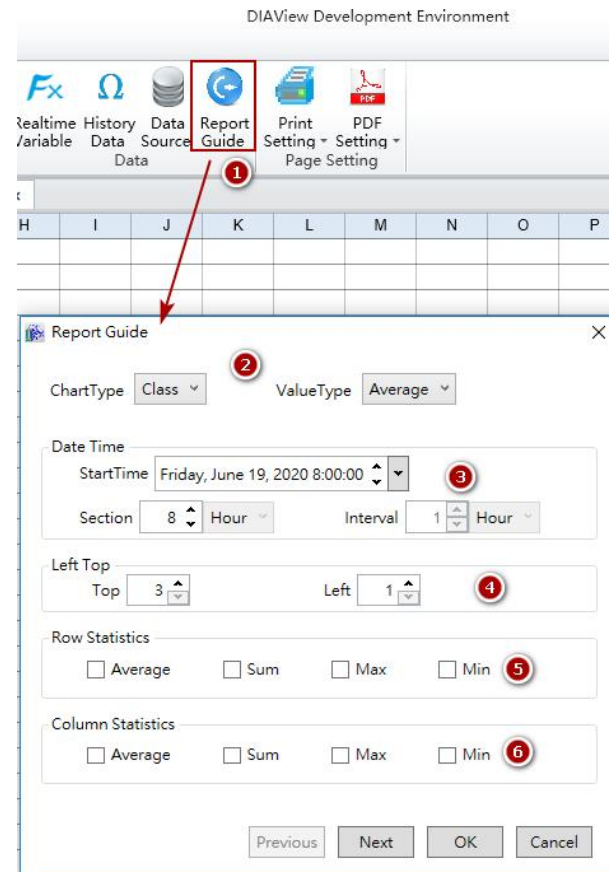
- ① Виберіть рядок меню шаблону звіту, щоб зв'язати історичні змінні
- ② Виберіть потрібні історичні змінні
- ③ Виберіть необхідну історичну змінну
- ④ Існує дві форми історичних змінних:
значення та час запуску



	A	B	C	D
1	#GetHisData("VarRecord.RecordVariable,Value")	#GetHisData("VarRecord.RecordVariable1,Value")	#GetHisData("VarRecord.RecordVariable2,Value")	#GetHisData("VarRecord.RecordVariable2,TriggerTime")

➤ Створіть посібник для звіту

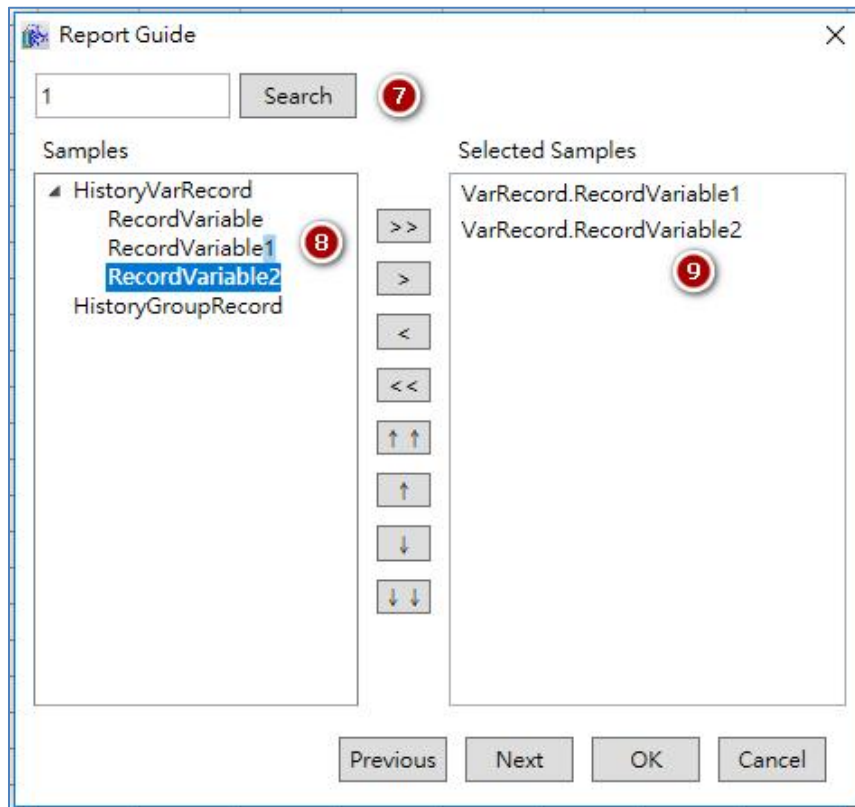
- ① Виберіть «Майстер звітів» на панелі меню шаблону звіту.
- ② Виберіть тип значка та тип значення.
- ③ Виберіть інтервал часу початку
- ④ Виберіть початкову позицію значка
- ⑤ Встановити статистику лінії
- ⑥ Встановити статистику стовпця



Створення звіту

➤ Створіть посібник для звіту

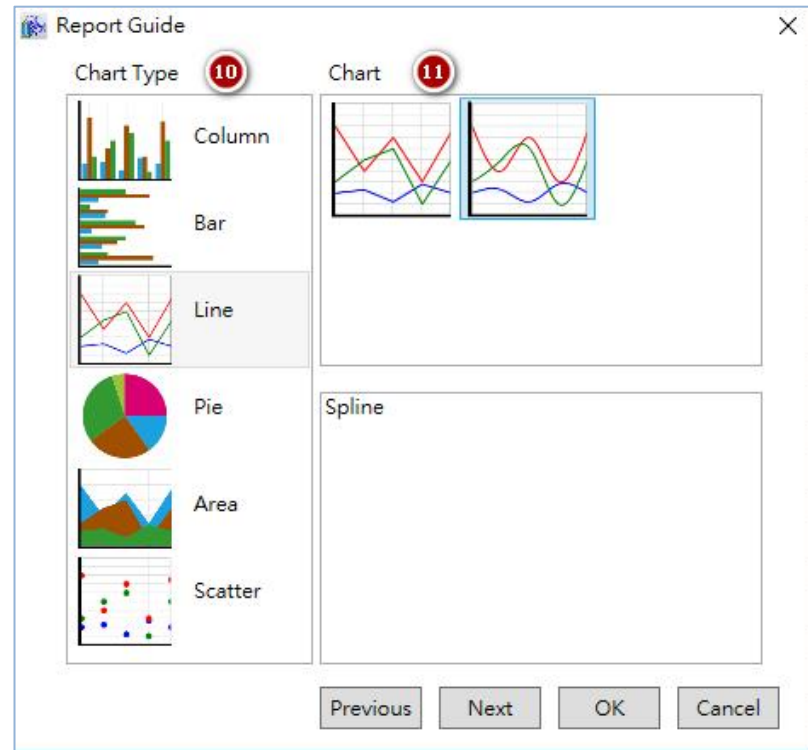
- ⑦ Швидко знайти потрібну змінну
- ⑧ Виберіть потрібну змінну
- ⑨ Визначте вибрані змінні та внесіть корективи



Створення звіту

➤ Створіть посібник для звіту

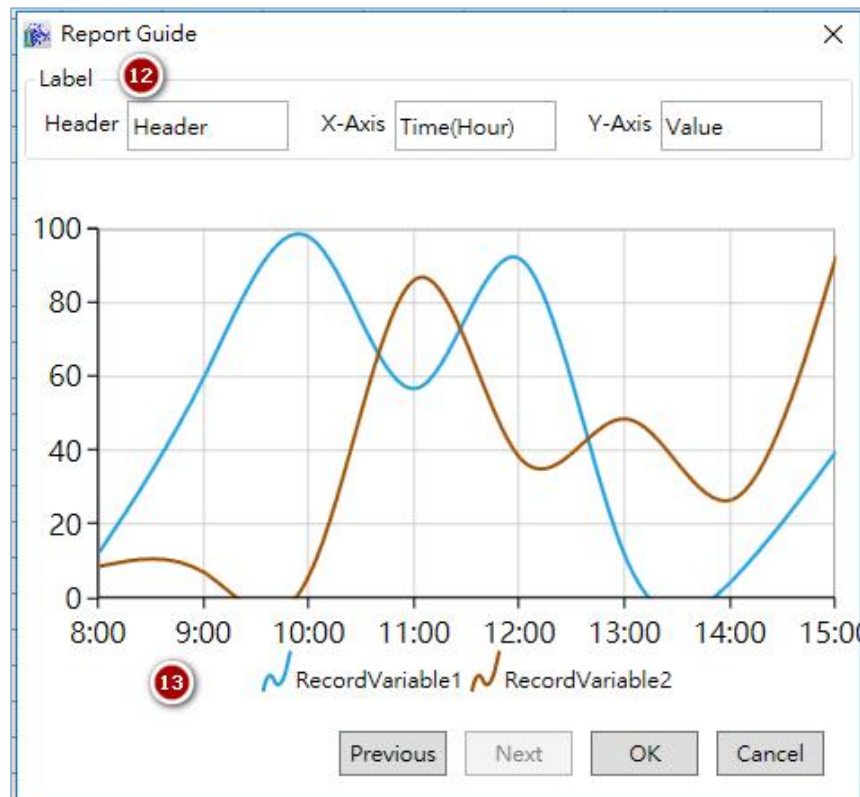
- ⑩ Виберіть тип створеної діаграми
- ⑪ Виберіть «Створити стиль діаграми».



Створення звіту

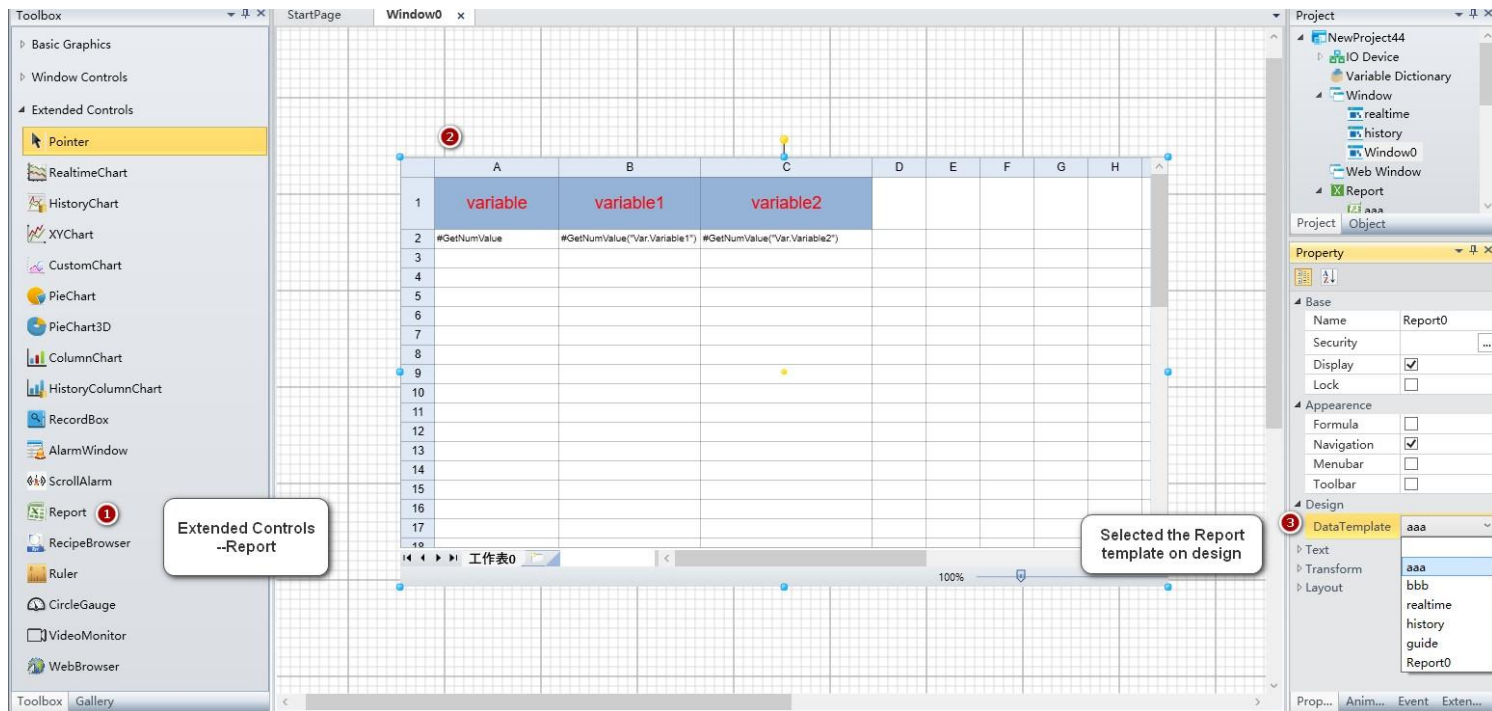
➤ Створіть посібник для звіту

- ⑫ Змініть заголовок і вміст осі XY
- ⑬ Створити попередній перегляд діаграми



➤ Приклад налаштування звіту

- Створіть готовий шаблон звіту та прив'язку звіту.



Extended Controls --Report

Selected the Report template on design

	A	B	C	D	E	F	G	H
1	variable	variable1	variable2					
2	#GetNumValue	#GetNumValue("Var.Variable1")	#GetNumValue("Var.Variable2")					
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								

Project: NewProject44

- IO Device
- Variable Dictionary
- Window
 - realtime
 - history
 - Window0
- Web Window
- Report
 - aaa

Property: Base

Name	Report0
Security	
Display	☒
Lock	☐

Design: DataTemplate

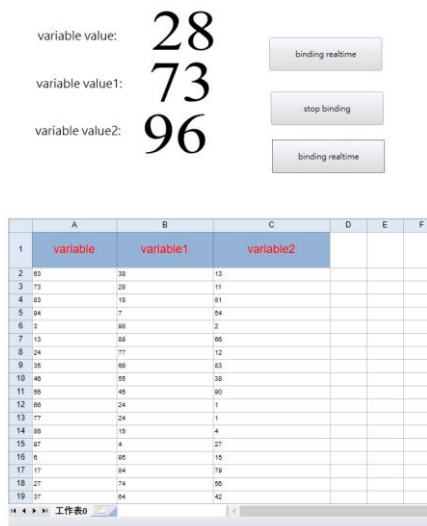
- aaa
- bbb
- realtime
- history
- guide
- Report0

- Огляд звіту
- Звіт про використання
- Шаблон звіту
- Створення звіту
- Звіт про налаштування конфігурації



➤ Приклад тесту Report Guide

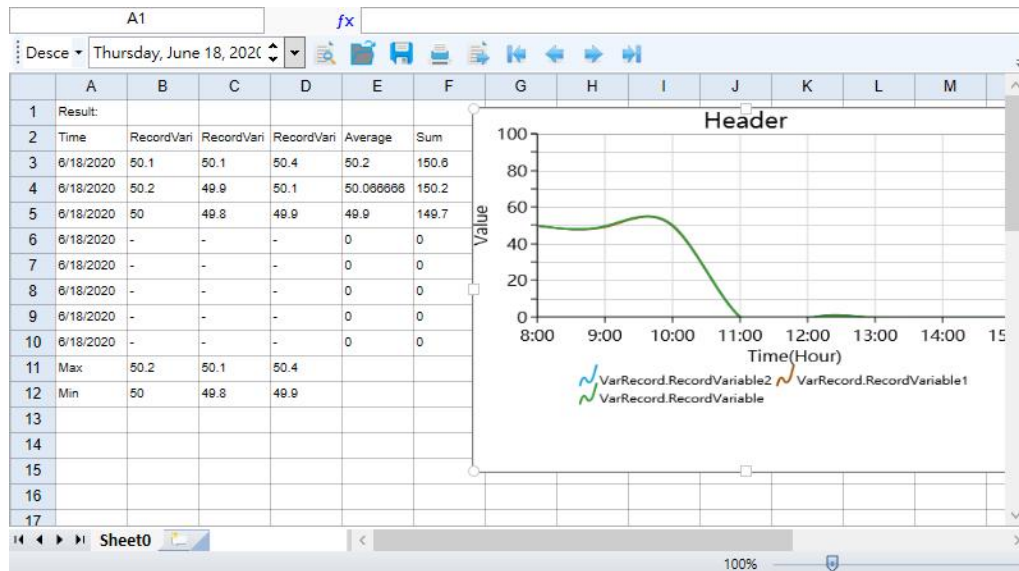
- Екран виконання звіту в реальному часі
- Оновлення даних у реальному часі після прив'язки змінних



Звіт про налаштування конфігурації

➤ Керівництво по звіту під час виконання

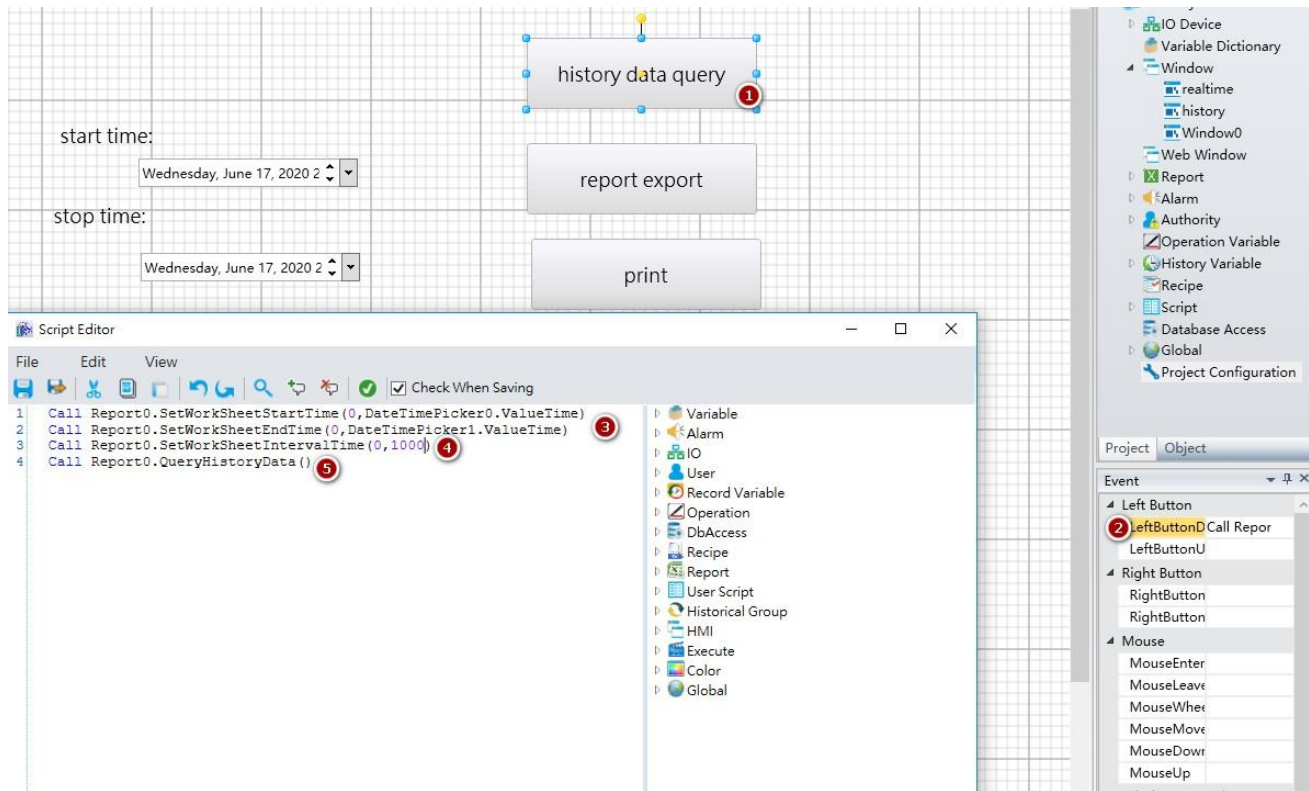
- Дата початку запиту - 18 червня, а дані за 8 годин запитується. Налаштовується середня, загальна, максимальна та мінімальна статистика, а також генерується діаграма статистики даних.



Приклад сценаріїв звітів

➤ У розробці часу

- ① Створіть кнопку.
- ② Налаштуйте ліву кнопку прес сценарій події.
- ③ Виберіть час початку та час закінчення
- ④ Встановити інтервал запиту
- ⑤ Запит історичних даних



✂Повідомлення про події, пов'язані з іншими сценаріями, можна переглянути в посібнику користувача 20.3.

Приклад сценаріїв звітів

➤ Під час виконання

① Виберіть час початку та час закінчення

② Натисніть Дані запиту кнопку

③ Запитаний історичний дані відображаються в таблиці

variable value: 40
variable value1: 61
variable value2: 73

start time:

Wednesday, June 17, 2020 2:25

stop time:

Thursday, June 18, 2020 2:25

2

history data query

1

report export

print

	A	B	C	D	E	F	G	H	I	J	K
1	variable	variable1	variable2	time							
2	0	0	0	2020/06/17 14:50:21							
3	18	83	12	2020/06/17 14:50:22							
4	29	72	89	2020/06/17 14:50:23							
5	39	82	34	2020/06/17 14:50:24							
6	50	51	45	2020/06/17 14:50:25							
7	61	40	78	2020/06/17 14:50:26							
8	72	29	22	2020/06/17 14:50:28							
9	81	20	43	2020/06/17 14:50:29							
10	92	9	43	2020/06/17 14:50:30							
11	2	99	86	2020/06/17 14:50:31							
12	12	89	52	2020/06/17 14:50:32							
13	22	79	12	2020/06/17 14:50:33							
14	33	58	86	2020/06/17 14:50:34							
15	43	58	6	2020/06/17 14:50:35							
16	53	48	2	2020/06/17 14:50:36							
17	63	38	67	2020/06/17 14:50:37							

✘Повідомлення про події, пов'язані з іншими сценаріями, можна переглянути в посібнику користувача 20.3.

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Керування користувачами

朱明杰

20.03.2020



Контур

- Огляд керування користувачами
- Інструменти керування користувачами
- Конфігурація зони безпеки
- Скрипт керування користувачами

призначення

У цьому розділі ви дізнаєтеся...

- ... Основні операції керування користувачами

- ... Керування користувачами, сценарії та функції зон безпеки

Контур

- Огляд керування користувачами
- Інструменти керування користувачами
- Конфігурація зони безпеки
- Скрипт керування користувачами

Огляд керування користувачами

➤ Визначення корости користувача в DIAView

- Управління користувачами стосується операцій і прав керування, які користувачі мають під час використання конфігураційного програмного забезпечення DIAView, включаючи «безпечну зону» та «користувача». За допомогою керування користувачами можна забезпечити безпечну та надійну роботу та стабільну роботу проекту, а користувачеві заборонено розробляти та запускати проект. Такі проблеми, як непрацездатність, несанкціонована робота, незаконна зміна налаштувань проекту тощо.

StartPage

Security

User Manage x

+ Add

▼ Insert

✖ Delete

🔑 Password Rule

	Name	Password	Security	Auto Log off Time (min)	User Type	Expire DateTime	User Level	Description
1	SystemAdmin	*****		0	System admin	Disabled	0	System admin
2	User	*****	Security	0	Admin	Disabled	0	
3	User1	*****	Security1	0	Admin	Disabled	0	
4	User2	*****	Security2	0	Operator	Expired Item	0	
5	User3	*****	Security3	0	Operator	Disabled ...	0	

Огляд керування користувачами

➤ Визначення зони безпеки в DIAView

- «Зона безпеки» — стратегія захисту безпеки роботи системи. У середовищі розробки конфігураційного програмного забезпечення DIAView, встановивши зону безпеки, права доступу до графічних об'єктів можна розділити та захистити. «Зона безпеки» зазвичай використовується в поєднанні з «користувачем».
- Користувач може мати одне або декілька прав на керування зоною безпеки, а також графічний об'єкт
підпорядковуються одній або декільком зонам безпеки; під час роботи системи користувачі можуть отримати доступ лише до графічних об'єктів у зоні безпеки, якою він володіє, щоб уникнути неправильної роботи користувача та неавторизованої роботи та забезпечити безпеку роботи системи та роботи.

Огляд керування користувачами

➤ Визначення користувача в DIAView

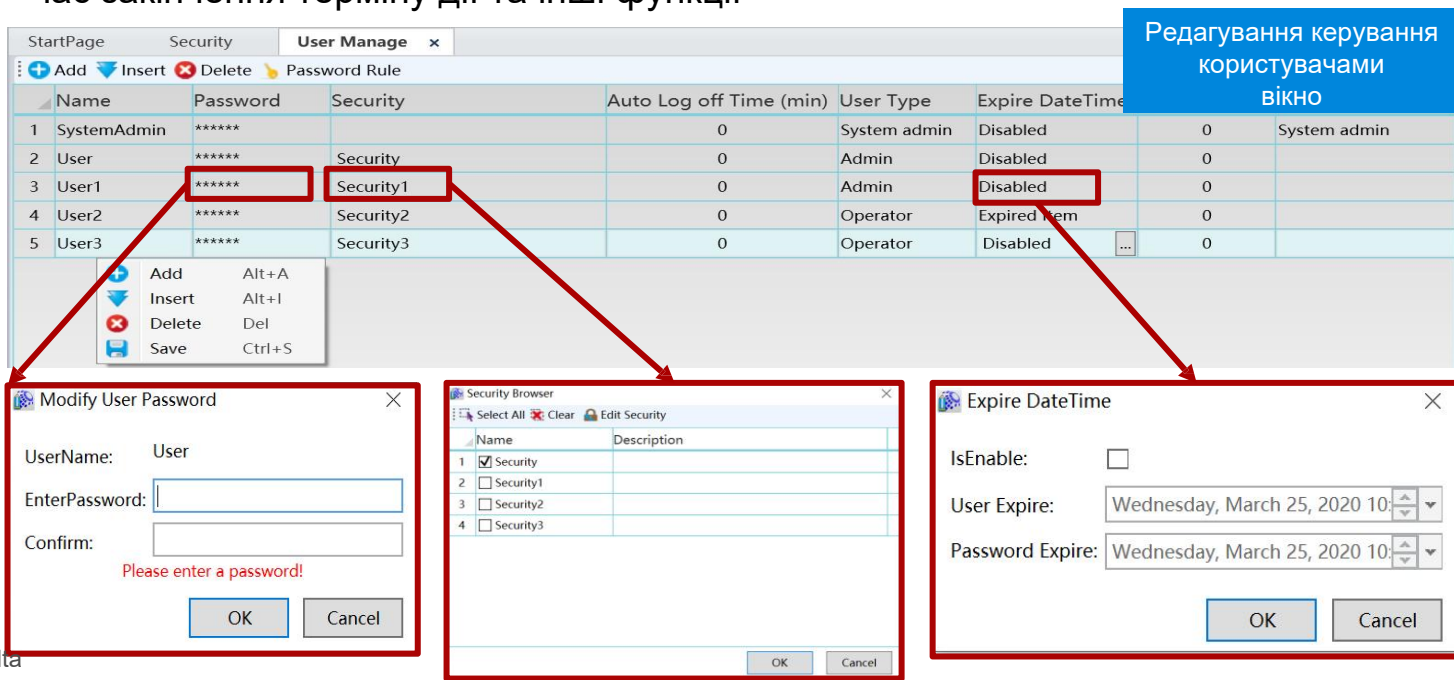
- «Користувачі» — оператори та менеджери з відповідними повноваженнями щодо розробки та експлуатації проекту. Користувачі в програмному забезпеченні DIAView поділяються на: «системного адміністратора», «оператора» та «адміністратора». «Системний адміністратор» є типовим у проекті. Має всі права на експлуатацію на найвищому рівні. У проекті є лише один системний адміністратор, і його неможливо видалити. «Оператор» — це загальний оператор проекту, який може налаштувати зону безпеки, щоб налаштувати свої повноваження на роботу; «Адміністратор» може керувати «оператором» на додаток до встановлених повноважень на роботу зони безпеки, а також може додавати, змінювати та видаляти інформацію про користувача «оператора».

- Огляд керування користувачами
- Засоби управління користувачами
- Конфігурація зони безпеки
- Скрипт керування користувачами

➤ Робота інструменту керування користувачами:

- Додавання, вставка, видалення та інші дії редагування користувачів і зон безпеки
- Реалізуйте правила пароля користувача, час автоматичного виходу з системи, час закінчення терміну дії та інші функції

Редагування керування користувачами ВІКНО



Name	Password	Security	Auto Log off Time (min)	User Type	Expire DateTime		
1 SystemAdmin	*****		0	System admin	Disabled	0	System admin
2 User	*****	Security	0	Admin	Disabled	0	
3 User1	*****	Security1	0	Admin	Disabled	0	
4 User2	*****	Security2	0	Operator	Expired Item	0	
5 User3	*****	Security3	0	Operator	Disabled	0	

Modify User Password

UserName: User

EnterPassword:

Confirm:

Please enter a password!

OK Cancel

Security Browser

Name	Description
☒ Security	
☐ Security1	
☐ Security2	
☐ Security3	

OK Cancel

Expire DateTime

IsEnable: ☐

User Expire: Wednesday, March 25, 2020 10:00

Password Expire: Wednesday, March 25, 2020 10:00

OK Cancel

Засоби управління користувачами

➤ Функціональне значення правил пароля:

Цикл зміни пароля днів	Розширення терміну дії пароля за замовчуванням у днях (Скидання пароля користувача: Час закінчення дії пароля = поточний системний час + дні циклу зміни пароля)
Кількість помилок пароля	Коли кількість помилок пароля під час входу досягне заданого значення, користувач буде автоматично заблоковано
Час блокування користувача (хвилина)	Тривалість часу, протягом якого користувач автоматично розблоковує пароль після того, як його було заблоковано певну кількість разів (Ця функція розблоковується після встановлення кількості помилок пароля)
Зберігайте історію паролів	Користувачеві не буде дозволено змінити пароль, якщо він збігається з попереднім паролем
Правила використання пароля користувача	Якщо надійність пароля ввімкнено, виберіть Вірно увімкнути перевірку надійності пароля
правила	Містить чотири правила, позначте одне або всі, потім символи пароля можна встановлювати лише відповідно до правил, якщо включена перевірка надійності

 Password Rule
 ✕

Change Password Cycle Days:

Number of Password Errors:

User Lock-in Time(Minute):

Keep Password History:

Passwords Remembered

Using User Password Rules:

False ▾

Rules:

☐ Must include numbers
☐ Must include special characters
☐ Must include upper and lower case letters
☐ Must be greater than or equal to 8 characters

OK

Cancel

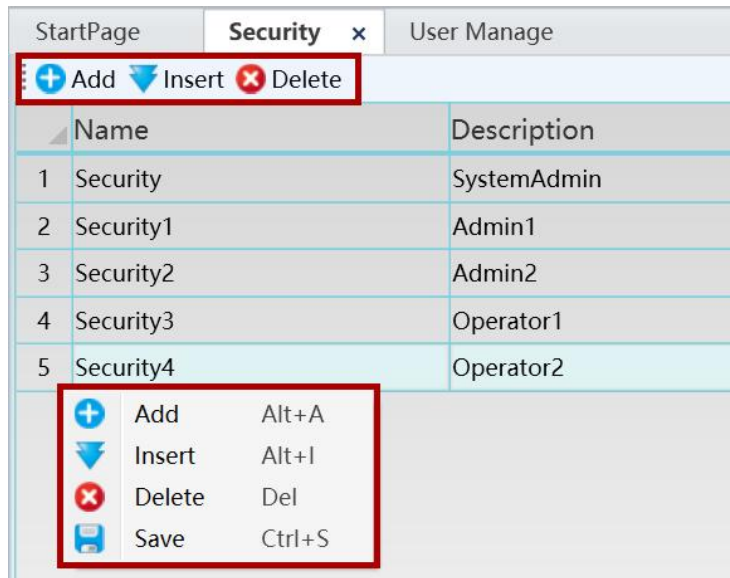
※Зверніться до розділу «20.3 Користування користувача» в посібнику користувача.

- Огляд керування користувачами
- Засоби управління користувачами
- **Конфігурація зони безпеки**
- Скрипт керування користувачами

Конфігурація зони безпеки

➤ Основна робота зони безпеки:

додати	Додайте зону безпеки, щойно додані змінні відображаються в списку області редагування зони безпеки
Вставка	Вставте зону безпеки, щойно вставлена зона безпеки відображається над вставленою зоною безпеки
Видалити	Видалити зону безпеки
зберегти	Зберегти конфігурацію зони безпеки



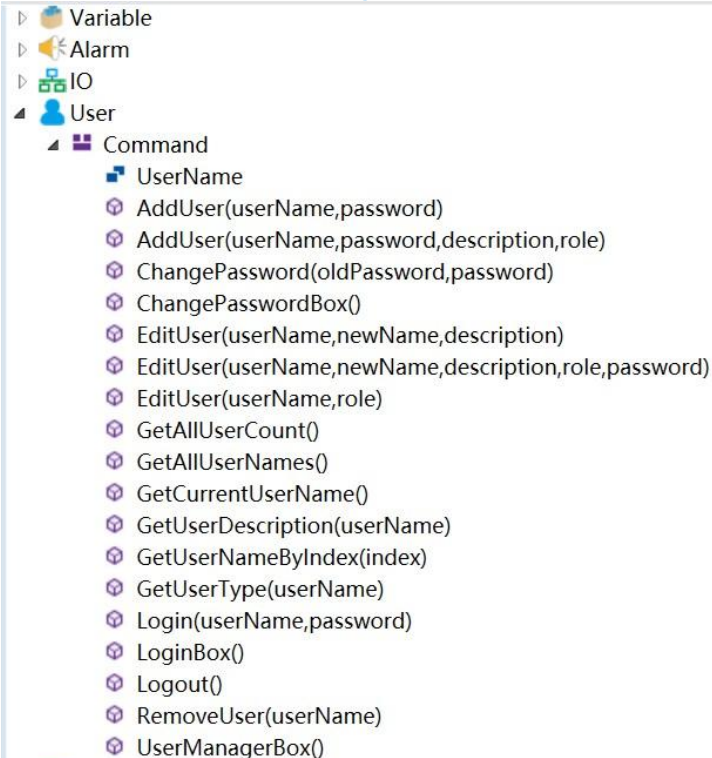
※Зверніться до розділу «12.2 Зона безпеки» в посібнику користувача.

- Огляд керування користувачами
- Засоби управління користувачами
- Конфігурація зони безпеки
- Скрипт керування користувачами

Скрипт керування користувачами

➤ Скрипт керування користувачами

- Коли користувачі налаштовують події віконних кнопок, екранні програми та сценарії програм користувача, програмне забезпечення надає прості та зручні у використанні атрибути сценаріїв і методи для виконання деяких спеціальних завдань і функцій.



※Див. розділ «20.3 Синтаксис і функції сценарію →
→ Користувачі та дозволи» в посібнику користувача.

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Запис про операцію

朱明杰

2020/04/10



Контур

- Огляд запису операцій
- Конфігурація маніпульованої змінної
- Конфігурація скриньки записів

призначення

У цьому розділі ви дізнаєтеся...

- ... Основні операції та керування керованими змінними

- ... Основні операції скриньки записів

Контур

- Огляд запису операцій
- Конфігурація операційної змінної
- Конфігурація скриньки записів

➤ Операційні змінні

- Операційна змінна — це змінна, яка зберігає та записує інформацію про технічні змінні, яку користувач змінює під час операції, пов'язану з подією та використовується для відстеження системної змінної за допомогою запису операції користувача.
- Інформація про збережений запис включає: **назву змінної, час зміни, значення до та після зміни змінної, інформацію про користувача** тощо.

StartPage

OperationVariableGroup0 x

Variable Dictionary

+

 Add

↓

 Insert

×

 Delete

↶

 Import

↷

 Export

	Name	Associated Variables	Description
1	OperationVariable	Var.Variable	
2	OperationVariable1	Var.Variable1	
3	OperationVariable2	Var.Variable2	

Variable Browser

+

 Add Group

✎

 Edit

Variable Dictionary

VariableGroup0

TypeFilter:

Analog

Digital

A String

Name	Type	Initial Val	Minimum	Maximum	De
Variable	Analog	0	0	10000	
Variable1	Analog	0	0	10000	
Variable2	Analog	0	0	10000	

➤ Скринька для записів

- Поле запису може записувати зміну інформації змінної операції, що зручно для користувача для фільтрації та перегляду інформації.

Operations and Ev	NearestOneHo						
Record Name	Variable Path	Record Type	Record Time	Record New Value	Record Old Value	Operator Name	Record :
Newuser		AddUser	5/14/2020 10:27:30 AM	Newuser		SystemAdmin	CNWJ6RI
Var.Variable	Var.Variable		5/14/2020 10:27:29 AM	321	123	SystemAdmin	CNWJ6RI
Var.Variable	Var.Variable		5/14/2020 10:27:28 AM	123	0	SystemAdmin	CNWJ6RI
		Login	5/14/2020 10:27:18 AM			SystemAdmin	CNWJ6RI
		StartProject	5/14/2020 10:27:04 AM				CNWJ6RI

Record Number: 5 New Record Position: Front No. 1 Page All 1 Page

- Огляд запису операцій
- Конфігурація операційної змінної
- Конфігурація скриньки записів

Конфігурація операційної змінної

➤ Основні операції та атрибути керованих змінних

додати	Додайте операційну змінну, щойно додана операційна змінна відображається в списку в області редагування операційної змінної
Вставка	Вставте керовану змінну, щойно вставлена керована змінна відображається над вставленою керованою змінною
Видалити	Видалити керовану змінну
Імпорт	Імпорт файлів конфігурації зовнішніх операційних змінних для підвищення ефективності конфігурації
Експорт	Інформацію про конфігурацію операційної змінної можна експортувати як файл Excel, який підтримує формати xls іxlsx
зберегти	Зберегти конфігурацію маніпульованої змінної

 Add  Insert  Delete  Import  Export			
	Name	Associated Variables	Description
1	OperationVariable	Var.Variable	
2	OperationVariable1	Var.Variable1	
3	OperationVariable2	Var.V	
4	OperationVariable3		
5	OperationVariable4		
6	OperationVariable5		
7	OperationVariable6		

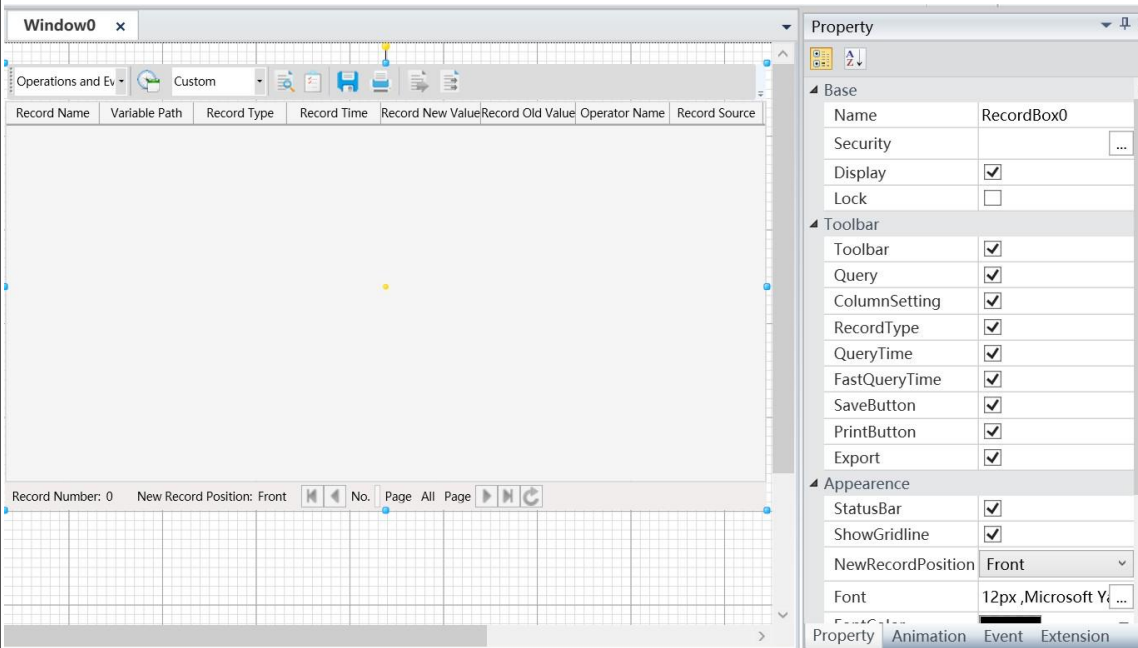
 Add Alt+A
 Insert Alt+I
 Delete Del
 Save Ctrl+S
 Import Ctrl+I
 Export Ctrl+E

※Для опису основних атрибутів керованих змінних дивіться розділ «Керовані змінні СН13» у посібнику користувача.

- Огляд запису операцій
- Конфігурація операційної змінної
- Конфігурація скриньки записів

Конфігурація скриньки записів

➤ Основні операції та атрибути ящика записів



- **Назва:** назва графіки в блокноті; назви графічних об'єктів в одному і тому ж блокноті не можуть повторюватися.
- **Безпека :** встановлює робочі повноваження для графіки; дозволяє графіці належати до однієї чи кількох зон безпеки, щоб виконувати керування правами користувачів.
- **Відображати :** відображати чи приховувати графіку під час виконання.
- **Блокувати:** чи блокувати графіку; як тільки його заблоковано, операції миші стануть недійсними, операції можна буде виконувати лише з іншими властивостями цієї графіки з кадру властивостей. Навколо заблокованої графіки з'явиться пунктирна лінія.

Для опису основних атрибутів кадру запису, будь ласка зверніться до посібника користувача «CH7.7.9 Record Frame» .

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Керування рецептами

朱明杰 杨阳

2020/04/24



Контур

- Огляд рецепта
- Конфігурація рецепта
- Використання формули
- Сценарій рецепта

призначення

Після цього розділу ви дізнаєтеся...

- ... Основні операції та керування рецептами

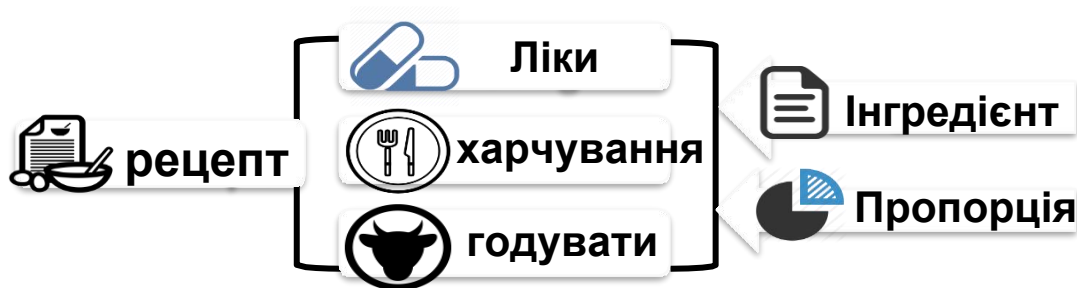
- ... Функція рецептів та їх сценаріїв

Контур

- Огляд рецепта
- Конфігурація рецепта
- Використання формули
- Сценарій рецепта

➤ Що таке рецепт?

- У промисловому виробництві рецепт — це припис, який визначає метод інгредієнтів продукту та пропорцію використаних матеріалів.
- У конфігураційному програмному забезпеченні DIAView є набір значень параметрів, що відповідають змінним у виробничому процесі. У конфігураційному програмному забезпеченні DIAView ви можете переглядати та підтримувати рецепти через **«Браузер рецептів»** управління, яке є простим і ефективним.



- Огляд рецепта
- Конфігурація рецепта
- Використання формули
- Сценарій рецепта

➤ Браузер рецептів

	RecipeItemName	Flour(g)	Suger(g)	Egg(g)	Water(g)	Vanilla(g)	Redbean(g)
1	VanillaCake	50	10	10	30	20	0
2	RedBeanCake	10	10	30	0	20	50

Recipe Name:Recipe0



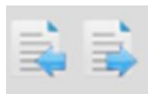
All Detail about the selected recipe



Change, switch columns and rows



Add, Edit, Delete recipes



Left: import recipe from xls
Right: export xls into recipe



Switch between recipes



Save the recipe browser to image



Save the recipe changes



Left: Write Recipe into Var.
Right: Write Var. into Recipe

➤ Конфігурація рецепта

Огляд рецепта

+ Add element ▾ Insert element ✖ Delete element Add item Insert item Delete item Import Export						
	Name	RecipeType	Associated variables	Description	VanillaCake	RedBeanCake
1	Flour	Analog	Var.Variable1	High quality flour	50	50
2	Suger	Analog	Var.Variable2	Crystal sugar	10	10

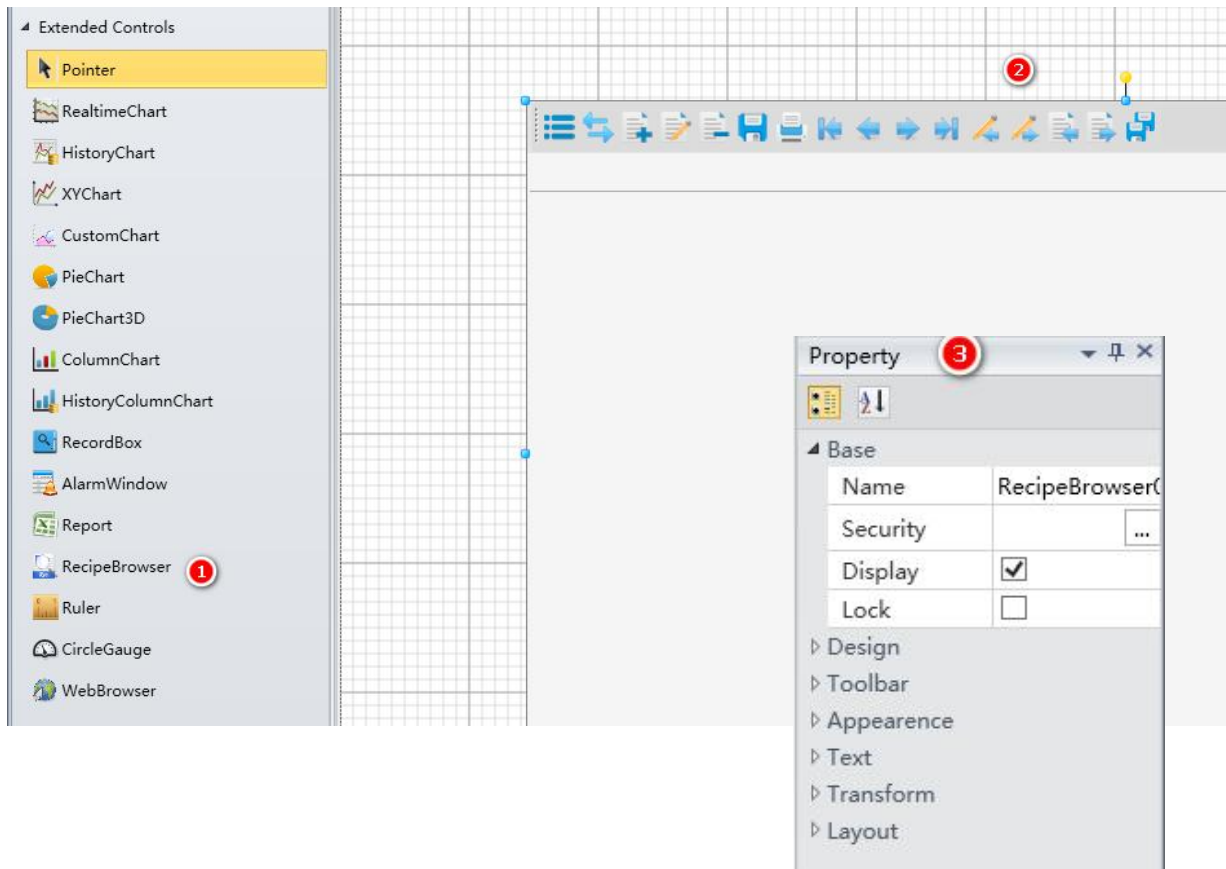
- **Додати елемент:** додайте інгредієнт рецепта, щойно доданий інгредієнт рецепта відображається у списку інгредієнтів в області редагування рецепту
- **Вставити елемент:** вставте інгредієнт рецепта, щойно вставлений інгредієнт рецепта відображається над вставленим інгредієнтом рецепту
- **Видалити елемент:** видалення інгредієнта рецепта
- **Додати елемент:** додайте елемент формули, щойно доданий елемент формули відображається у списку інгредієнтів в області редагування формули
- **Вставити елемент:** вставте елемент формули, щойно вставлений елемент формули відображається перед вставленим елементом формули
- **Видалити елемент:** видалити елемент рецепту
- **Імпорт:** імпорт зовнішніх файлів конфігурації рецептів для підвищення ефективності конфігурації рецептів
- **Експорт:** інформацію про конфігурацію рецепта можна експортувати як файл Excel, який підтримує формати xls іxlsx
- **Зберегти:** зберегти конфігурацію рецепта

- Огляд рецепта
- Конфігурація рецепта
- Використання формули
- Сценарій рецепта

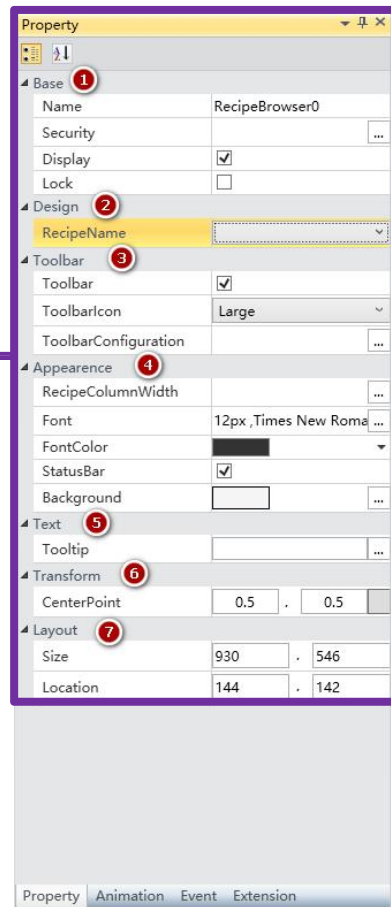
Створіть браузер рецептів

➤ Як створити браузер рецептів?

- ① Натисніть на рецепт у панель інструментів
- ② Перетягніть у потрібне положення на полотні
- ③ Налаштувати властивості рецепта



- ① **База:**
Основні властивості, такі ж, як і в інших графічних елементів керування
- ② **Дизайн:**
Створіть і встановіть рецепт, прив'язаний до браузера ре
- ③ **Панель інструментів:** панель інструментів визначає, чи відображати панель інструментів, конфігурацію панелі інструментів і розмір панелі інструментів
- ④ **Зовнішній вигляд:**
Зовнішній вигляд може спроектувати фон кольору шрифту формули, тощо
- ⑤ **Текст:**
Текстова інформація про рецепт
- ⑥ **Трансформувати:**
Містить центральну точку елемента керування рецептом
- ⑦ **Макет:**
Розмір і положення формули на полотні можна регулювати числовими значеннями



The screenshot shows a 'Property' window for an object named 'RecipeBrowser0'. The window is divided into several sections, each with a red circle and a number corresponding to the list items:

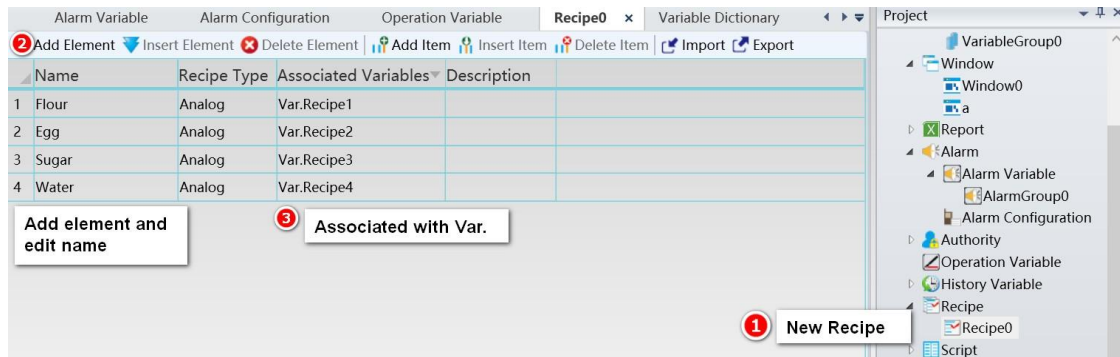
- 1 Base:** Contains properties like Name (RecipeBrowser0), Security, Display (checked), and Lock (unchecked).
- 2 Design:** Contains the RecipeName property, which is highlighted with a yellow background.
- 3 Toolbar:** Contains properties like Toolbar (checked), ToolbarIcon (Large), and ToolbarConfiguration.
- 4 Appearance:** Contains properties like RecipeColumnWidth, Font (12px, Times New Roma), FontColor (black), StatusBar (checked), and Background.
- 5 Text:** Contains the Tooltip property.
- 6 Transform:** Contains the CenterPoint property, set to 0.5, 0.5.
- 7 Layout:** Contains the Size property (930, 546) and the Location property (144, 142).

At the bottom of the window, there are tabs for 'Property', 'Animation', 'Event', and 'Extension'.

Приклад налаштування рецепта

➤ Приклад: як створити рецепт?

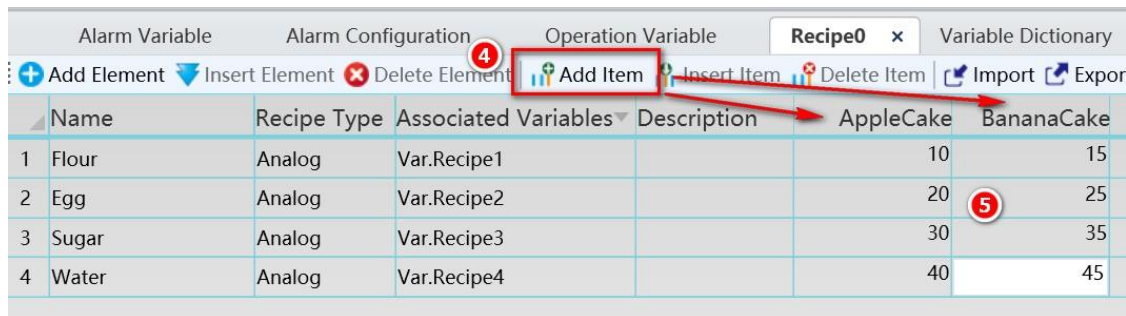
- Крок 1: Створіть **Recipe0** і додати елемент (елемент означає різний тип вмісту в одному рецепті)



The screenshot shows the 'Recipe0' configuration window. It has tabs for 'Alarm Variable', 'Alarm Configuration', 'Operation Variable', and 'Variable Dictionary'. The 'Operation Variable' tab is active, displaying a table with 4 ingredients and their associated variables. A 'New Recipe' button is visible in the bottom right corner.

Name	Recipe Type	Associated Variables	Description
1 Flour	Analog	Var.Recipe1	
2 Egg	Analog	Var.Recipe2	
3 Sugar	Analog	Var.Recipe3	
4 Water	Analog	Var.Recipe4	

- Крок 2: Додати товар означає додати інший рецепт, наприклад, тут ми додали яблучний і банановий пиріг , два види рецептів.

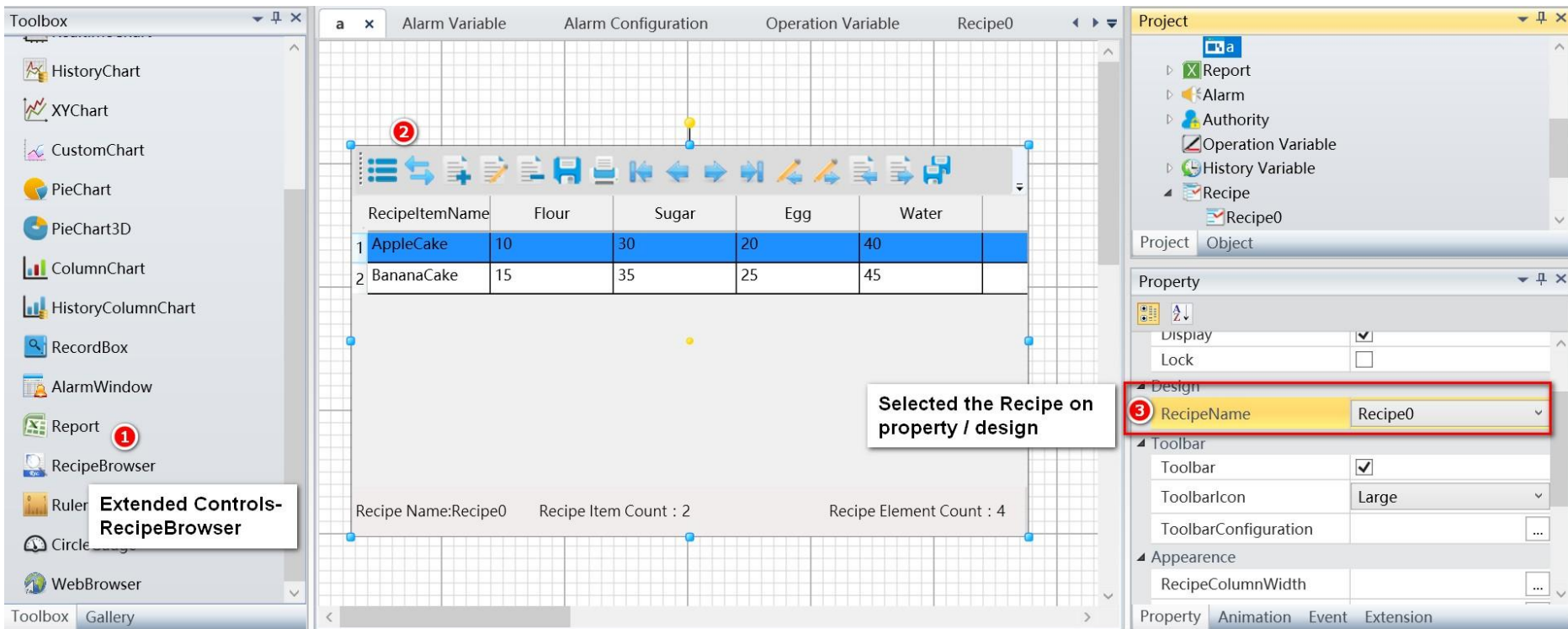


The screenshot shows the 'Recipe0' configuration window after adding new recipes. The 'Add Item' button is highlighted with a red box and a red arrow pointing to the 'AppleCake' and 'BananaCake' columns. The table now includes these two new recipes.

Name	Recipe Type	Associated Variables	Description	AppleCake	BananaCake
1 Flour	Analog	Var.Recipe1		10	15
2 Egg	Analog	Var.Recipe2		20	25
3 Sugar	Analog	Var.Recipe3		30	35
4 Water	Analog	Var.Recipe4		40	45

Приклад налаштування рецепта

- Останній крок: створіть завершений рецепт і прив'язку браузера рецептов.



Toolbox

- HistoryChart
- XYChart
- CustomChart
- PieChart
- PieChart3D
- ColumnChart
- HistoryColumnChart
- RecordBox
- AlarmWindow
- Report
- RecipeBrowser
- Ruler
- Circle
- WebBrowser

Alarm Variable Alarm Configuration Operation Variable Recipe0

RecipeItemName	Flour	Sugar	Egg	Water
1 AppleCake	10	30	20	40
2 BananaCake	15	35	25	45

Selected the Recipe on property / design

Recipe Name:Recipe0 Recipe Item Count : 2 Recipe Element Count : 4

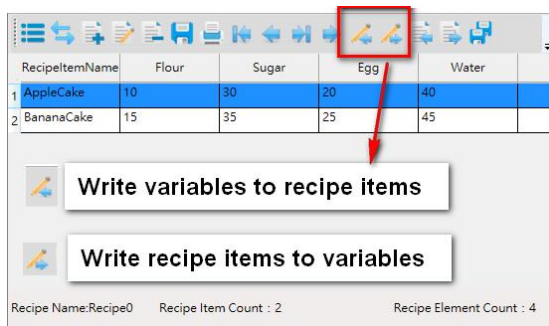
Property

- Display ☒
- Lock ☐
- Design
 - RecipeName Recipe0
- Toolbar
 - Toolbar ☒
 - ToolbarIcon Large
 - ToolbarConfiguration ...
- Appearance
 - RecipeColumnWidth ...

Property Animation Event Extension

Extended Controls-RecipeBrowser

Переглядач рецептів під час виконання

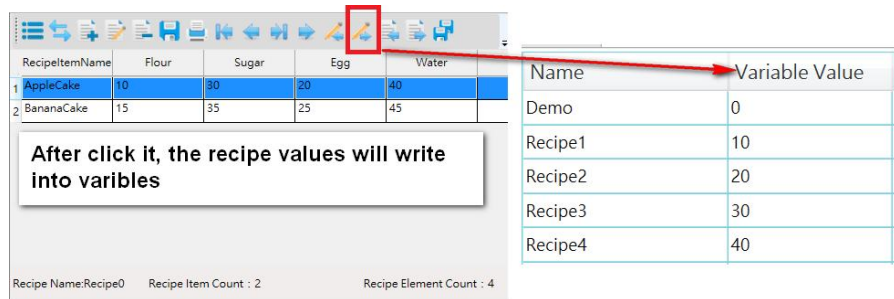


RecipeItemName	Flour	Sugar	Egg	Water
1 AppleCake	10	30	20	40
2 BananaCake	15	35	25	45

Write variables to recipe items

Write recipe items to variables

Recipe Name: Recipe0 Recipe Item Count : 2 Recipe Element Count : 4



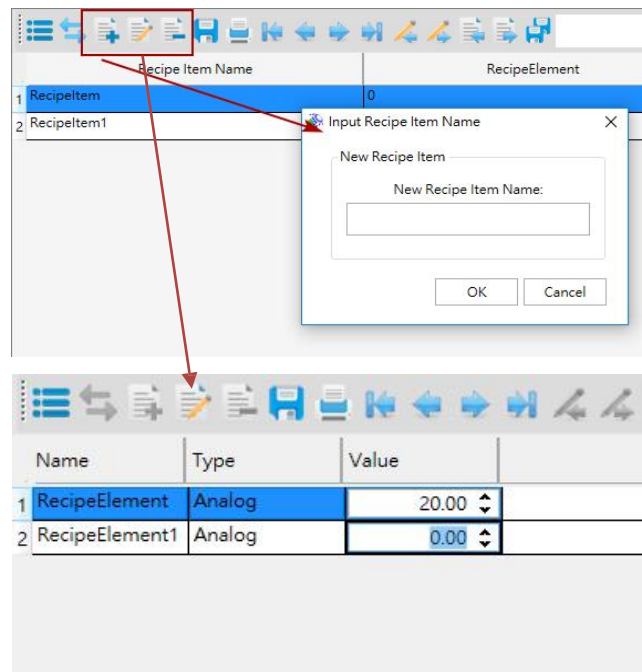
RecipeItemName	Flour	Sugar	Egg	Water
1 AppleCake	10	30	20	40
2 BananaCake	15	35	25	45

After click it, the recipe values will write into variables

Name	Variable Value
Demo	0
Recipe1	10
Recipe2	20
Recipe3	30
Recipe4	40

Recipe Name: Recipe0 Recipe Item Count : 2 Recipe Element Count : 4

Запишіть змінні до елементів рецепта



RecipeItemName	RecipeElement
1 RecipeItem	0
2 RecipeItem1	

Input Recipe Item Name

New Recipe Item

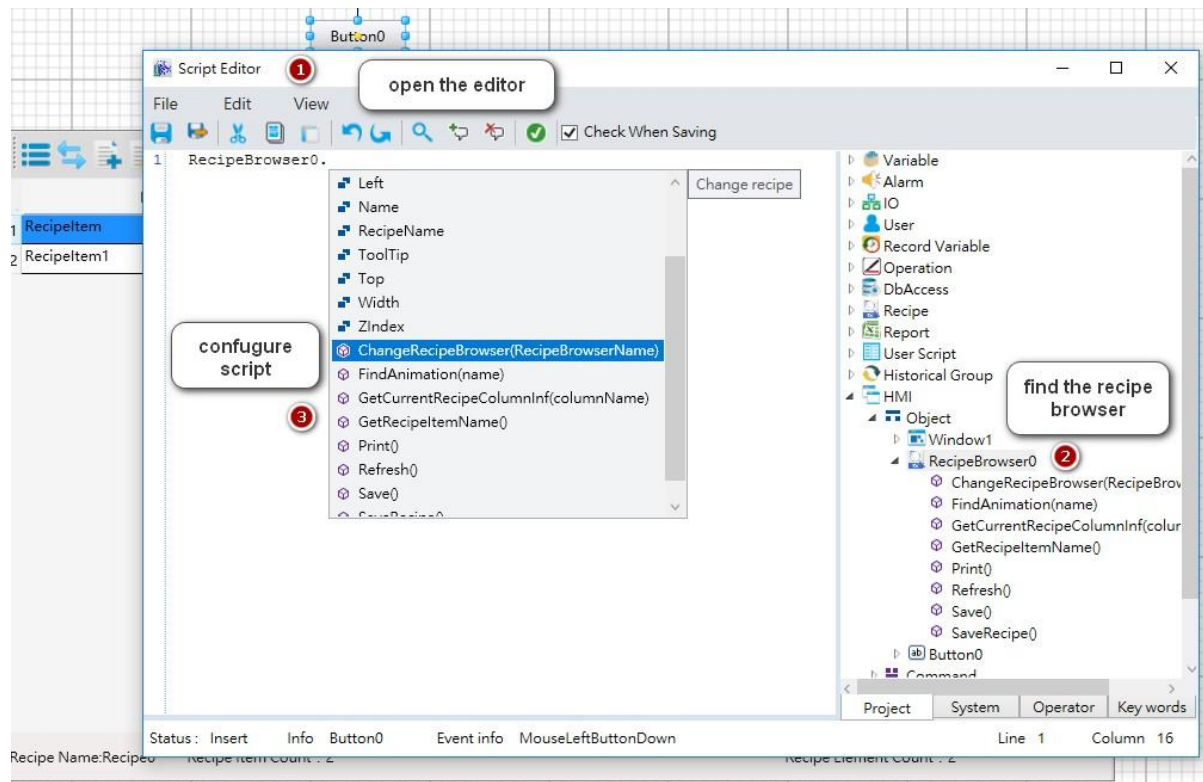
New Recipe Item Name:

OK Cancel

Name	Type	Value
1 RecipeElement	Analog	20.00
2 RecipeElement1	Analog	0.00

Recipe Name: Recipe0 Recipe Item Count : 2 Recipe Element Count : 4

Додайте, видаліть або змініть елементи рецепта і елемент рецепту.



※ Сценарій керування браузером рецептів може посилатися на посібник користувача Ch20.3

- Огляд рецепта
- Конфігурація рецепта
- Використання формули
- Сценарій рецепта

➤ Сценарій рецепта

- Коли користувачі налаштовують події віконних кнопок, екранні програми та сценарії програм користувача, програмне забезпечення надає прості та зручні у використанні властивості сценаріїв і методи для виконання деяких спеціальних завдань і функцій.



※ Щоб отримати основні інструкції щодо роботи зі сценарієм рецепту, зверніться до посібника користувача «20.3 Синтаксис і функції сценарію → Модель програмування → Рецепт».

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Основи програмування

朱明杰

2020/05/15



Контур

- Огляд редактора сценаріїв
- Базове використання синтаксису

призначення

Після цього розділу ви дізнаєтеся...

... Використання базового синтаксису VBScript

Контур

- Огляд редактора сценаріїв
- Базове використання синтаксису

Огляд редактора сценаріїв

- Конфігураційне програмне забезпечення DIAView є комплексним і потужним, але функції, необхідні клієнтам, відрізняються. Деякі функції налаштовуються відповідно до потреб клієнта. Конфігураційне програмне забезпечення DIAView може писати відповідні програми через редактор сценаріїв для виконання деяких спеціальних завдань і функцій.
- Конфігурація подій і програма користувача в програмному забезпеченні конфігурації DIAView повинні використовувати редактор сценаріїв для написання сценаріїв. Конфігураційне програмне забезпечення DIAView використовує мову VB Script. Користувачі можуть писати програми керування логікою відповідно до специфікацій граматики мови VB Script, щоб виконувати спеціальні функції та покращувати зручність використання системи.

- Огляд редактора сценаріїв
- Базове використання синтаксису

➤ Використання умовних операторів

- Варіант оператора **If...Then...Else** дозволяє вибрати з кількох умов, а саме додавання пропозиції **Else If** для розширення функціональності оператора **If...Then...Else** ,
- Дозволяє контролювати потік програми на основі багатьох можливих. наприклад:

Тьяне значення

Якщо значення = 0, то

значення MsgBox

Значення Elself = 1 Тоді

значення MsgBox

Значення Elself = 2, тоді

значення Msgbox

Інакше

Msgbox "Значення поза діапазоном!"

Кінець Якщо

➤ Використання операторів циклу

- Оператор **For...Next** використовується для виконання блоку операторів задану кількість разів. Змінна лічильника використовується в циклі, і значення змінної збільшується або зменшується з кожним циклом.
- Наприклад: у наступному прикладі процедура **MyProc** повторюється 50 разів. Оператор For визначає змінну лічильника x та її початкове та кінцеве значення. Оператор **Next** збільшує змінну лічильника на 1.

```
Sub DoMyProc50Times()
```

```
    Дим x
```

```
    Для x = 1 до 50
```

```
        MyProc
```

```
    Далі
```

```
Кінець Під
```

Базове використання синтаксису

➤ Використання процесу Function

- **Функціональні** процедури — це набір операторів VBScript, що містяться між операторами Function і End Function. Функціональні процедури подібні до підпроцедур, але функціональні процедури можуть повертати значення. Функціональні процедури можуть використовувати параметри (константи, змінні або вирази, передані процедурою виклику).
- Якщо процедура Function не має параметрів, оператор Function має містити порожні дужки (). Процедура Function повертає значення за назвою функції. Це значення присвоюється імені функції в операторі процедури. Тип даних значення, що повертається функцією, завжди **Variant**.

➤ Використання процесу Fuction

- У наступному прикладі:

Функція Цельсія перетворює градуси Фаренгейта на **Цельсій** . Підпроцедура **ConvertTemp** Коли викликається ця функція, змінна, що містить значення параметра, передається функції. Результат перетворення повертається до процесу виклику та відображається у вікні повідомлення.

```
Sub ConvertTemp()  
    temp = InputBox("Будь ласка, введіть температуру у Фаренгейтах.", 1)  
    MsgBox "Температура: "& Цельсій(temp) & "Цельсій."  
Кінець Під  
Функція Цельсія(fГрадусів) Цельсія =  
    (fГрадусів-32) * 5/9  
Кінцева функція
```

➤ Використання підпроцесу

- **Підпроцедури** — це набір операторів VBScript, що містяться між операторами **Sub i End Sub** , які виконують операції, але не повертають значення. Підпроцедури можуть використовувати параметри (константи, змінні або вирази, передані процедурою виклику). Якщо процедура Sub не має параметрів, оператор Sub повинен містити порожні дужки ().
- Наступна підпроцедура використовує дві властиві (або вбудовані) функції VBScript, **MsgBox** і **InputBox** , щоб запропонувати користувачеві ввести інформацію. Потім відобразить результати розрахунків на основі цієї інформації. Розрахунок виконується за допомогою процедури Function, створеної за допомогою VBScript.

```
Sub ConvertTemp()  
    temp = InputBox("Будь ласка, введіть температуру у Фаренгейтах.", 1)  
    MsgBox "Температура: "& Цельсій(temp) & "Цельсій."  
Кінець Під
```

※Зверніться до розділу "20.3 Синтаксис і функції сценарію" в користувачеві у посібнику для детальної інформації про використання базового синтаксису VBScript.

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Скрипт користувача

Венді
2020/03/23



Контур

- Огляд сценарію користувача
- Сценарій умов
- Сценарій часу
- Глобальна функція

призначення

У цьому розділі ви дізнаєтеся....

- ... Конфігурація та використання умовних програм

- ... Конфігурація та використання програми часу

- ... Конфігурація, керування та виклик глобальних функцій

Контур

- Огляд сценарію користувача
- Сценарій умов
- Сценарій часу
- Глобальна функція

➤ Що таке сценарій користувача?

- Сценарій користувача — це сценарій зі спеціальними функціями, написаний користувачем, який може вдосконалювати та розширювати функції конфігураційного програмного забезпечення DIAView для досягнення складного логічного керування, тим самим покращуючи та оптимізуючи робочий процес системи та ефективність виконання.
- Список підтримки сценаріїв користувача:

① Сценарій умов

② Сценарій часу

③ Спеціальна глобальна функція

④ Шифрування скрипта

StartPage Condition Script x			
+ Add Insert Delete			
Name	IsEnable	Expression	Trigger Mode
1 ConditionScript	✓	Var.Variable	Value change
2 ConditionScript1	✓	Var.Variable2	While true
3 ConditionScript2	✓	Var.Variable3	While false
4 ConditionScript3	✓	Var.Variable4	On true
5 ConditionScript4	✓	Var.Variable5	On false

StartPage GlobalFunctionModule x		
+ Add Insert Delete		
Name	Method Type	Script Content
1 Method	Function	Function Method()
2 Method1	Sub	Sub Method1()

StartPage Time Script x					
+ Add Insert Delete					
Name	IsEnable	Trigger Mode	Configuration	Script Content	Description
1 TimeScript	✓	Start	N/A	Sub TimeScript_Tick()	
2 TimeScript1	✓	Stop	N/A	Sub TimeScript1_Tick()	
3 TimeScript2	✓	One time	3/31/2020 12:00:00 AM	Sub TimeScript2_Tick()	
4 TimeScript3	✓	Continue	12:00:00 AM	Sub TimeScript3_Tick()	
5 TimeScript4	✓	Weekly	12:00:00 AM	Sub TimeScript4_Tick()	
6 TimeScript5	✓	Monthly	12:00:00 AM	Sub TimeScript5_Tick()	

➤ Що таке сценарій користувача?

- **Сценарій умови:** підтримка функції запуску сценарію виразу умови
- **Сценарій часу:** підтримує функцію сценарію запуску часу, яка зазвичай використовується в режимі планування часу
- Сценарій умови та сценарій часу виконуються як дві групи за замовчуванням паралельно окремі потоки і не впливатимуть один на одного.
- Сценарії умов, сценарії часу та сценарії екрана відокремлені від основного потоку користувальницький інтерфейс і не призведе до зупинок інтерфейсу користувача
- **Глобальна функція:** запакуйте повторюваний код сценарію у функцію для інших функцій, щоб уникнути надто розпорошеного коду, зменшивши витрати на обслуговування та труднощі з налагодженням

- Огляд сценарію користувача
- Сценарій умов
- Сценарій часу
- Глобальна функція

➤ Сценарій умов

- Програми умов — це спеціальні програми, які виконуються системою, коли задані умови виконуються
- познайомилися в проекті.

Режим запуску	
Зміна значення	Запустити виконання сценарію, коли змінюється значення виразу
Поки правда	Ініціювати виконання сценарію, коли вираз значення змінюється на true
Поки фальшиві	Запустити виконання сценарію, коли значення виразу змінюється на false
По правді	Виконання сценарію запускається, коли значення виразу залишається істинним
На помилкові	Запустити виконання сценарію, коли значення виразу залишається false

Condition Script						
Add Insert Delete						
	Name	IsEnable	Expression	Trigger Mode	Time Interval (ms)	Script Content
1	ConditionScript	✓	Var.Variable	Value change	N/A	Sub ConditionScript
2	ConditionScript1	✓	Var.Variable2	While true	N/A	Sub ConditionScript
3	ConditionScript2	✓	Var.Variable3	While false	N/A	Sub ConditionScript
4	ConditionScript3	✓	Var.Variable4	On true	1000	Sub ConditionScript
5	ConditionScript4	✓	Var.Variable5	On false	1000	Sub ConditionScript

✗Див. розділ «16.2 Сценарій умов» у посібнику користувача.

- Огляд сценарію користувача
- Сценарій умов
- Сценарій часу
- Глобальна функція

➤ Сценарій часу

- Програма часу – це визначена користувачем програма, яка виконується системою під час запуску проекту, під час роботи або при зупинці.

Режим запуску	
старт	Запустити виконання сценарію під час запуску
СТІЙ	Запустити виконання сценарію після зупинки
Одного разу	Ініціювати виконання сценарної програми у фіксований час
Продовжити	Безперервно виконувати програму сценарію відповідно до встановленого інтервалу часу
Щотижня	Виконуйте програму сценарію безперервно щотижня відповідно до встановленого інтервалу часу
Щомісяця	Виконуйте програму сценарію безперервно щомісяця відповідно до встановленого часу інтервал

Time Script x					
Add Insert Delete					
Name	IsEnable	Trigger Mode	Configuration	Script Content	
1 TimeScript	✓	Start	N/A	Sub TimeScript_Tick()	
2 TimeScript1	✓	Stop	N/A	Sub TimeScript1_Tick()	
3 TimeScript2	✓	One time	3/31/2020 12:00:00 AM	Sub TimeScript2_Tick()	
4 TimeScript3	✓	Continue	12:00:00 AM	Sub TimeScript3_Tick()	
5 TimeScript4	✓	Weekly	12:00:00 AM	Sub TimeScript4_Tick()	
6 TimeScript5	✓	Monthly	12:00:00 AM	Sub TimeScript5_Tick()	

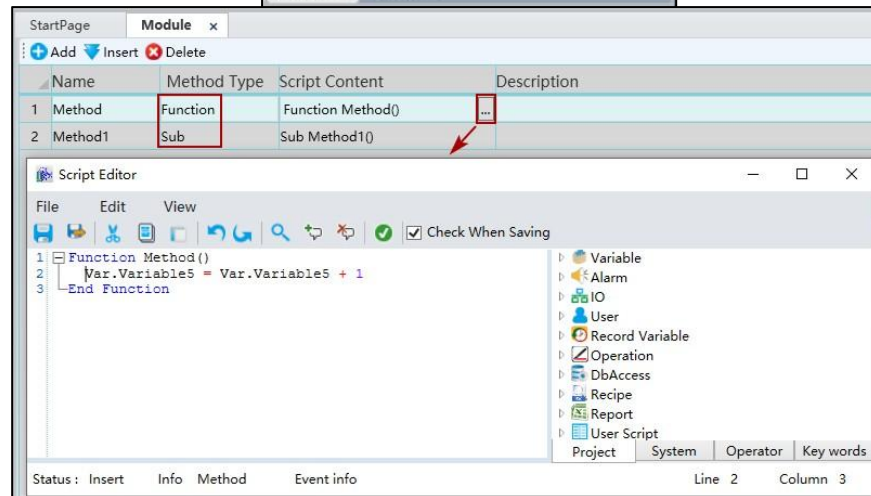
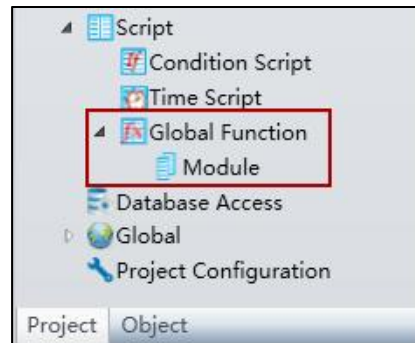
※Зверніться до розділу «Сценарій часу 16.3» у посібнику користувача.

Контур

- Огляд сценарію користувача
- Сценарій умов
- Сценарій часу
- Глобальна функція

➤ Глобальний огляд функцій:

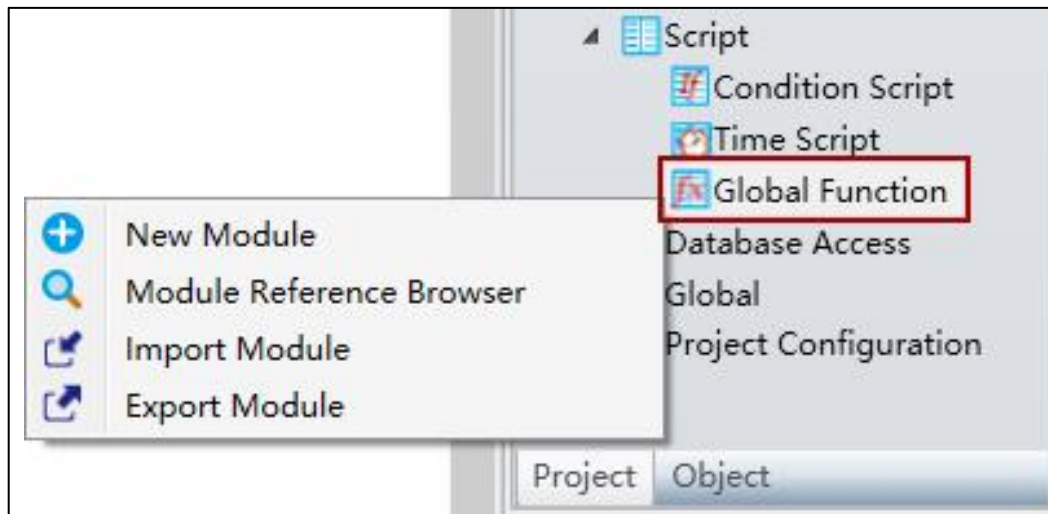
- Основна мета глобальної функції полягає в тому, щоб інкапсулювати повторюваний код сценарію у функцію для інших функцій, щоб уникнути надто розпорошеного коду, зменшивши витрати на обслуговування та труднощі з налагодженням.
- Глобальні функції складаються з модулів.
- Модулі — це набір функцій. Функції між різними модулями можуть викликати одна одну.
 - **Модуль:** модуль складається з низки функцій;
 - **Функція:** функція має повернення значення
 - **Sub:** функція без значення, що повертається



Глобальна функція

➤ Глобальне управління функціями

- **Новий модуль:** додайте глобальну функцію
модуль
- **Браузер довідкових модулів:**
Глобальна функція перегляду довідників
- **Модуль імпорту:** глобальний імпорт
функціональний модуль
- **Модуль експорту:** експорт
глобального функціонального
модуля

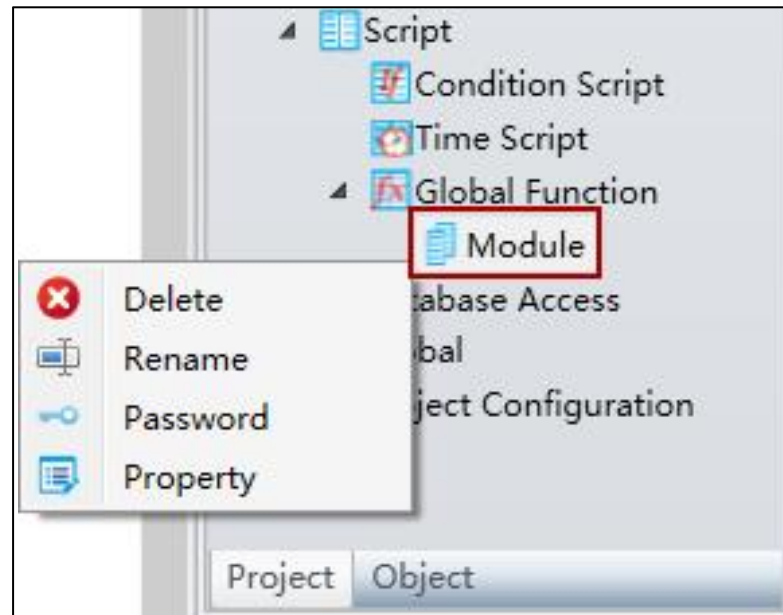


※Див. розділ «16.4 Глобальна функція» в посібнику користувача.

Глобальна функція

➤ Управління модулями

- **Видалити:** видалити глобальний функціональний модуль
- **Перейменувати:** перейменувати модуль
- **Пароль:** зашифрувати модуль
- **Властивість:** змінити властивості модуля



※Див. розділ «16.4 Глобальна функція» в посібнику користувача.

➤ Гнучкий виклик глобальних функцій

- Користувачі налаштовують умови запуску, писати програми для виклику глобальних функцій і реалізувати нові функції за допомогою логічного керування;
- Підтримка умовних програм і часу програми для виклику глобальних функцій

Умовні програми викликають глобальні функції

Name	IsEnable	Expression	Trigger Mode	Time Interval (ms)	Script Content
1 ConditionScript	✓	Var.Var.a5.2	Value change	N/A	Sub ConditionScript_DataChangedOrTimeout()
2 ConditionScript1	✓	Var.Variable2	While true	N/A	Sub ConditionScript1_DataChangedOrTimeout()
3 ConditionScript2	✓	Var.Variable3	While false	N/A	Sub ConditionScript2_DataChangedOrTimeout()
4 ConditionScript3	✓	Var.Variable4	On true	1000	Sub ConditionScript3_DataChangedOrTimeout()
5 ConditionScript4	✓	Var.Variable5	On false	1000	Sub ConditionScript4_DataChangedOrTimeout()

Script Editor

```

1 Sub ConditionScript_DataChangedOrTimeout()
2   Var.VariableGroup0.Variable1 = Calculator.Gs_Sum(Var.VariableGroup0.Variable1, 2)
3 End Sub
  
```

Variable

- Alarm
- IO
- User
- Record Variable
- Operation
- DbAccess
- Recipe
- Report
- User Script
- Historical Group
- Global Function

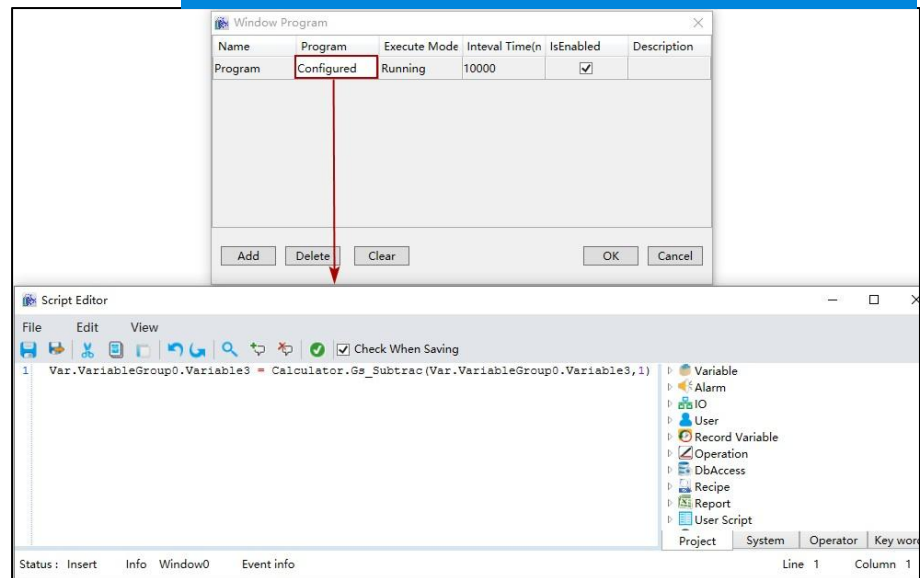
Status: Insert Info ConditionScript Event info DataChangedOrTimeout Line 2 Column 2

Глобальна функція

➤ Гнучкий виклик глобальних функцій

- Підтримка програми Windows для виклику глобальної функції
- Підтримка подій керування вікном для виклику глобальні функції

Програма Windows викликає глобальні функції



Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Віконна програма та програма заходів

рубін

2020/05/01



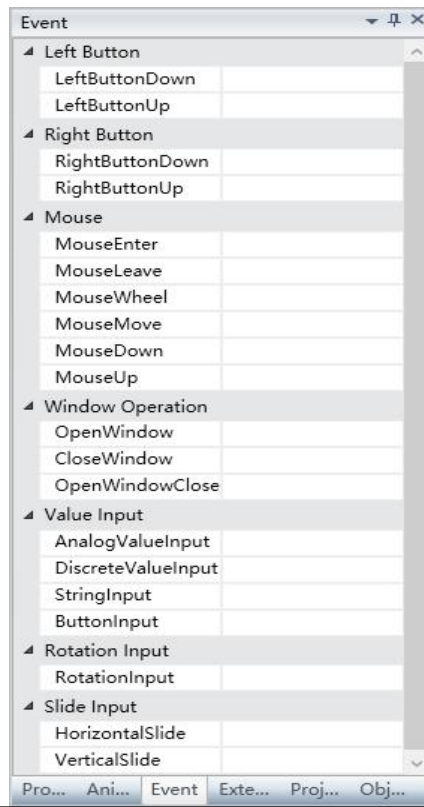
- Поняття події
- Подія лівої кнопки
- Подія правої кнопки
- Подія миші
- Подія роботи вікна
- Подія введення значення
- Подія ковзного введення
- Подія введення повороту
- Подія віконної програми
- Контрольна подія
- Клавіатура

У цьому розділі ви дізнаєтеся....

- ... докладніше про подію
- ... детальніше про десять типів подій DIAView

- Поняття події
- Подія лівої кнопки
- Подія правої кнопки
- Подія миші
- Подія роботи вікна
- Подія введення значення
- Подія ковзного введення
- Подія введення повороту
- Подія віконної програми
- Контрольна подія
- Клавіатура

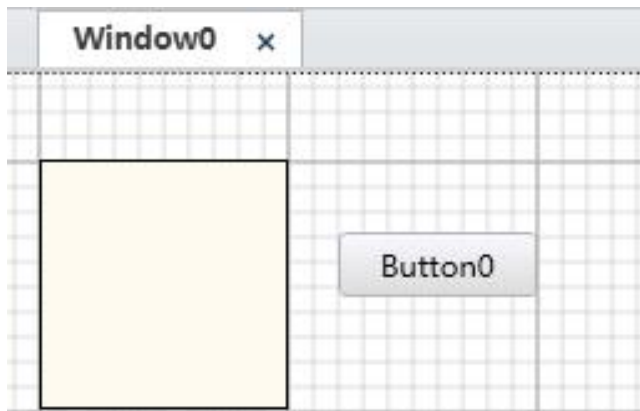
Події — це операції, які можуть розпізнаватися та реагувати на графічні об'єкти, і поділяються на системні події та події користувача. Події в DIAView, як правило, є подіями користувача, тобто діями користувача над різними графічними об'єктами у вікні, а потім керують графічними елементами керування для виконання певної функції.



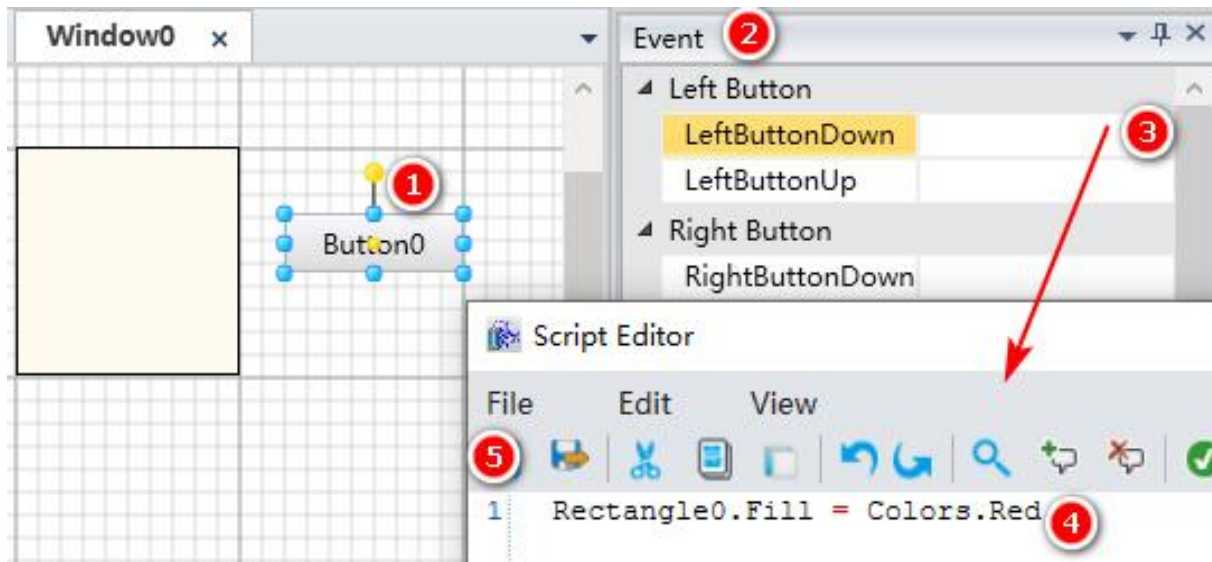
- Поняття події
- Подія лівої кнопки
- Подія правої кнопки
- Подія миші
- Подія роботи вікна
- Подія введення значення
- Подія ковзного введення
- Подія введення повороту
- Подія віконної програми
- Контрольна подія
- Клавіатура

➤ події натискання лівої кнопки :

Створіть червоний прямокутник за допомогою події натискання лівої кнопки (1) Створіть Rectangle0 і Button0 у Window0

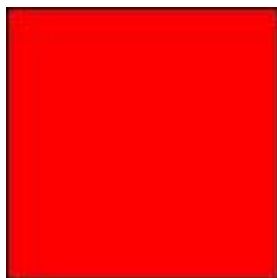


(2) Налаштуйте подію натискання лівої кнопки Button0



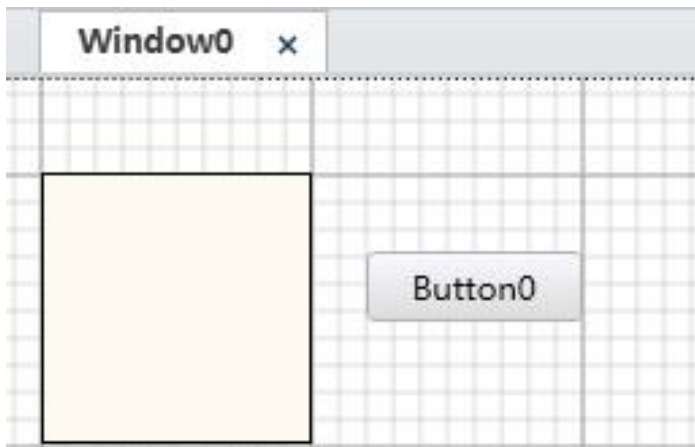
※Див. розділ «10.2 Подія лівої кнопки» в посібнику користувача.

(3) Запустіть Window0, клацніть лівою кнопкою миші на “ Button0 ”, потім Rectangle0 стане червоним

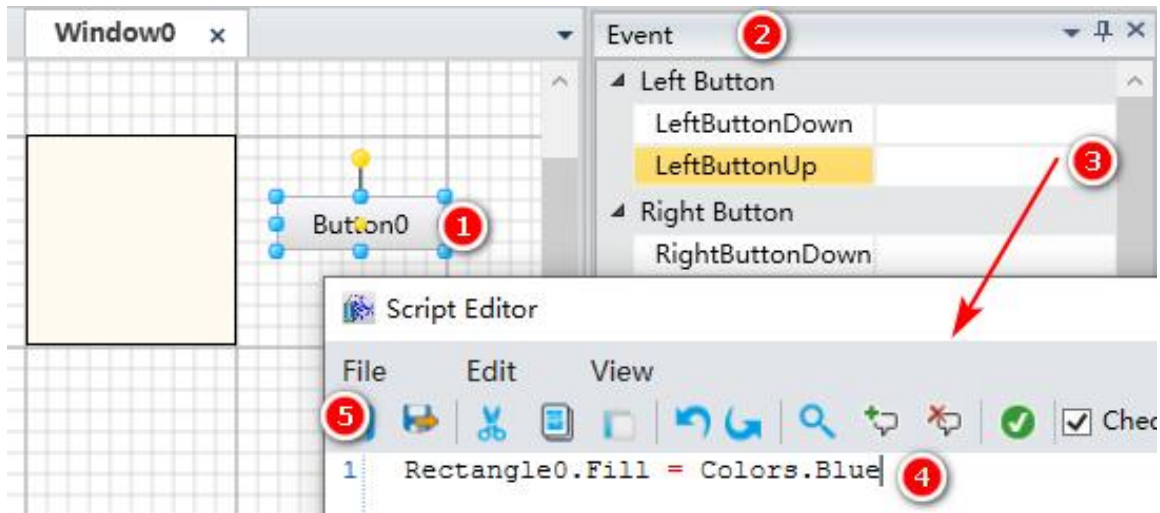


➤ події вгору лівої кнопки :

Створіть синій прямокутник із подією вгору лівої кнопки : (1) Створіть Rectangle0 і Button0 у Window0

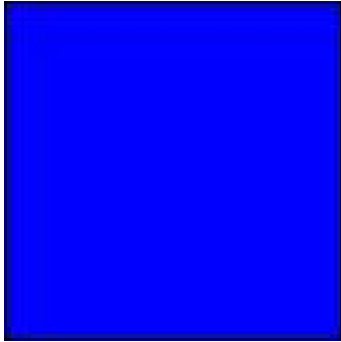


(2) Налаштуйте подію підняття лівої кнопки Button0



※Див. розділ «10.2 Подія лівої кнопки» в посібнику користувача.

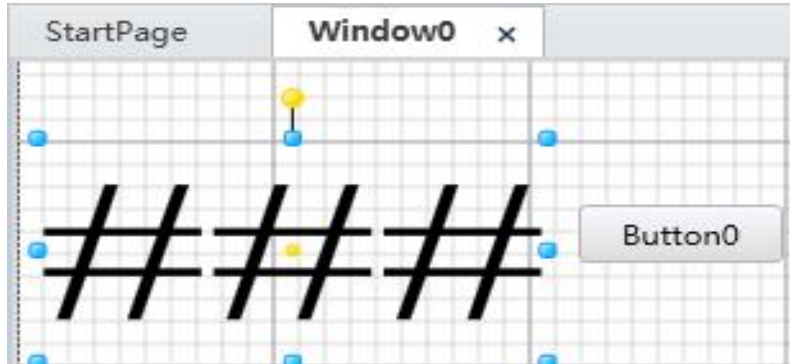
(3) Запустіть Window0, клацніть лівою кнопкою миші на “ Button0 ”, коли миша натиснута, колір Rectangle0 не змінюється, потім відпустіть мишу, колір Rectangle0 стає синім



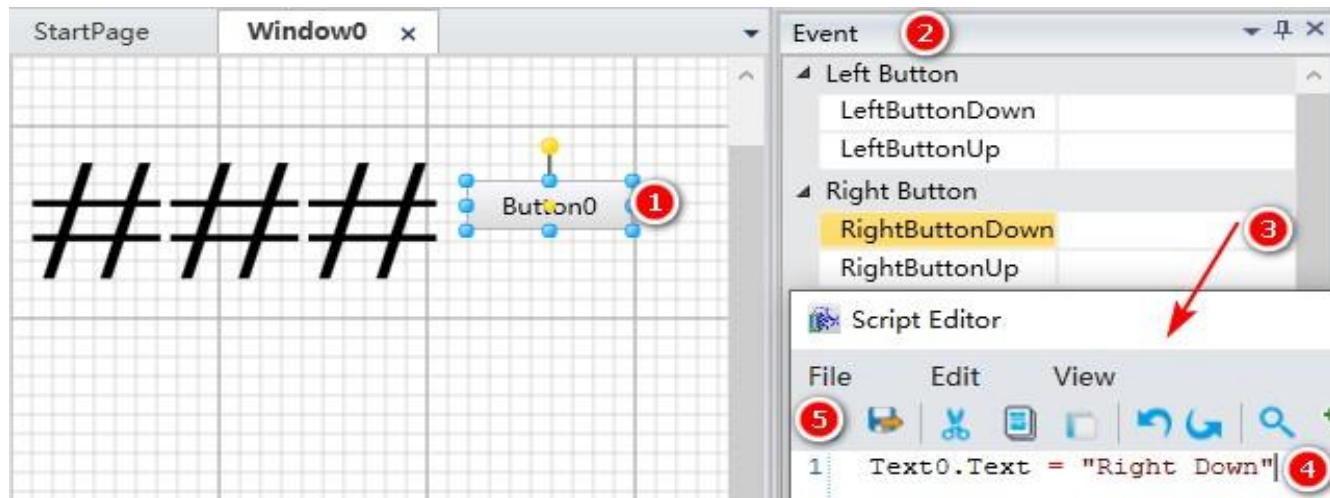
- Поняття події
- Подія лівої кнопки
- Подія правої кнопки
- Подія миші
- Подія роботи вікна
- Подія введення значення
- Подія ковзного введення
- Подія введення повороту
- Подія віконної програми
- Контрольна подія
- Клавіатура

➤ події натискання правої кнопки :

Створіть текст і встановіть його вміст за допомогою події натискання правої кнопки : (1) Створіть Text0 і Button0 у Window0



(2) Налаштуйте подію натискання правої кнопки для Button0



※Див. розділ «10.3 Подія правої кнопки» в посібнику користувача.

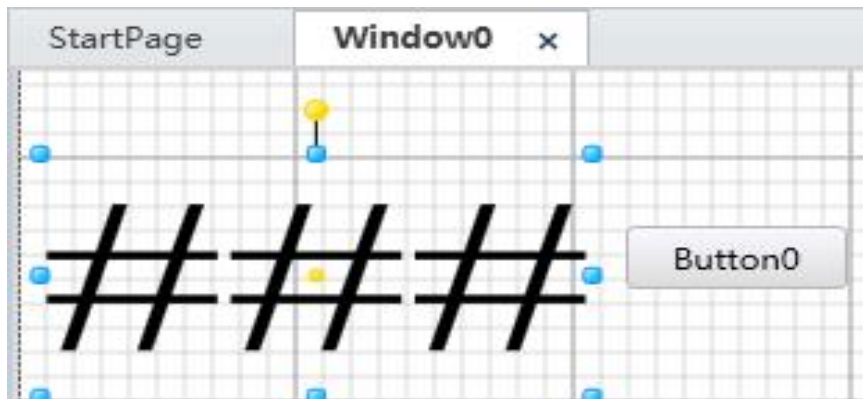
(3) Запустіть Window0, клацніть правою кнопкою миші « Button0 », а потім вміст Text0 стає « Праворуч »

Right Down

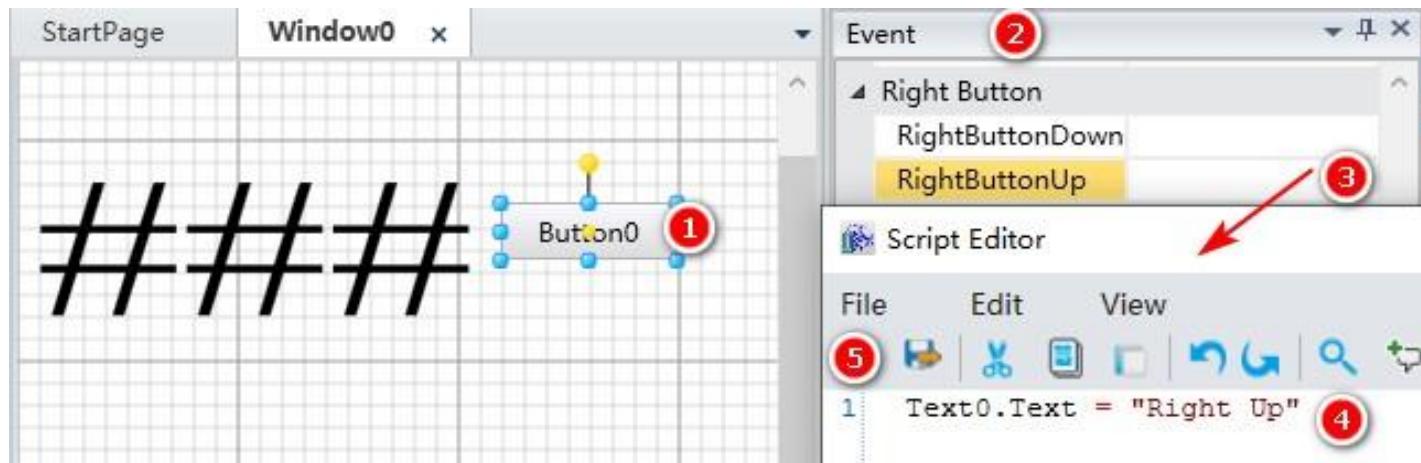


➤ Приклад події «Вгору правої кнопки »:

Створіть текст і встановіть його вміст за допомогою події натискання правої кнопки : (1) Створіть Text0 і Button0 у Window0



(2) Налаштуйте подію натискання правої кнопки Button0



※Див. розділ «10.3 Подія правої кнопки» в посібнику користувача.

Подія правої кнопки — вгору правої кнопки

(3) Запустіть Window0, клацніть правою кнопкою миші на “ Button0 ”, коли миша натиснута, вміст Text0 не змінюється, потім відпустіть мишу, вміст Text0 стає “Right Up”

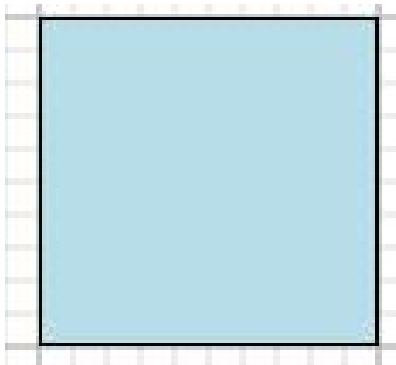
Right Up

A rectangular button with rounded corners, labeled 'Button0'. A mouse cursor is pointing at the bottom right corner of the button.

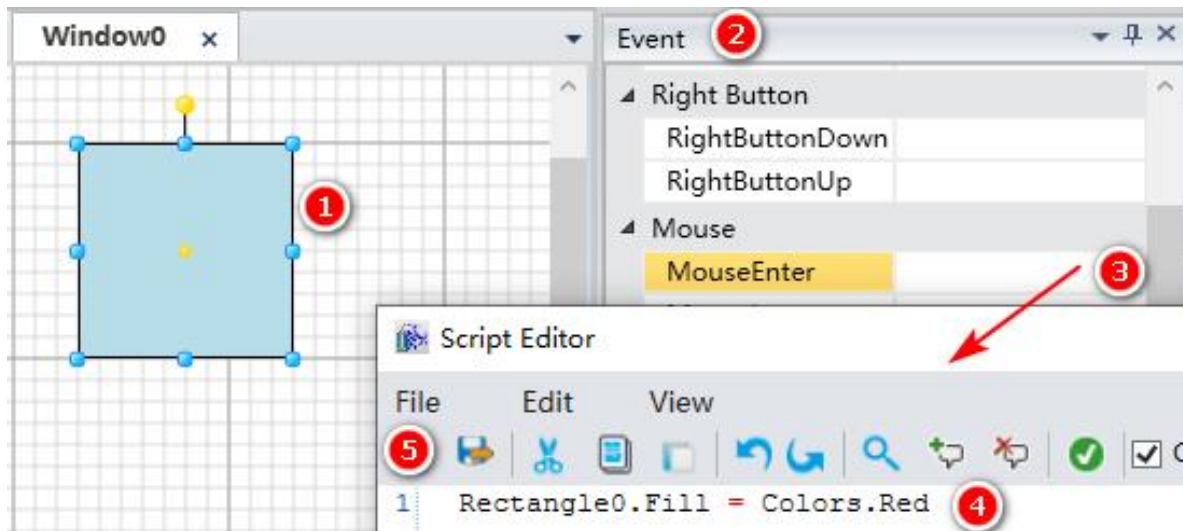
- Поняття події
- Подія лівої кнопки
- Подія правої кнопки
- Подія миші
- Подія роботи вікна
- Подія введення значення
- Подія ковзного введення
- Подія введення повороту
- Подія віконної програми
- Контрольна подія
- Клавіатура

➤ **Миша Введіть** приклад події:

Створіть прямокутник і встановіть його колір за допомогою події введення миші : (1) Створіть Rectangle0 у Window0

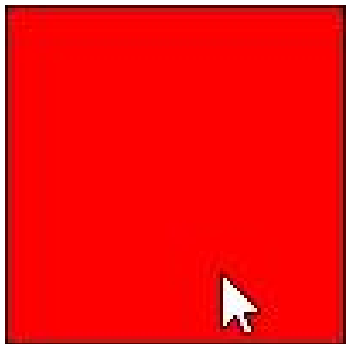


(2) Налаштуйте подію введення миші Rectangle0



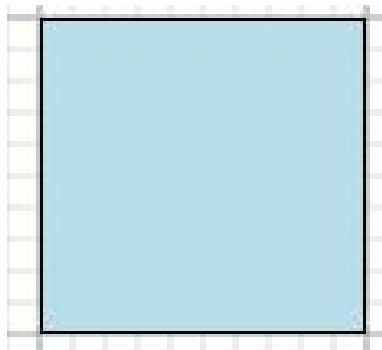
※Зверніться до розділу «10.4 Подія миші» в посібнику користувача.

(3) Запустіть Window0, перемістіть мишу, коли миша ввійде в Rectangle0, тоді Rectangle0 стане червоним

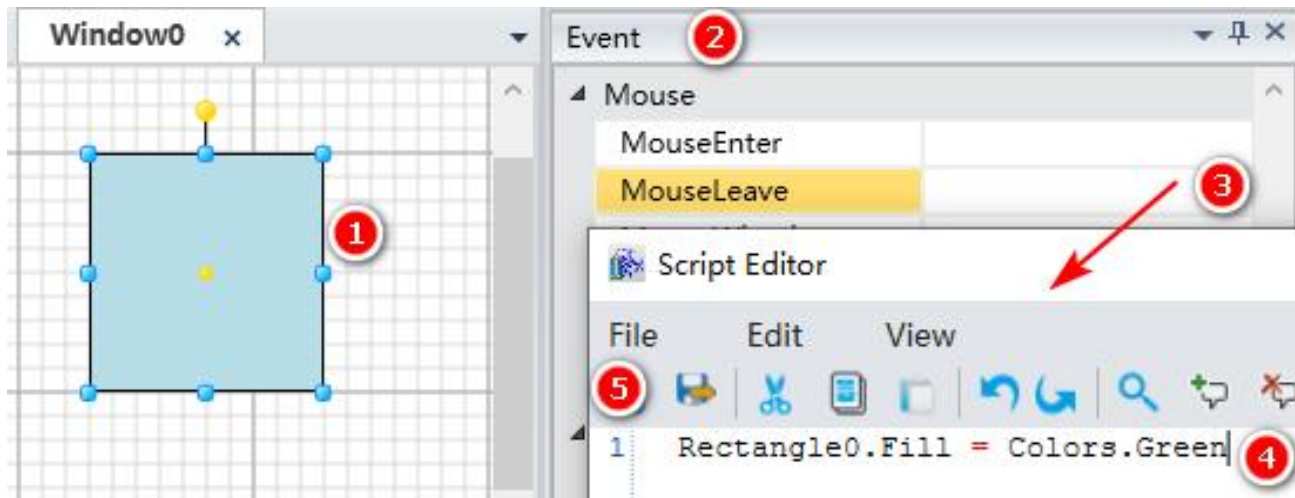


➤ події виходу миші :

Створіть прямокутник і встановіть його колір за допомогою події виходу миші : (1) Створіть Rectangle0 у Window0

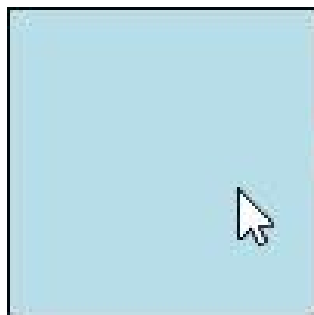


(2) Налаштувати подію виходу миші для Rectangle0

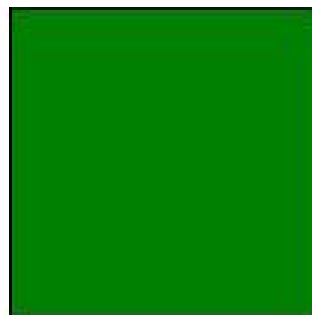


※Зверніться до розділу «10.4 Подія миші» в посібнику користувача.

(3) Запустіть Window0, перемістіть мишу, коли миша увійде в Rectangle0, колір Rectangle0 не зміниться; коли миша залишає Rectangle0, Rectangle0 стає зеленим



1



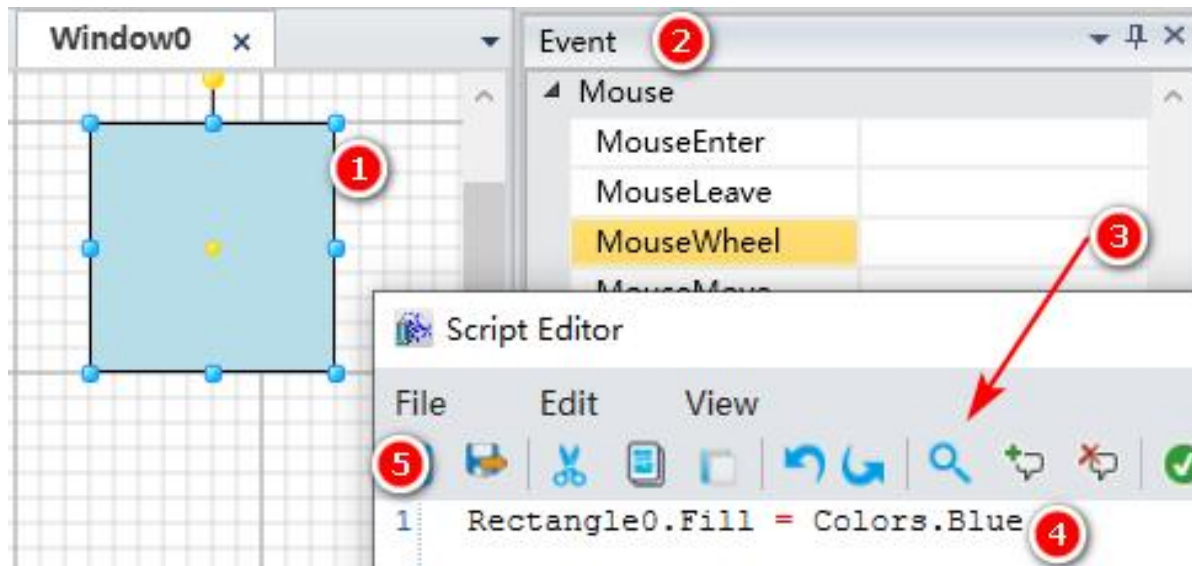
2

➤ події колеса миші :

Створіть прямокутник і встановіть його колір за допомогою події коліщатка миші : (1) Створіть Rectangle0 у Window0

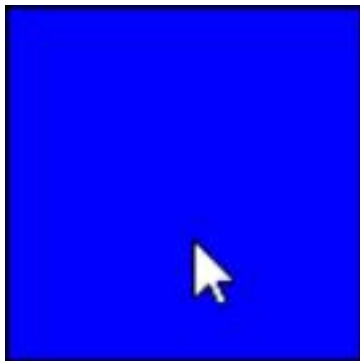


(2) Налаштувати подію коліщатка миші Rectangle0



※Зверніться до розділу «10.4 Подія миші» в посібнику користувача.

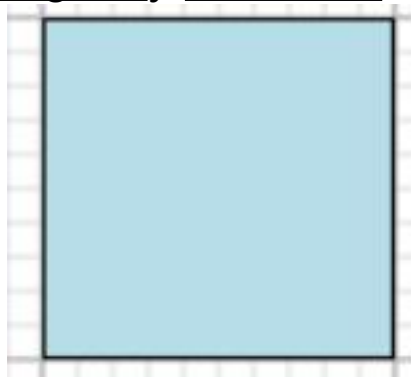
(3) Запустіть Window0, перемістіть мишу, коли миша знаходиться на Rectangle0, обертаючи мишею, Rectangle0 стає синім



➤ події Mouse Move :

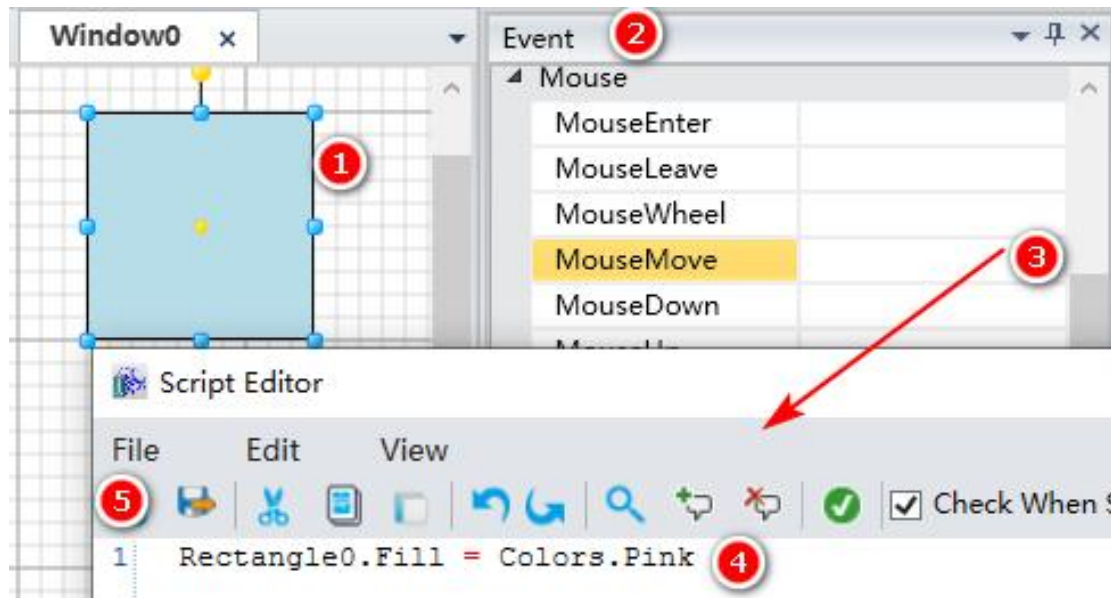
Створіть прямокутник і встановіть його колір за допомогою події переміщення миші :

(1) Створіть Rectangle0 у Window0

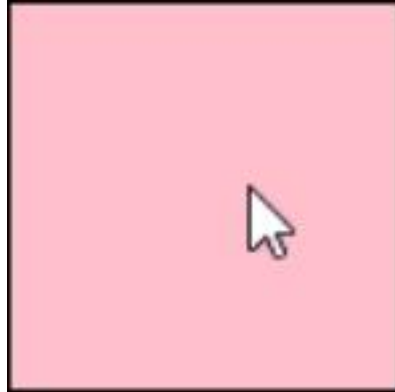


※Зверніться до розділу «10.4 Подія миші» в посібнику користувача.

(2) Налаштувати подію переміщення миші Rectangle0



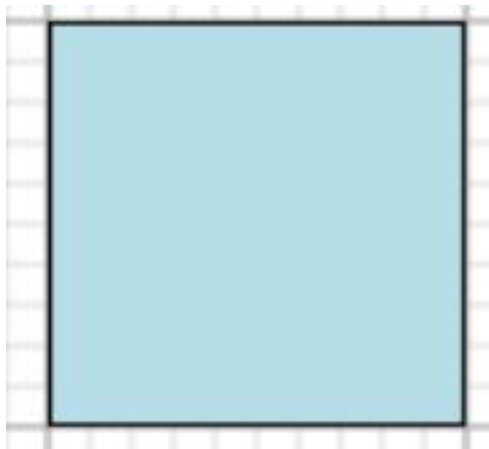
(3) Запустіть Window0, перемістіть мишу, коли миша знаходиться на Rectangle0, перемістіть мишу, Rectangle0 стане рожевим



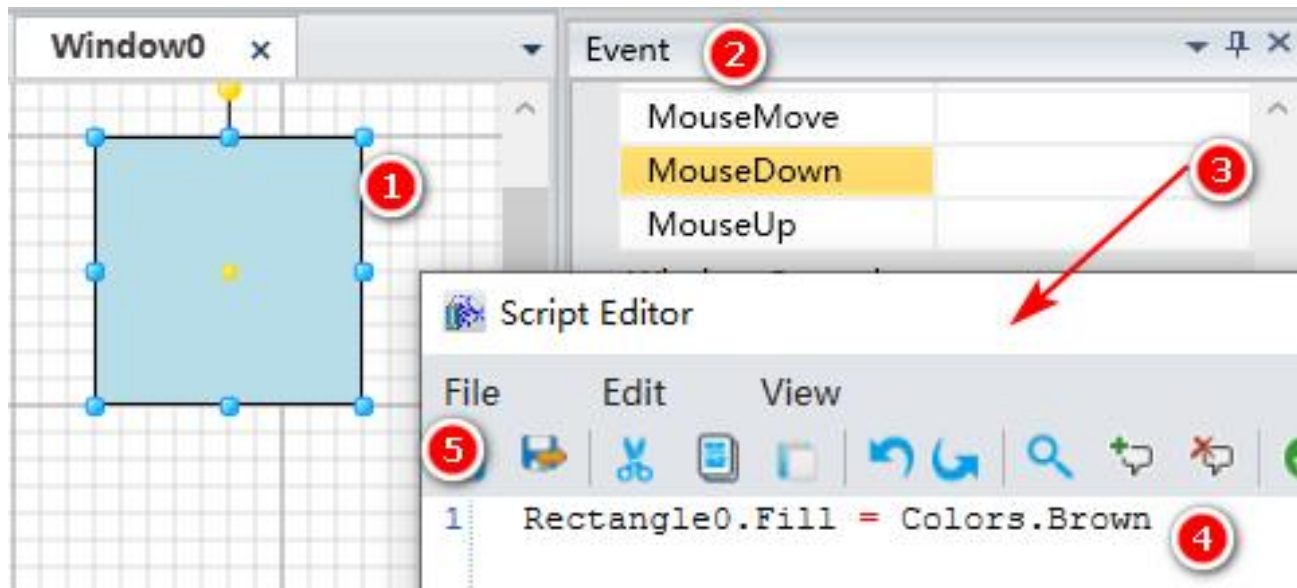
➤ події Mouse Down :

Створіть прямокутник і встановіть його колір за допомогою події миші вниз :

(1) Створіть Rectangle0 у Window0

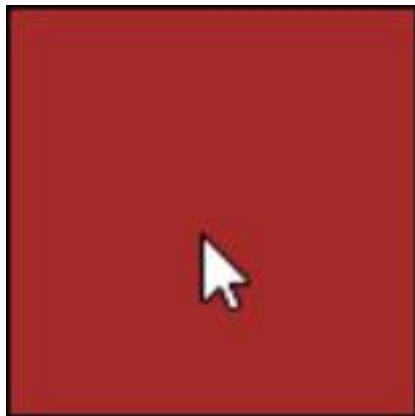


(2) Налаштувати подію миші вниз Rectangle0



※Зверніться до розділу «10.4 Подія миші» в посібнику користувача.

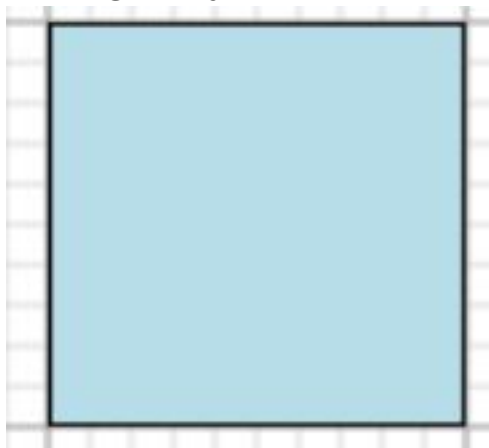
(3) Запустіть Window0 , перемістіть мишу, коли миша знаходиться на Rectangle0 , натисніть ліву або праву кнопку миші, Rectangle0 стане коричневим



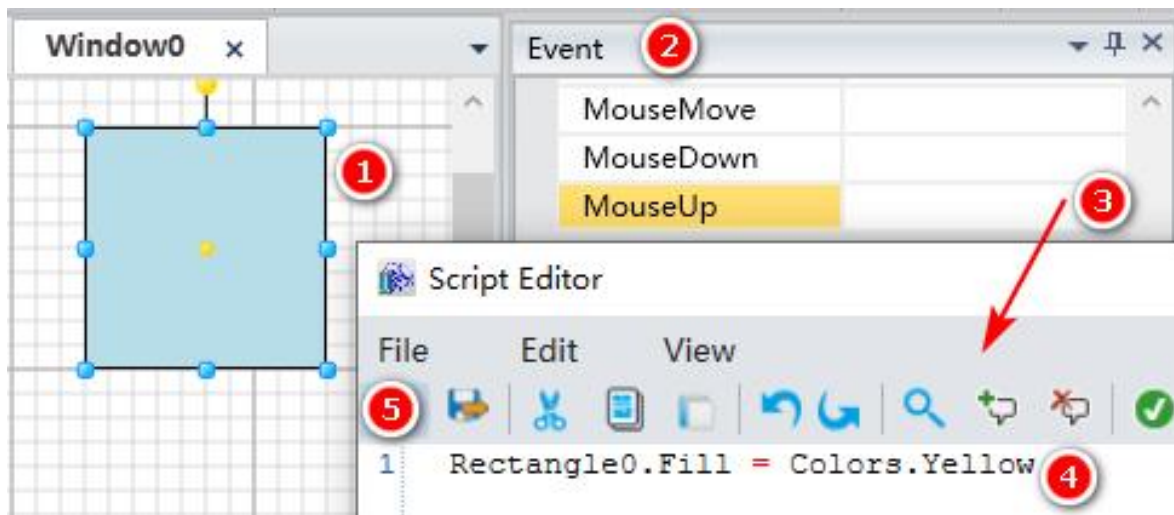
➤ події Mouse Up :

Створіть прямокутник і встановіть його колір за допомогою події підняття миші :

(1) Створіть Rectangle0 у Window0

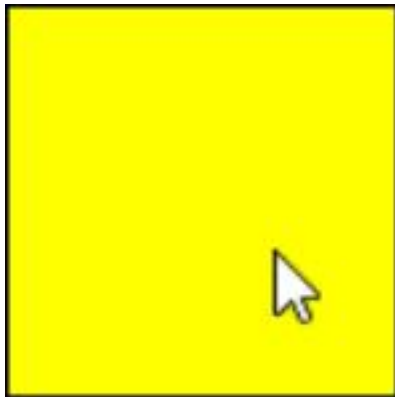


(2) Налаштуйте мишу вгору Rectangle0



※Зверніться до розділу «10.4 Подія миші» в посібнику користувача.

(3) Запустіть Window0 , перемістіть мишу, коли миша знаходиться на Rectangle0 , натисніть ліву або праву кнопку миші, колір Rectangle0 не зміниться, потім відпустіть мишу, Rectangle0 стане жовтим



- Поняття події
- Подія лівої кнопки
- Подія правої кнопки
- Подія миші
- Подія роботи вікна
- Подія введення значення
- Подія ковзного введення
- Подія введення повороту
- Подія віконної програми
- Контрольна подія
- Клавіатура

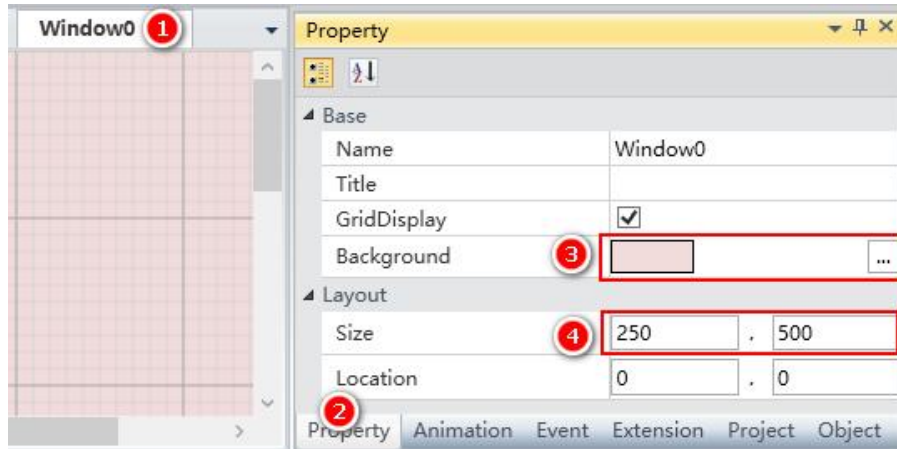
➤ Приклад події Open Window

(1) Створіть два вікна (Window0, Window1) у проекті

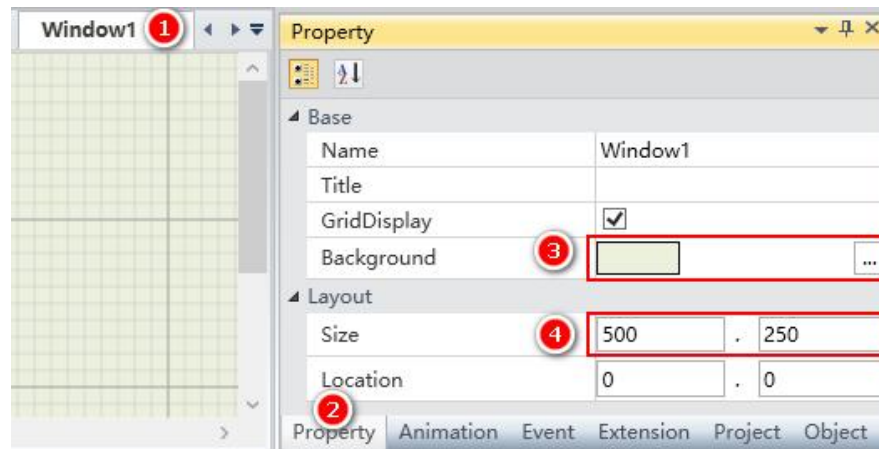


※Див. розділ «7.2.1.1 Додати вікно» в посібнику користувача.

(2) Встановить властивості двох вікон



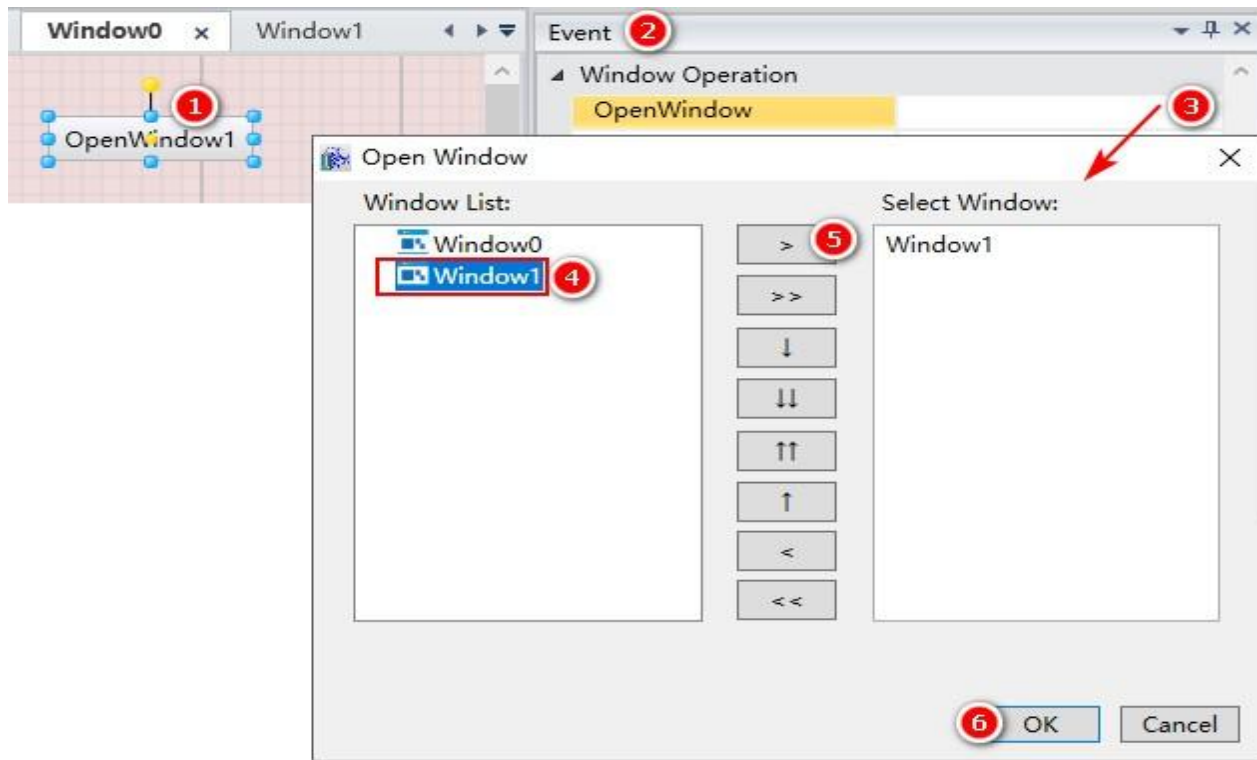
Вікно0



Вікно1

Подія операції з вікном — відкрите вікно

(3) Створіть кнопку у Window0, налаштуйте її подію відкриття вікна



(4) Запускайте лише Window0 за замовчуванням. Потім натисніть кнопку « OpenWindow1 », Вікно1 відкрито. Два вікна всі запущені



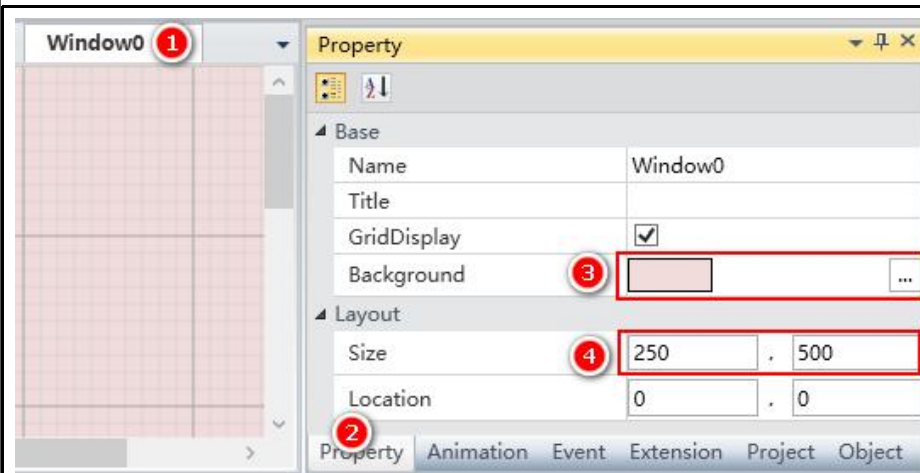
➤ Приклад події Close Window

(1) Створіть два вікна (Window0 , Window1) у проекті

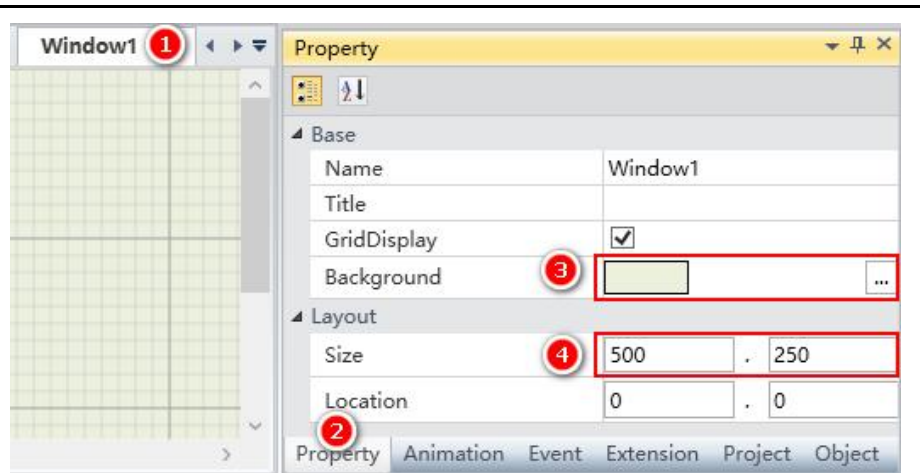


※Див. розділ «7.2.1.1 Додати вікно» в посібнику користувача.

(2) Встановіть властивості двох вікон

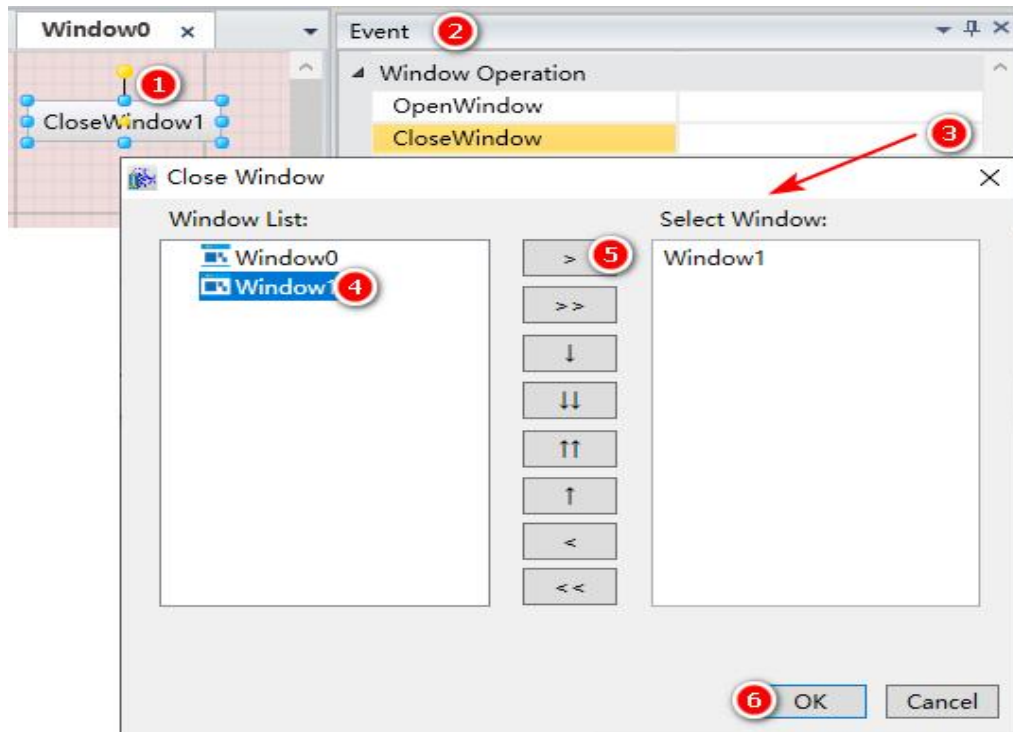


Вікно0



Вікно1

(3) Створіть кнопку у Window0, налаштуйте її подію закриття вікна



(4) Запустіть два вікна за замовчуванням. Потім натисніть кнопку « CloseWindow1 », Window1 буде закрито. Працює лише Window0

Подія операції вікна — відкрити вікно та закрити інші

➤ Приклад події «Відкрити вікно та закрити інші».

(1) Створіть два вікна (Window0 , Window1) у проекті

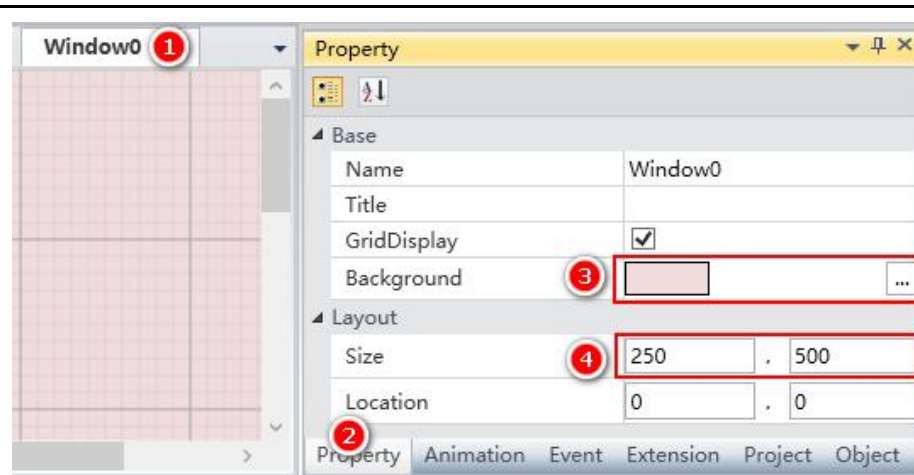


※Див. розділ «7.2.1.1 Додати вікно» в посібнику користувача.

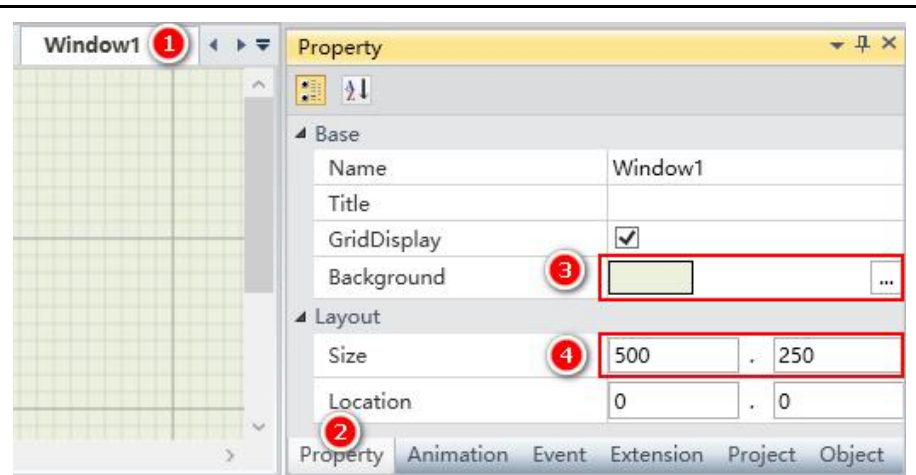


Подія операції вікна — відкрити вікно та закрити інші

(2) Встановіть властивості двох вікон



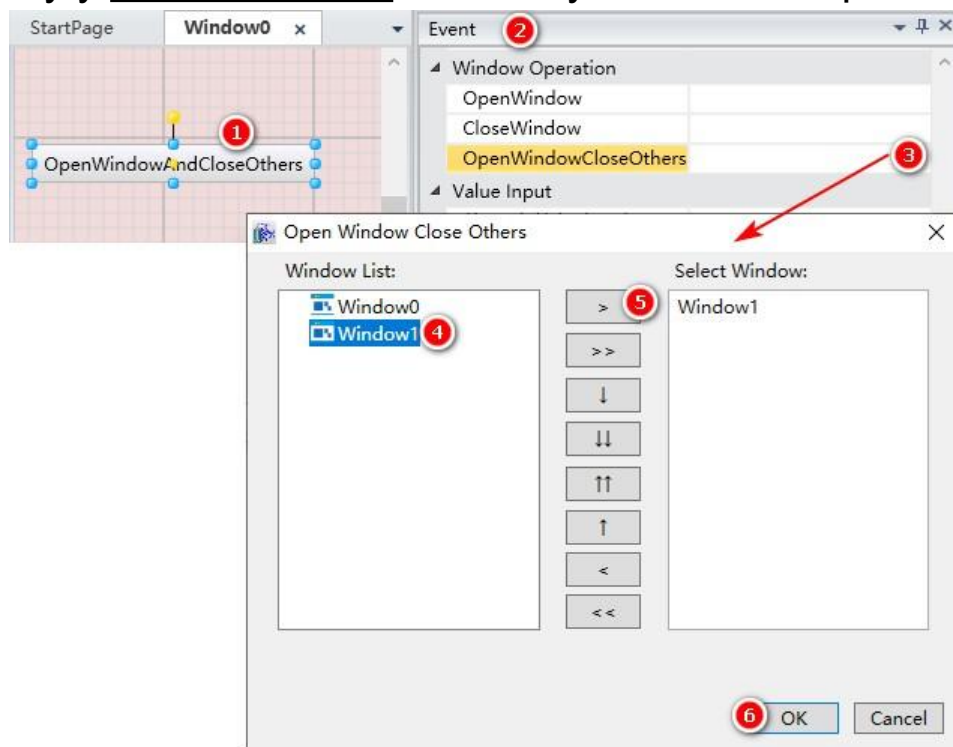
Вікно0



Вікно1

Подія операції вікна — відкрити вікно та закрити інші

(3) Створіть кнопку у вікні Window0, налаштуйте його відкритий вікно та закрийте інші події



(4) Запускайте лише Window0 за замовчуванням. Потім натисніть « OpenWindowAndCloseOthers »
Вікно1 відкривається, вікно0 закривається. Запущено лише Window1

- Поняття події
- Подія лівої кнопки
- Подія правої кнопки
- Подія миші
- Подія роботи вікна
- Подія введення значення
- Подія ковзного введення
- Подія введення повороту
- Подія віконної програми
- Контрольна подія
- Клавіатура

Подія ковзного введення— горизонтальне ковзання

➤ Приклад події горизонтального ковзання

Створіть текст і перемістіть його горизонтально та відобразіть відстань переміщення:
(1) Створіть аналогову змінну: Variable0



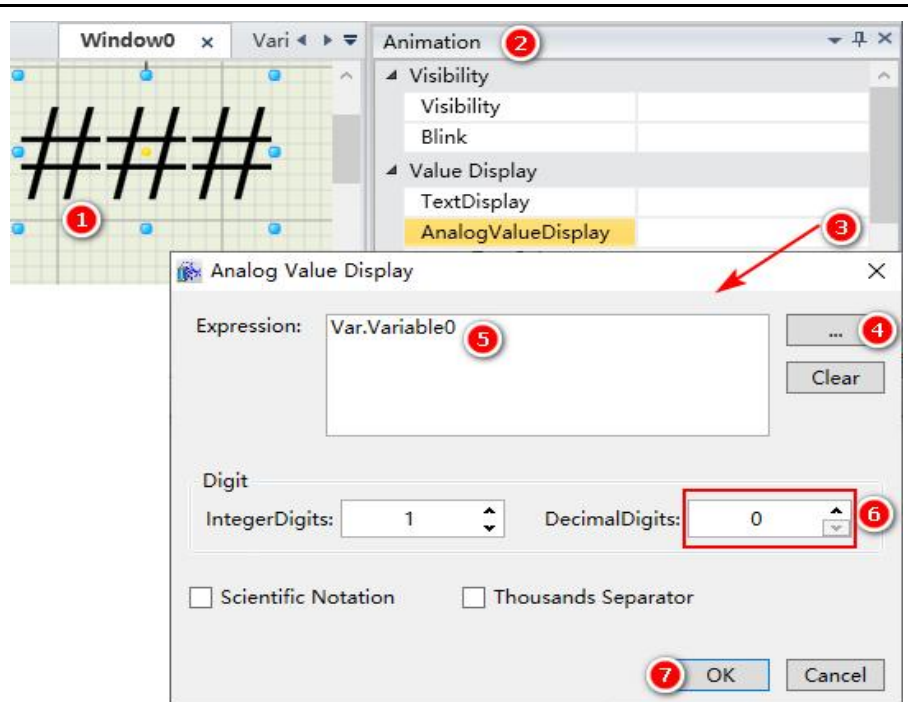
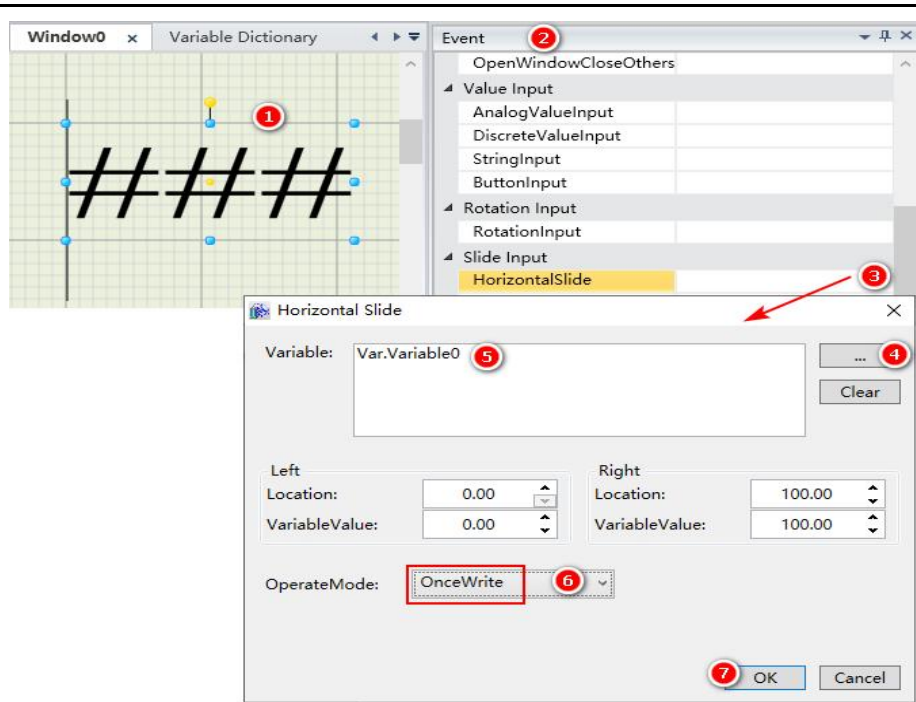
The screenshot shows the Delta software interface. On the left, there is a table with the following columns: Name, Variable Type, Initial Value, and Retentive Value. The table contains one row with the name 'Variable0', type 'Analog', and initial value '0'. On the right, there is a tree view showing the project structure: EventProject, IO Device, Variable Dictionary (circled with a red '1'), Window, and Window0. A red arrow points from the 'Variable Dictionary' node in the tree view to the 'Variable0' row in the table. The table also has a toolbar at the top with buttons: Add (circled with a red '2'), Insert, Delete, Batch Add, Import, and Export. The 'Variable Type' column header is circled with a red '4', and the 'Analog' value in the first row is circled with a red '3'.

	Name	Variable Type	Initial Value	Retentive Value
1	Variable0	Analog	0	

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

Подія ковзного введення— горизонтальне ковзання

(2) Створіть Text0 і Line0 у Window0. Налаштуйте подію та анімацію Text0

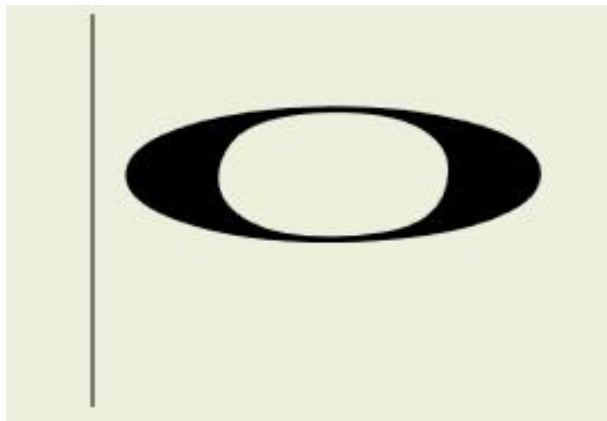


Подія горизонтального слайда

Анімація відображення аналогового значення

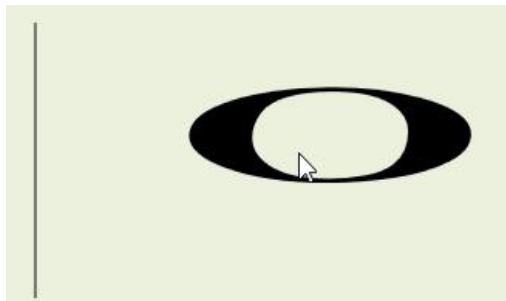
Подія ковзного введення— горизонтальне ковзання

(3) Запустіть Window0 . Початковий дисплей виглядає наступним чином



Подія ковзного введення— горизонтальне ковзання

(4) Перетягніть Text0 праворуч за допомогою миші. Під час процесу перетягування вміст Text0 залишається незмінним (Малюнок 1). Коли миша відпущена, вміст Text0 змінюється, показуючи відстань перетягування. (Малюнок 2)



Фігура 1



Малюнок 2

Подія ковзного введення — вертикальне ковзання

➤ Приклад події вертикального ковзання

Створіть текст і перемістіть його вертикально та відобразіть відстань переміщення:

(1) Створіть аналогову змінну: Variable0



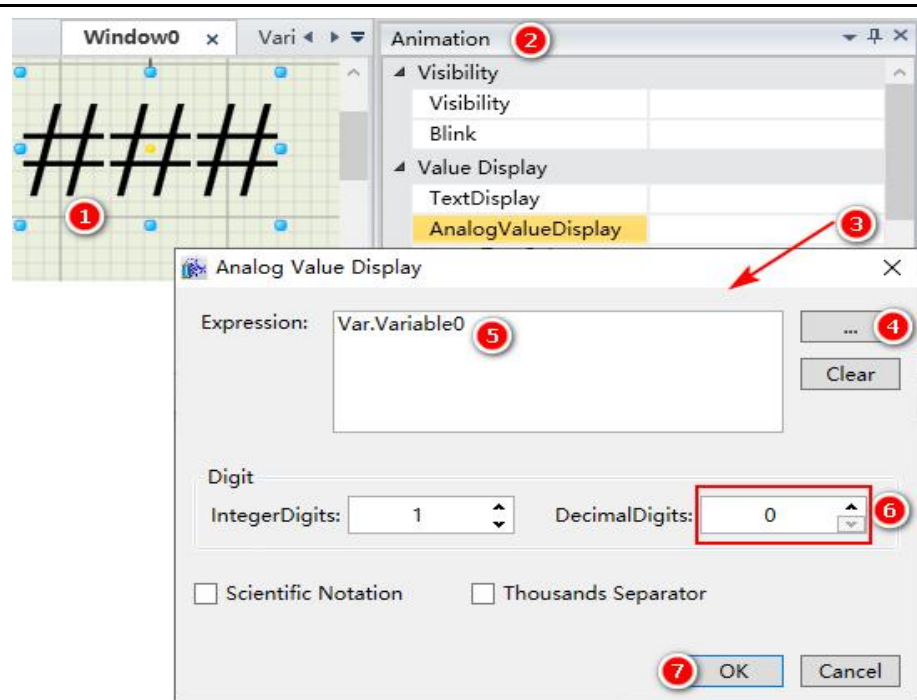
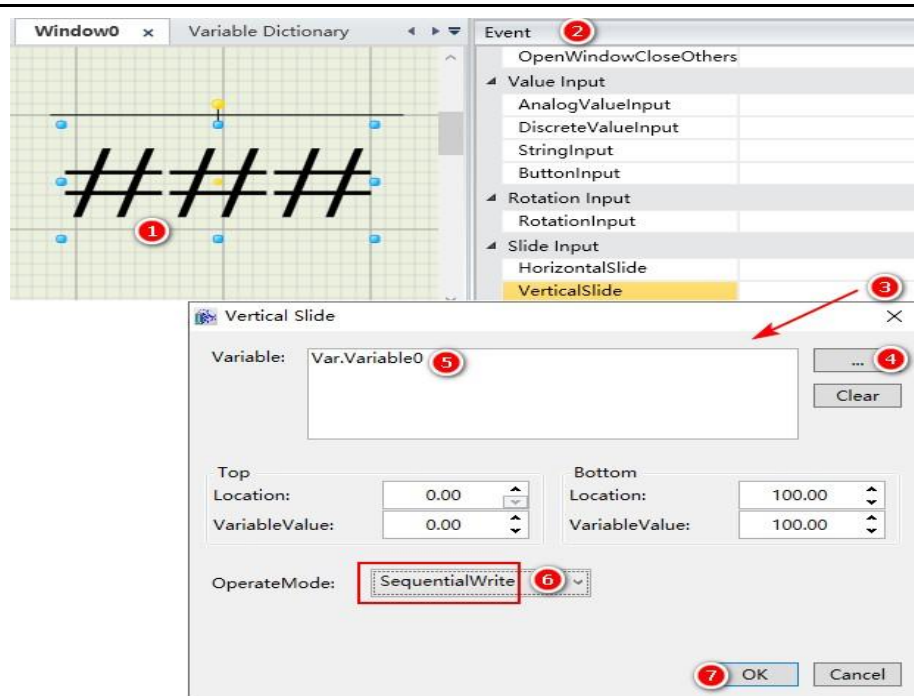
The screenshot shows the Delta software interface. On the right, the 'Variable Dictionary' is expanded, showing a tree structure with 'EventProject', 'IO Device', 'Variable Dictionary' (circled with a red 1), 'Window', and 'Window0'. A red arrow points from the 'Variable Dictionary' to the table on the left. The table has columns: Name, Variable Type, Initial Value, Retentive Value, and M. The first row is highlighted with a red circle 3 around the 'Variable0' name and a red circle 4 around the 'Analog' type. The table also has a toolbar at the top with buttons: Add (circled with a red 2), Insert, Delete, Batch Add, Import, and Export.

	Name	Variable Type	Initial Value	Retentive Value	M
1	Variable0	Analog	0		

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

Подія ковзного введення — вертикальне ковзання

(2) Створіть Text0 і Line0 у Window0. Налаштуйте подію та анімацію тексту0 _



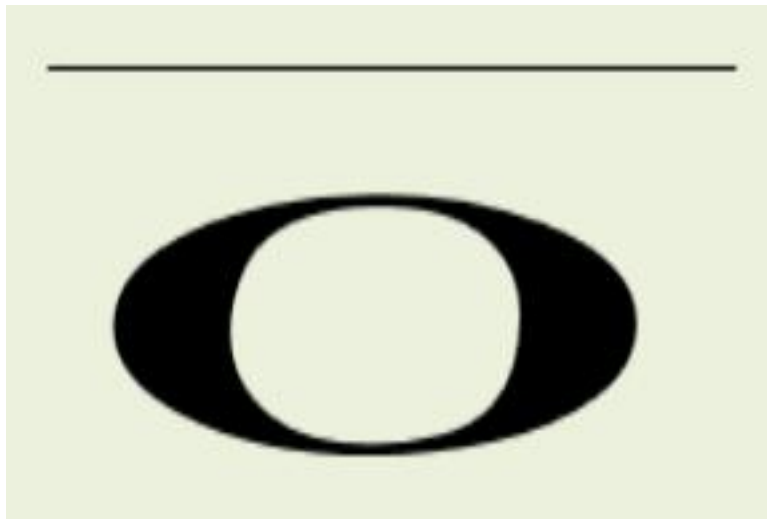
Подія вертикального слайда

Конфіденційно Delta

Анімація відображення аналогового значення

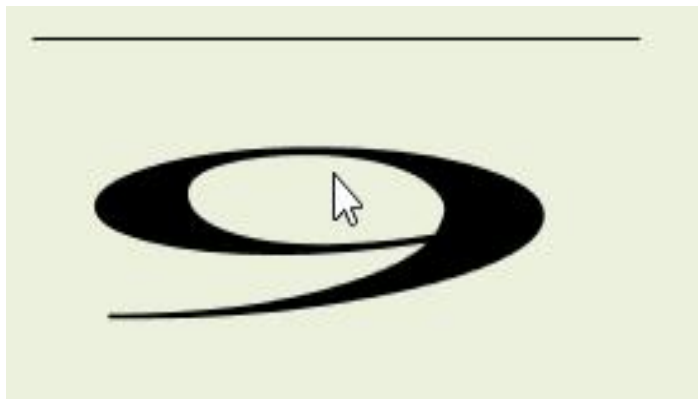
Подія ковзного введення — вертикальне ковзання

(3) Запустіть Window0 . Початковий дисплей виглядає наступним чином



Подія ковзного введення — вертикальне ковзання

(4) Перетягніть Text0 вниз за допомогою миші. Під час процесу перетягування вміст Text0 змінюється відповідно до зміни відстані перетягування, тобто Text0 відображає відстань перетягування в реальному часі



- Поняття події
- Подія лівої кнопки
- Подія правої кнопки
- Подія миші
- Подія роботи вікна
- Подія введення значення
- Подія ковзного введення
- Подія введення повороту
- Подія віконної програми
- Контрольна подія
- Клавіатура

Подія введення значення— введення аналогового значення

➤ Приклад події введення аналогового значення

Введіть аналогове значення в текст за допомогою події введення аналогового значення:

(1) Створіть аналогову змінну: Variable0



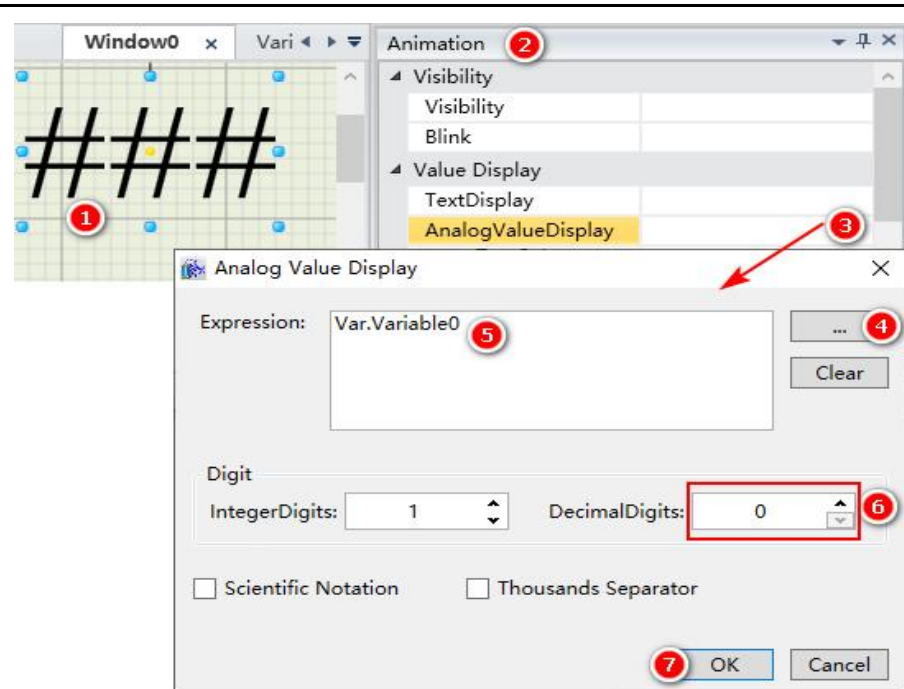
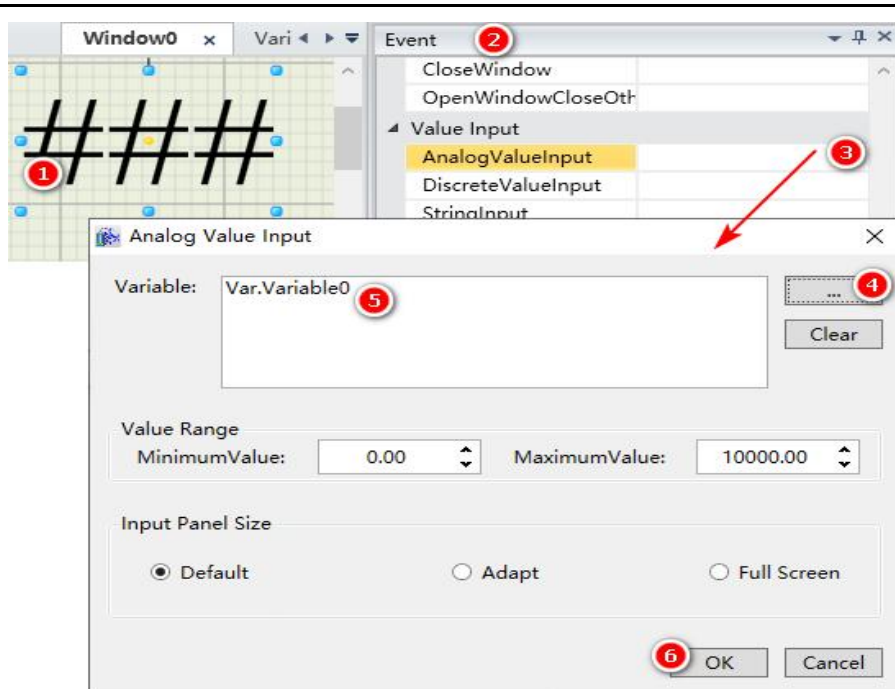
The screenshot shows the Delta software interface. On the right, the 'Variable Dictionary' is expanded, showing a tree structure with 'EventProject', 'IO Device', 'Variable Dictionary' (highlighted with a red circle 1), 'Window', and 'Window0'. A red arrow points from the 'Variable Dictionary' to the table on the left. The table has columns: Name, Variable Type, Initial Value, and Retentive Value. The first row is highlighted with a red circle 3, showing 'Variable0' with an 'Analog' type (highlighted with a red circle 4) and an 'Initial Value' of '0'.

	Name	Variable Type	Initial Value	Retentive Value
1	Variable0	Analog	0	

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

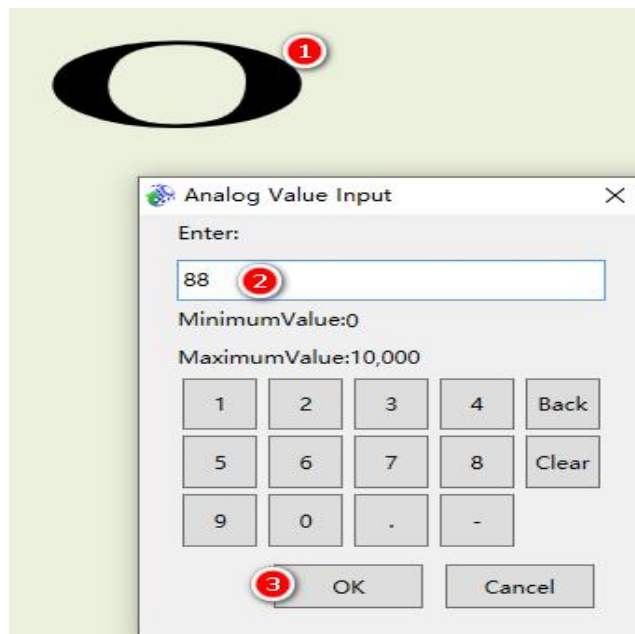
Подія введення значення— введення аналогового значення

(2) Створіть Text0 у Window0. Налаштуйте подію та анімацію Text0



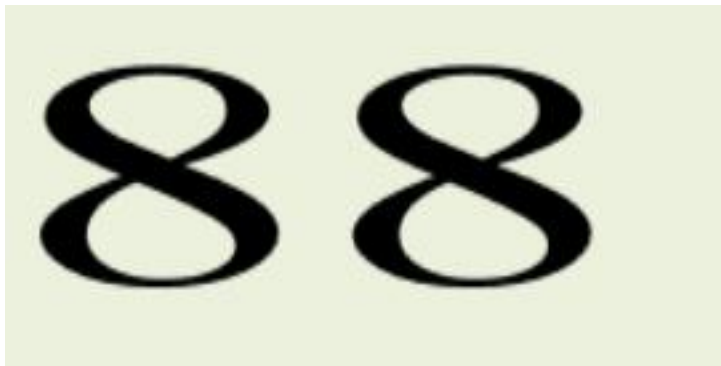
Подія введення значення— введення аналогового значення

(3) Запустіть Window0. Натисніть Text0, щоб ввести 88 у Text0



Подія введення значення— введення аналогового значення

(4) Text0 відображає 88.





Подія введення значення — введення дискретного значення

➤ Приклад події введення дискретного значення

Введіть дискретне значення в текст за допомогою події введення дискретного значення:

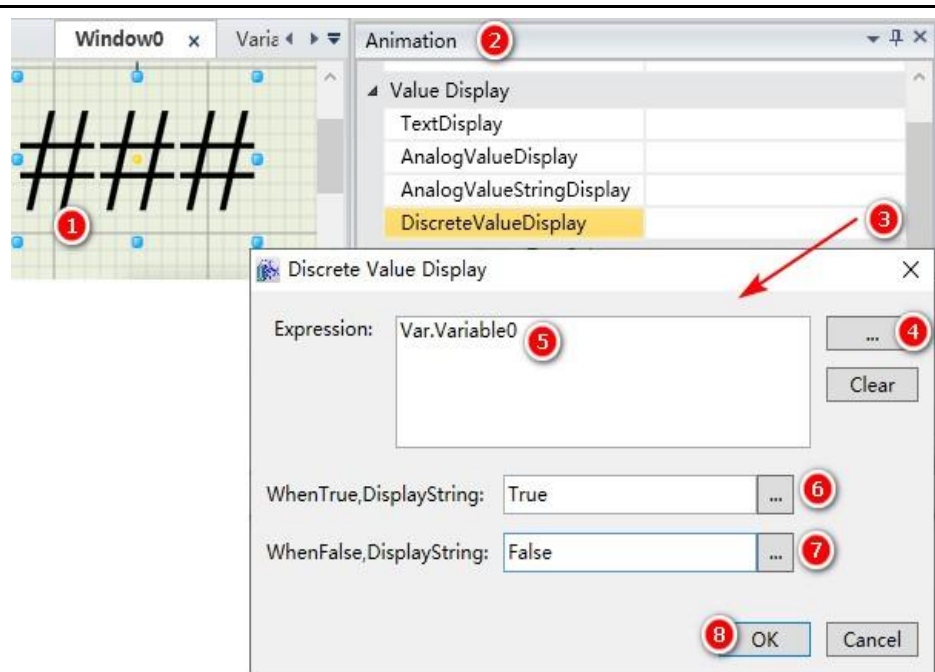
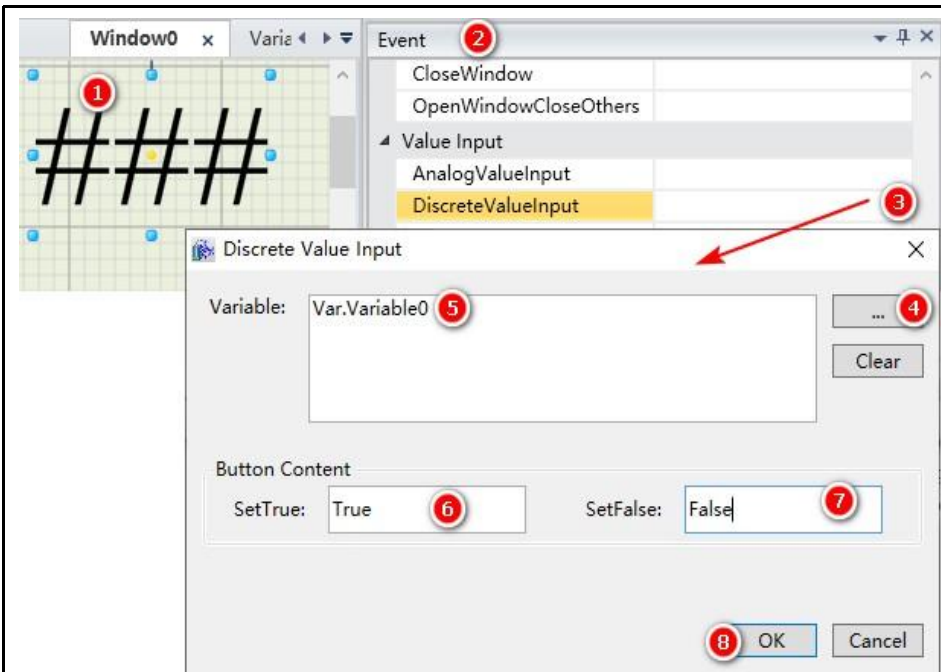
(1) Створіть цифрову змінну: Variable0

	Name	Variable Type	Initial Value	Retentive Value	Min
1	Variable0	Digital	False		

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

Подія введення значення — введення дискретного значення

(2) Створіть Text0 у Window0 . Налаштуйте подію та анімацію Text0



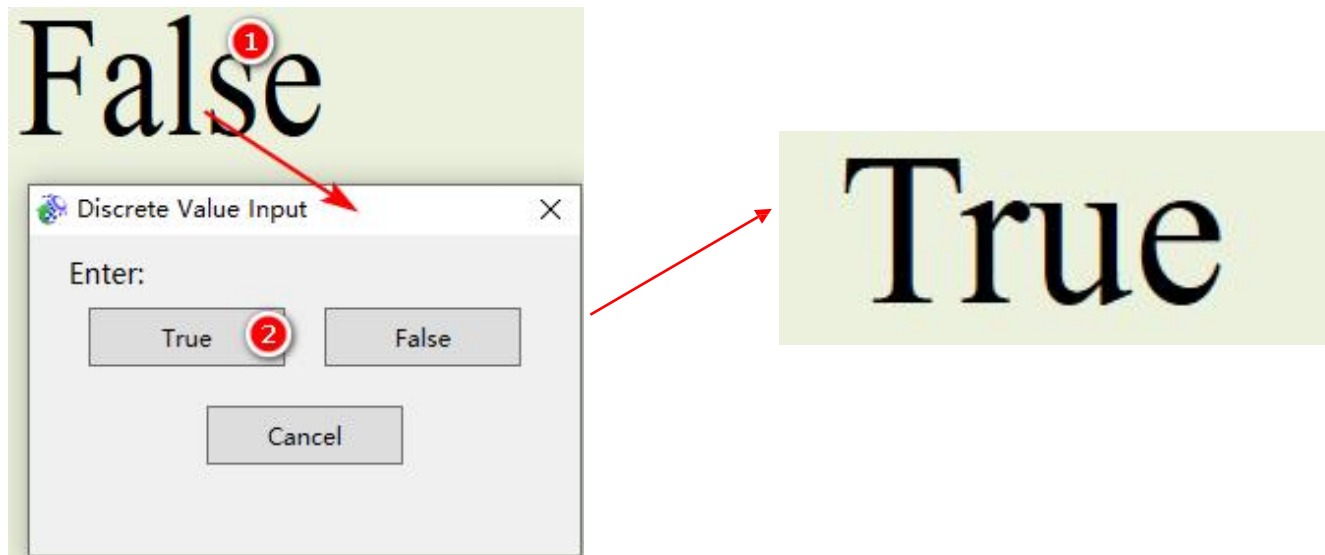
Подія введення значення — введення дискретного значення

(3) Запустіть Window0 . Text0 за замовчуванням відображає « False ».

The word 'False' is displayed in a large, black, serif font, centered on a light green rectangular background.

Подія введення значення — введення дискретного значення

(4) Клацніть Text0, щоб ввести True, тоді Text0 відобразить «True»



➤ Приклад події String Input

Введіть рядок у текст за допомогою події string input :

(1) Створіть рядкову змінну: Variable0



The screenshot shows the Delta software interface. On the right, the 'Variable Dictionary' is expanded, showing a tree structure with 'EventProject', 'IO Device', 'Variable Dictionary' (marked with a red circle 1), 'Window', and 'Window0'. On the left, a table lists variables. The table has columns: Name, Variable Type, Initial Value, Retentive Value, and Minimum Value. The first row is highlighted and contains the following data:

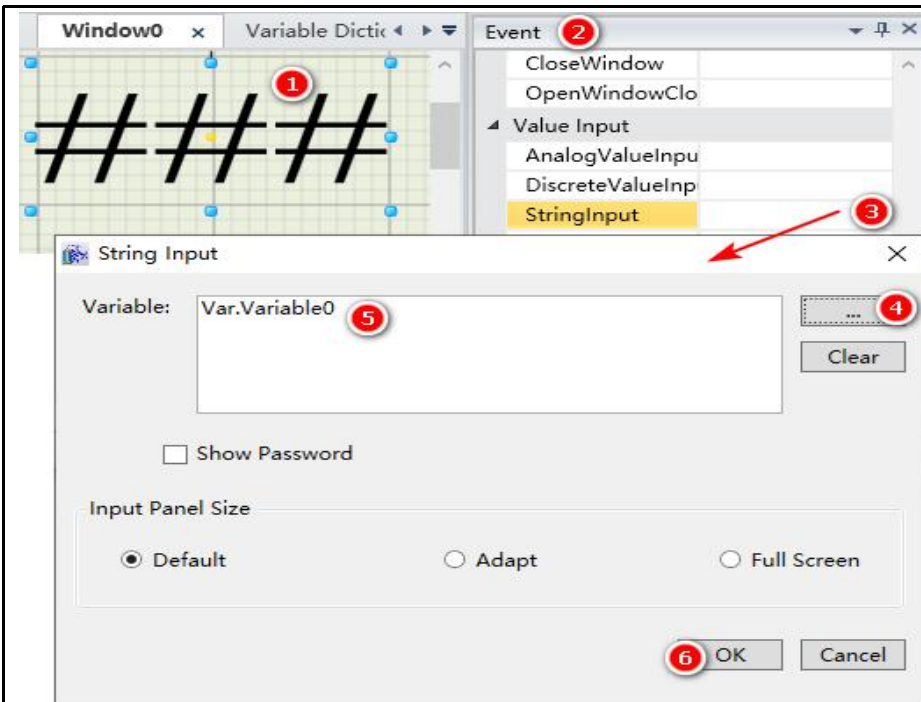
	Name	Variable Type	Initial Value	Retentive Value	Minimum Value
1	Variable0 (3)	String (4)	StringTest (5)		N

Red circles with numbers 1 through 5 highlight specific elements: 1 points to 'Variable Dictionary' in the tree, 2 points to the 'Add' button, 3 points to 'Variable0', 4 points to 'String', and 5 points to 'StringTest'.

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

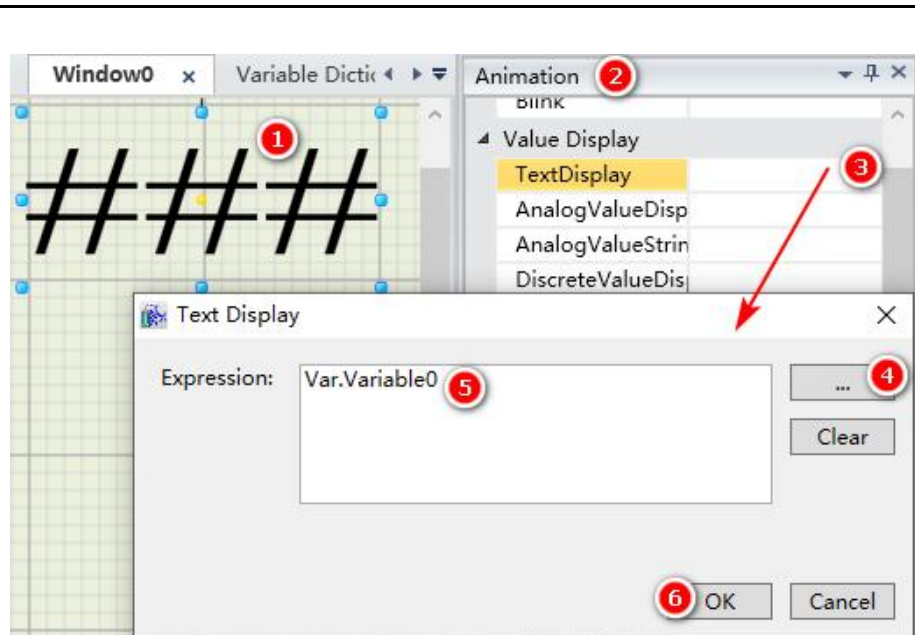
Подія введення значення — введення рядка

(2) Створіть Text0 у Window0 . Налаштуйте подію та анімацію Text0



Подія введення рядка

Конфіденційно Delta



Анімація відображення тексту

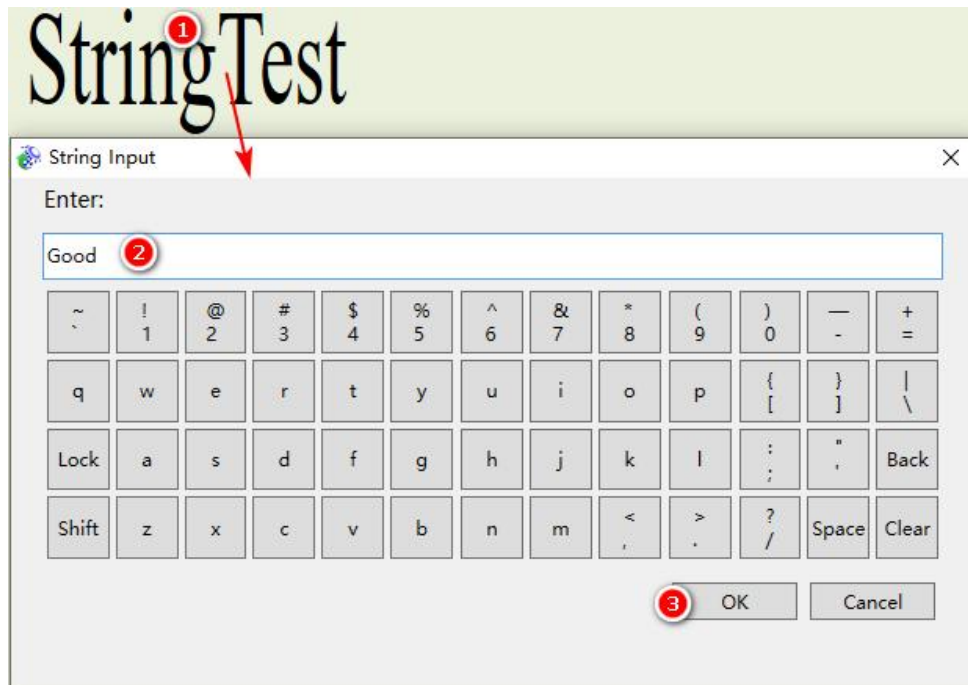
Подія введення значення — введення рядка

(3) Запустіть Window0 . Text0 за замовчуванням відображає « StringTest »

StringTest

Подія введення значення — введення рядка

(4) Клацніть Text0 , щоб ввести в нього Good, тоді Text0 відобразить « Good »



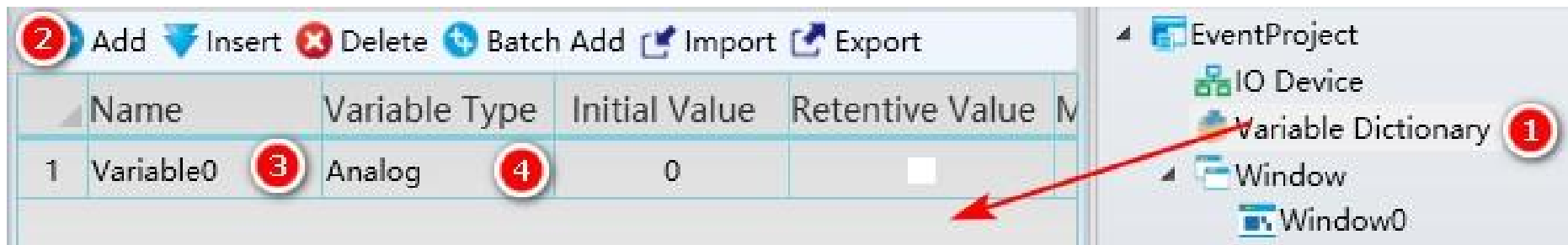
Good

Подія введення значення — введення за допомогою кнопки

➤ Приклад події введення кнопки1

Встановіть значення змінної за допомогою події введення кнопки:

(1) Створіть рядкову змінну: Variable0



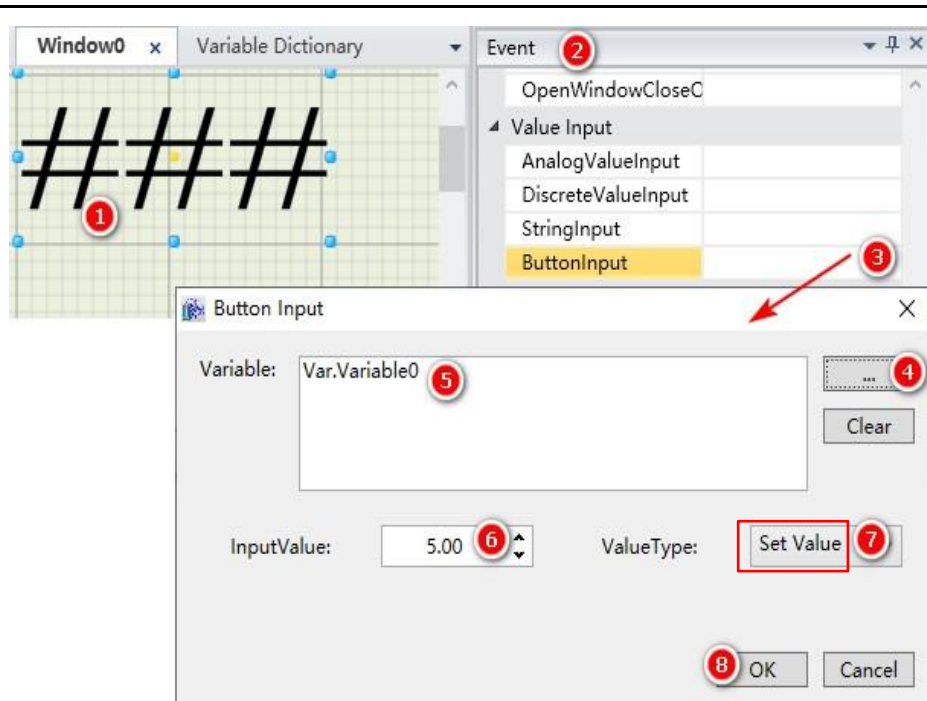
The screenshot shows the Delta software interface. On the right, the 'Variable Dictionary' is expanded, showing a tree structure with 'EventProject', 'IO Device', 'Variable Dictionary' (marked with a red circle 1), 'Window', and 'Window0'. A red arrow points from the 'Variable Dictionary' to the 'Variable Declaration' table on the left. The table has columns: Name, Variable Type, Initial Value, Retentive Value, and M. The first row is 'Variable0' (marked with a red circle 3), 'Analog' (marked with a red circle 4), '0', and an empty 'Retentive Value' field. The table also has buttons: Add (marked with a red circle 2), Insert, Delete, Batch Add, Import, and Export.

	Name	Variable Type	Initial Value	Retentive Value	M
1	Variable0	Analog	0		

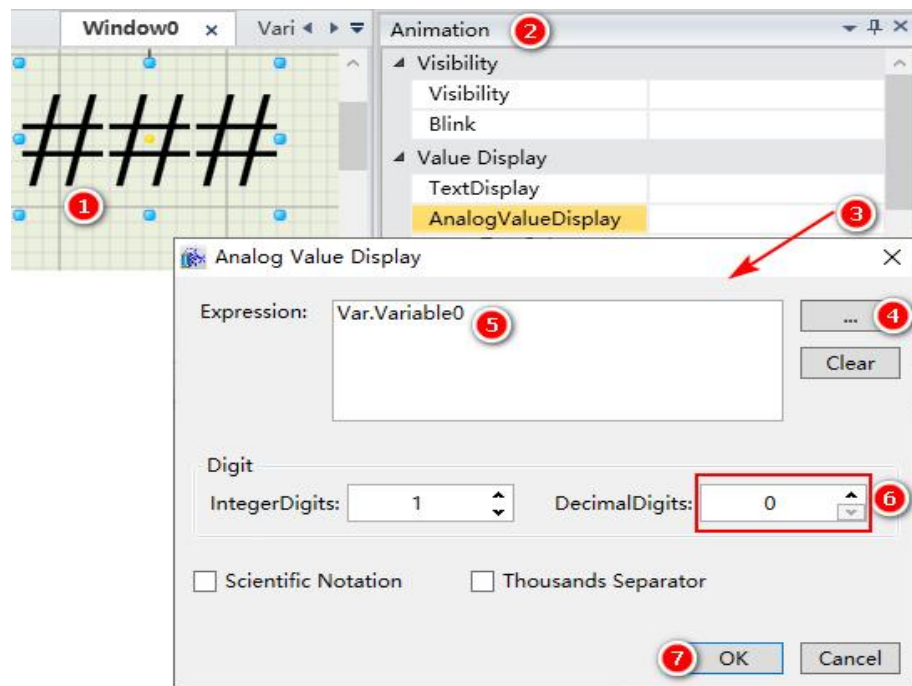
※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

Подія введення значення — введення за допомогою кнопки

(2) Створіть Text0 у Window0 . Налаштуйте подію та анімацію Text0



Подія введення кнопки



Анімація відображення аналогового значення

Подія введення значення — введення за допомогою кнопки

(3) Запустіть Window0 . Text0 за замовчуванням відображає «0».



Подія введення значення — введення за допомогою кнопки

(4) Клацніть Text0, потім Text0 відобразить «5»



Подія введення значення — введення за допомогою кнопки

➤ Приклад події введення кнопки2

Установіть значення змінної, збільшуючи кожен раз на 5:

(1) Створіть аналогову змінну: Variable0



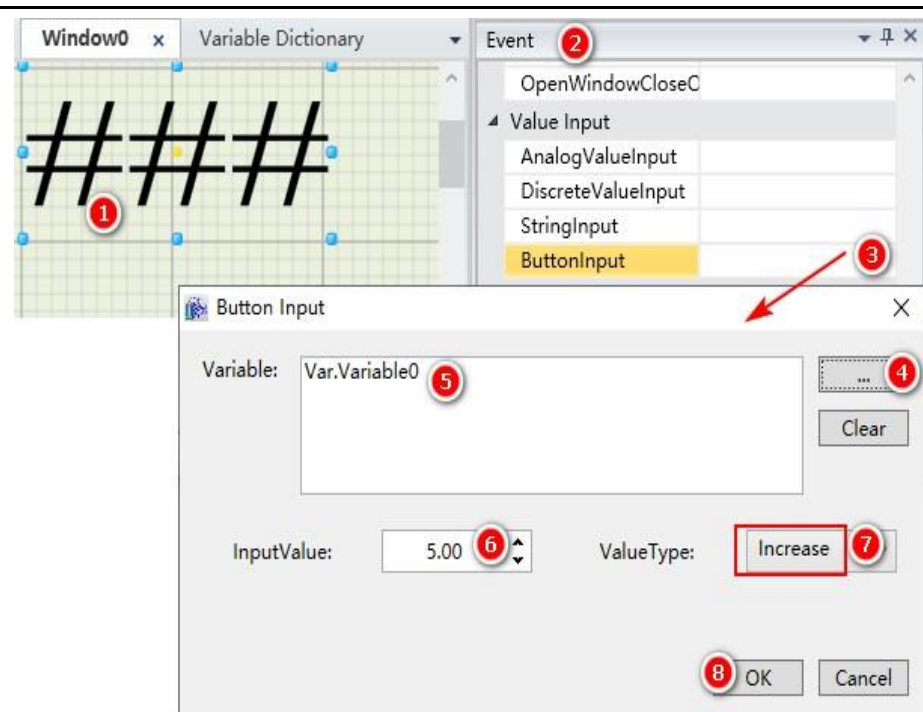
The screenshot shows the Delta software interface. On the right, the 'Variable Dictionary' is expanded, showing a tree structure with 'EventProject', 'IO Device', 'Variable Dictionary' (circled with a red 1), 'Window', and 'Window0'. A red arrow points from the 'Variable Dictionary' to the table on the left. The table has columns: Name, Variable Type, Initial Value, and Retentive Value. The first row is 'Variable0' (circled with a red 3), 'Analog' (circled with a red 4), '0', and an empty box. The table also has buttons at the top: Add (circled with a red 2), Insert, Delete, Batch Add, Import, and Export.

	Name	Variable Type	Initial Value	Retentive Value
1	Variable0	Analog	0	

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

Подія введення значення — введення за допомогою кнопки

(2) Створіть Text0 у Window0. Налаштуйте подію та анімацію Text0



Window0 x Variable Dictionary

Event 2

- OpenWindowCloseC
- Value Input
 - AnalogValueInput
 - DiscreteValueInput
 - StringInput
 - ButtonInput

Button Input

Variable: Var.Variable0 5

Clear

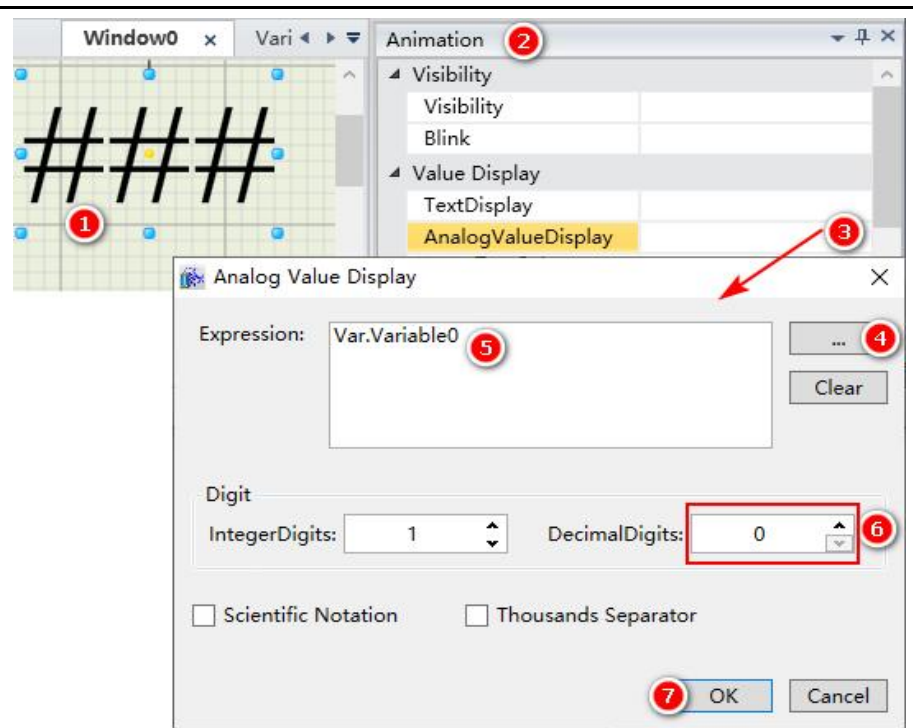
InputValue: 5.00 6

ValueType: Increase 7

OK 8 Cancel

Подія введення кнопки

Конфіденційно Delta



Window0 x Vari

Animation 2

- Visibility
 - Visibility
 - Blink
- Value Display
 - TextDisplay
 - AnalogValueDisplay

Analog Value Display

Expression: Var.Variable0 5

Clear

Digit

IntegerDigits: 1

DecimalDigits: 0 6

☐ Scientific Notation ☐ Thousands Separator

OK 7 Cancel

Анімація відображення аналогового значення

Подія введення значення — введення за допомогою кнопки

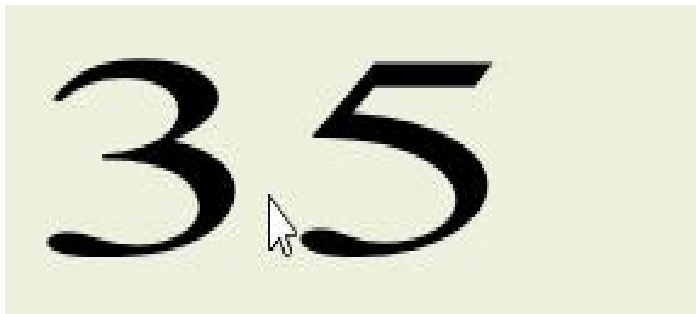
(3) Запустіть Window0. Text0 за замовчуванням відображає «0».





Подія введення значення — введення за допомогою кнопки

(4) Клацніть Text0 кілька разів, кожного разу, коли відображуване значення Text0 збільшується на 5



- Поняття події
- Подія лівої кнопки
- Подія правої кнопки
- Подія миші
- Подія роботи вікна
- Подія введення значення
- Подія ковзного введення
- Подія введення повороту
- Подія віконної програми
- Контрольна подія
- Клавіатура

➤ Приклад події Rotation Input

Створіть прямокутник, який можна обертати:

(1) Створіть аналогову змінну: Variable0

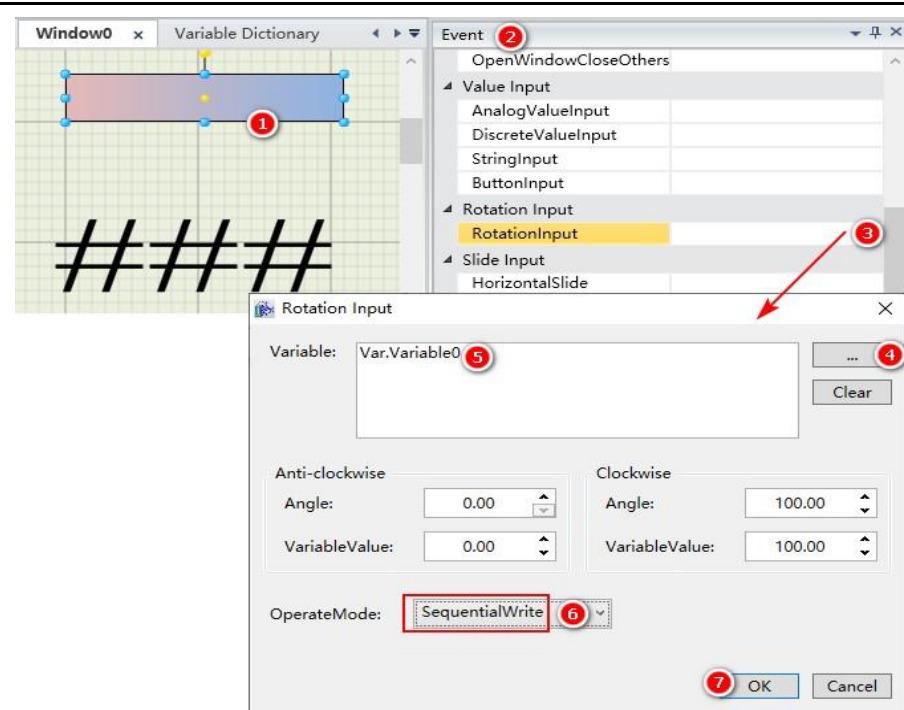


The screenshot shows the Delta software interface. On the right, the 'Variable Dictionary' tree is visible, with 'Variable0' highlighted under the 'Window' folder. A red arrow points from this tree to the table on the left. The table has columns: Name, Variable Type, Initial Value, and Retentive Value. The first row shows 'Variable0' with an 'Analog' type and an initial value of '0'. Red circles with numbers 1 through 4 highlight specific elements: 1 on the Variable Dictionary tree, 2 on the 'Add' button, 3 on the 'Variable0' cell, and 4 on the 'Analog' cell.

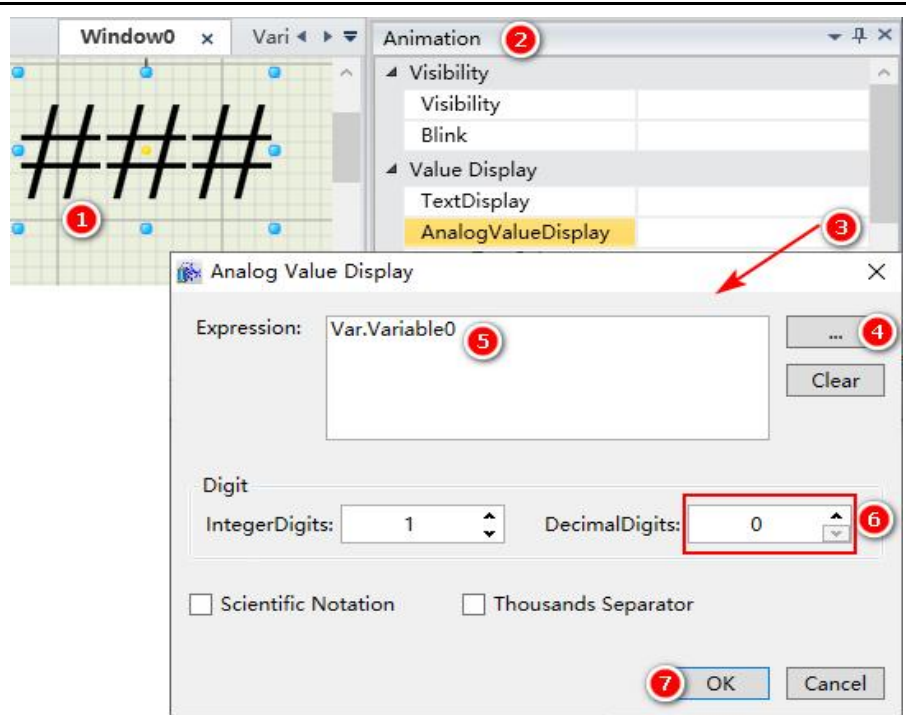
	Name	Variable Type	Initial Value	Retentive Value
1	Variable0	Analog	0	

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

(2) Створіть Rectangle0 і Text0 у Window0 . Налаштуйте вхідну подію обертання Rectangle0 та анімацію відображення аналогового значення Text0

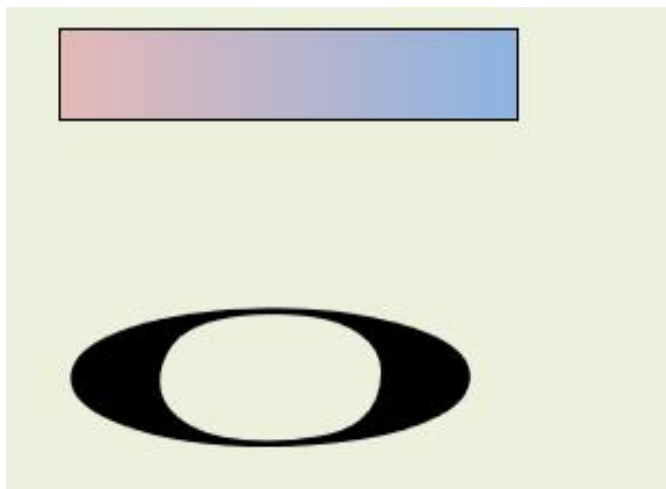


Подія введення кнопки

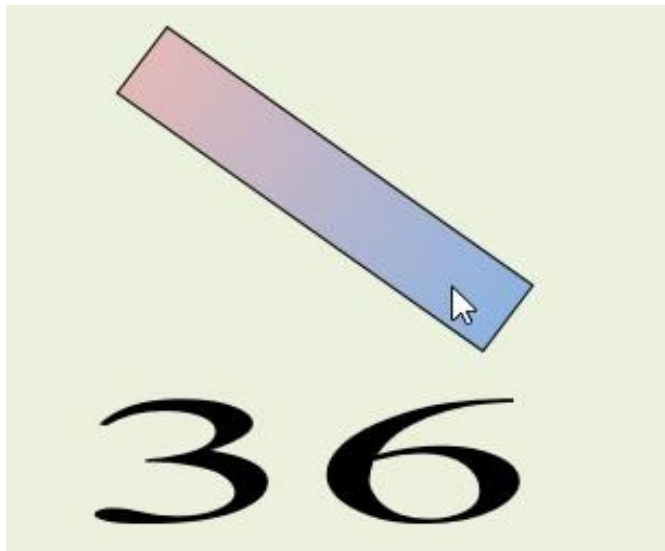


Анімація відображення аналогового значення

(3) Запустіть Window0 . Початковий дисплей виглядає наступним чином



(4) Перетягніть Rectangle0 за годинниковою стрілкою за допомогою миші. Під час процесу перетягування вміст Text0 змінюється зі зміною кута повороту, тобто Text0 відображає кут повороту Rectangle0 у реальному часі



- Поняття події
- Подія лівої кнопки
- Подія правої кнопки
- Подія миші
- Подія роботи вікна
- Подія введення значення
- Подія ковзного введення
- Подія введення повороту
- Подія віконної програми
- Контрольна подія
- Клавіатура

➤ Приклад події Window Program

Створіть віконну програму для виконання під час виконання:

(1) Створіть аналогову змінну: Variable0

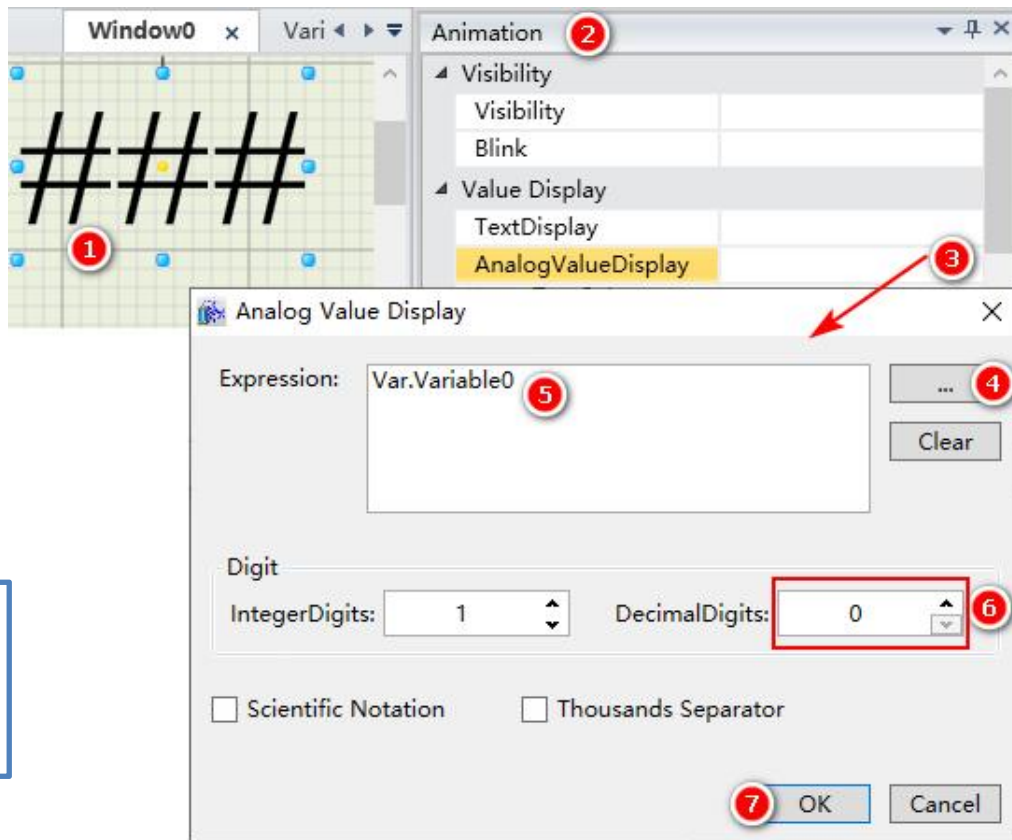


The screenshot displays the Delta software interface. On the left, a table lists variables. The first row shows 'Variable0' with an 'Analog' type and an initial value of '0'. On the right, a tree view shows the project structure: 'EventProject' contains 'IO Device', 'Variable Dictionary', 'Window', and 'Window0'. A red arrow points from the 'Variable Dictionary' node in the tree to the 'Variable0' entry in the table. Red circles with numbers 1 through 5 highlight specific elements: 1 on 'Variable Dictionary', 2 on the 'Add' button, 3 on the 'Variable0' name, 4 on the 'Analog' type, and 5 on the 'Initial Value' column header.

	Name	Variable Type	Initial Value	Retentive Value
1	Variable0	Analog	0	

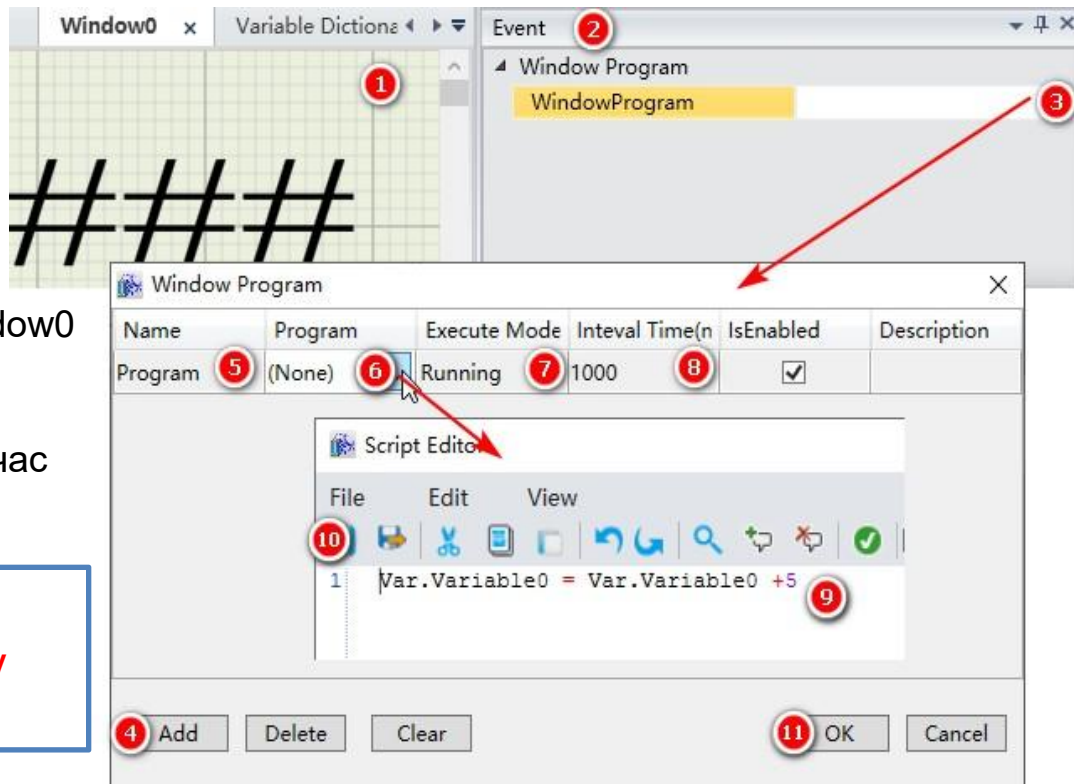
※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

(2) Створіть Text0 у Window0 . Анімація відображення аналогового значення Text0
асоційована змінна0



※ Див. розділ «9.9 Анімація відображення значень» у посібнику користувача.

(3) Створіть віконну програму у Window0 (див. розділ «10.9 Window програмна подія» в посібнику користувача)



- ① Натисніть будь-яке порожнє місце у Window0
- ② Відкрити вікно події
- ⑤ Назвіть віконну програму
- ⑦⑧ Встановити виконання щосекунди під час виконання

※ Див. розділ «10.9 Подія програми Window» у посібнику користувача.

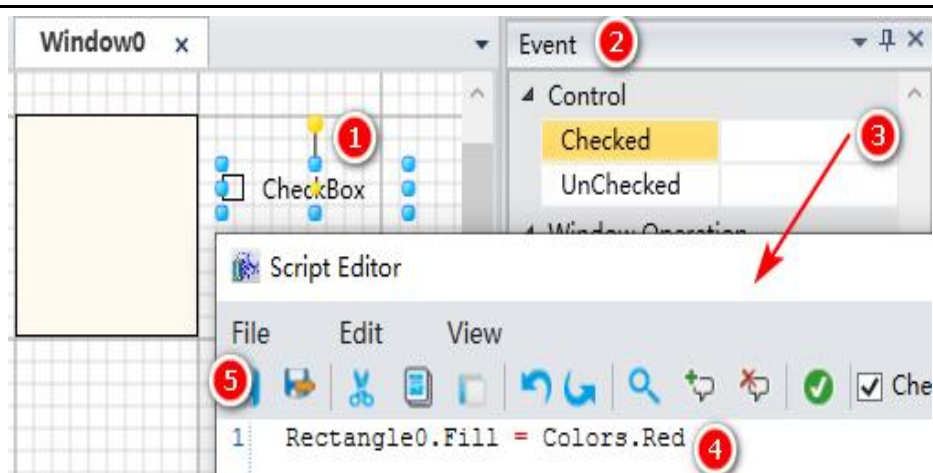
(4) Запустіть Window0 . Text0 спочатку відображає 0, а відображене значення збільшується на 5 за секунду

A large, black, serif-style number '35' displayed on a light green rectangular background.

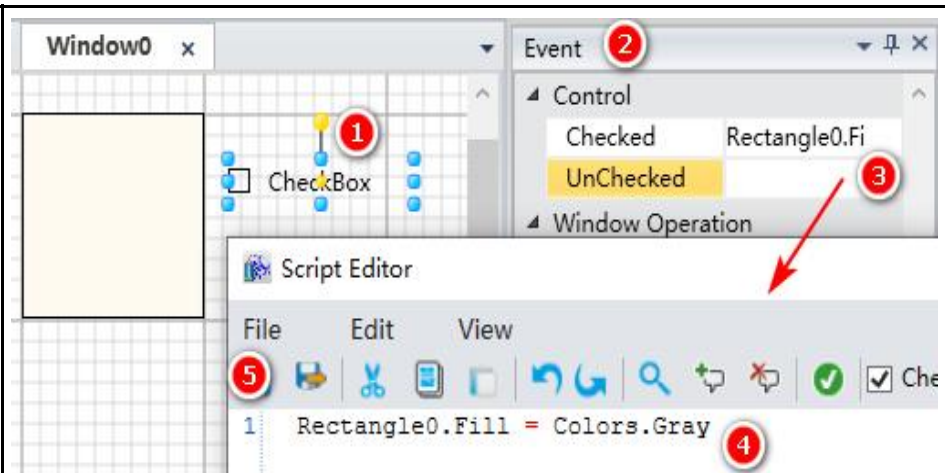
- Поняття події
- Подія лівої кнопки
- Подія правої кнопки
- Подія миші
- Подія роботи вікна
- Подія введення значення
- Подія ковзного введення
- Подія введення повороту
- Подія віконної програми
- Контрольна подія
- Клавіатура

➤ Приклад перевіреної/неперевіреної події

Установіть колір прямокутника за допомогою контрольної події прапорця:
(1) Створіть Checkbox0 і Rectangle0 у Window0. Налаштуйте подію керування Checkbox0

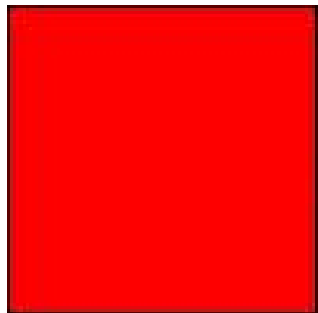


Червоний



Сірий

(2) Запустіть Window0



Червоний



Сірий

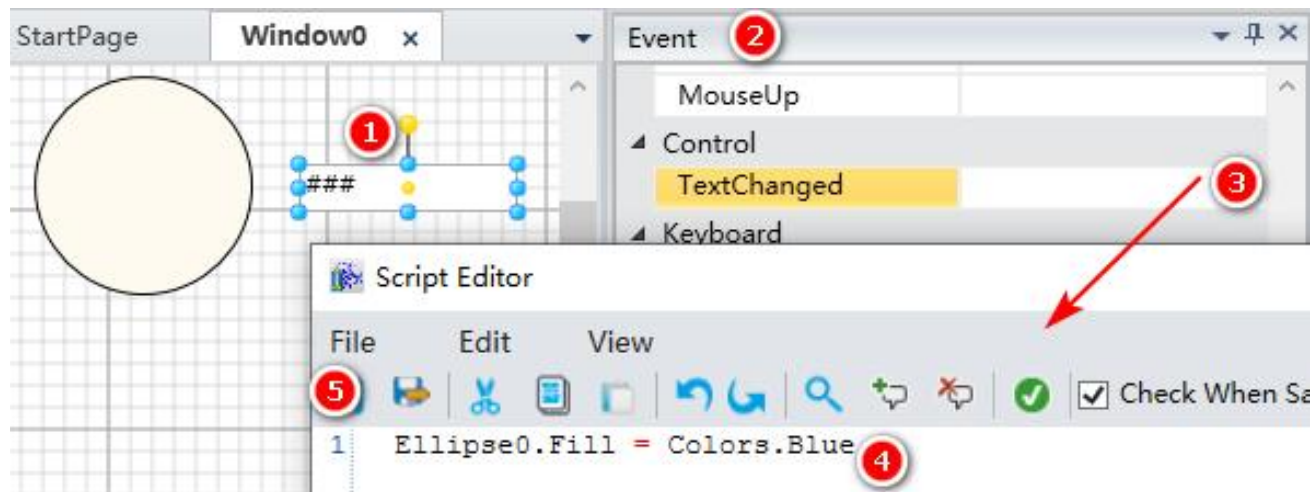
① Поставте прапорець 0, Rectangle0 стане червоним

② Зніміть прапорець 0, прямокутник 0 стане сірим

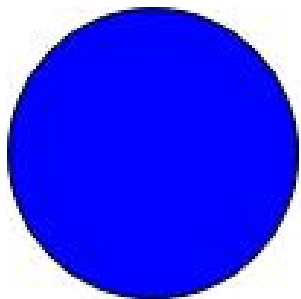
➤ Приклад події зміни тексту

Установіть колір еліпса за допомогою контрольної події текстового поля:

(1) Створіть Ellipse0 і TextBox0 у Window0 . Налаштуйте подію зміни тексту для TextBox0 .



(2) Запустіть Window0. Змініть вміст TextBox0, тоді Ellipse0 стане синім.



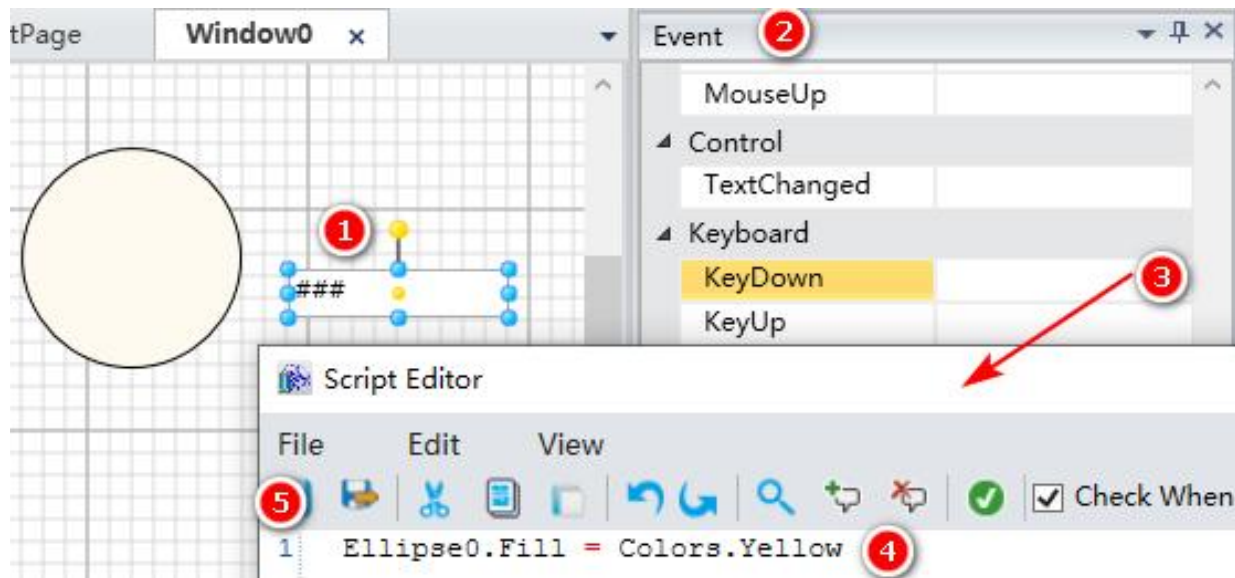
###test

- Поняття події
- Подія лівої кнопки
- Подія правої кнопки
- Подія миші
- Подія роботи вікна
- Подія введення значення
- Подія ковзного введення
- Подія введення повороту
- Подія віконної програми
- Контрольна подія
- Клавіатура

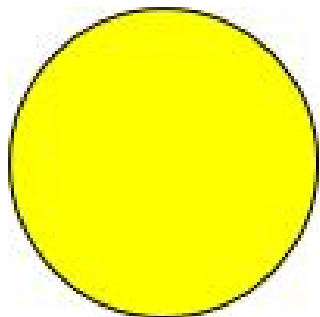
➤ Приклад події Key Down

Встановіть колір прямокутника за допомогою контрольної події текстового поля:

(1) Створіть Ellipse0 і TextBox0 у Window0. Налаштуйте подію натискання клавіші TextBox0.



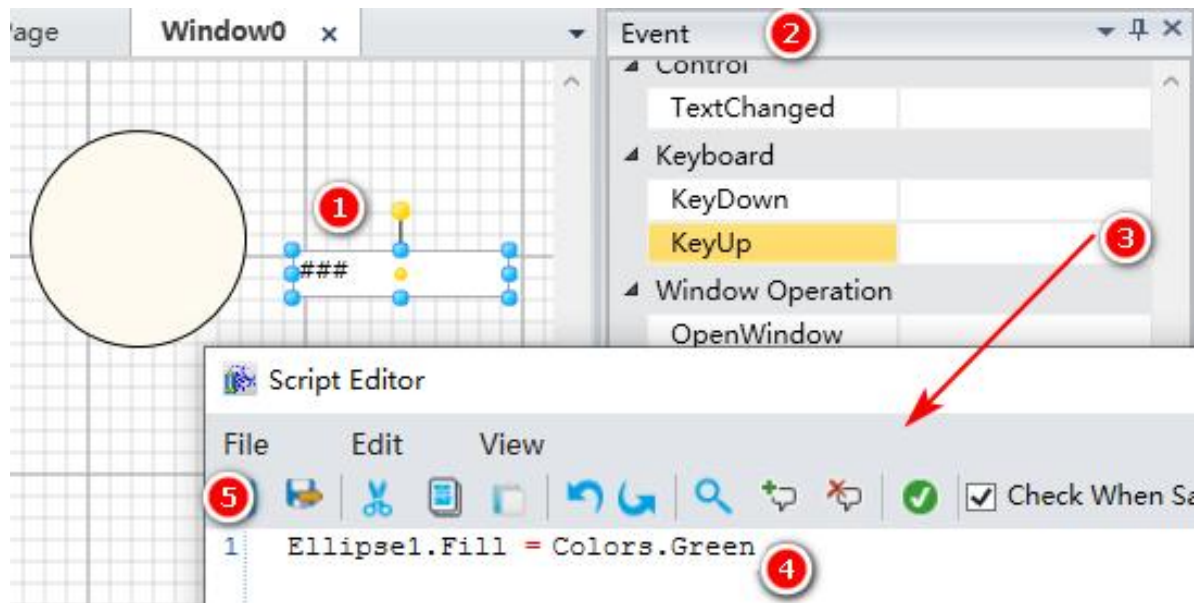
(2) Запустіть Window0. Коли фокус миші знаходиться в TextBox0, натисніть будь-яку клавішу на клавіатурі Ellipse0 стає жовтим



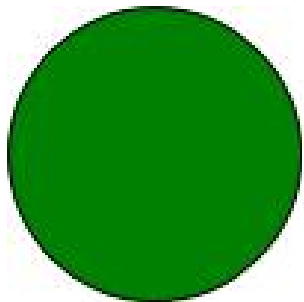
➤ Приклад події Key Up

Встановіть колір прямокутника за допомогою контрольної події текстового поля:

(1) Створіть Ellipse0 і TextBox0 у Window0 . Налаштуйте подію відкриття ключа для TextBox0 .



(2) Запустіть Window0 . Коли фокус миші знаходиться в TextBox0 , натисніть будь-яку клавішу на клавіатурі, колір Ellipse0 не зміниться. Коли клавішу відпустити, Ellipse0 стане зеленим.



Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta,
відвідайте www.deltaww.com.



DIAView

Загальні сценарії

рубін

2020/06/01



- Поняття сценаріїв
- Скрипти базової графіки
- Сценарії елементів керування вікном
- Сценарії Extend Controls
- Скрипти дій
- Сценарії вікна
- Кольорові сценарії _

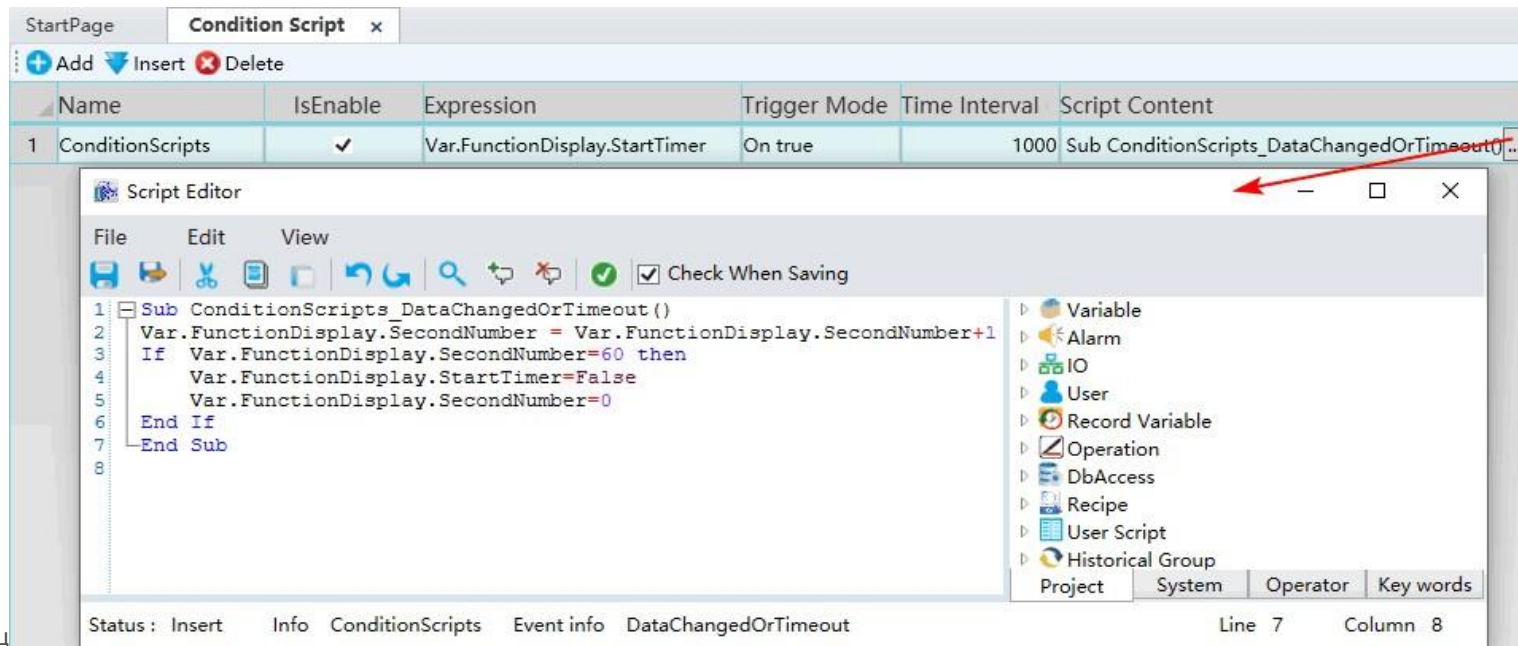
У цьому розділі ви дізнаєтеся....

- ... докладніше про типові сценарії
- ... докладніше про шість типів скриптів у DIAView

- Поняття сценаріїв
- Скрипти базової графіки
- Сценарії елементів керування вікном
- Сценарії Extend Controls
- Скрипти дій
- Сценарії вікна
- Кольорові сценарії _

Хоча функції DIAView є всеосяжними та потужними, функції, які потрібні клієнтам, відрізняються. Деякі функції потрібно налаштувати відповідно до потреб клієнта. DIAView може писати відповідні програми через редактор сценаріїв для виконання деяких спеціальних завдань і функцій.

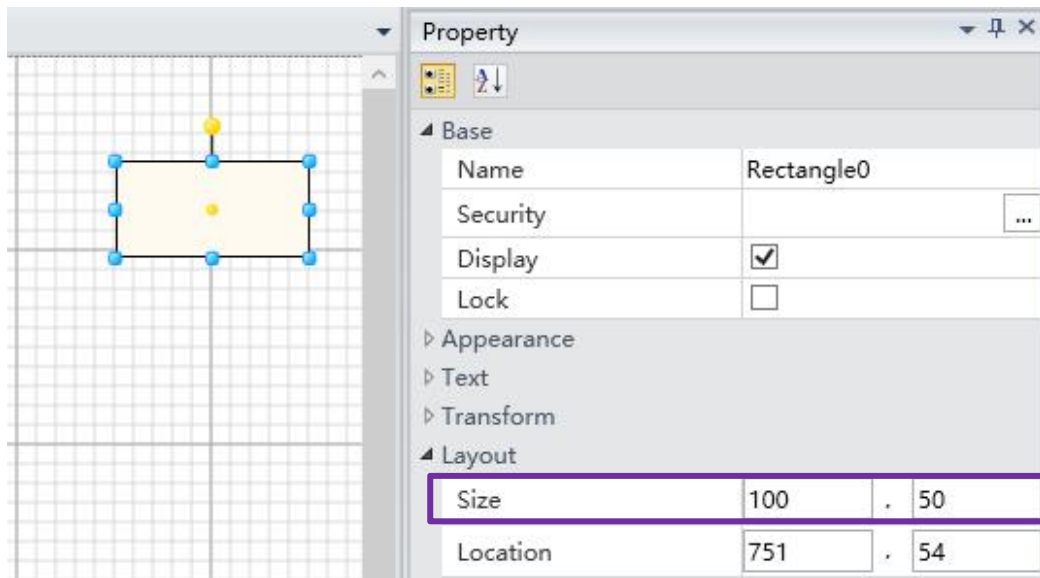
Конфігурація події, віконна програма та програма користувача в DIAView потребують використання редактора сценаріїв для написання сценаріїв. DIAView використовує мову VB Script, користувачі можуть писати програми керування логікою відповідно до специфікації граматики мови VB Script, щоб виконувати певні функції та покращувати зручність використання системи.



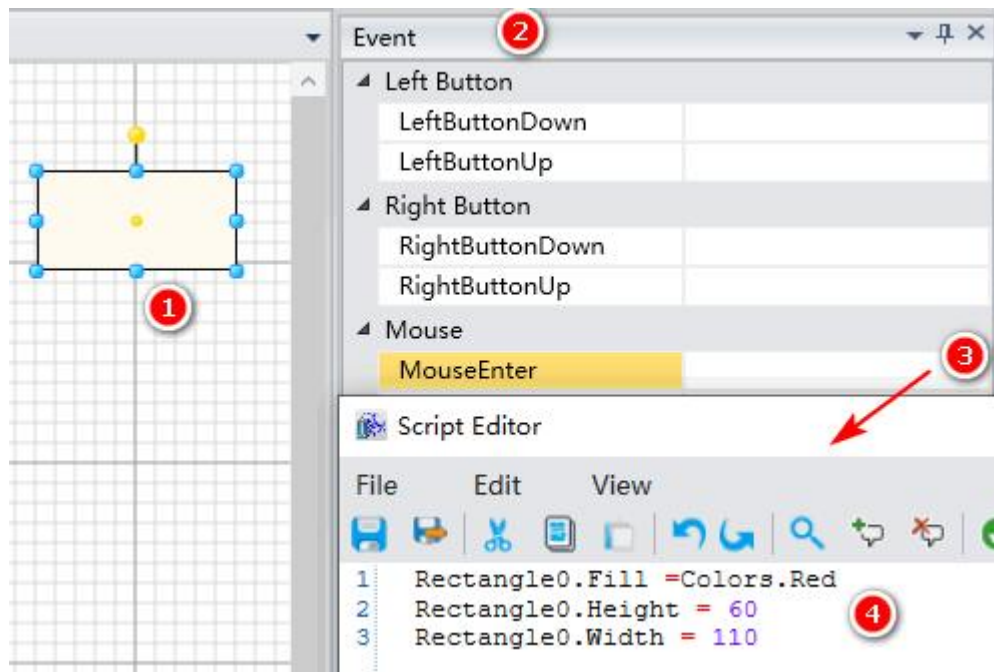
- Поняття сценаріїв
- Скрипти базової графіки
- Сценарії елементів керування вікном
- Сценарії Extend Controls
- Скрипти дій
- Сценарії вікна
- Кольорові сценарії _

➤ Заливка, висота, ширина приклад:

Створіть прямокутник і встановіть його властивості за допомогою сценаріїв (1)Створіть Rectangle0 (Розмір: 100*50) у Window0



_) Налаштуйте подію MouseEnter Rectangle0

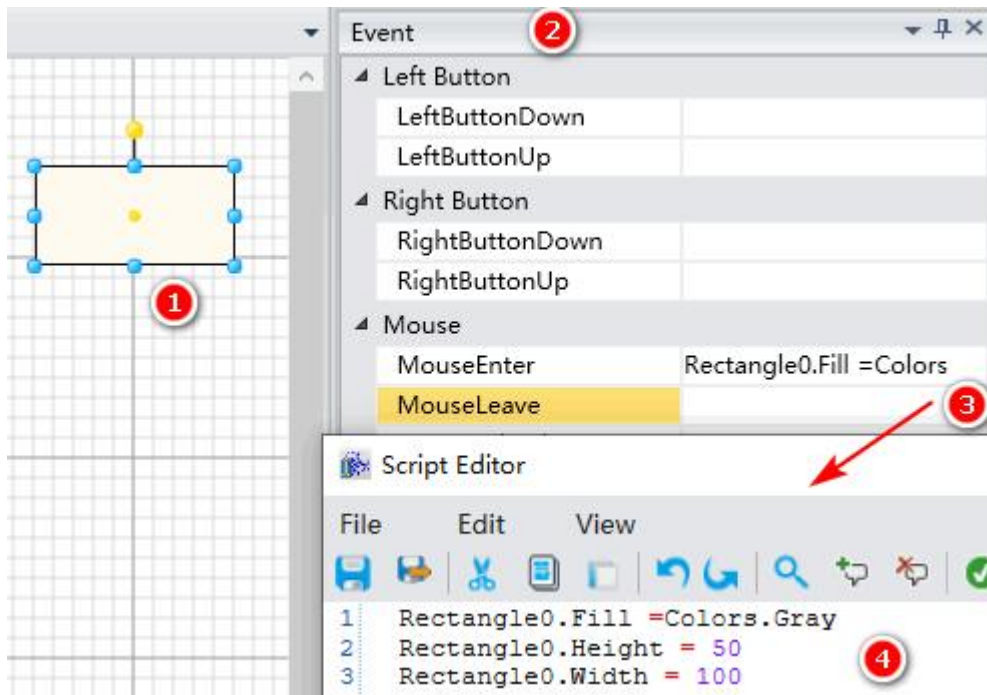


The screenshot shows the Delta IDE interface. On the left, a yellow rectangle is positioned on a grid, with a red circle labeled '1' at its bottom center. On the right, the 'Event' panel (labeled '2') is open, showing a list of events for 'Rectangle0'. The 'MouseEnter' event is selected and highlighted in yellow, with a red circle labeled '3' next to it. A red arrow points from the 'MouseEnter' event to the 'Script Editor' panel below it. The 'Script Editor' panel has a menu bar with 'File', 'Edit', and 'View'. Below the menu bar is a toolbar with various icons. The script editor contains the following code:

```
1 Rectangle0.Fill = Colors.Red
2 Rectangle0.Height = 60
3 Rectangle0.Width = 110
```

A red circle labeled '4' is next to the third line of code.

3.) Налаштуйте подію MouseLeave Rectangle0



The screenshot displays the Delta IDE interface. On the left, a yellow rectangle labeled 'Rectangle0' is shown on a grid, with a red circle '1' highlighting its bottom-right corner. On the right, the 'Event' window (labeled with a red circle '2') lists various events. The 'Mouse' section is expanded, and the 'MouseLeave' event is highlighted with a yellow background and a red circle '3'. Below the Event window, the 'Script Editor' window is open, showing a menu bar (File, Edit, View) and a toolbar. The script contains three lines of code, with a red arrow pointing from the 'MouseLeave' event to the first line. A red circle '4' highlights the end of the script.

```
1 Rectangle0.Fill =Colors.Gray
2 Rectangle0.Height = 50
3 Rectangle0.Width = 100
```

(4) Запустіть поточний проект.



1



2

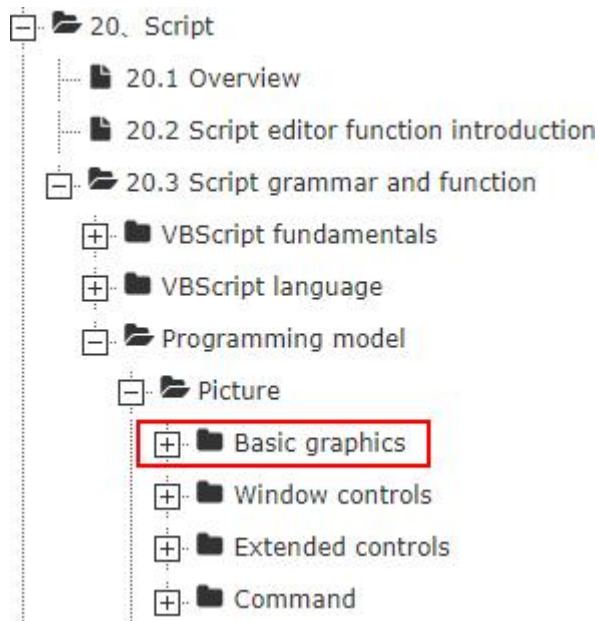
Червоний, 110*60

Сірий, 100*50

- ① Коли миша входить у Rectangle0 , Rectangle0 стає червоним, а розмір стає 110*60
- ② Коли миша залишає Rectangle0 , Rectangle0 стає сірим, а його розмір стає 100*50

※ Для отримання додаткової інформації зверніться до розділу «20.3 Граматика та функції сценарію» в посібнику користувача.

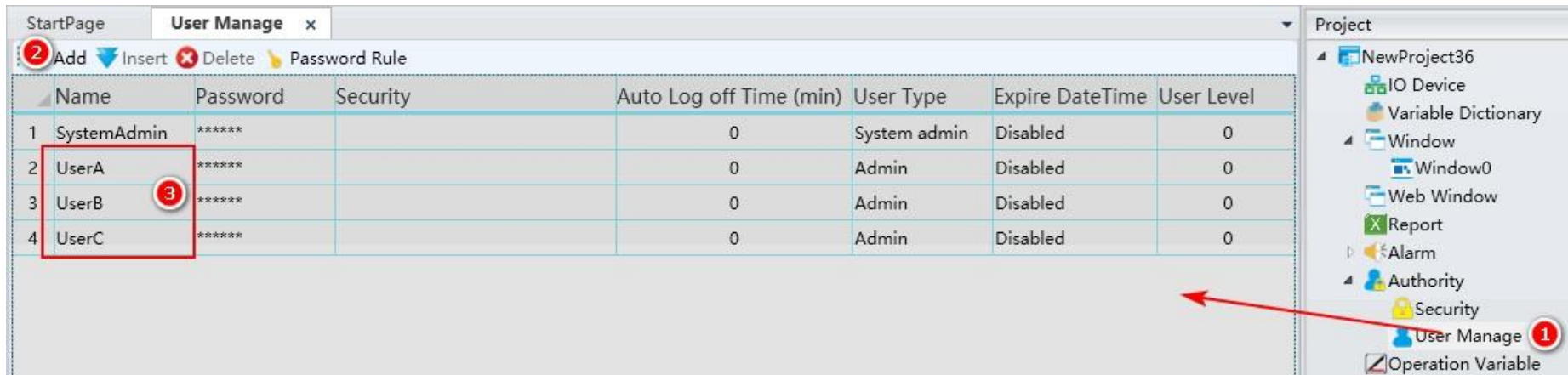
Щоб отримати докладніші відомості про використання сценаріїв базової графіки, зверніться до розділу « 20.3 Граматика та функції сценаріїв » посібника користувача. Як показано на малюнку нижче:



- Поняття сценаріїв
- Скрипти базової графіки
- **Сценарії елементів керування вікном**
- Сценарії Extend Controls
- Скрипти дій
- Сценарії вікна
- Кольорові сценарії _

➤ AddItems приклад1:

Отримайте імена користувачів у поточному проєкті за допомогою сценарію AddItems (1) Створіть 3 користувачів: UserA , UserB , UserC

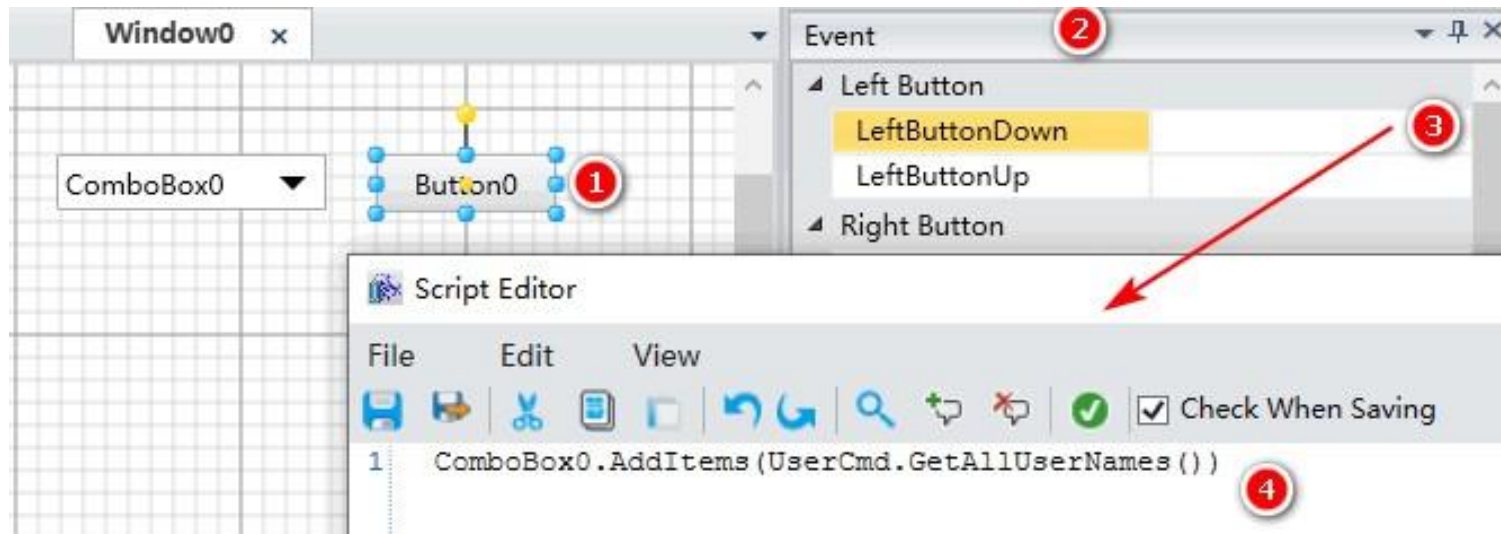


The screenshot displays the 'User Manage' window. At the top, there are tabs for 'StartPage' and 'User Manage'. Below the tabs is a toolbar with buttons for 'Add' (circled with a red '2'), 'Insert', 'Delete', and 'Password Rule'. The main area contains a table with the following columns: Name, Password, Security, Auto Log off Time (min), User Type, Expire DateTime, and User Level. The table lists four users: SystemAdmin, UserA, UserB, and UserC. The 'UserA', 'UserB', and 'UserC' rows are highlighted with a red box, and a red circle with the number '3' is placed over the 'UserB' row. On the right side, there is a 'Project' tree view showing a hierarchy of items: NewProject36, IO Device, Variable Dictionary, Window, Window0, Web Window, Report, Alarm, Authority, Security, User Manage (circled with a red '1'), and Operation Variable. A red arrow points from the 'User Manage' item in the tree to the 'User Manage' window.

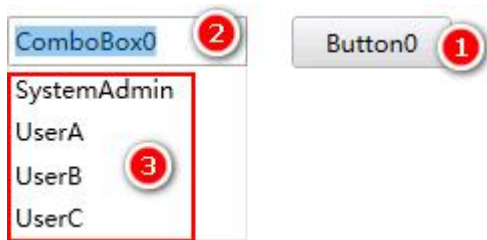
	Name	Password	Security	Auto Log off Time (min)	User Type	Expire DateTime	User Level
1	SystemAdmin	*****		0	System admin	Disabled	0
2	UserA	*****		0	Admin	Disabled	0
3	UserB	*****		0	Admin	Disabled	0
4	UserC	*****		0	Admin	Disabled	0

※Зверніться до розділу «12.3 Користувач» у посібнику користувача.

(2) Створіть ComboBox0 і Button0 у Window0 , налаштуйте Подія LeftButtonDown Button0 .



(3) Запустіть поточний проект.

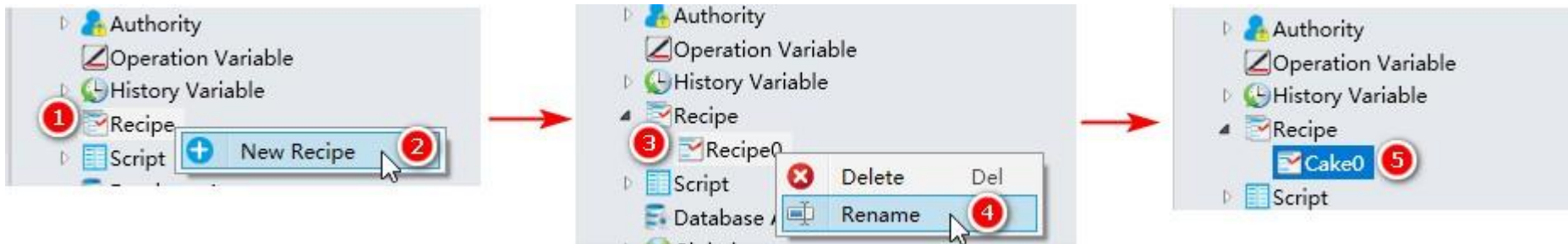


① Натисніть кнопку Button0

②③ ComboBox0 відображає імена всіх користувачів у поточному проекті—
SystemAdmin , UserA , UserB , UserC

➤ AddItems приклад2:

Отримайте інформацію про рецепт за допомогою сценарію AddItems (1)Створіть 1 рецепт: Cake0

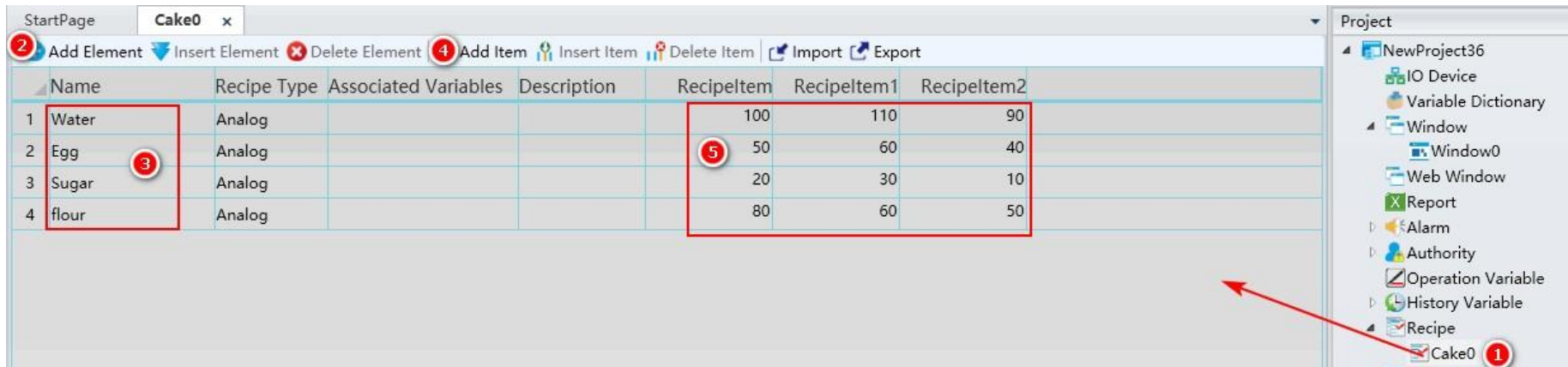


① ③ Клацніть правою кнопкою миші

② ④ Натисніть

⑤ Перейменувати на «Cake0»

(2) Налаштуйте параметри для cake0



StartPage Cake0 x

2 Add Element Insert Element Delete Element 4 Add Item Insert Item Delete Item Import Export

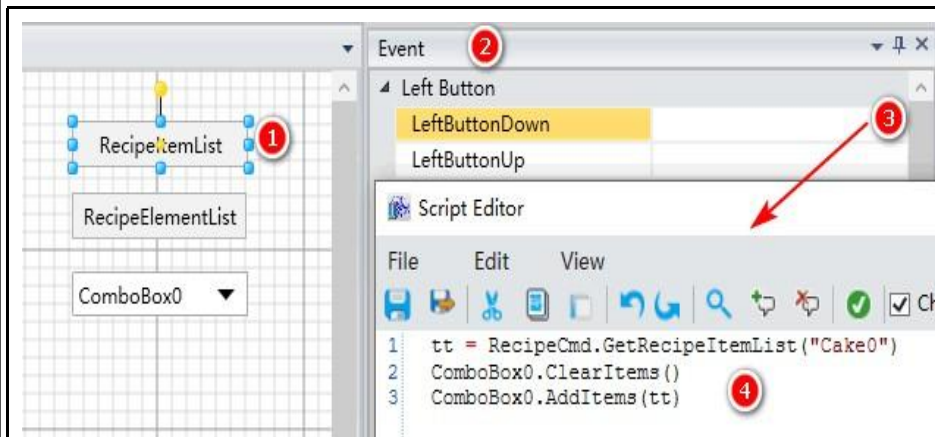
	Name	Recipe Type	Associated Variables	Description	Recipeltem	Recipeltem1	Recipeltem2
1	Water	Analog			100	110	90
2	Egg	Analog			50	60	40
3	Sugar	Analog			20	30	10
4	flour	Analog			80	60	50

Project

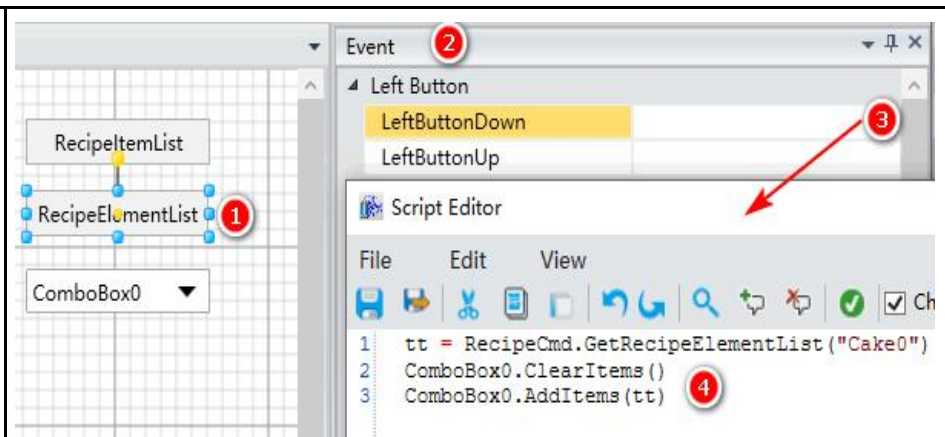
- NewProject36
 - IO Device
 - Variable Dictionary
 - Window
 - Window0
 - Web Window
 - Report
 - Alarm
 - Authority
 - Operation Variable
 - History Variable
 - Recipe
 - Cake0 1

※Див. розділ «15.2 Конфігурація рецепта» в посібнику користувача.

(3) Створіть ComboBox0 і дві кнопки (RecipeItemList , RecipeElementList) у Window0 , налаштуйте подію LeftButtonDown для двох кнопок.

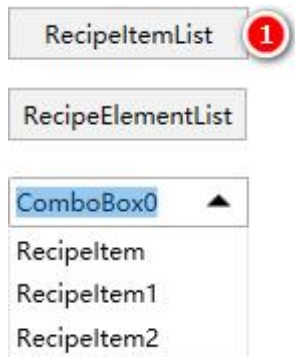


RecipeItemList



RecipeElementList

(4) Запустіть поточний проект.



RecipeItemList



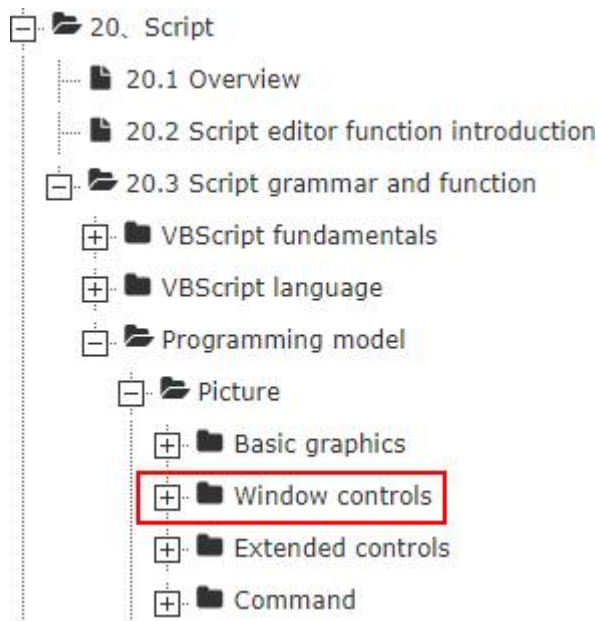
RecipeElementList

① Натисніть кнопку «RecipeItemList», у ComboBox0 відобразяться елементи рецепту Cake0—RecipeItem, RecipeItem1, RecipeItem2

② Натисніть кнопку «RecipeElementList», у ComboBox0 відобразяться елементи рецепту Cake0—Вода, яйце, цукор, борошно

Скрипти елементів керування вікном

Щоб отримати докладніші відомості про використання сценаріїв елементів керування вікнами, зверніться до розділу « 20.3 Граматика та функції сценаріїв » посібника користувача. Як показано на малюнку нижче:

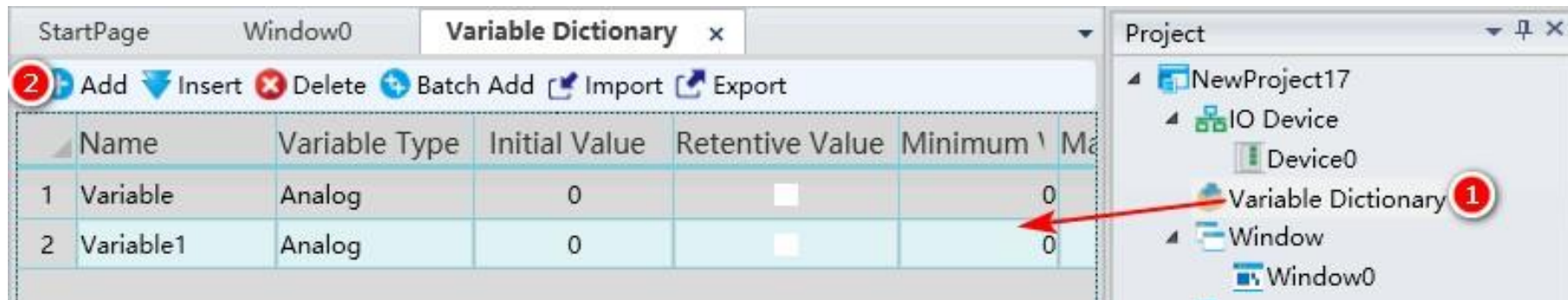


- Поняття сценаріїв
- Скрипти базової графіки
- Сценарії елементів керування вікном
- **Сценарії Extend Controls**
- Скрипти дій
- Сценарії вікна
- Кольорові сценарії _

➤ QueryHistoryDate приклад :

Запитувати дані історії з моменту запуску проекту до поточного часу (1)

Створити 2 змінні : змінна , змінна 1



The screenshot shows the Delta software interface. The 'Variable Dictionary' window is open, displaying a table with two variables. The 'Project' tree on the right shows the hierarchy of the project, with 'Variable Dictionary' highlighted under 'NewProject17'.

	Name	Variable Type	Initial Value	Retentive Value	Minimum \	Ma
1	Variable	Analog	0	☐	0	
2	Variable1	Analog	0	☐	0	

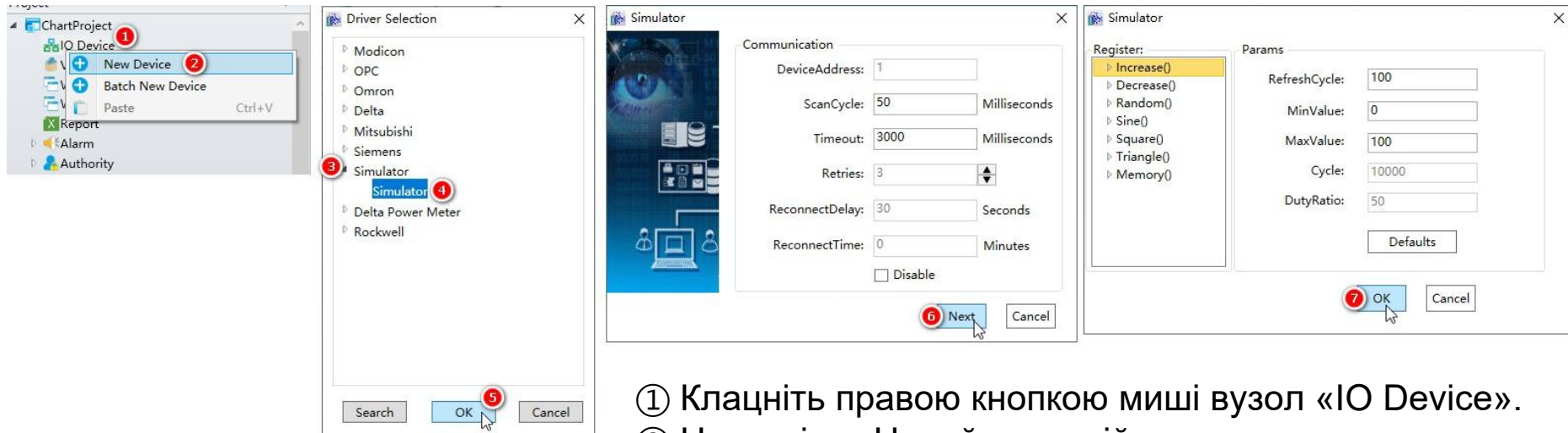
The 'Project' tree on the right shows the following structure:

- NewProject17
 - IO Device
 - Device0
 - Variable Dictionary (1)
 - Window
 - Window0

A red arrow points from the 'Variable Dictionary' node in the Project tree to the 'Variable Dictionary' window.

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

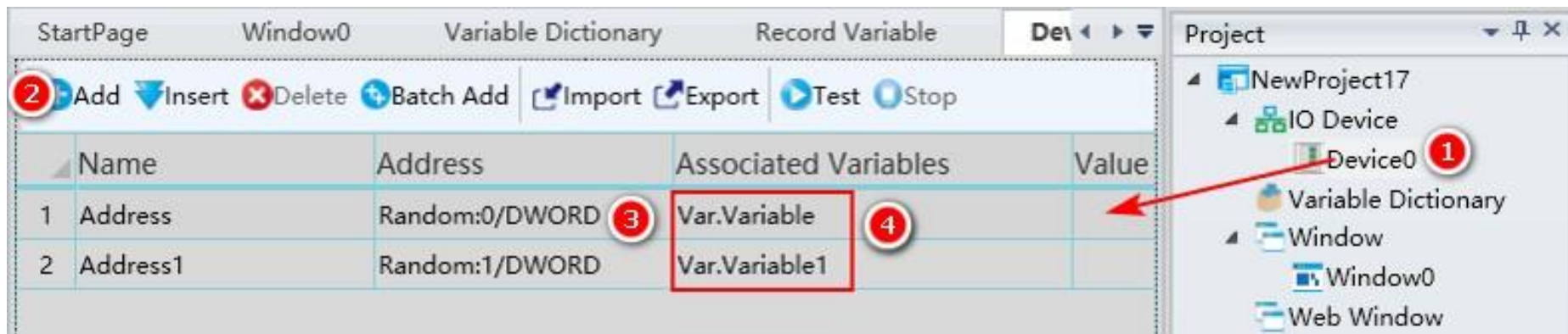
(2) Створіть імітований пристрій : Device0



- ① Клацніть правою кнопкою миші вузол «IO Device».
- ② Натисніть «Новий пристрій»
- ③④ Двічі клацніть «Симулятор»

※Див. розділ «5.10.1 Симулятор» у посібнику користувача.

(3) Створіть дві адреси моделювання в Device0, пов'язані зі змінною, змінною1 відповідно



The screenshot displays the Delta software interface. The 'Variable Dictionary' tab is active, showing a table with the following data:

	Name	Address	Associated Variables	Value
1	Address	Random:0/DWORD	Var.Variable	
2	Address1	Random:1/DWORD	Var.Variable1	

The 'Associated Variables' column for both rows is highlighted with a red box. The 'Project' tree on the right shows the hierarchy: NewProject17 > IO Device > Device0 (circled with a red 1). A red arrow points from the 'Device0' node to the 'Associated Variables' column of the table.

(4) Створіть дві історичні змінні в Record Variable, які пов'язані зі змінною, Variable1 відповідно

2 Add Insert Delete Import Export

Name	Associated Variables	Mode	Timer	Deadband	D
1 RecordVariable	Var.Variable 3	Timing	s1 4	N/A	
2 RecordVariable1	Var.Variable1	Timing	s1	N/A	

Timer Browser

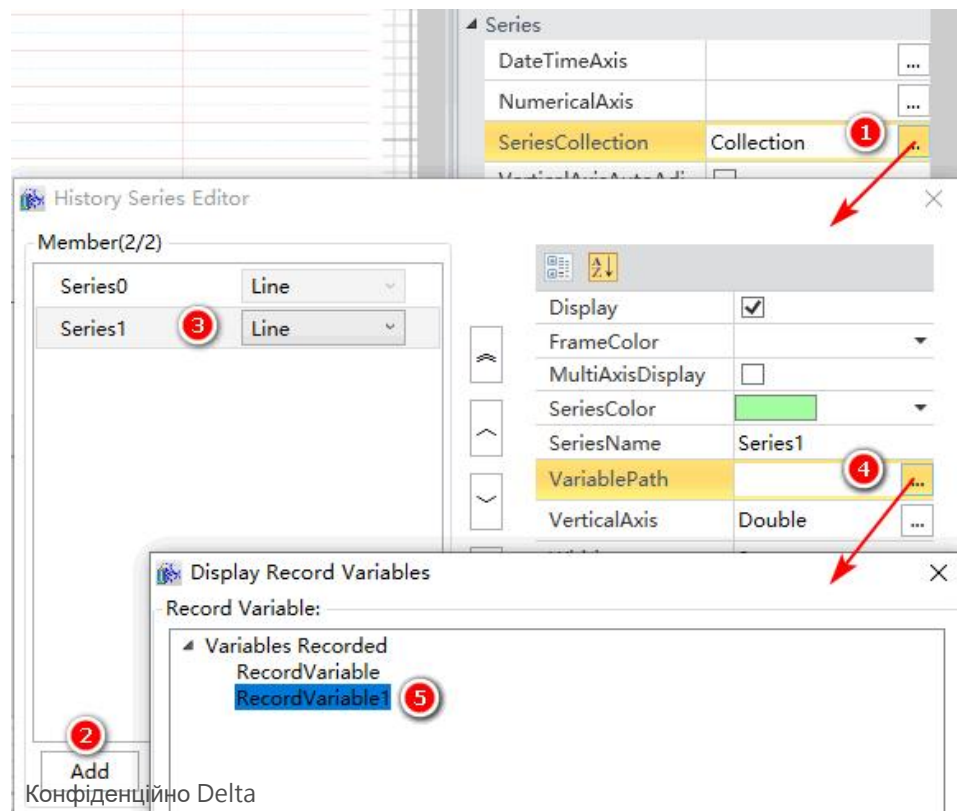
	IsEnable	Name	Timer Unit	Timer Coefficient	Relative Time	Description
1 5	☒	s1	Second	1	0001-01-01 00:00:00	Triggers every second
2	☐	s10	Second	10	0001-01-01 00:00:00	Triggers every 10 seconds
3	☐	s30	Second	30	0001-01-01 00:00:00	Triggers every 30 seconds
4	☐	m1	Minute	1	0001-01-01 00:00:00	Triggers every minute
5	☐	m30	Minute	30	0001-01-01 00:00:00	Triggers every 30 minutes

ChartProject

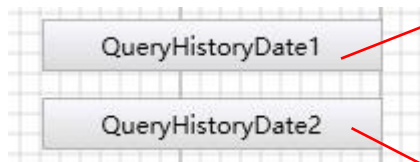
- IO Device
 - Device0
- Variable Dictionary
- Window
 - Window0
 - Window1
- Web Window
- Report
- Alarm
- Authority
- Operation Variable
- History Variable 1
 - Record Variable
 - Historical Group

※Зверніться до розділу «14.2 Налаштування змінної запису історії» в посібнику користувача.

(5) Створіть HistoryChart0 у Window0 і додайте 2 криві в HistoryChart0, Series0 пов'язану RecordVariable, Series1 пов'язану RecordVariable1



(6) Створіть дві кнопки у Window0 , налаштуйте подію LeftButtonDown для дві кнопки



Script Editor

File Edit View

Icons: Save, Undo, Cut, Copy, Paste, Find, Add, Remove, Check, Check When Saving

```
1 Call HistoryChart0.QueryStartTime (Sys.StartTime)
2 'Set the query start time to the moment when the project starts
3 Call HistoryChart0.QueryEndTime (Sys.Now)
4 'Set the query end time to the system current time
5 Call HistoryChart0.SetVarRecordRulerName ("s1")
6 Call HistoryChart0.QueryHistoryDate ()
```

Script Editor

File Edit View

Icons: Save, Undo, Cut, Copy, Paste, Find, Add, Remove, Check, Check When Saving

```
1 Call HistoryChart0.SetVarRecordRulerName ("s1")
2 Call HistoryChart0.QueryHistoryDate (Sys.StartTime, Sys.Now, 1, 0)
```

(7) Запустіть Window0



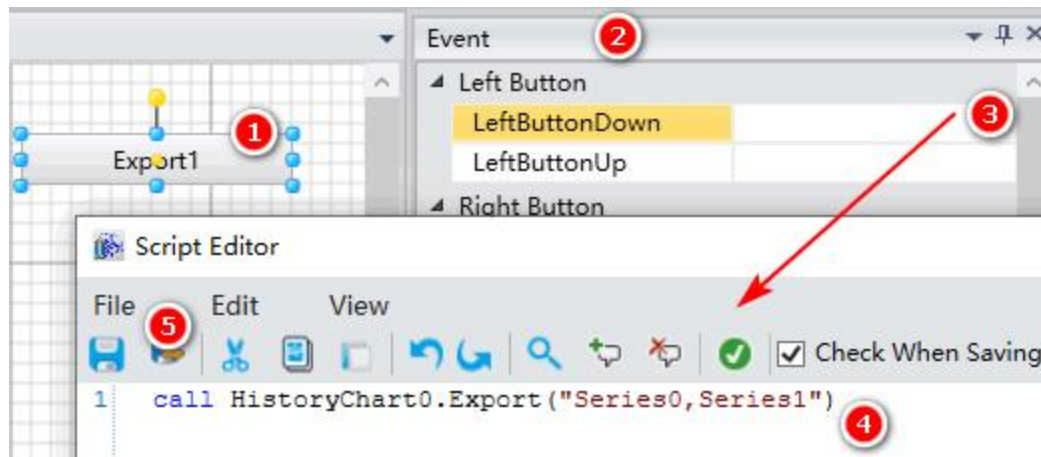
Натисніть кнопку « QueryHistoryDate1 » або « QueryHistoryDate2 » кнопку , HistoryChart0 відображає дані історії з моменту запуску проекту до поточного часу

➤ Експорт приклад1 :

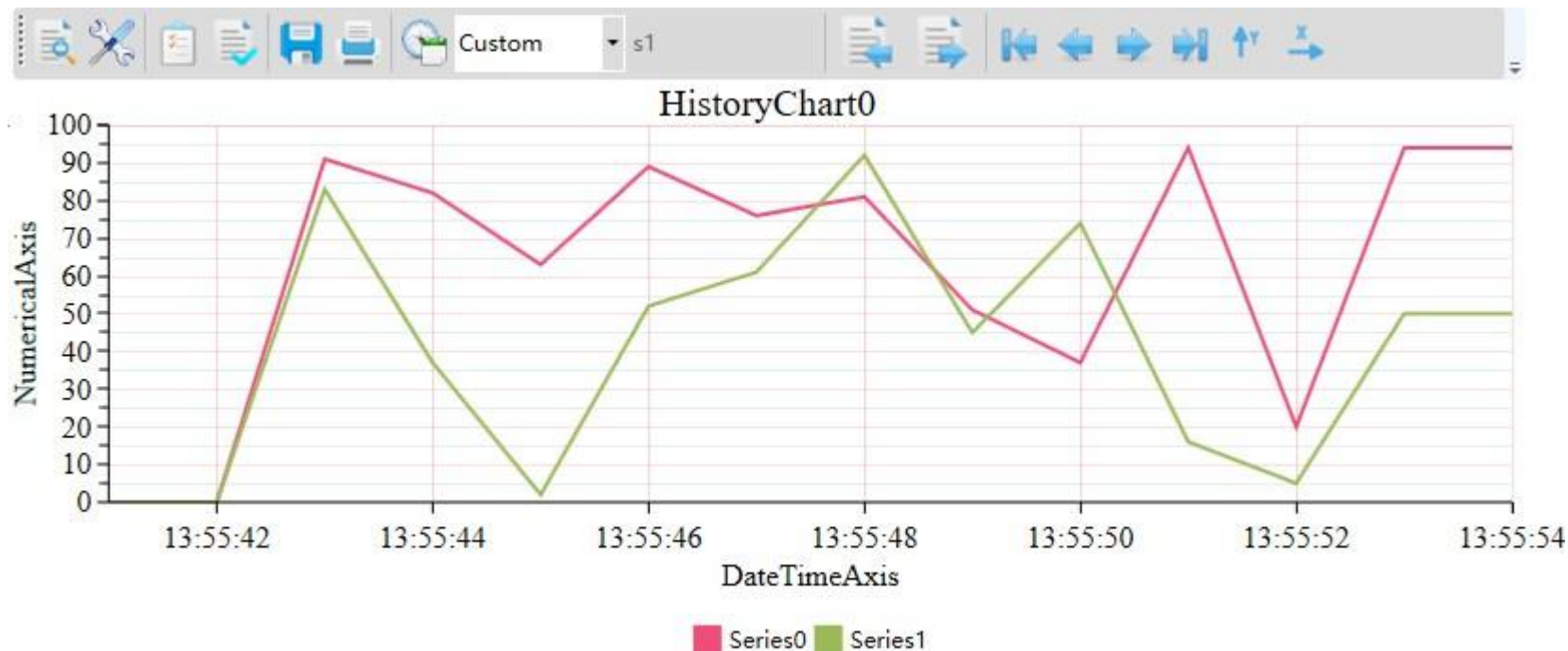
Експорт даних історії за певний період часу

Перші 5 кроків такі самі, як (1)(2)(3)(4)(5) кроки прикладу QueryHistoryDate (6)

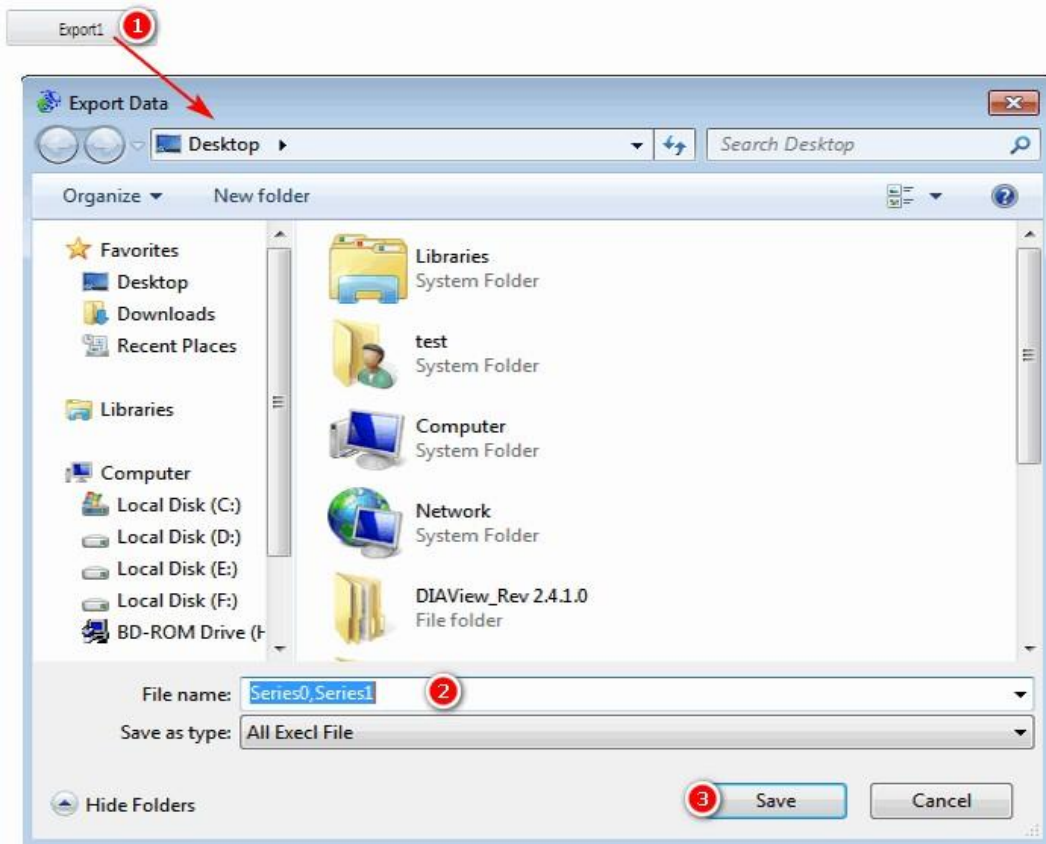
Створіть одну кнопку (Export1) у Window0 , налаштуйте подію LeftButtonDown для кнопки



(7) Запустіть Window0, дані історії запитів



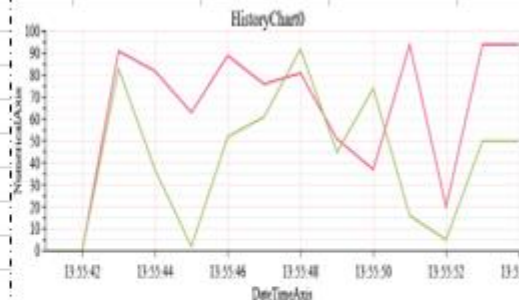
(8) Експортуйте дані історії, запитувані на попередньому кроці



Натисніть кнопку «Export1», після чого Series0 і Series1 HistoryChart0 будуть успішно експортовані на робочий стіл, а ім'я файлу Excel буде «Серія0, Серія1»

(9) Експортований файл Excel показано нижче

	A	B	C	D	E	F	G	H
1	TriggerTime	/varRecord.RecordVariable	/varRecord.RecordVariable1					
2	2020-06-15 13:55:41	0	0					
3	2020-06-15 13:55:42	0	0					
4	2020-06-15 13:55:43	91	83					
5	2020-06-15 13:55:44	82	37					
6	2020-06-15 13:55:45	63	2					
7	2020-06-15 13:55:46	89	52					
8	2020-06-15 13:55:47	76	61					
9	2020-06-15 13:55:48	81	92					
10	2020-06-15 13:55:49	51	45					
11	2020-06-15 13:55:50	37	74					
12	2020-06-15 13:55:51	94	16					
13	2020-06-15 13:55:52	20	5					
14	2020-06-15 13:55:53	94	50					
15	2020-06-15 13:55:54	94	50					
16								





➤ Експорт приклад 2 :

Експорт даних історії за певний період за шаблоном звіту

Перші 5 кроків такі самі, як (1)(2)(3)(4)(5) кроки **QueryHistoryDate** приклад (6) Створіть шаблон звіту (Report0) і налаштуйте для нього дані історії

The screenshot displays the Delta software interface. The top menu bar includes options such as 'Wrap Text', 'Merge and Center', 'Insert', 'Delete', 'Format', 'Realtime Variable', 'History Data', 'Data Source', 'Report Guide', 'Print Setting', and 'PDF Setting'. The main workspace shows a 'History Variables' dialog box with the following fields: 'DataPoint' (with a red circle 5), 'BaseLineTime', 'Year' (set to 'Year'), 'Month' (set to 'Month'), 'Day' (set to 'Day'), 'Hour' (set to 'Hour'), 'Minute' (set to 'Minute'), 'Second' (set to 'Second'), and 'Category' (set to 'TriggerTime' with a red circle 4). The 'OK' button is highlighted with a red circle 9. To the right, the 'Report0' project window is visible, showing a tree view of the project structure. The 'Report0' item is highlighted with a red circle 1. The 'History Variables' dialog box also shows 'HistoryVariables' and 'Historical Group' with a red circle 6, and 'RecordVariable' and 'RecordVariable1' with a red circle 7. The 'OK' button at the bottom right of the dialog box is highlighted with a red circle 8.

(7) Результат конфігурації Report0 такий

StartPage	Window0	Report0	x										
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	#GetHisData	#GetHisDa	#GetHisData("VarRecord.RecordVariable1,Year,Month,Day,Hour,Minute,Second,Value")										
2													
3													
4													
5													
6													

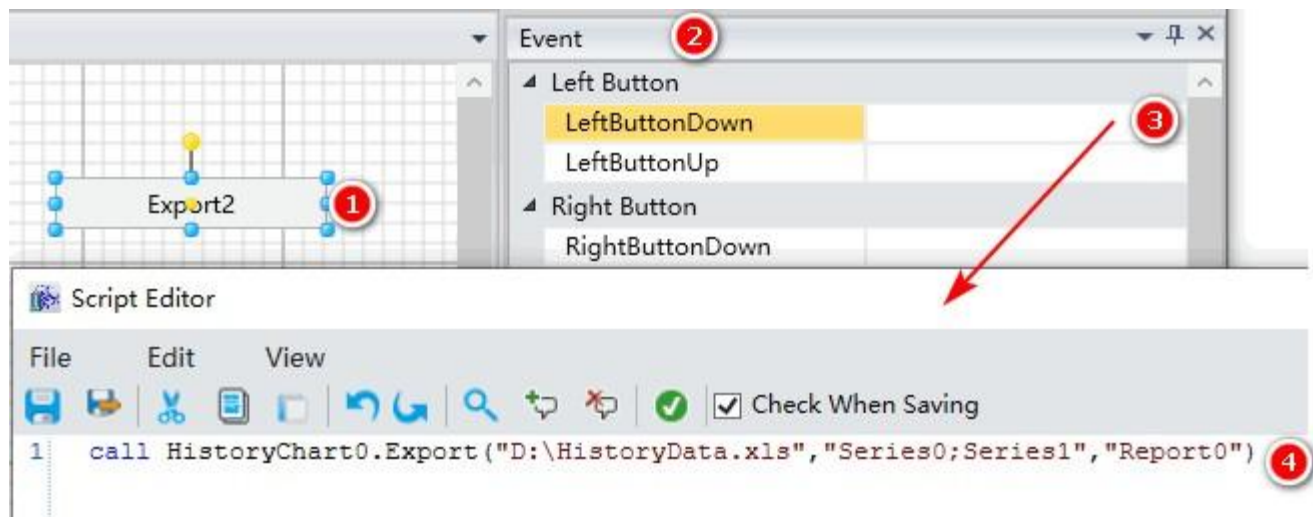
#GetHisData("VarRecord.RecordVariable1,Year,Month,Day,Hour,Minute,Second,Value")

#GetHisData("VarRecord.RecordVariable,Year,Month,Day,Hour,Minute,Second,Value")

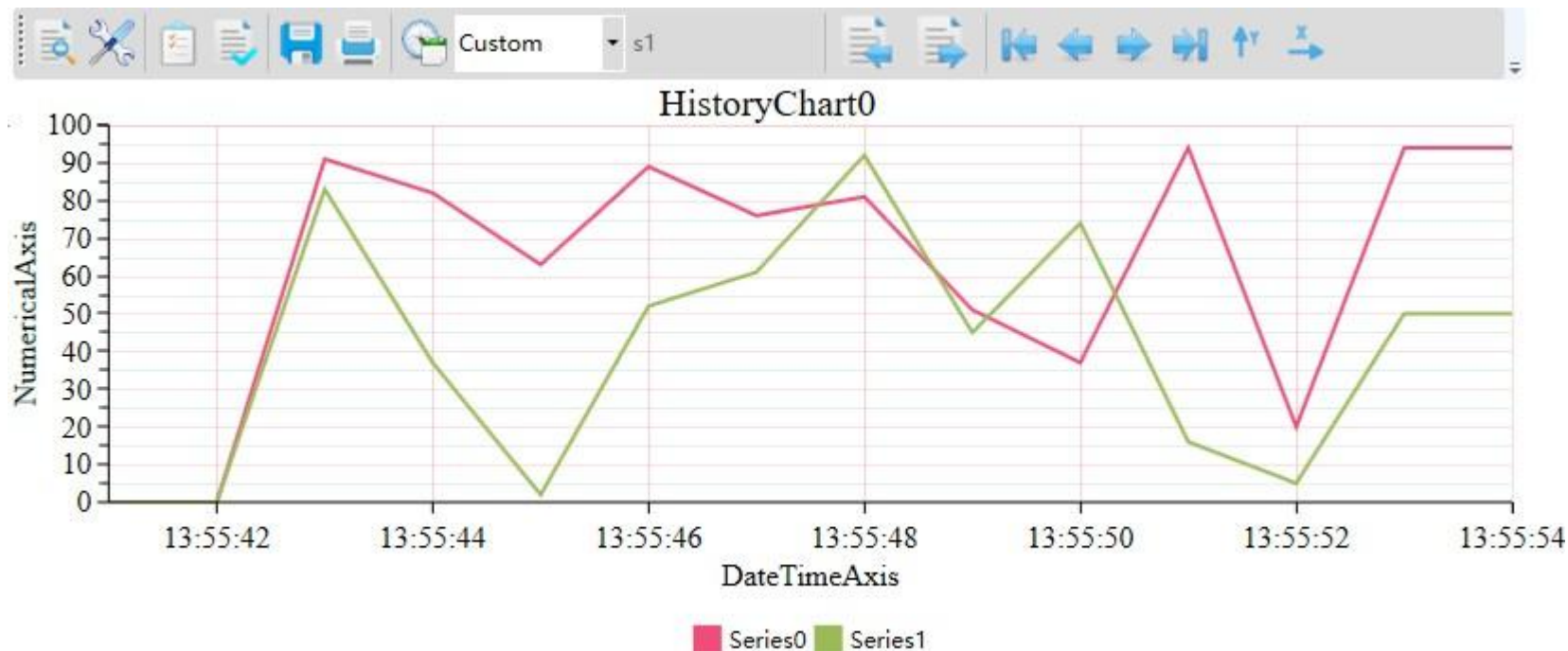
#GetHisData("VarRecord.RecordVariable,Year,Month,Day,Hour,Minute,Second,TriggerTime")

(8) Налаштуйте дві області для відображення зображень для Report0

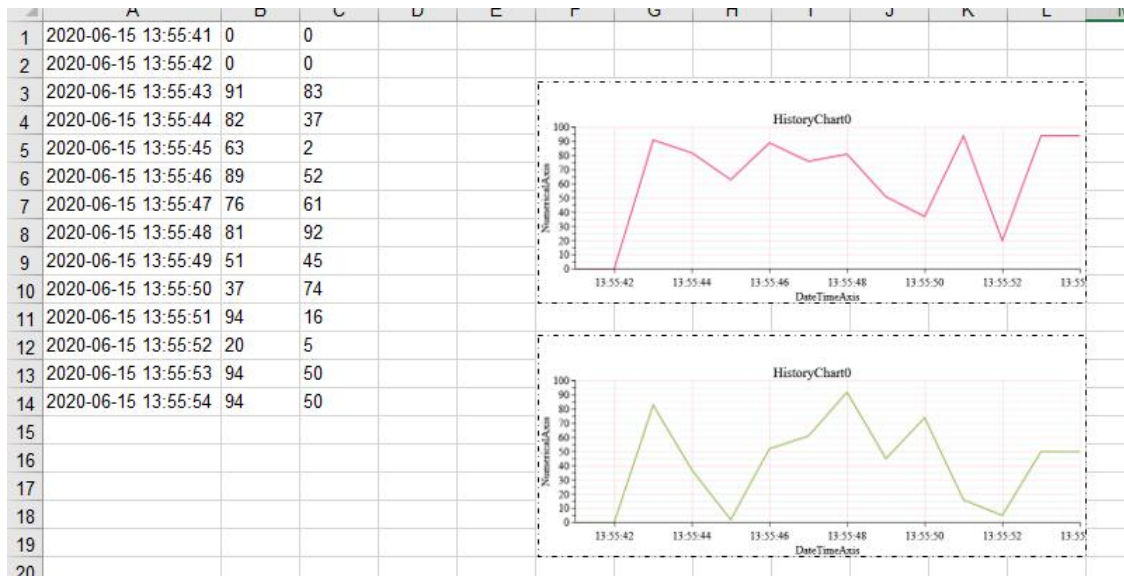
(9) Створіть одну кнопку (Export2) у Window0 , налаштуйте подію LeftButtonDown кнопки



(10) Запустіть Window0, дані історії запитів



(11) Експортуйте дані історії за шаблоном звіту, експортований файл Excel показано нижче



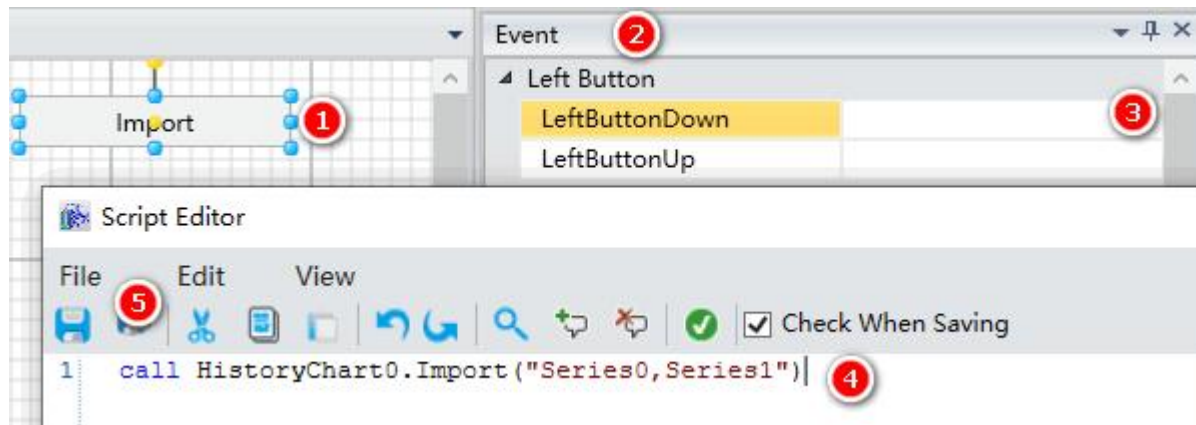
Натисніть кнопку «Export2», після чого Series0 і Series1 HistoryChart0 будуть успішно експортовані на диск D, а ім'я файлу буде « HistoryData » .

➤ Імпорт приклад :

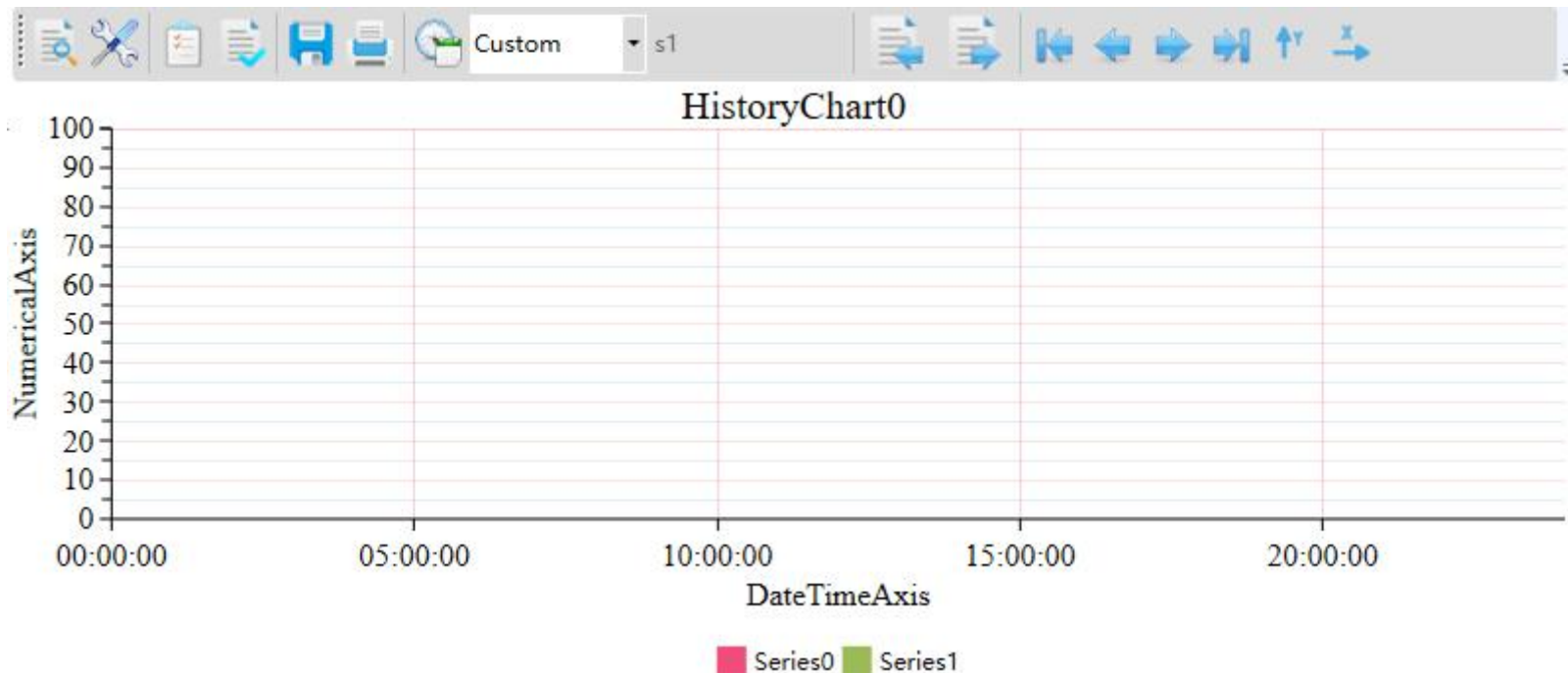
Імпортуйте дані історії з файлу Excel в діаграму історії

Перші 5 кроків такі самі, як (1)(2)(3)(4)(5) кроки прикладу QueryHistoryDate

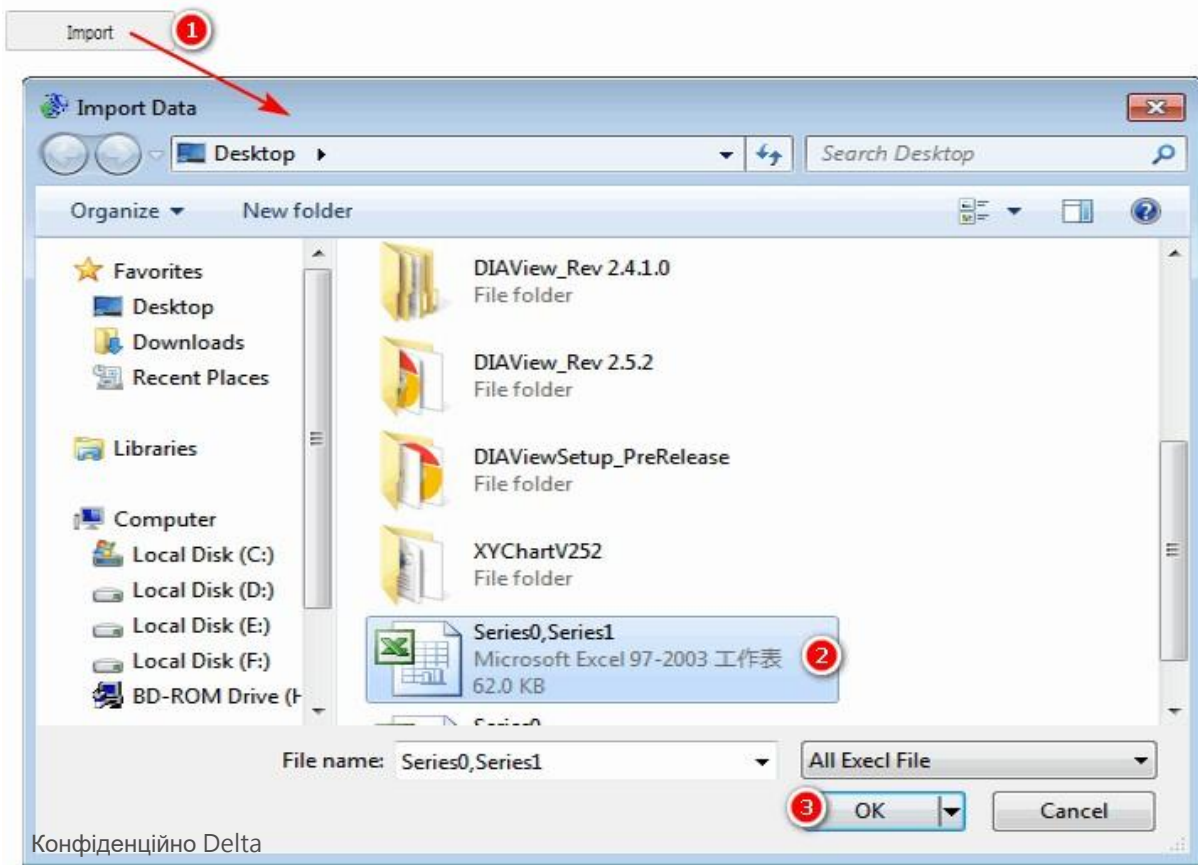
(6) Створіть одну кнопку (Імпорт) у Window0 , налаштуйте подію LeftButtonDown кнопки



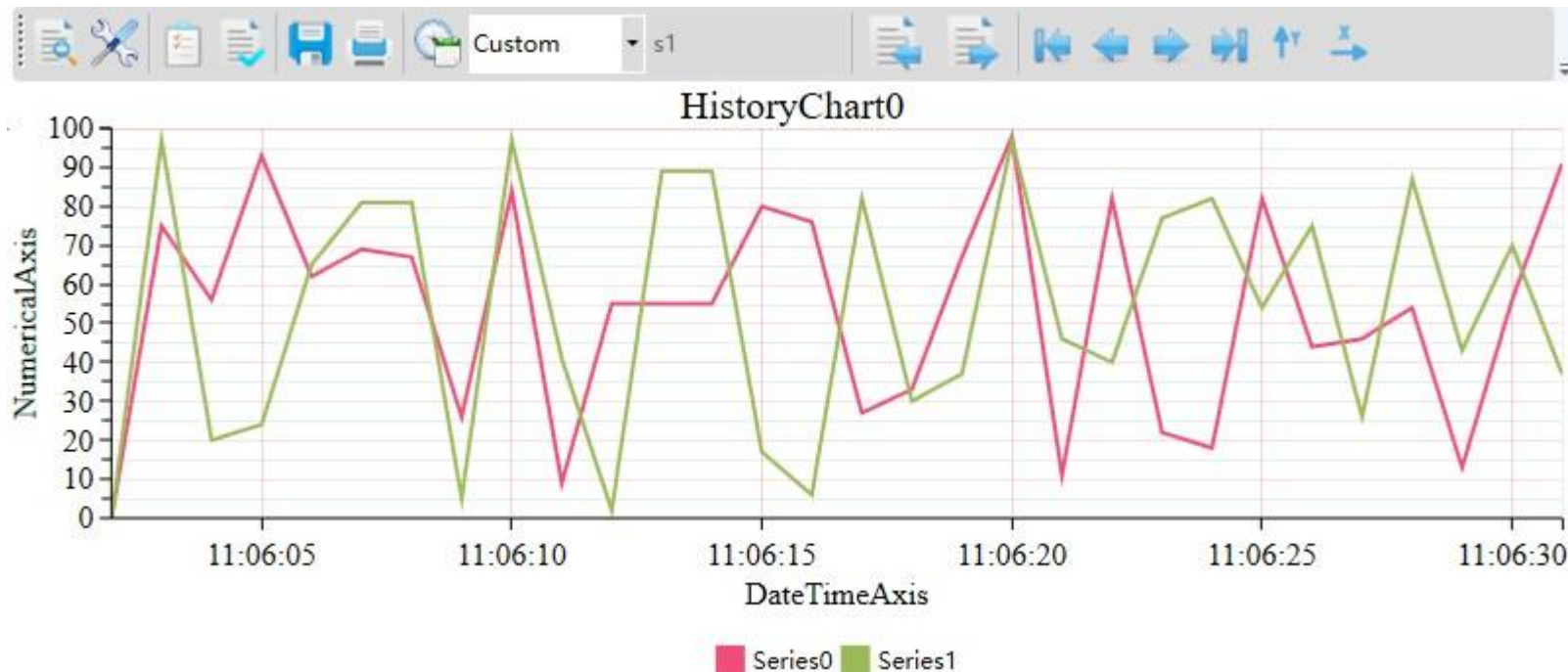
(7) Запустіть Window0



(8) Натисніть кнопку « Імпортувати », щоб імпортувати дані історії в HistoryChart0



(9) Дані історії успішно імпортовано до HistoryChart0, як показано нижче

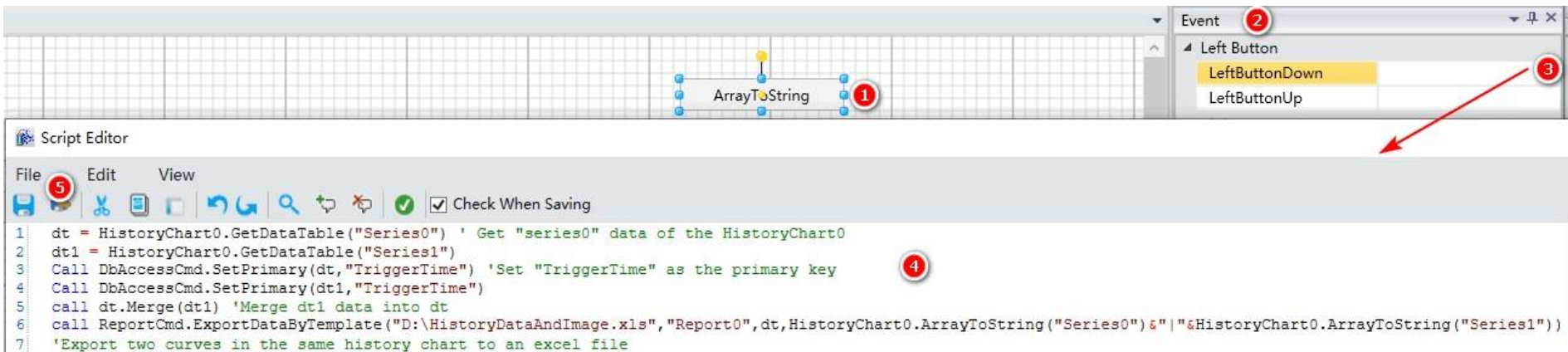


➤ ArrayToString приклад :

Експортуйте дві криві в одній діаграмі історії в Excel

Перші 8 кроків такі самі, як перші 8 кроків експорту приклад2

(9) Створіть одну кнопку (ArrayToString) у Window0 , налаштуйте подію LeftButtonDown кнопки

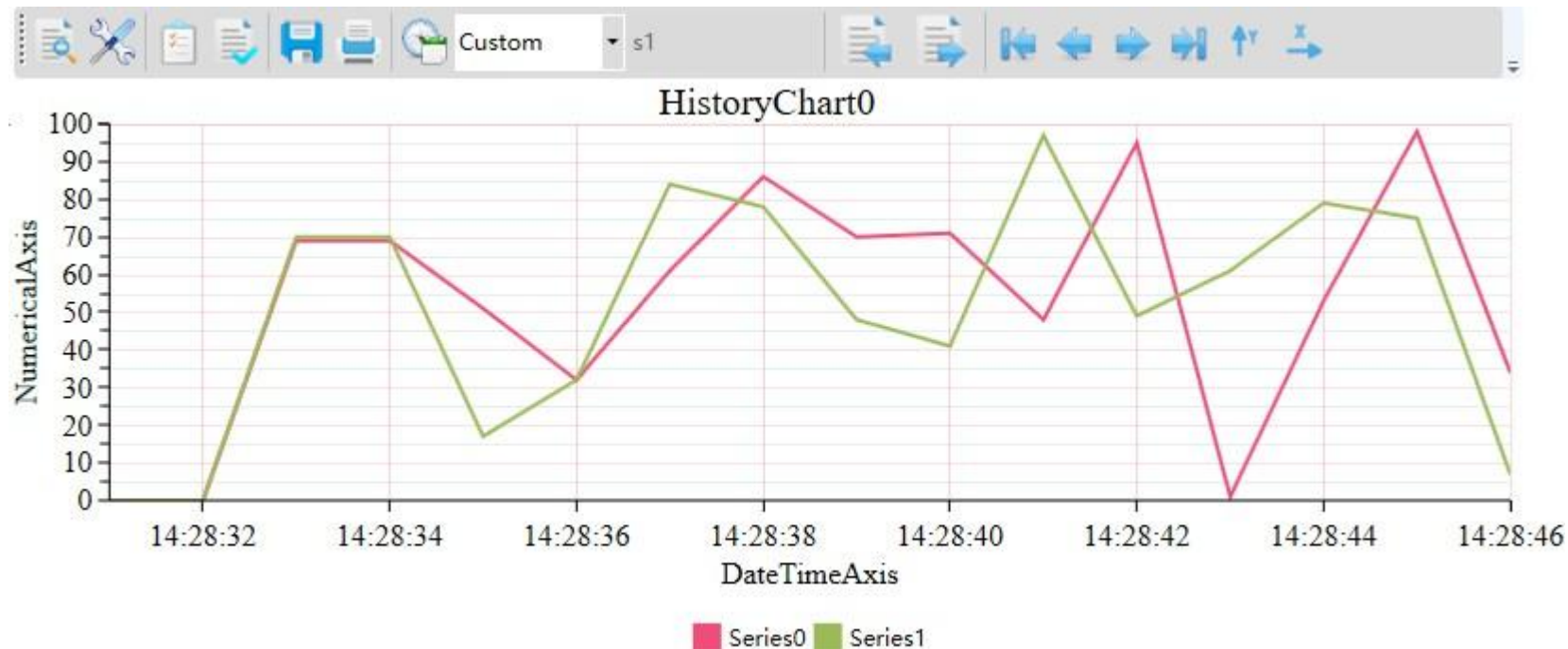


The screenshot displays the Delta software interface. At the top, a grid contains a yellow circle and a blue square, with an 'ArrayToString' script block (marked with a red circle 1) positioned below them. To the right, the 'Event' panel (marked with a red circle 2) shows the 'Left Button' event, with 'LeftButtonDown' selected (marked with a red circle 3). Below the grid is the 'Script Editor' window. It features a menu bar with 'File' (marked with a red circle 5), 'Edit', and 'View'. The toolbar includes icons for saving, undo, redo, and other editing functions, along with a 'Check When Saving' checkbox. The script text is as follows:

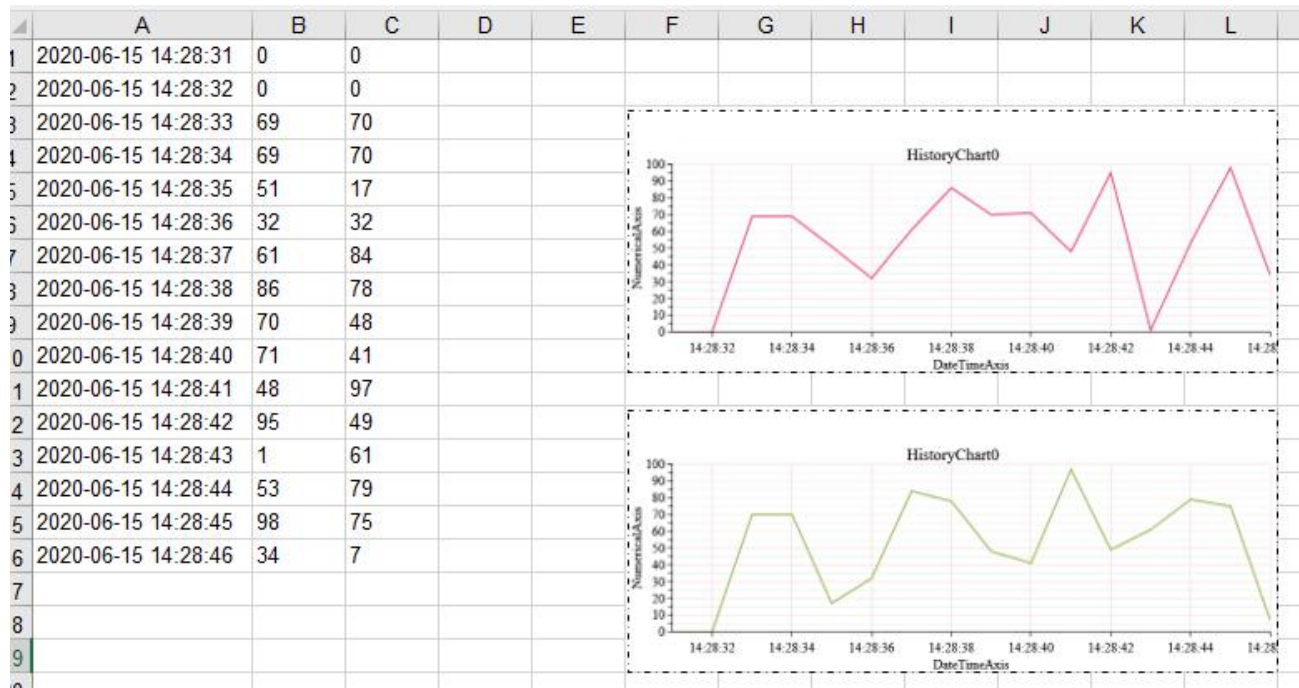
```
1 dt = HistoryChart0.GetDataTable("Series0") ' Get "series0" data of the HistoryChart0
2 dt1 = HistoryChart0.GetDataTable("Series1")
3 Call DbAccessCmd.SetPrimary(dt,"TriggerTime") 'Set "TriggerTime" as the primary key
4 Call DbAccessCmd.SetPrimary(dt1,"TriggerTime")
5 call dt.Merge(dt1) 'Merge dt1 data into dt
6 call ReportCmd.ExportDataByTemplate("D:\HistoryDataAndImage.xls", "Report0",dt,HistoryChart0.ArrayToString("Series0") & "|" & HistoryChart0.ArrayToString("Series1"))
7 'Export two curves in the same history chart to an excel file
```

A red arrow points from the 'LeftButtonDown' event in the Event panel to the script editor, indicating the script is triggered by this event. A red circle 4 is placed next to the script text.

(10) Запустіть Window0 , дані історії запитів



(11) Експортуйте дані історії та зображення, експортований файл Excel показано нижче



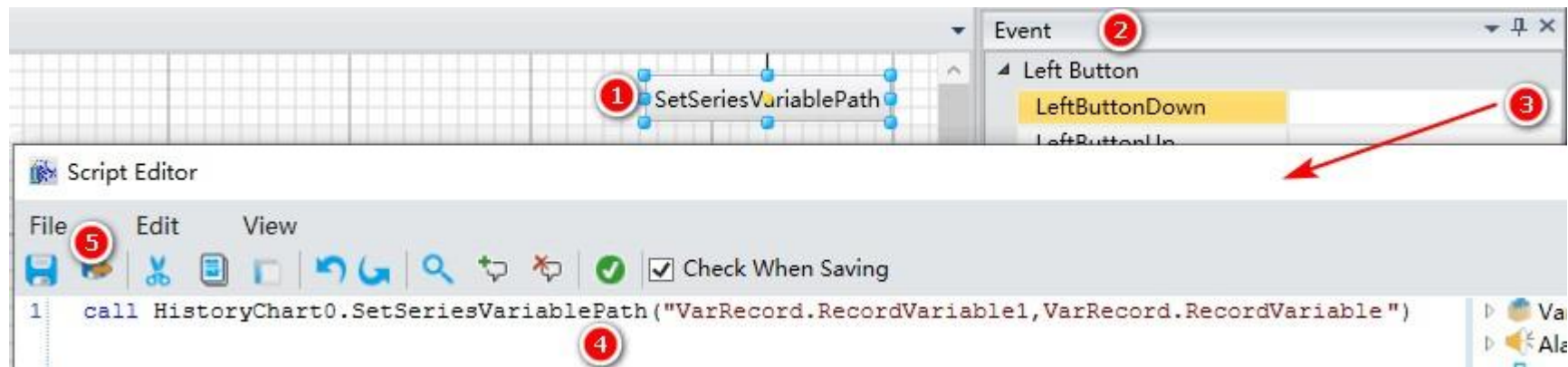
Натисніть кнопку « ArrayToImage », після чого Series0 і Series1 HistoryChart0 будуть успішно експортовані на диск D, а ім'я файлу – « HistoryDataAndImage » .

➤ SetSeriesVariablePath приклад :

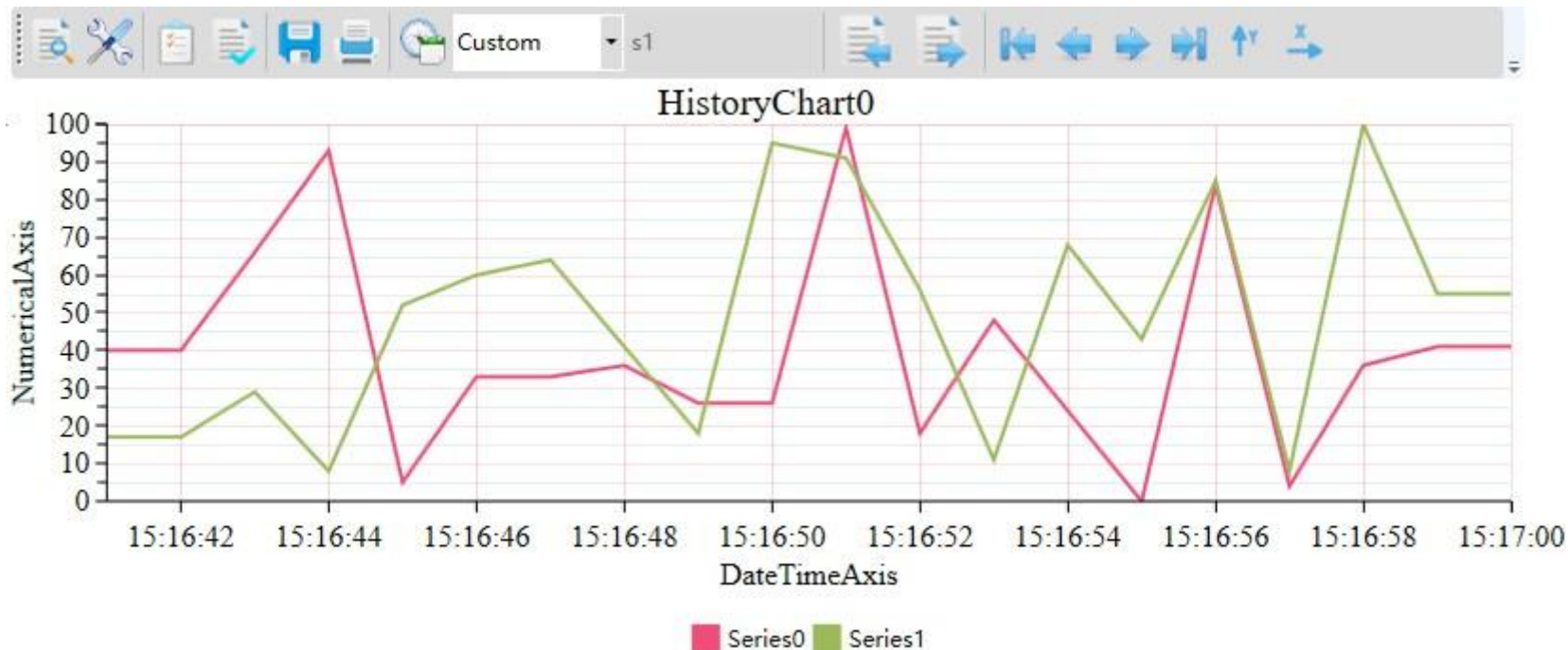
Змінні історії обміну, пов'язані з двома кривими

Перші 5 кроків такі самі, як (1)(2)(3)(4)(5) кроки QueryHistoryDate приклад (6)

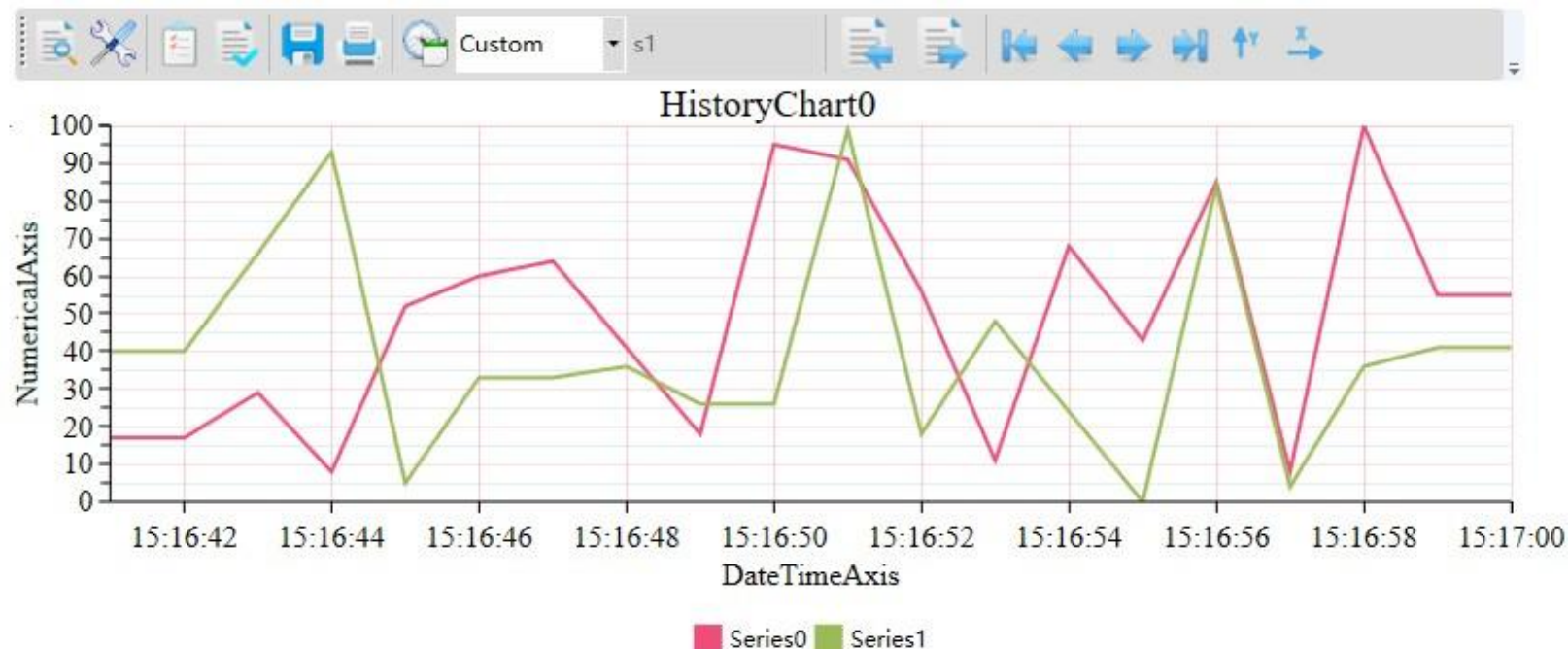
Створіть одну кнопку (SetSeriesVariablePath) у Window0 , налаштуйте подію LeftButtonDown кнопки



(7) Запустіть Window0, дані історії запитів



(8) Натисніть кнопку « SetSeriesVariablePath » , а потім знову запитайте дані історії , як показано нижче



Порівнюючи зображення на попередньому кроці, можна зробити висновок, що історичні змінні, пов'язані з двома кривими, помінялися місцями.

➤ AddPoint, UpdatePoint, DeletePoint, DeletevalueAll приклад :

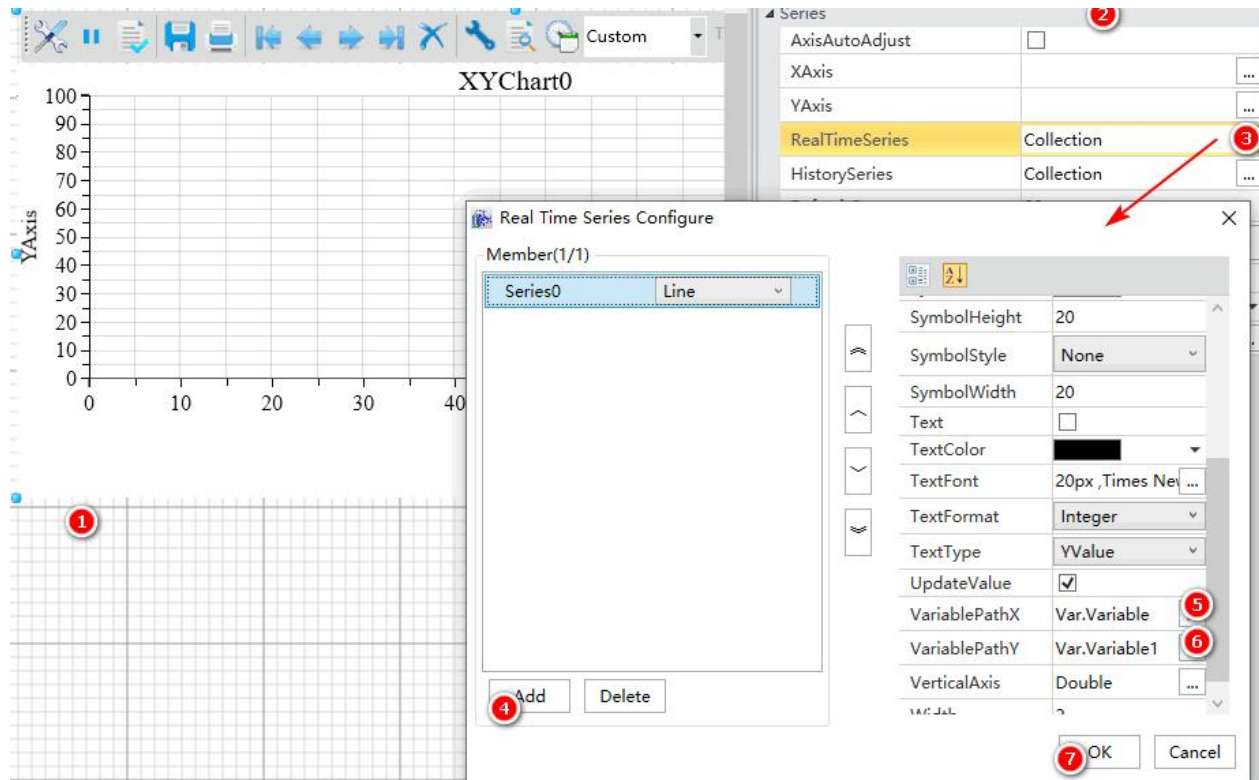
Додайте точку в XYChart

(1) Створіть 2 змінні : змінна , змінна1

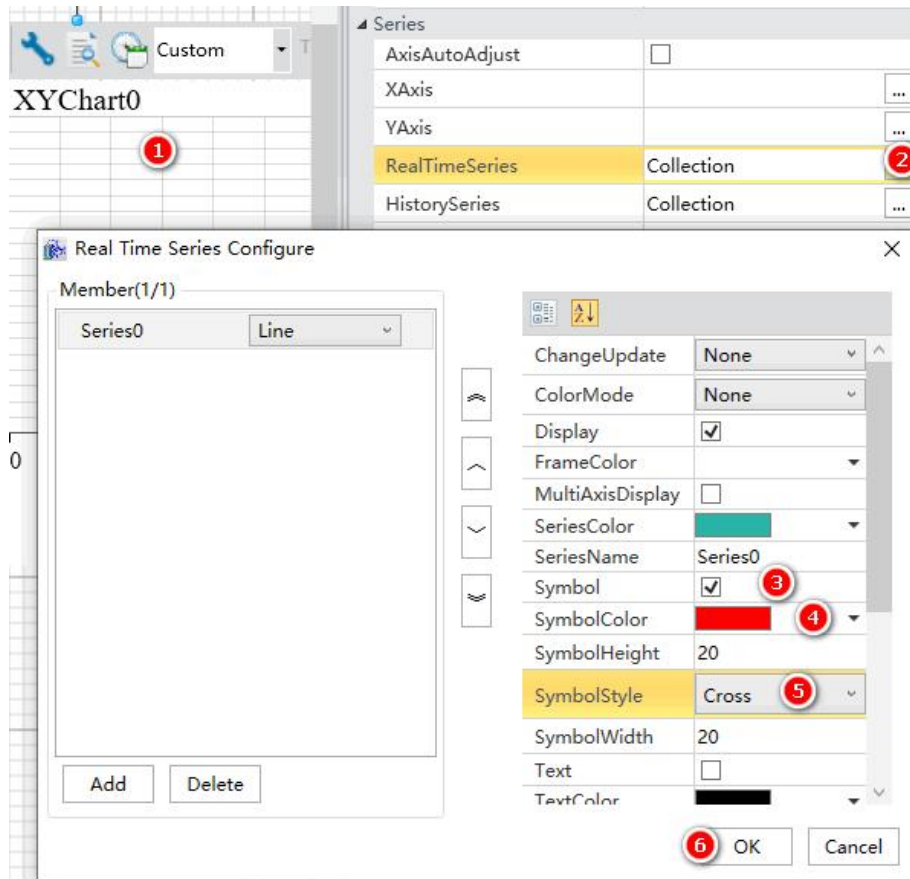
(2) Створіть змодельований пристрій : Device0

(3) Створіть дві адреси моделювання в Device0 , пов'язані зі змінною , змінною1 відповідно

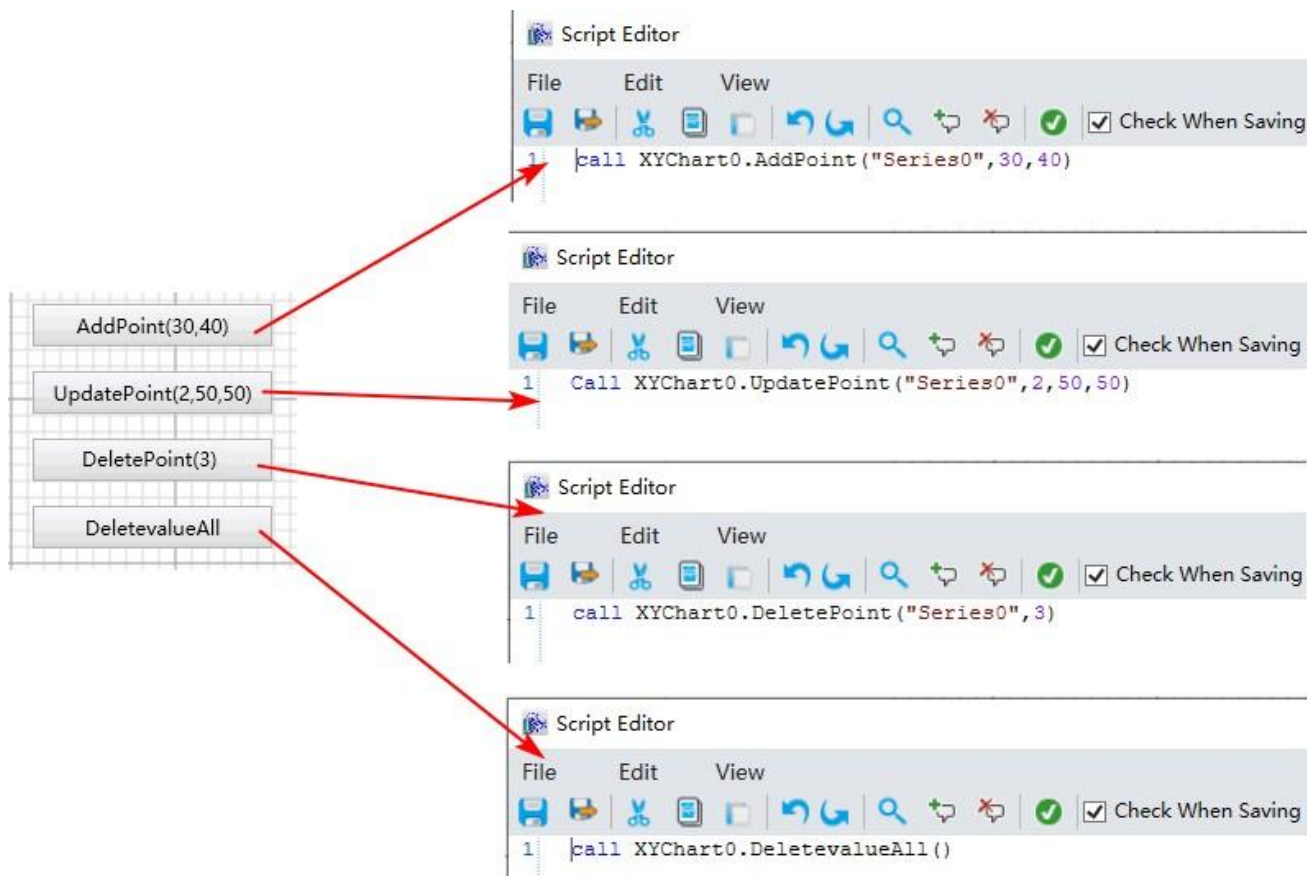
(4) Створіть XYChart0 у вікні Window0 і додайте одну криву в XYChart0 , VariablePathX Series0 , пов'язану зі змінною , VariablePathY Series0 , пов'язану зі змінною1



(5) Налаштуйте відображення символів Series0



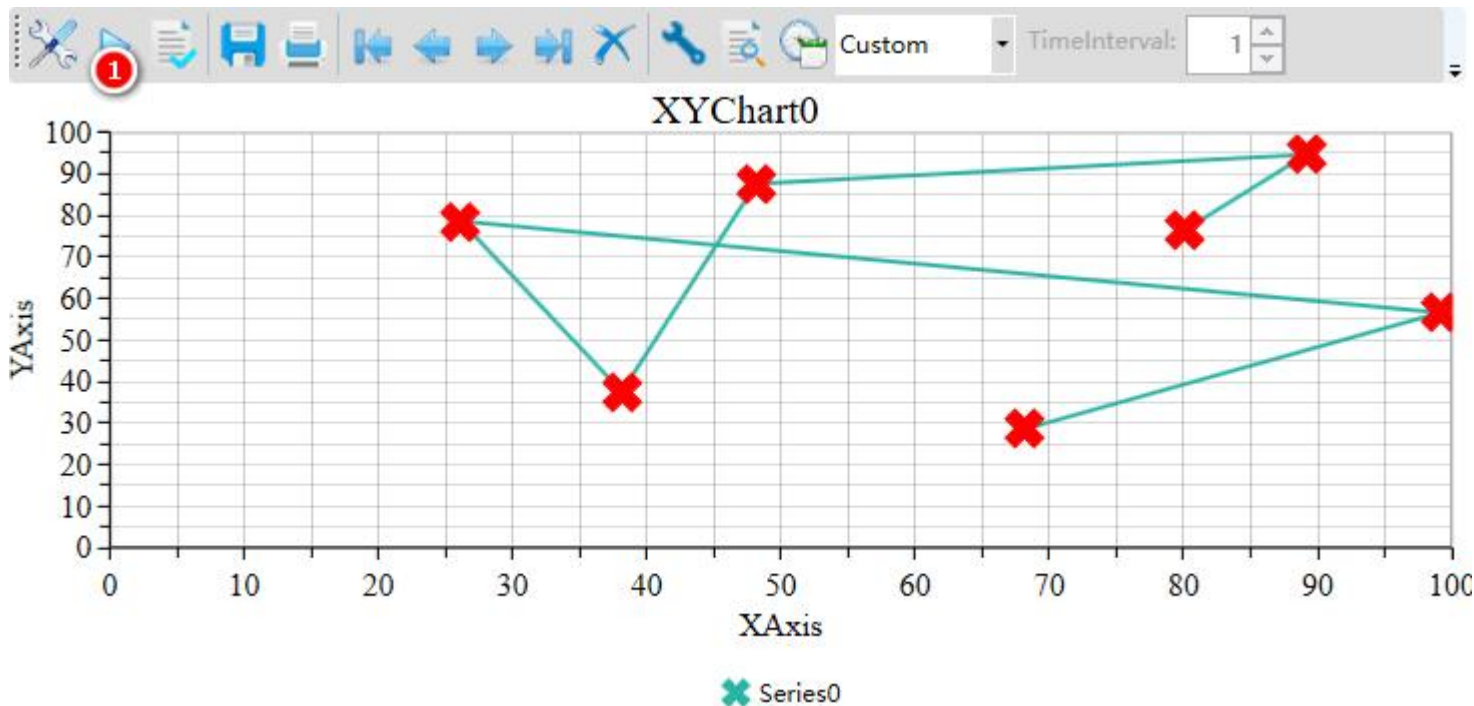
(6) Створіть 4 кнопки у вікні, налаштуйте подію LeftButtonDown для 4 КНОПКИ



The diagram illustrates the configuration of four buttons for the XYChart script editor. Each button is shown with its corresponding script code in a separate Script Editor window. Red arrows indicate the mapping from the button to its script.

- AddPoint(30,40)** maps to the script: `call XYChart0.AddPoint("Series0",30,40)`
- UpdatePoint(2,50,50)** maps to the script: `Call XYChart0.UpdatePoint("Series0",2,50,50)`
- DeletePoint(3)** maps to the script: `call XYChart0.DeletePoint("Series0",3)`
- DeletevalueAll** maps to the script: `call XYChart0.DeletevalueAll()`

(7) Запустіть Window0 на деякий час, потім натисніть кнопку паузи в XYChart0, щоб зупинити оновлення даних



(8) Виконувати сценарії

- ① Натисніть кнопку « AddPoint(30,40) » , Series0 додасть точку даних (30,40) у кінці себе
- ② Натисніть кнопку « UpdatePoint(2,50,50) » , точку даних з індексом 2 на Series0 буде оновлено до (50,50)
- ③ Натисніть кнопку « DeletePoint(3) » , точка даних з індексом 3 на Series0 буде видалена
- ④ Натисніть кнопку « DeletevalueAll » , усі точки даних на XYChart0 буде видалено

➤ SetDateConditionListStartTime приклад :

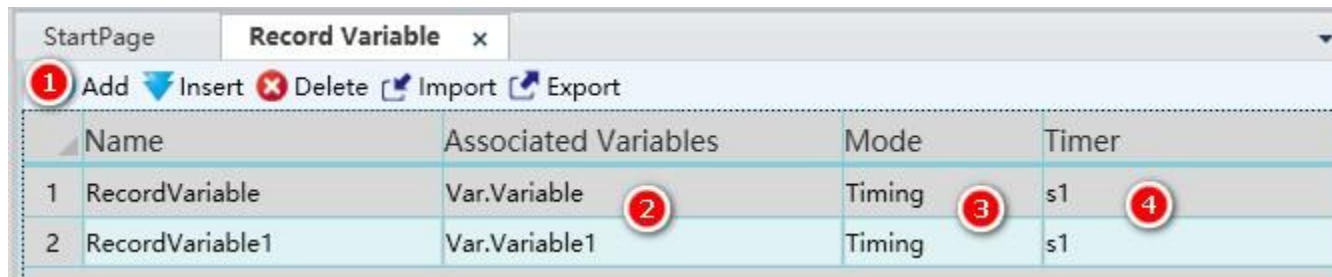
Запит мінімальних, середніх і максимальних даних двох елементів на HistoryColumnChart за певний період часу

(1) Створіть 2 змінні : змінна , змінна1

(2) Створіть змодельований пристрій : Device0

(3) Створіть дві адреси моделювання в Device0 , пов'язані зі змінною , змінною1 відповідно

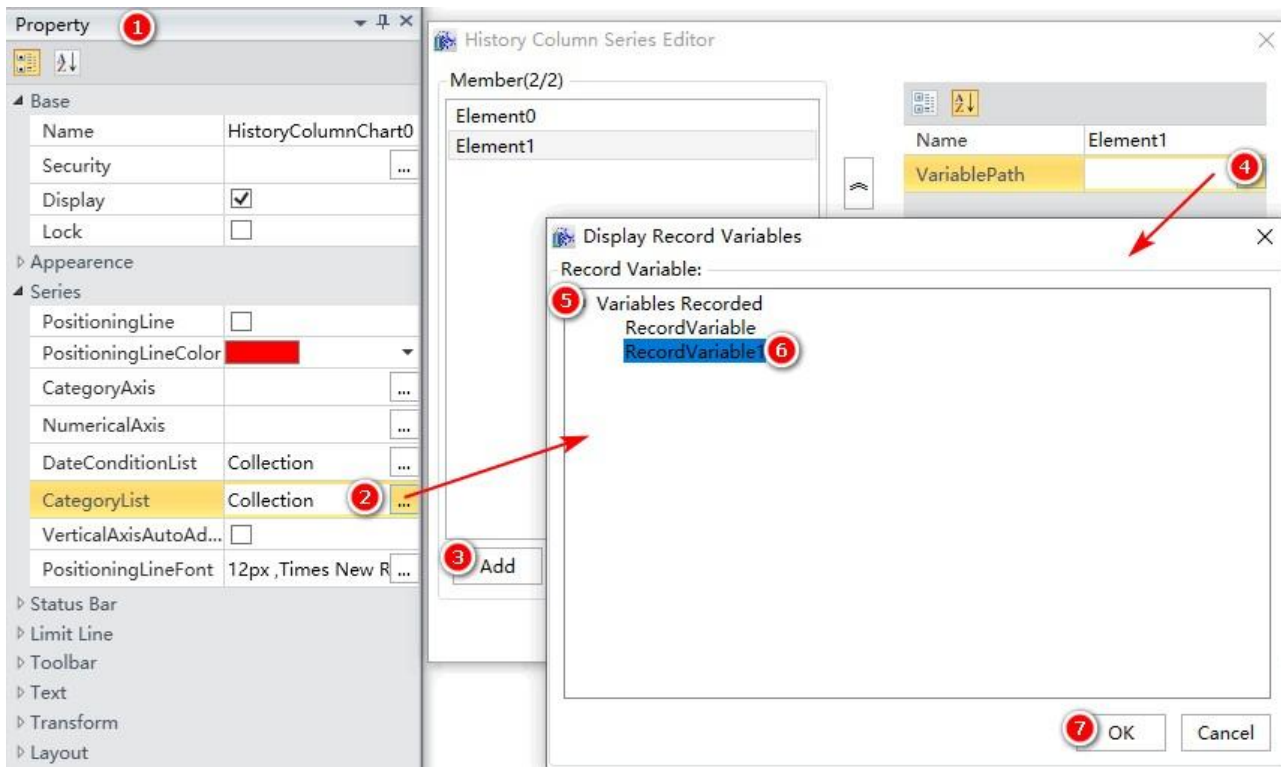
(4) Створіть дві історичні змінні (RecordVariable, RecordVariable1) у Запис вузла Variable , пов'язаного зі змінною , змінною1 відповідно



	Name	Associated Variables	Mode	Timer
1	RecordVariable	Var.Variable	Timing	s1
2	RecordVariable1	Var.Variable1	Timing	s1

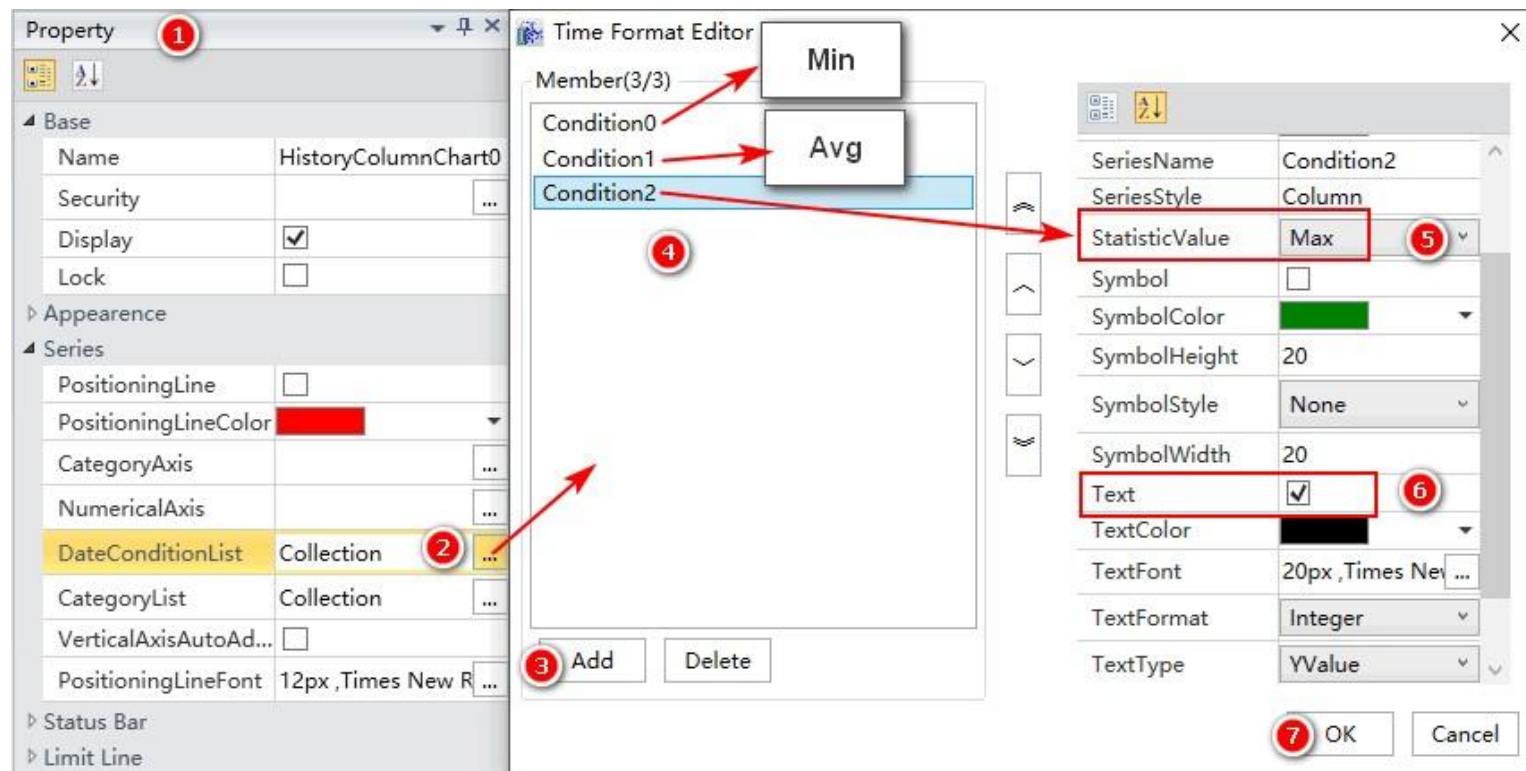
Скрипти HistoryColumnChart

(5) Створіть HistoryColumnChart0 у Window0 та додайте 2 елементи в HistoryColumnChart0, Element0 пов'язану RecordVariable, Element1 пов'язану RecordVariable1



(6) Задайте властивості умов

StatisticValue Condition0, Condition1, Condition2 є Min, Avg, Max відповідно

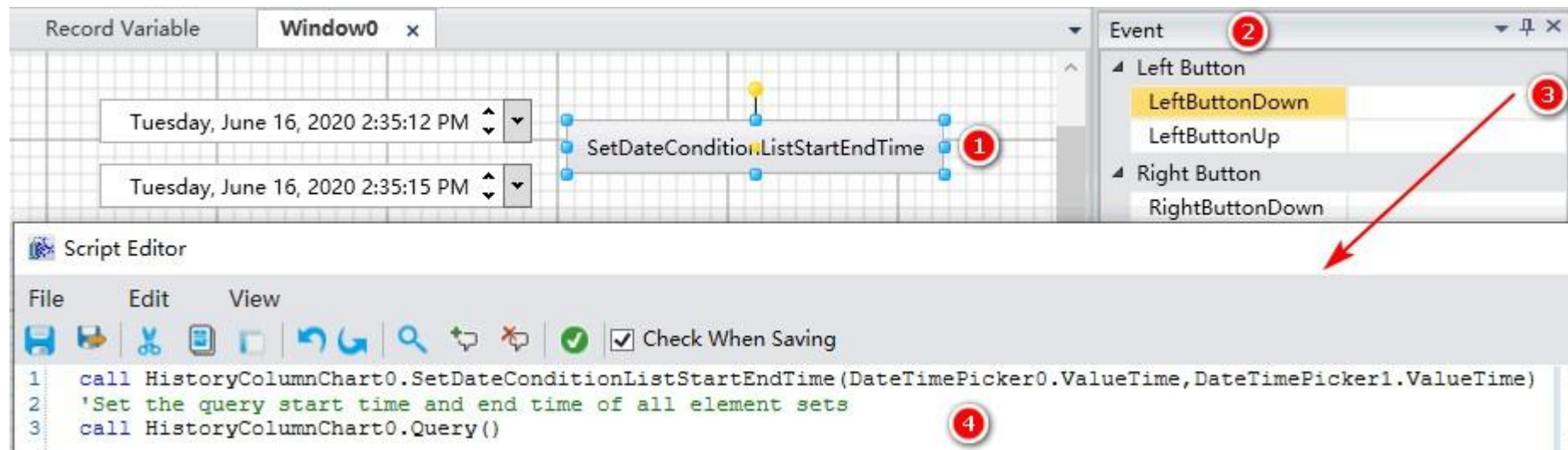


The screenshot illustrates the configuration of the HistoryColumnChart. The interface is divided into several panels:

- Property (1):** Shows the chart's base and series properties. The 'DateConditionList' property is highlighted with a red circle 2.
- Time Format Editor (2):** Shows the conditions list. The 'Condition2' row is selected with a red circle 4.
- Min, Avg, Max (3):** These buttons are used to assign statistical values to the conditions. Red arrows point from the buttons to the corresponding condition rows in the Time Format Editor.
- Condition2 (5):** The detailed properties for Condition2 are shown on the right. The 'StatisticValue' is set to 'Max' (red circle 5) and the 'Text' property is checked (red circle 6).
- Buttons (7):** The 'Add' and 'Delete' buttons are located at the bottom of the Time Format Editor. The 'OK' and 'Cancel' buttons are at the bottom of the right panel.

Скрипти HistoryColumnChart

(7) Створіть два DateTimePicker(DateTimePicker0, DateTimePicker1) і кнопку у Window0 , налаштуйте подію LeftButtonDown кнопки



Record Variable: Window0 x

Timeline:

- DateTimePicker0: Tuesday, June 16, 2020 2:35:12 PM
- Script: SetDateConditionListStartEndTime (1)
- DateTimePicker1: Tuesday, June 16, 2020 2:35:15 PM

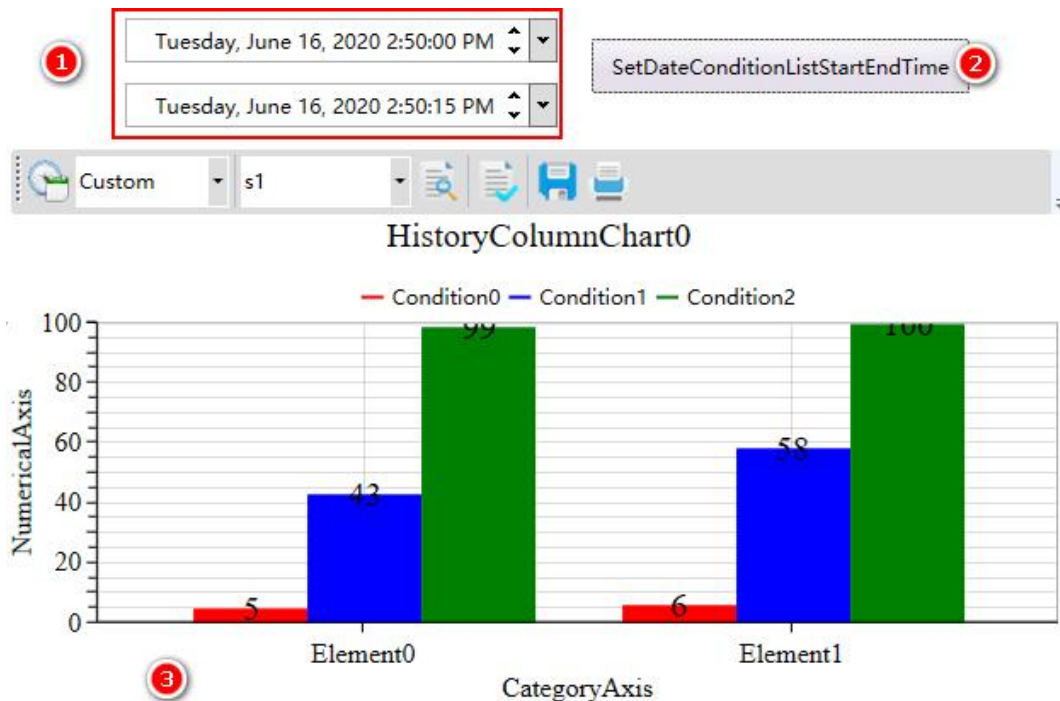
Event (2):

- Left Button
 - LeftButtonDown (3)
 - LeftButtonUp
- Right Button
 - RightButtonDown

Script Editor (4):

```
1 call HistoryColumnChart0.SetDateConditionListStartEndTime(DateTimePicker0.ValueTime, DateTimePicker1.ValueTime)
2 'Set the query start time and end time of all element sets
3 call HistoryColumnChart0.Query()
```

(8) Запустіть Window0



① Встановіть час початку та завершення запиту

② Застосуйте встановлений період запиту до HistoryColumnChart0, а потім запитайте дані

③ Результат запиту: мінімальне, середнє та максимальне значення Element0 становлять 5,43,99; мінімальне, середнє та максимальне значення Element1 становлять 6,58,100

➤ GetCurrentAlarmColumnInf приклад :

Отримайте вміст рядка 2 і стовпця 7 вікна тривоги в реальному часі

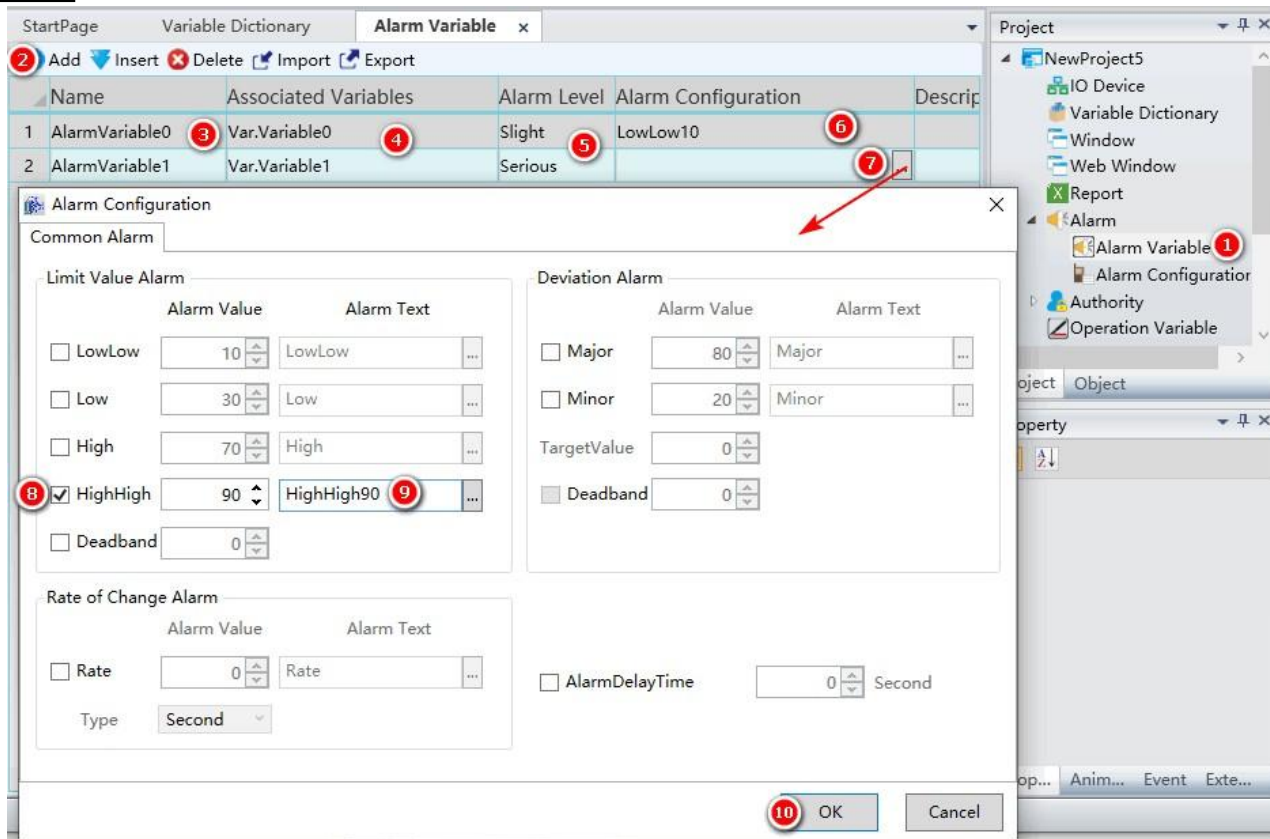
(1) Створіть 2 змінні : Variable0 , Variable1



	Name	Variable Type	Initial Value	Retentive Value	Minimum Value	Maximum Value	Deadband
1	Variable0	Analog	0		0	10000	
2	Variable1	Analog	0		0	10000	

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

(2) Створіть 2 змінні сигналу (AlarmVariable0 , AlarmVariable1), пов'язані зі змінною Variable0 , Variable1 відповідно



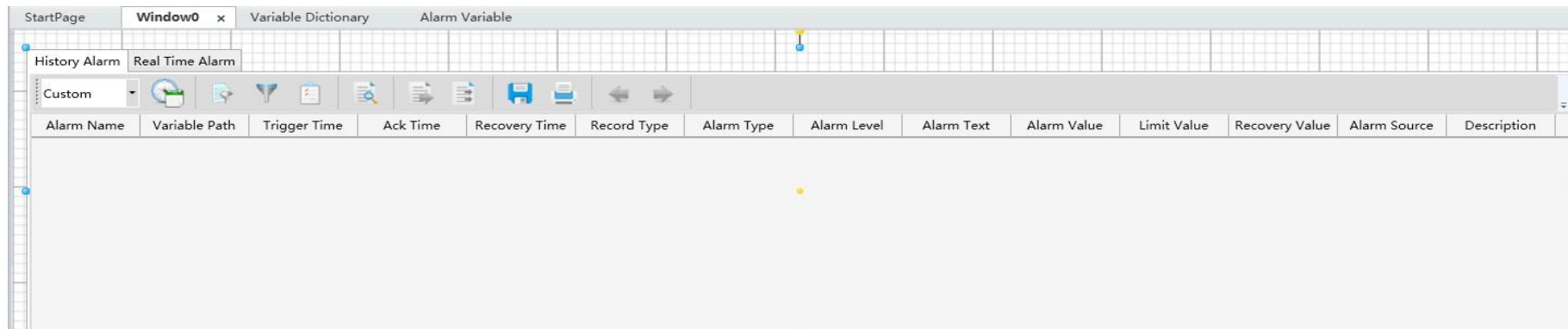
The screenshot shows the 'Alarm Configuration' window with the following details:

- Table:**

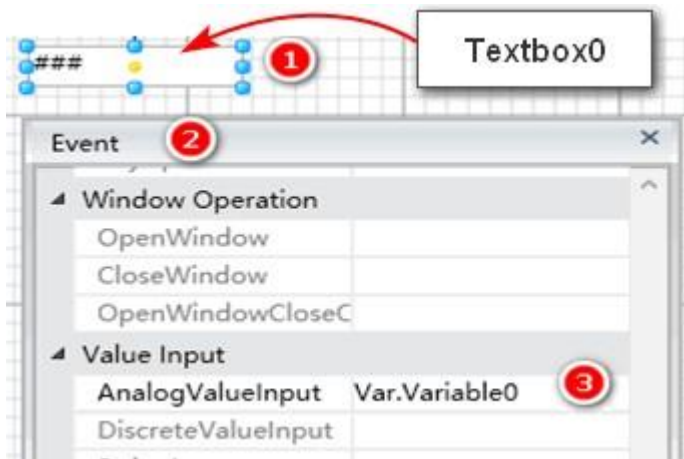
Name	Associated Variables	Alarm Level	Alarm Configuration	Description
1 AlarmVariable0	Var.Variable0	Slight	LowLow10	
2 AlarmVariable1	Var.Variable1	Serious		
- Alarm Configuration Panel:**
 - Common Alarm:**
 - Limit Value Alarm:**
 - ☐ LowLow
 - ☐ Low
 - ☐ High
 - ☒ HighHigh (Alarm Value: 90, Alarm Text: HighHigh90)
 - ☐ Deadband
 - Rate of Change Alarm:**
 - ☐ Rate
 - Type: Second
 - Deviation Alarm:**
 - ☐ Major
 - ☐ Minor
 - TargetValue: 0
 - Deadband: 0

※Зверніться до розділу «11.3 Змінна тривоги» в посібнику користувача.

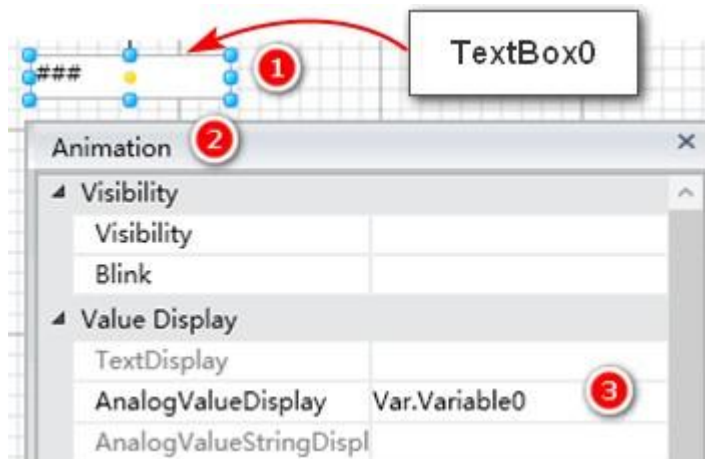
(3) Створіть AlarmWindow0 у Window0



(4) Створіть 2 текстові поля (Textbox0 , TextBox1) у Window0 , подія введення аналогового значення та анімація відображення аналогового значення для TextBox0 , TextBox1 пов'язані зі змінною0, змінною1 відповідно

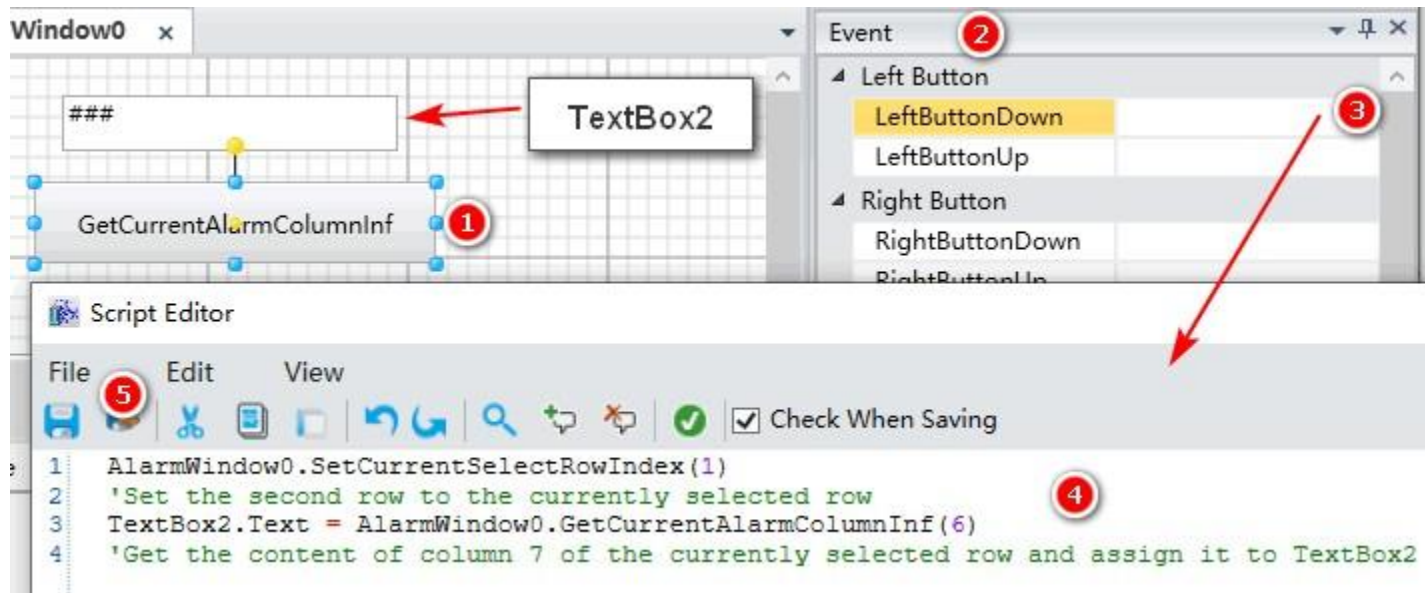


Введення аналогового значення



Відображення аналогового значення

(5) Створіть TextBox2 і кнопку (GetCurrentAlarmColumnInf) у Window0 ,
налаштувати подію кнопки LeftButtonDown

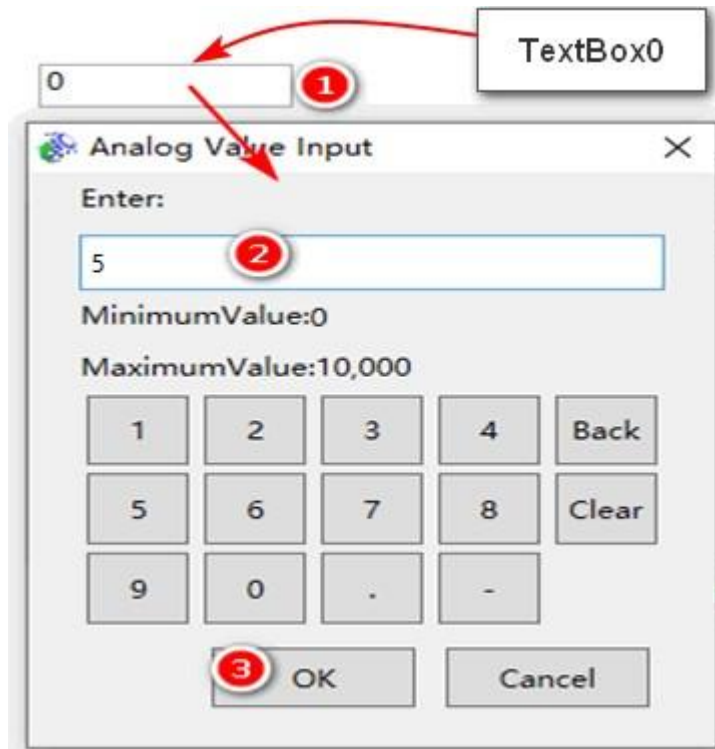


The screenshot displays the development environment for the AlarmWindow. The top view is the design surface, showing a grid with a yellow dot representing a button and a text box labeled "TextBox2". A red arrow points from the button to the text box. Below the design surface is the Script Editor, which contains the following code:

```
1 AlarmWindow0.SetCurrentSelectRowIndex(1)
2 'Set the second row to the currently selected row
3 TextBox2.Text = AlarmWindow0.GetCurrentAlarmColumnInf(6)
4 'Get the content of column 7 of the currently selected row and assign it to TextBox2
```

The Event window on the right shows the "Left Button" event, with "LeftButtonDown" selected. A red arrow points from the "LeftButtonDown" event to the script editor. The Script Editor has a menu bar (File, Edit, View) and a toolbar with icons for saving, undo, redo, and other actions. A red circle with the number 5 is next to the Save icon in the toolbar.

(6) Запустіть Window0 . У Textbox0 введіть послідовно 3,13,5; У Textbox1 послідовно введіть 96,86,97



(7) Вікно будильника в реальному часі відображається наступним чином

1

5

2

97

###

GetCurrentAlarmColumnInf

History Alarm Real Time Alarm 3

4

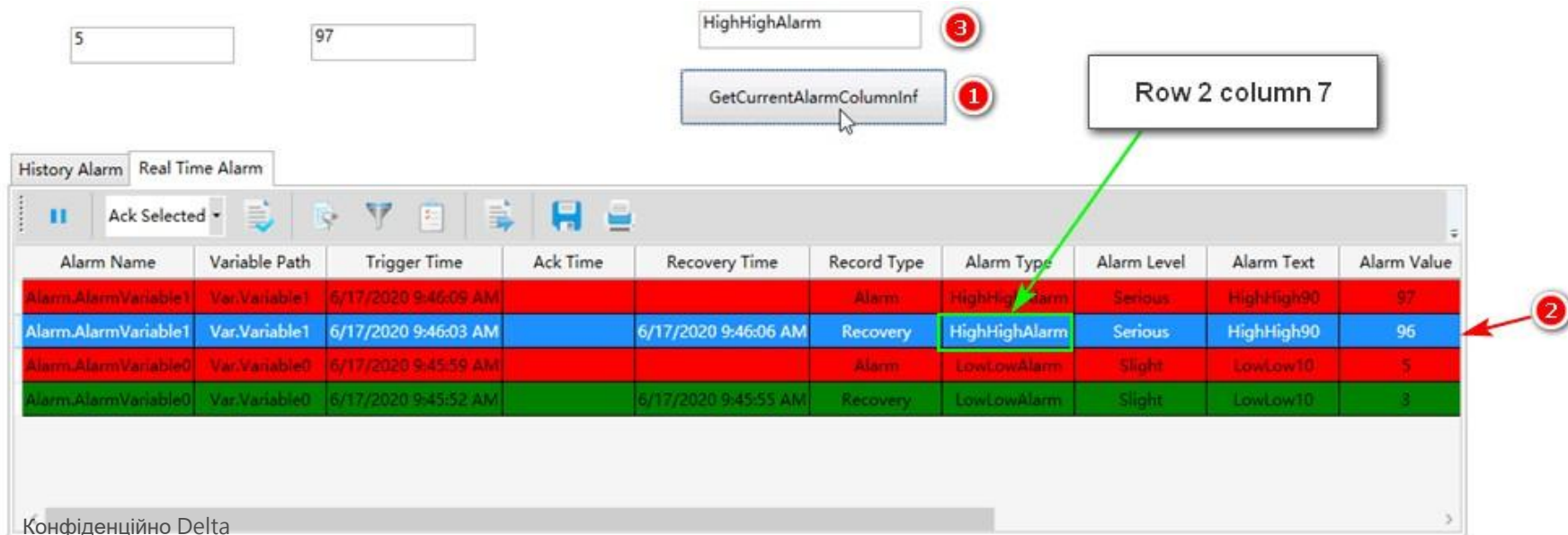
Ack Selected

Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	Alarm Text	Alarm Valu
Alarm.AlarmVariable1	Var.Variable1	6/17/2020 10:24:31 AM			Alarm	HighHighAlarm	Serious	HighHigh90	97
Alarm.AlarmVariable1	Var.Variable1	6/17/2020 10:24:25 AM		6/17/2020 10:24:28 AM	Recovery	HighHighAlarm	Serious	HighHigh90	96
Alarm.AlarmVariable0	Var.Variable0	6/17/2020 10:24:22 AM			Alarm	LowLowAlarm	Slight	LowLow10	5
Alarm.AlarmVariable0	Var.Variable0	6/17/2020 10:24:16 AM		6/17/2020 10:24:19 AM	Recovery	LowLowAlarm	Slight	LowLow10	3

(8) Виконайте сценарії

① Натисніть кнопку «GetCurrentAlarmColumnInf».

③ TextBox2 відображає вміст стовпця 7 поточного вибраного рядка -- HighHighAlarm



HighHighAlarm 3

GetCurrentAlarmColumnInf 1

Row 2 column 7

History Alarm Real Time Alarm

Ack Selected

Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	Alarm Text	Alarm Value
Alarm.AlarmVariable1	Var.Variable1	6/17/2020 9:46:09 AM			Alarm	HighHighAlarm	Serious	HighHigh90	97
Alarm.AlarmVariable1	Var.Variable1	6/17/2020 9:46:03 AM		6/17/2020 9:46:06 AM	Recovery	HighHighAlarm	Serious	HighHigh90	96
Alarm.AlarmVariable0	Var.Variable0	6/17/2020 9:45:59 AM			Alarm	LowLowAlarm	Slight	LowLow10	5
Alarm.AlarmVariable0	Var.Variable0	6/17/2020 9:45:52 AM		6/17/2020 9:45:55 AM	Recovery	LowLowAlarm	Slight	LowLow10	3

Конфіденційно Delta

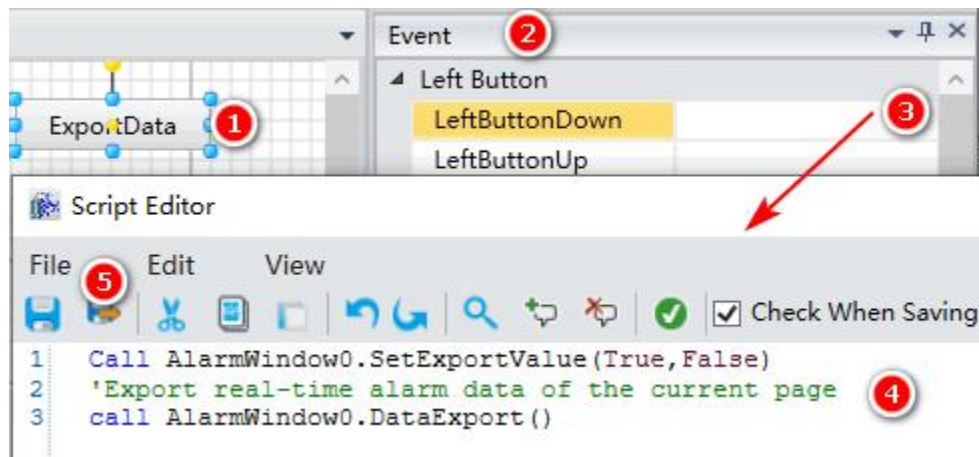
➤ ExportData приклад1 :

Експорт даних тривоги поточної сторінки в реальному часі

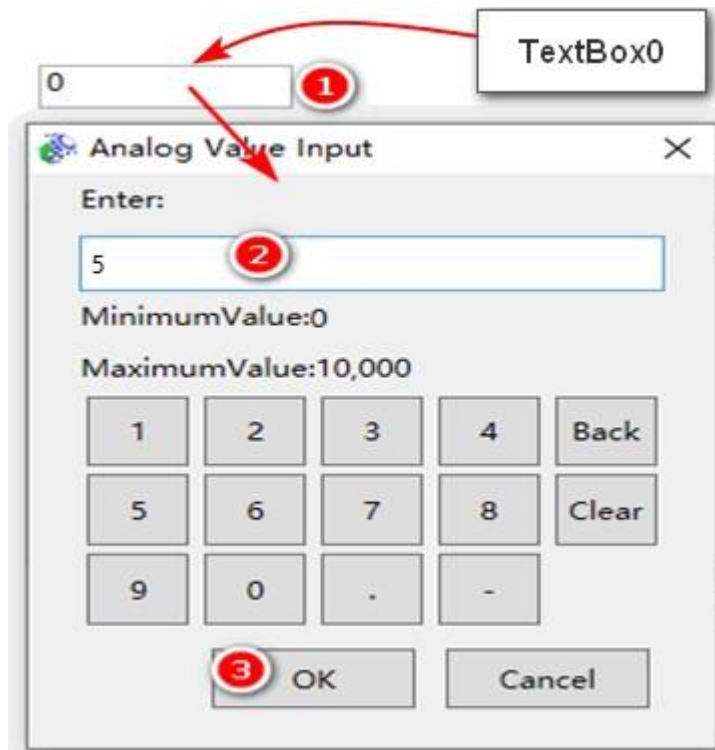
Перші 4 кроки такі самі, як (1)(2)(3)(4) кроки GetCurrentAlarmColumnInf

приклад

(5) Створіть кнопку (ExportData) у Window0 , налаштуйте подію LeftButtonDown кнопки



(6) Запустіть Window0 . У Textbox0 введіть послідовно 3,13,5; У Textbox1 послідовно введіть 96,86,97



(7) Вікно будильника в реальному часі відображається наступним чином

1 5 2 97 ExportData

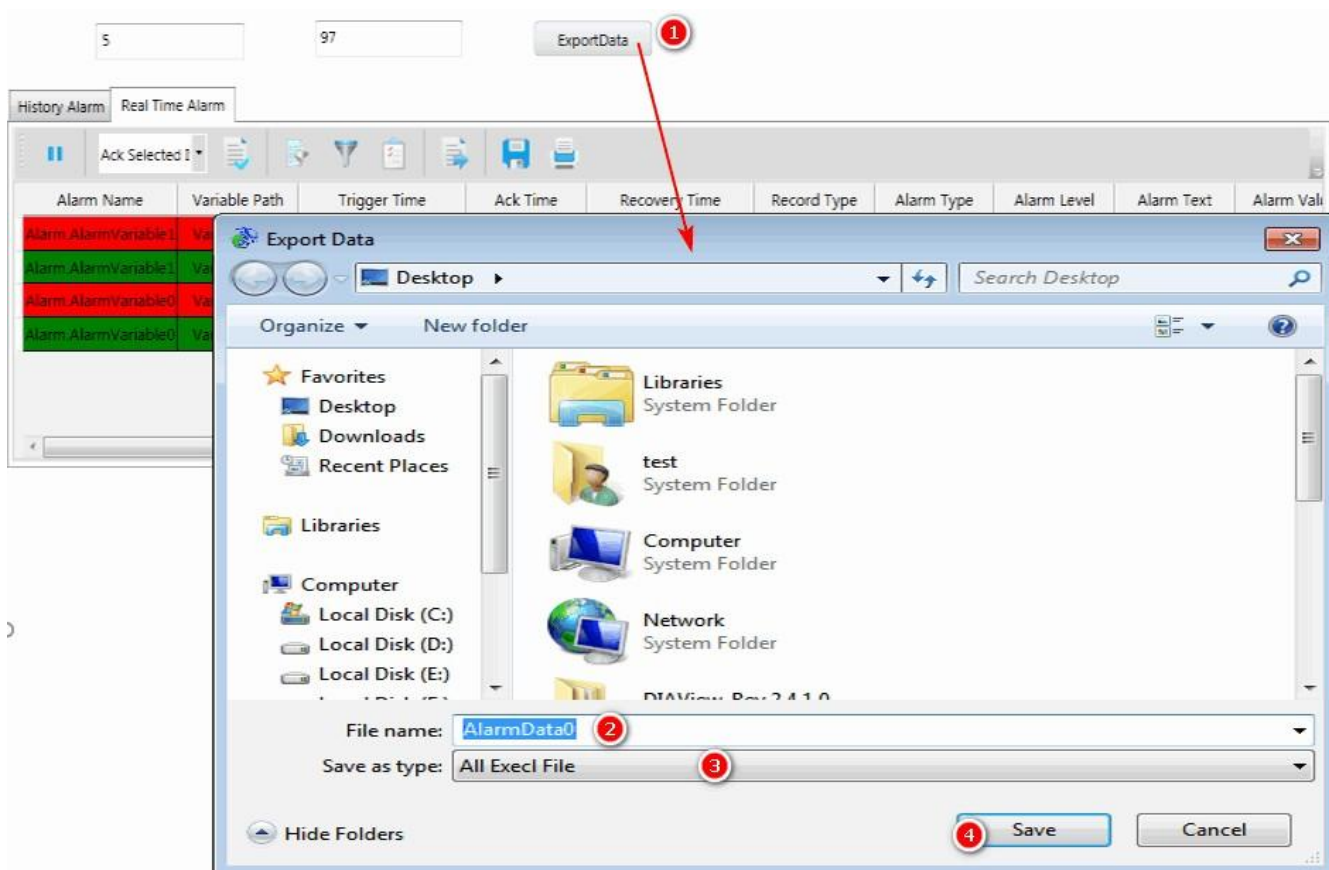
History Alarm Real Time Alarm 3

4 Ack Selected

Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	Alarm Text	Alarm Value
Alarm.AlarmVariable1	Var.Variable1	6/17/2020 11:58:34 AM			Alarm	HighHighAlarm	Serious	HighHigh90	97
Alarm.AlarmVariable1	Var.Variable1	6/17/2020 11:58:28 AM		6/17/2020 11:58:31 AM	Recovery	HighHighAlarm	Serious	HighHigh90	96
Alarm.AlarmVariable0	Var.Variable0	6/17/2020 11:58:23 AM			Alarm	LowLowAlarm	Slight	LowLow10	5
Alarm.AlarmVariable0	Var.Variable0	6/17/2020 11:58:18 AM		6/17/2020 11:58:21 AM	Recovery	LowLowAlarm	Slight	LowLow10	3

Сценарії AlarmWindow

(8) Виконайте сценарії. Натисніть кнопку « ExportData », щоб експортувати дані тривоги у файл Excel



(9) Експортований файл excel відображається так

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	Alarm Text	Alarm Value	Limit Value	Current Value	Recovery Value	Alarm Source	Description
2	Alarm.AlarmVariable1	Var.Variable1	2020-06-17 11:55:35			Alarm	HighHighAlarm	Serious	HighHigh90	97	90	97		test-PC	
3	Alarm.AlarmVariable1	Var.Variable1	2020-06-17 11:55:26		2020-06-17 11:55:30	Recovery	HighHighAlarm	Serious	HighHigh90	96	90	97	86	test-PC	
4	Alarm.AlarmVariable0	Var.Variable0	2020-06-17 11:55:19			Alarm	LowLowAlarm	Slight	LowLow10	5	10	5		test-PC	
5	Alarm.AlarmVariable0	Var.Variable0	2020-06-17 11:55:13		2020-06-17 11:55:17	Recovery	LowLowAlarm	Slight	LowLow10	3	10	5	13	test-PC	
6															

➤ ExportData example2 :

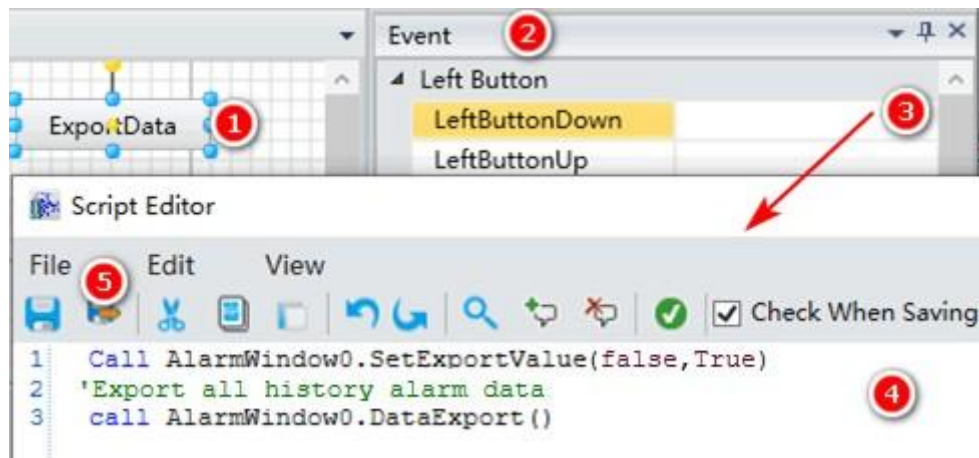
Експортувати всі дані історії

тривоги

Перші 4 кроки такі самі, як (1)(2)(3)(4) кроки GetCurrentAlarmColumnInf

приклад

(5) Створіть кнопку (ExportData) у Window0 , налаштуйте подію LeftButtonDown кнопки



(6) Запустіть Window0. У Textbox0 введіть послідовно 3,13,5;

У Textbox1 послідовно введіть 96,86,97

(7) Вікно нагадування історії відображається наступним чином

1

2

ExportData

3

History Alarm

Real Time Alarm

NearestOne

4

5

No.1Page

Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	Alarm Text	Alarm Value
1 AlarmVariable	Var.Variable0	6/18/2020 9:52:50 AM		6/18/2020 9:52:52 AM	Recovery	LowLowAlarm	Slight	LowLow10	3
2 AlarmVariable	Var.Variable0	6/18/2020 9:52:54 AM			Alarm	LowLowAlarm	Slight	LowLow10	5
3 AlarmVariable1	Var.Variable1	6/18/2020 9:53:03 AM		6/18/2020 9:53:07 AM	Recovery	HighHighAlarm	Serious	HighHigh90	96
4 AlarmVariable1	Var.Variable1	6/18/2020 9:53:12 AM			Alarm	HighHighAlarm	Serious	HighHigh90	97

<

Конфіденційно Delta

>

(8) Виконайте сценарії. Натисніть кнопку «Експортувати дані», щоб експортувати дані тривоги у файл Excel (9) Експортований файл Excel відображається наступним чином

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Alarm Name	Variable Path	Trigger Time	Ack Time	Recovery Time	Record Type	Alarm Type	Alarm Level	Alarm Text	Alarm Value	Limit Value	Recovery Value	Alarm Source	Description
2	Alarm.Alarm	Var.Variable0	2020-06-18 09:52:50		2020-06-18 09:52:52	Recovery	LowLowAlarm	Slight	LowLow10	3	10	13	CNWJ6IAF	
3	Alarm.Alarm	Var.Variable0	2020-06-18 09:52:54			Alarm	LowLowAlarm	Slight	LowLow10	5	10		CNWJ6IAF	
4	Alarm.Alarm	Var.Variable1	2020-06-18 09:53:03		2020-06-18 09:53:07	Recovery	HighHighAlarm	Serious	HighHigh90	96	90	86	CNWJ6IAF	
5	Alarm.Alarm	Var.Variable1	2020-06-18 09:53:12			Alarm	HighHighAlarm	Serious	HighHigh90	97	90		CNWJ6IAF	

➤ SetRealTimeVariableChange приклад :

звіти відображають дані в реальному часі

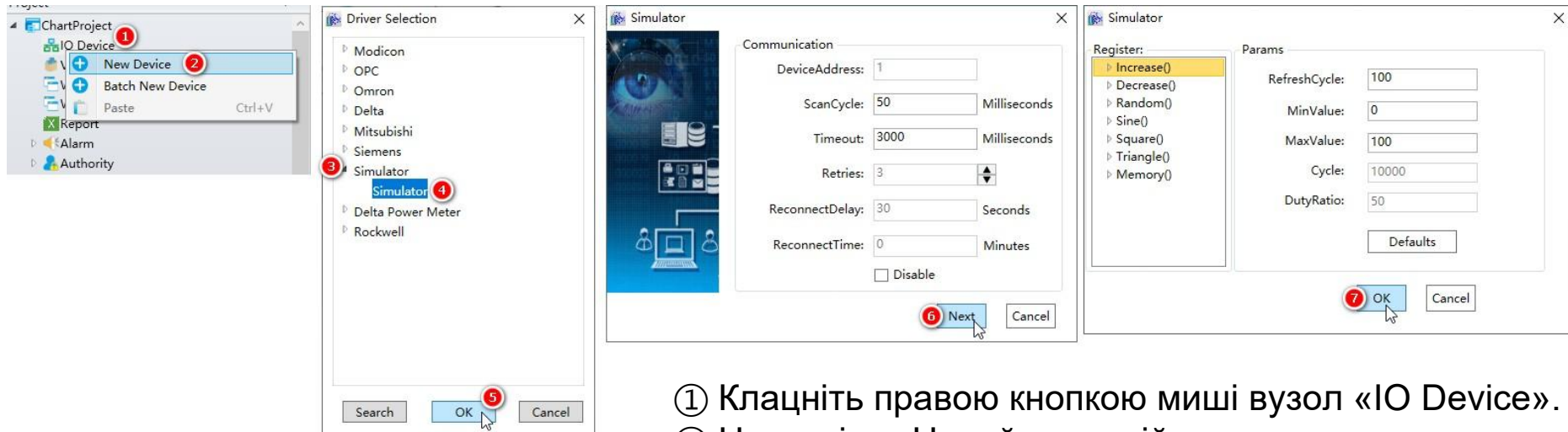
(1) Створіть 2 змінні : Variable0 , Variable



	Name	Variable Type	Initial Value	Retentive Value	Minimum Value	Maximum Value	Deadba
1	Variable0	Analog	0		0	10000	
2	Variable1	Analog	0		0	10000	

※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

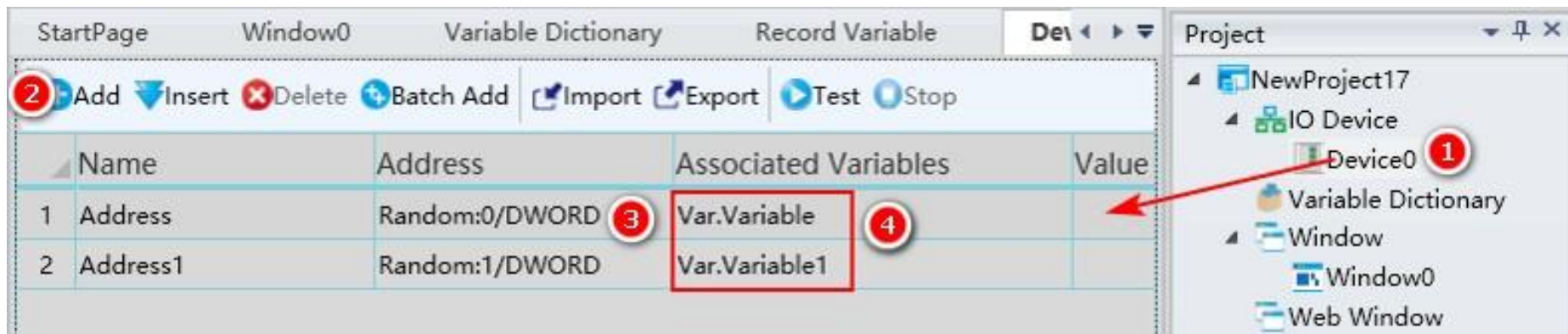
(2) Створіть імітований пристрій : Device0



- ① Клацніть правою кнопкою миші вузол «IO Device».
- ② Натисніть «Новий пристрій»
- ③④ Двічі клацніть «Симулятор»

※Див. розділ «5.10.1 Симулятор» у посібнику користувача.

(3) Створіть дві адреси моделювання в Device0, пов'язані зі змінною, змінною1 відповідно



The screenshot displays the Delta software interface. The 'Variable Dictionary' tab is active, showing a table with the following data:

	Name	Address	Associated Variables	Value
1	Address	Random:0/DWORD	Var.Variable	
2	Address1	Random:1/DWORD	Var.Variable1	

The 'Project' panel on the right shows a tree structure with 'Device0' highlighted. A red arrow points from the 'Device0' node to the 'Var.Variable' cell in the table. Red circles with numbers 1 through 4 indicate the sequence of actions: 1. Select 'Device0' in the Project panel. 2. Click the 'Add' button. 3. Enter the variable name 'Var.Variable' in the 'Associated Variables' column. 4. Enter the variable name 'Var.Variable1' in the 'Associated Variables' column.



(4) Створіть шаблон звіту (RealtimeData) і налаштуйте для нього змінні



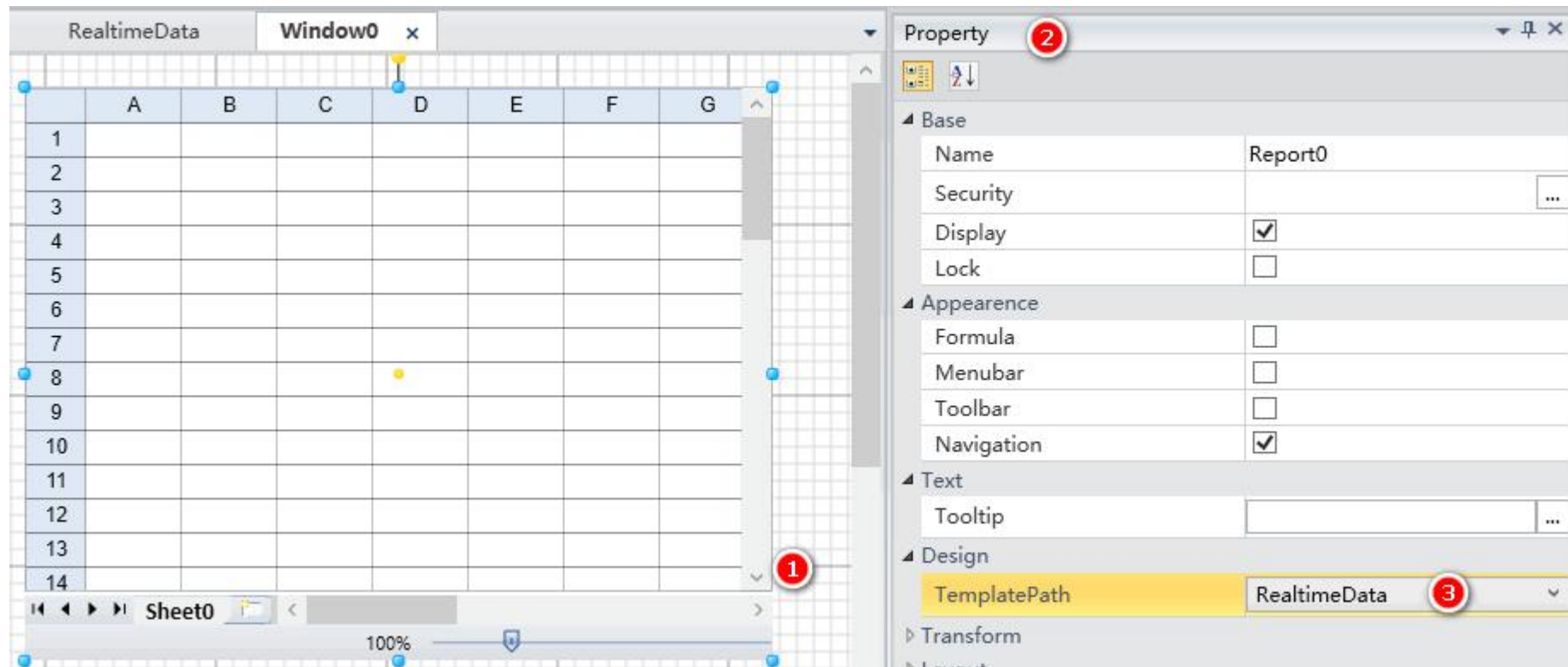
(5) Результат конфігурації RealtimeData такий

StartPage	RealtimeData x				Window0			
	A	B	C	D	E	F	G	H
1	#GetNumV	#GetNumValue("Var.Variable1")						
2								
3								

#GetNumValue("Var.Variable0")

#GetNumValue("Var.Variable1")

(6) Створіть Report0 у Window0 та прив'яжіть шаблон звіту RealtimeData



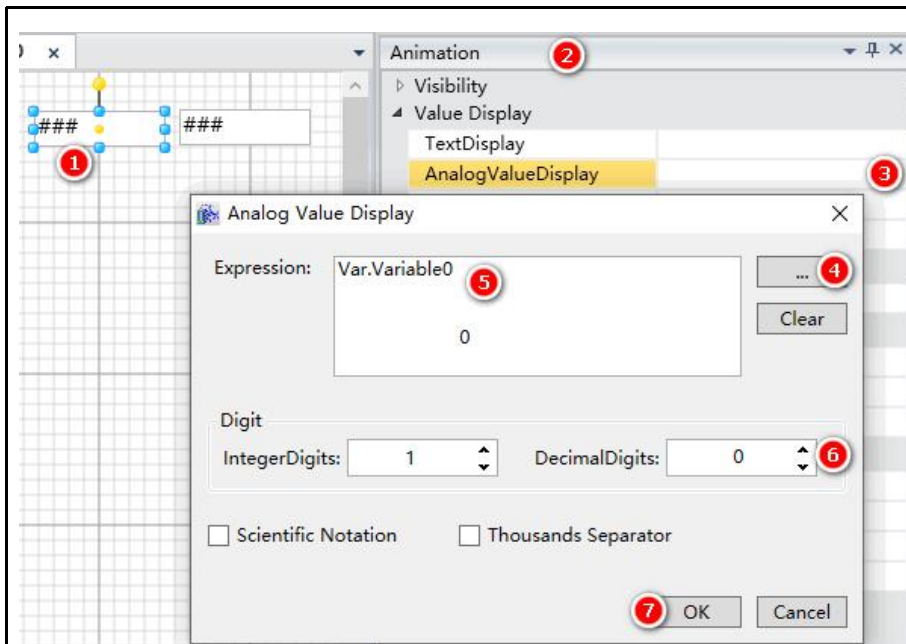
The screenshot displays the software interface with a spreadsheet on the left and a 'Property' window on the right. The spreadsheet has columns A through G and rows 1 through 14. A yellow dot is located in cell D8. The 'Property' window is titled 'Property' and contains several sections: 'Base', 'Appearance', 'Text', and 'Design'. The 'Base' section includes 'Name' (Report0), 'Security', 'Display' (checked), and 'Lock'. The 'Appearance' section includes 'Formula', 'Menubar', 'Toolbar', and 'Navigation' (checked). The 'Text' section includes 'Tooltip'. The 'Design' section includes 'TemplatePath' (RealtimeData). Red circles with numbers 1, 2, and 3 highlight specific elements: 1 points to the bottom right corner of the spreadsheet, 2 points to the 'Property' window title bar, and 3 points to the 'TemplatePath' dropdown menu.

	A	B	C	D	E	F	G
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							

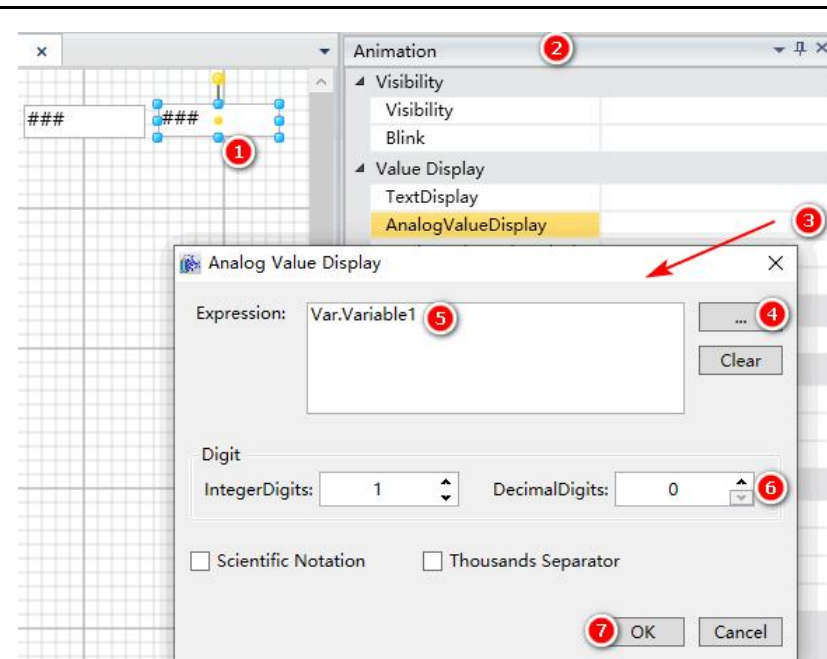
Property

- Base
 - Name: Report0
 - Security: ...
 - Display: ☒
 - Lock: ☐
- Appearance
 - Formula: ☐
 - Menubar: ☐
 - Toolbar: ☐
 - Navigation: ☒
- Text
 - Tooltip: ...
- Design
 - TemplatePath: RealtimeData

(7) Створіть 2 текстові поля (Textbox0 , TextBox1) у Window0 , анімація відображення аналогового значення TextBox0 , TextBox1 пов'язана зі змінною0, змінною1 відповідно

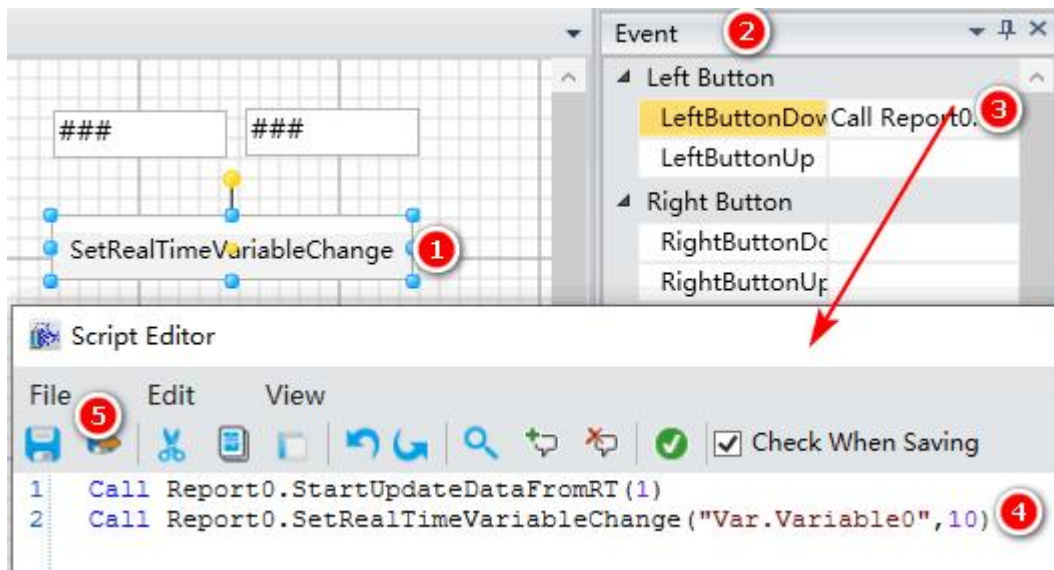


Відображення аналогового значення



Відображення аналогового значення

(8) Створіть кнопку (SetRealTimeVariableChange) у Window0 , налаштуйте Подія кнопки LeftButtonDown



(9) Запустіть Window0, натисніть кнопку « SetRealTimeVariableChange », після чого перші 10 рядків стовпця A та B звіту Report0 відображатимуть значення Variable0 та Variable 1 у реальному часі

	A	B	C	D	E	F	G
1	95	59					
2	86	10					
3	65	40					
4	18	49					
5	83	58					
6	85	45					
7	78	29					
8	37	85					
9							
10							
11							
12							
13							
14							

37 85

SetRealTimeVariableChange

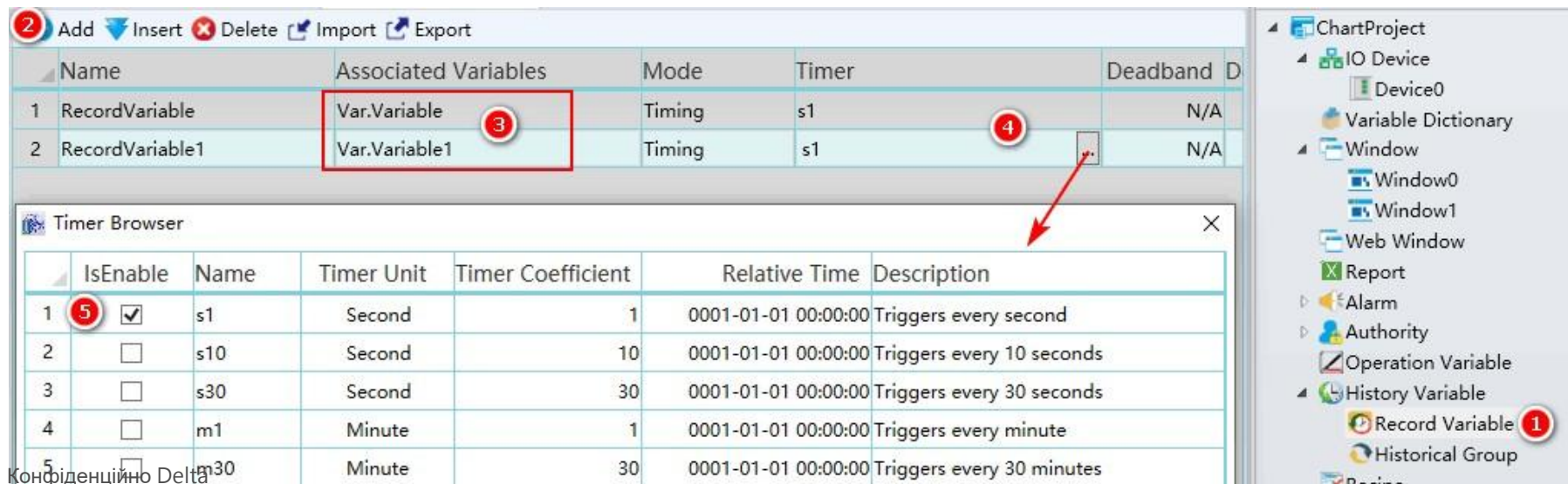
➤ QueryHistoryData, QueryHistoryDataByCommon приклад : звіт

про запит і відображення даних історії

Перші 3 кроки такі самі, як (1)(2)(3) кроки SetRealTimeVariableChange

приклад

(4) Створіть дві історичні змінні в Record Variable , пов'язані з Variable , Variable1 відповідно



The screenshot displays the Delta software interface. At the top, a toolbar includes buttons for 'Add', 'Insert', 'Delete', 'Import', and 'Export'. Below this is a table with columns: 'Name', 'Associated Variables', 'Mode', 'Timer', 'Deadband', and 'D'. The table contains two rows:

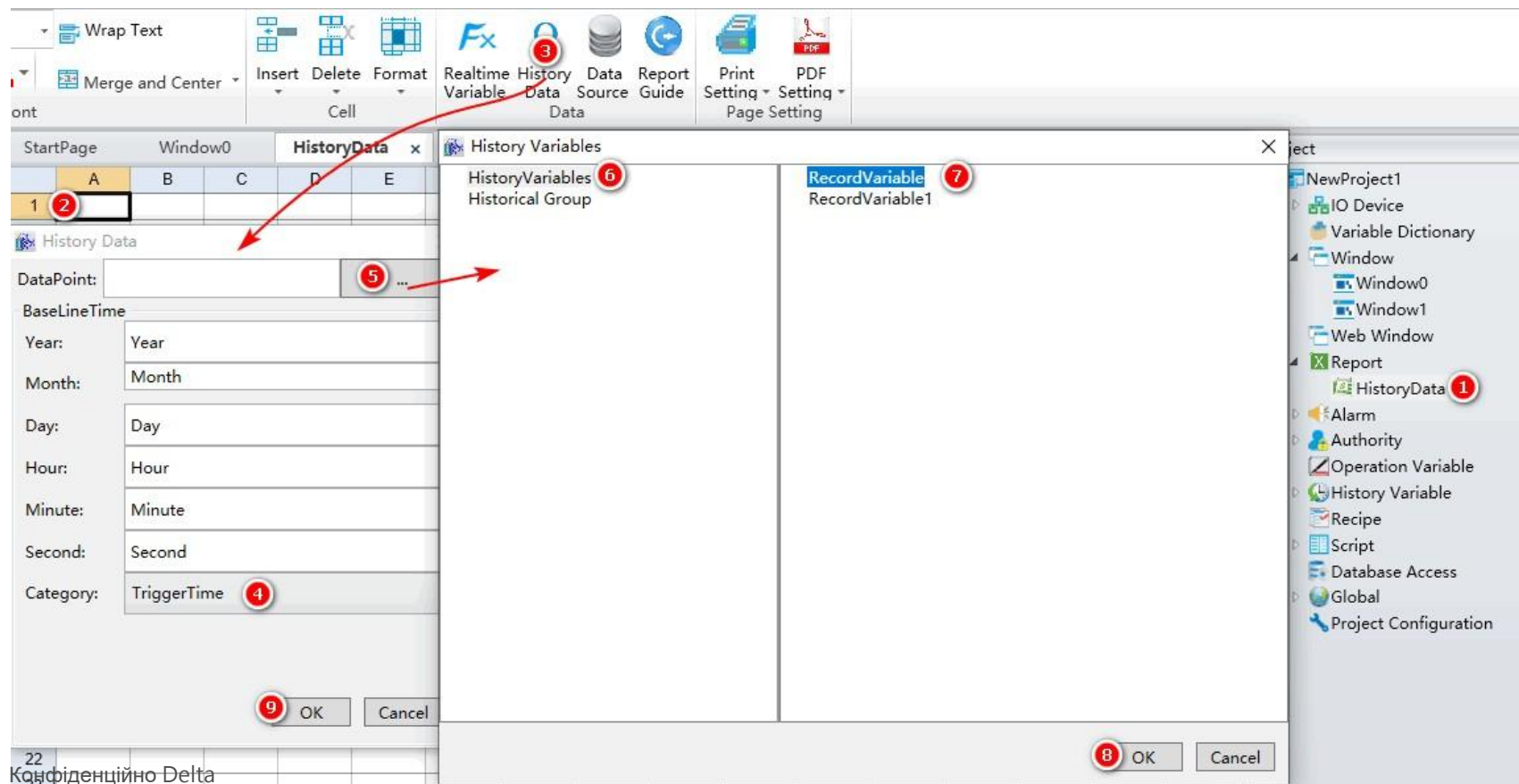
	Name	Associated Variables	Mode	Timer	Deadband	D
1	RecordVariable	Var.Variable	Timing	s1	N/A	
2	RecordVariable1	Var.Variable1	Timing	s1	N/A	

Below the table is a 'Timer Browser' dialog box with a table of timer configurations:

	IsEnable	Name	Timer Unit	Timer Coefficient	Relative Time	Description
1	☒	s1	Second	1	0001-01-01 00:00:00	Triggers every second
2	☐	s10	Second	10	0001-01-01 00:00:00	Triggers every 10 seconds
3	☐	s30	Second	30	0001-01-01 00:00:00	Triggers every 30 seconds
4	☐	m1	Minute	1	0001-01-01 00:00:00	Triggers every minute
5	☐	m30	Minute	30	0001-01-01 00:00:00	Triggers every 30 minutes

On the right side of the interface is a project tree showing the hierarchy: ChartProject > IO Device > Device0 > Variable Dictionary > Window > Window0 > Window1 > Web Window > Report > Alarm > Authority > Operation Variable > History Variable > Record Variable > Historical Group > Device0.

(5) Створіть шаблон звіту (HistoryData) і налаштуйте для нього змінні історії

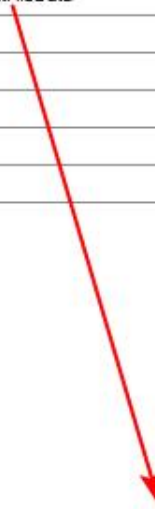


The screenshot shows the Delta software interface with the 'HistoryData' report template selected. The 'History Variables' dialog box is open, allowing configuration of historical data variables. Red numbered circles (1-9) indicate the following steps:

1. Select 'HistoryData' in the Project tree.
2. Click on the 'HistoryData' tab in the worksheet.
3. Click on the 'History Variables' button in the toolbar.
4. Select 'TriggerTime' as the Category.
5. Click on the 'DataPoint' field.
6. Select 'HistoryVariables' as the Historical Group.
7. Select 'RecordVariable' as the Record Variable.
8. Click the 'OK' button in the 'History Variables' dialog.
9. Click the 'OK' button in the main configuration window.

(6) Результат конфігурації HistoryData такий

StartPage	Window0	HistoryData x											
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	#GetHisData	#GetHisDa	#GetHisData("VarRecord.RecordVariable1,Year,Month,Day,Hour,Minute,Second,Value")										
2													
3													
4													
5													
6													

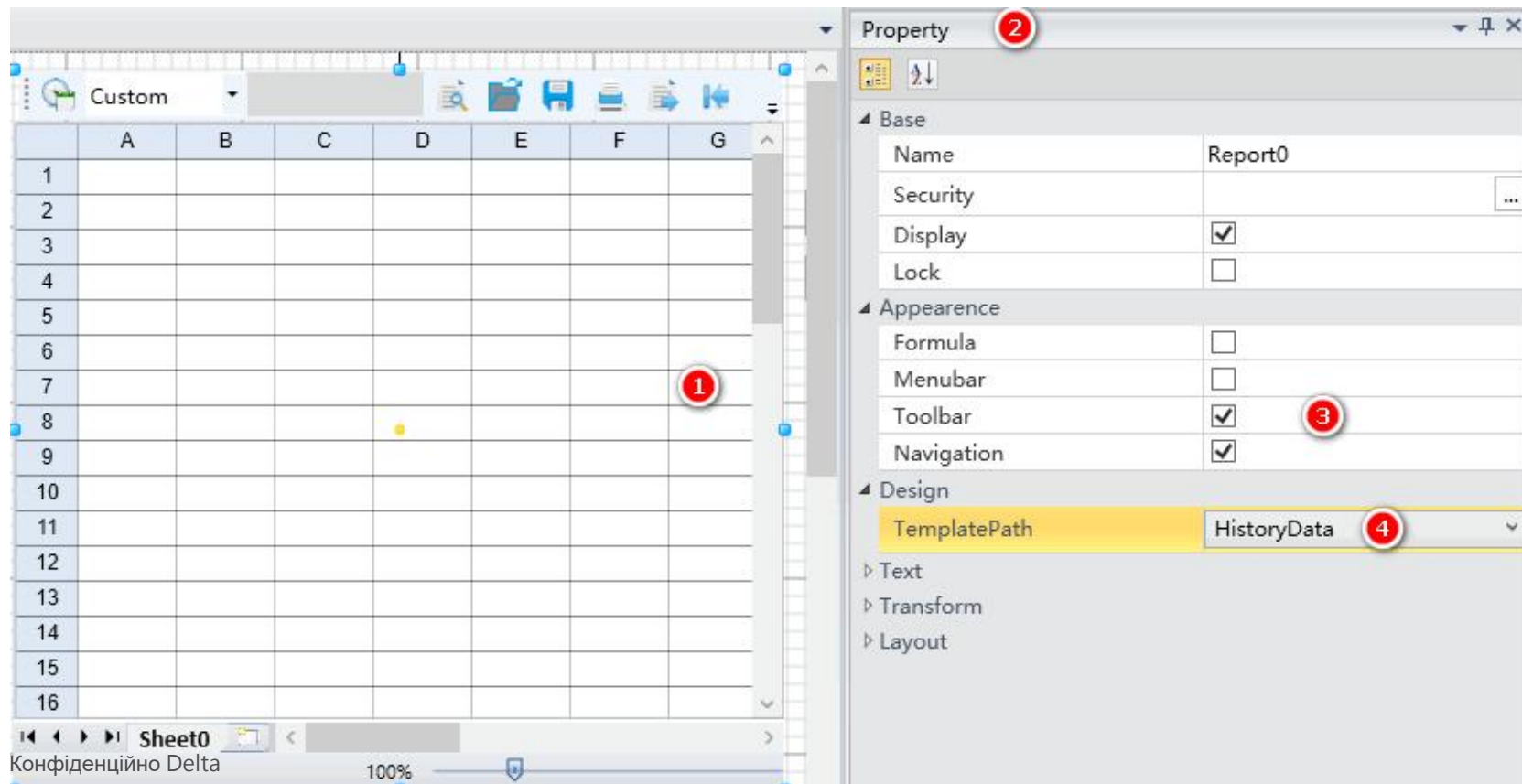


#GetHisData("VarRecord.RecordVariable1,Year,Month,Day,Hour,Minute,Second,Value")

#GetHisData("VarRecord.RecordVariable,Year,Month,Day,Hour,Minute,Second,Value")

#GetHisData("VarRecord.RecordVariable,Year,Month,Day,Hour,Minute,Second,TriggerTime")

(7) Створіть Report0 у Window0 та прив'яжіть шаблон звіту HistoryData



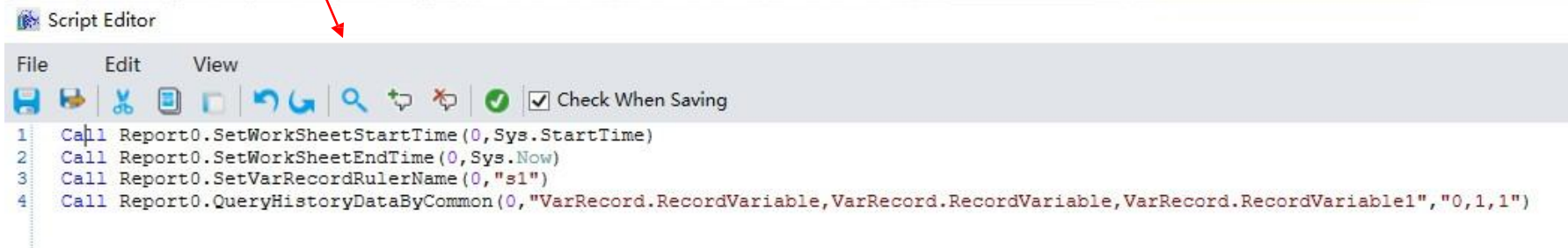
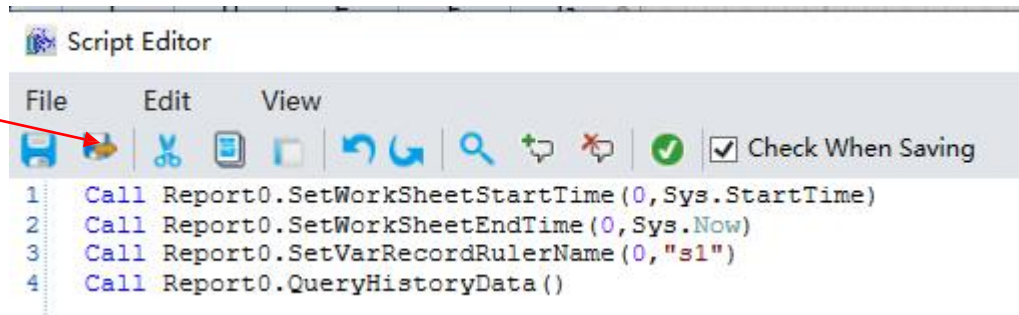
The screenshot displays the Delta software interface. On the left is a spreadsheet with columns A through G and rows 1 through 16. A red circle with the number 1 is positioned over the intersection of column D and row 8. On the right is the 'Property' window, which is divided into several sections. A red circle with the number 2 is over the 'Property' title bar. The 'Base' section contains fields for 'Name' (set to 'Report0'), 'Security', 'Display' (checked), and 'Lock' (unchecked). The 'Appearance' section contains fields for 'Formula' (unchecked), 'Menubar' (unchecked), 'Toolbar' (checked), and 'Navigation' (checked). A red circle with the number 3 is over the 'Toolbar' field. The 'Design' section contains a 'TemplatePath' field set to 'HistoryData', with a red circle with the number 4 over the dropdown arrow. Below the 'Design' section are expandable sections for 'Text', 'Transform', and 'Layout'. The bottom status bar shows 'Sheet0', 'Конфіденційно Delta', and a 100% zoom level.

	A	B	C	D	E	F	G
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							

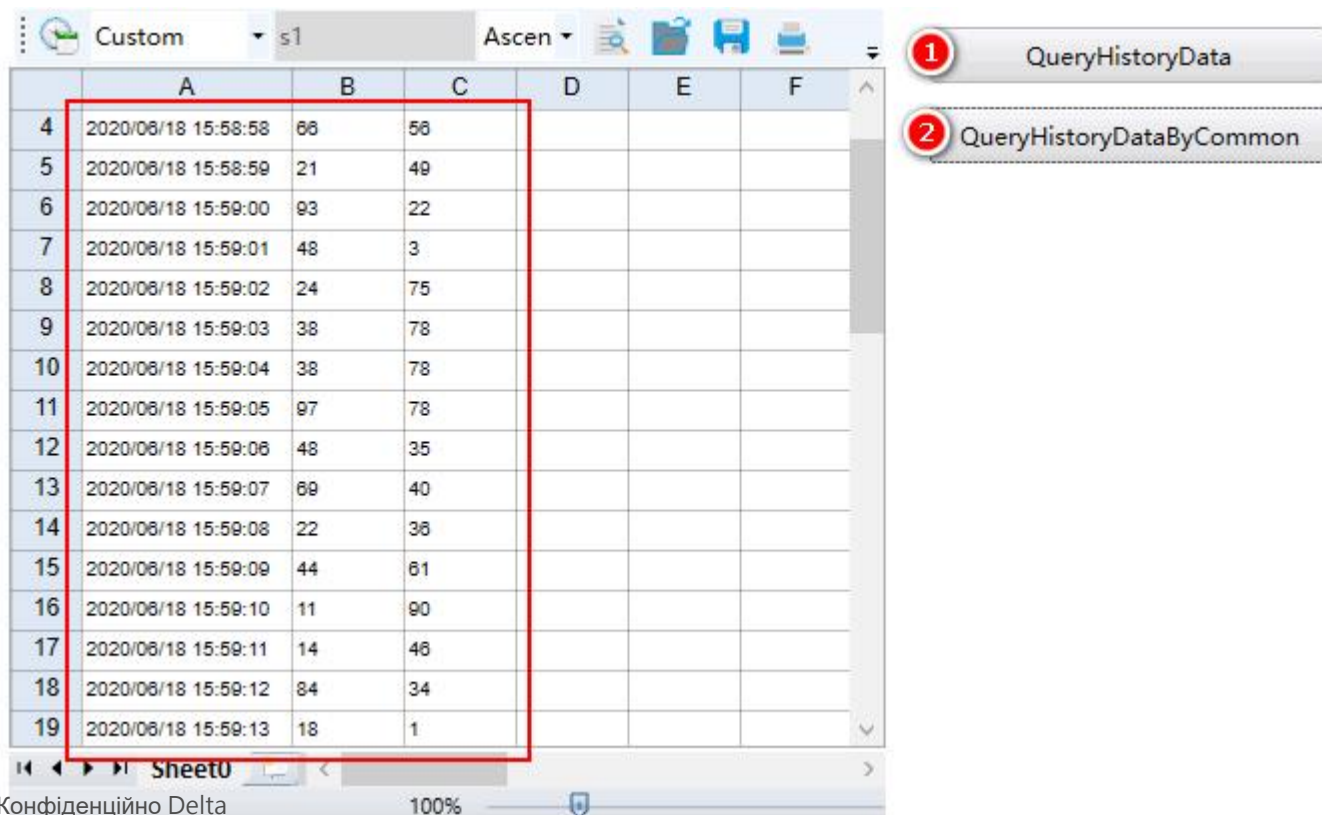
Property

- Base**
 - Name: Report0
 - Security: ...
 - Display: ☒
 - Lock: ☐
- Appearance**
 - Formula: ☐
 - Menubar: ☐
 - Toolbar: ☒
 - Navigation: ☒
- Design**
 - TemplatePath: HistoryData
- Text
- Transform
- Layout

(8) Створіть дві кнопки у Window0 , налаштуйте подію LeftButtonDown для дві кнопки



(9) Запустіть Window0 , клацніть кнопку « QueryHistoryData » або « QueryHistoryDataByCommon », тоді Report0 відобразить дані історії від часу початку до поточного часу



The screenshot displays a software window with a data table and two buttons. The table has columns A, B, and C, and rows 4 through 19. A red rectangle highlights the data in columns A, B, and C for rows 4 through 19. To the right of the table are two buttons: 'QueryHistoryData' (labeled 1) and 'QueryHistoryDataByCommon' (labeled 2). The 'QueryHistoryDataByCommon' button is dashed, indicating it is not active. The bottom of the window shows 'Sheet0' and a status bar with 'Конфіденційно Delta' and '100%'.

	A	B	C	D	E	F
4	2020/06/18 15:58:58	66	56			
5	2020/06/18 15:58:59	21	49			
6	2020/06/18 15:59:00	93	22			
7	2020/06/18 15:59:01	48	3			
8	2020/06/18 15:59:02	24	75			
9	2020/06/18 15:59:03	38	78			
10	2020/06/18 15:59:04	38	78			
11	2020/06/18 15:59:05	97	78			
12	2020/06/18 15:59:06	48	35			
13	2020/06/18 15:59:07	69	40			
14	2020/06/18 15:59:08	22	36			
15	2020/06/18 15:59:09	44	61			
16	2020/06/18 15:59:10	11	90			
17	2020/06/18 15:59:11	14	46			
18	2020/06/18 15:59:12	84	34			
19	2020/06/18 15:59:13	18	1			

1 QueryHistoryData

2 QueryHistoryDataByCommon

Sheet0

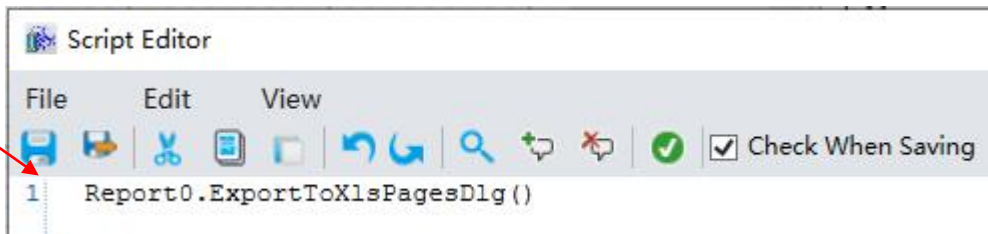
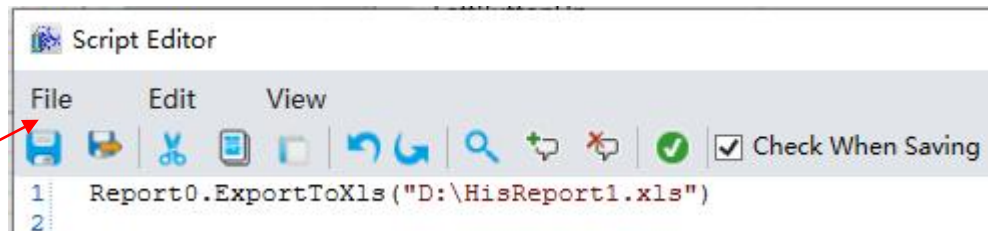
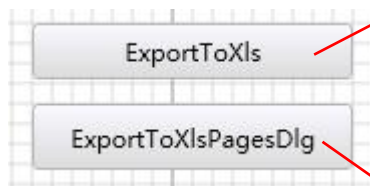
Конфіденційно Delta 100%

➤ **ExportToXls, ExportToXlsPagesDlg** приклад :

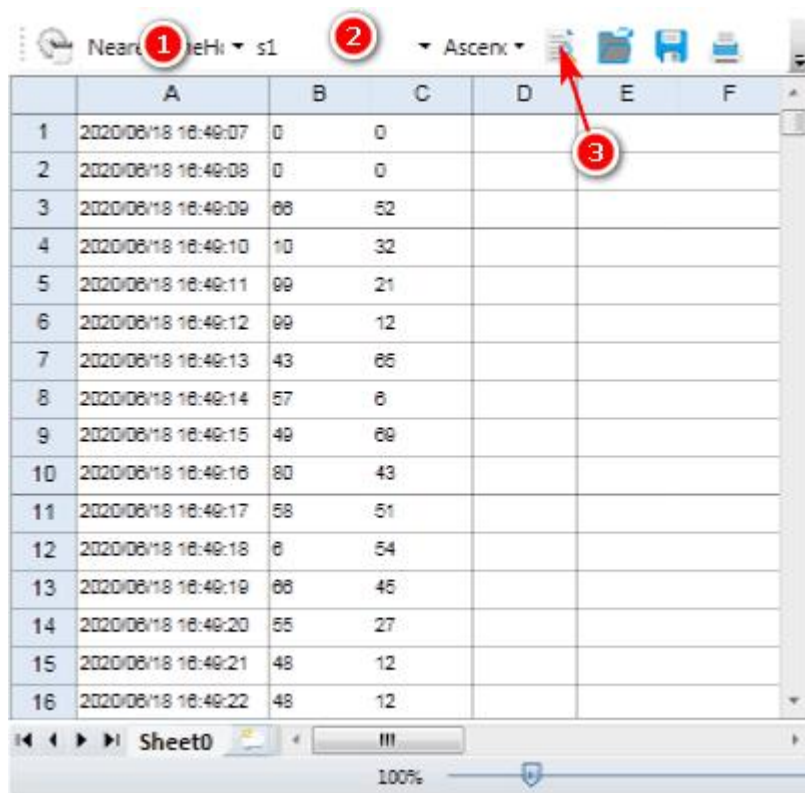
Експорт даних історії, які відображаються у звіті

Перші 7 кроків такі самі, як перші 7 кроків **QueryHistoryData**,
QueryHistoryDataByCommon приклад

(8) Створіть дві кнопки у Window0 , налаштуйте подію LeftButtonDown для дві кнопки



(9) Запустіть Window0, дані історії запитів за допомогою кнопок панелі інструментів

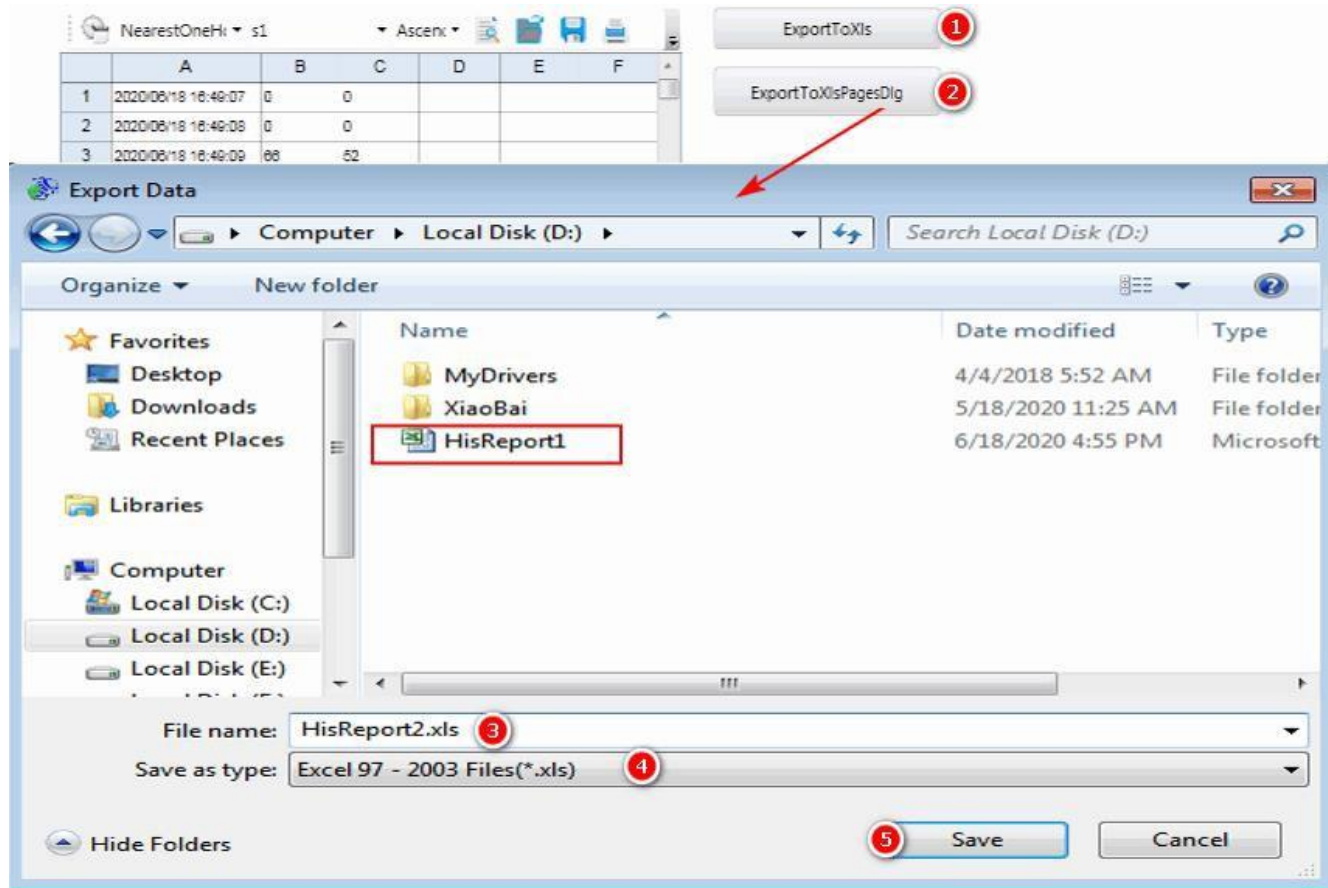


	A	B	C	D	E	F
1	2020/08/18 18:49:07	0	0			
2	2020/08/18 18:49:08	0	0			
3	2020/08/18 18:49:09	68	52			
4	2020/08/18 18:49:10	10	32			
5	2020/08/18 18:49:11	99	21			
6	2020/08/18 18:49:12	99	12			
7	2020/08/18 18:49:13	43	66			
8	2020/08/18 18:49:14	57	6			
9	2020/08/18 18:49:15	49	69			
10	2020/08/18 18:49:16	80	43			
11	2020/08/18 18:49:17	58	51			
12	2020/08/18 18:49:18	6	54			
13	2020/08/18 18:49:19	68	45			
14	2020/08/18 18:49:20	55	27			
15	2020/08/18 18:49:21	48	12			
16	2020/08/18 18:49:22	48	12			

(10) Виконайте сценарії

① Натисніть кнопку « ExportToXls », дані у Report0 експортуються у файл Excel на диску D, ім'я файлу HisReport1

② Натисніть « ExportDataToXlsByDialog », спливаюче діалогове вікно експорту даних, як показано праворуч



➤ **ImportDataToControl, ImportDataToControlByDialog** приклад :

Імпортуйте зовнішній файл Excel до Report0

Перші 7 кроків такі самі, як перші 7 кроків **QueryHistoryData, QueryHistoryDataByCommon** приклад

(8) Створіть дві кнопки у Window0, налаштуйте подію LeftButtonDown для двох кнопок



Script Editor



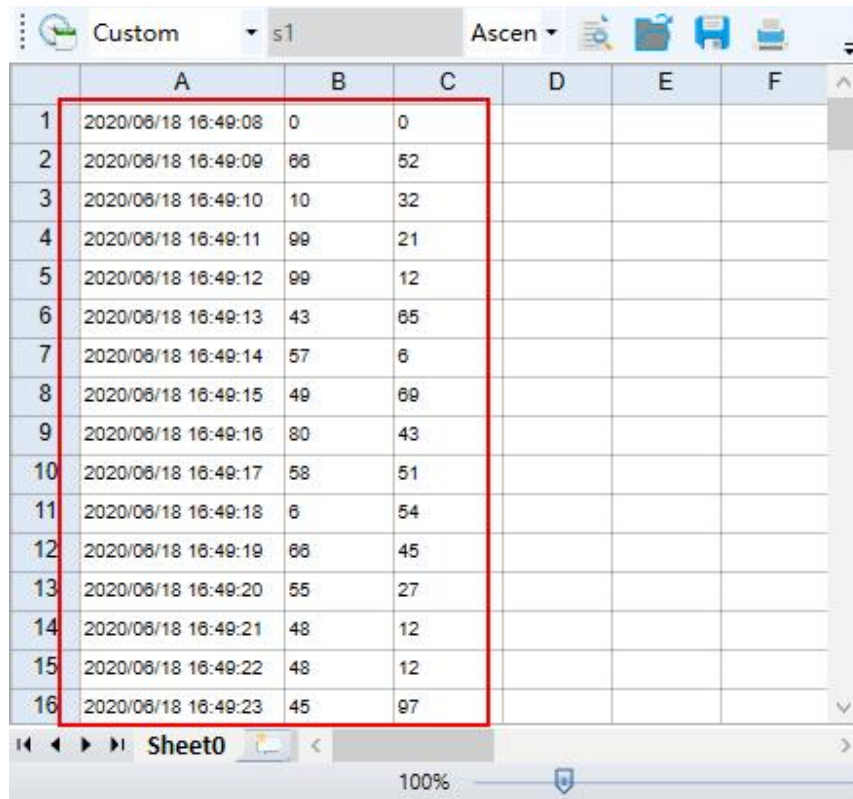
```
1 call Report0.ClearCellContent()  
2 'clear the contents of the Report0  
3 call Report0.ImportDataToControl("D:\Hisreport1.xls")  
4 'Import external Excel file(D:\Hisreport1.xls) to Report0
```

Script Editor

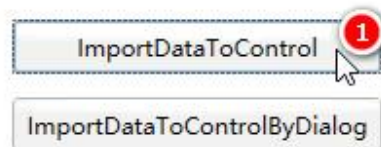


```
1 call Report0.ClearCellContent()  
2 'clear the contents of the Report0  
3 call Report0.ImportDataToControlByDialog()  
4 'Select the Excel file to be imported through the pop-up dialog
```

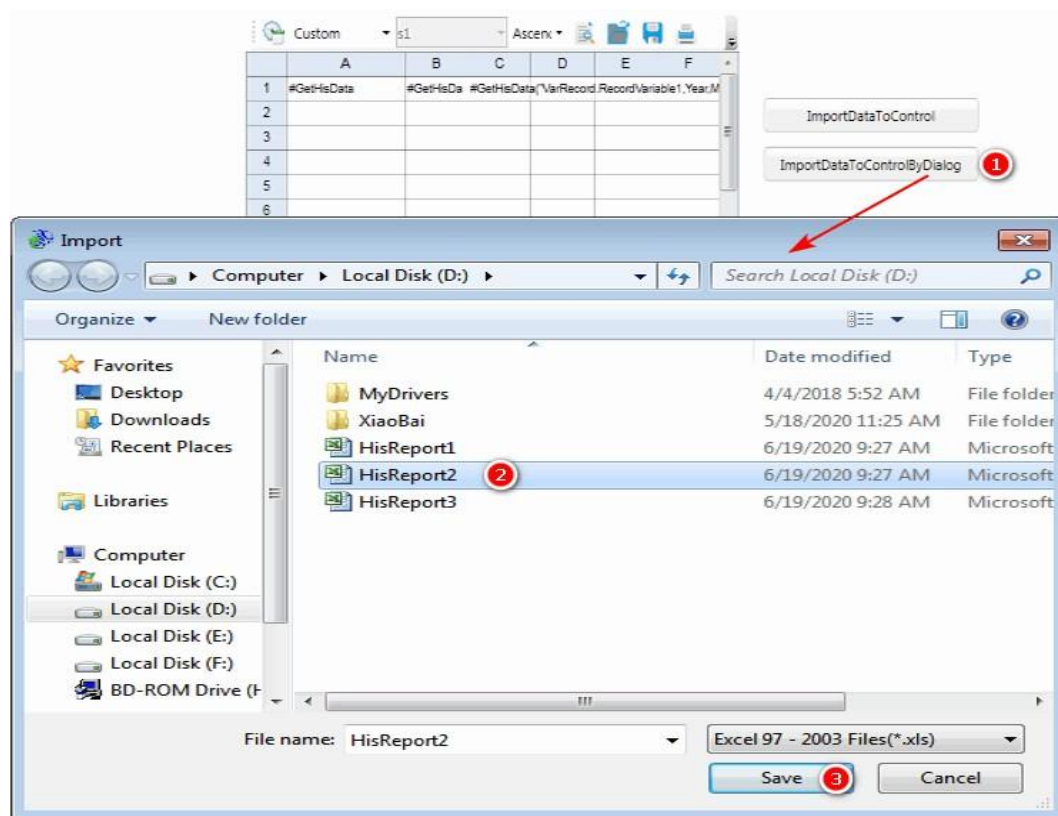

(9) Запустіть Window0 , натисніть « ImportDataToControl » , щоб імпортувати зовнішній файл Excel (D:\Hisreport1.xls) до Report0



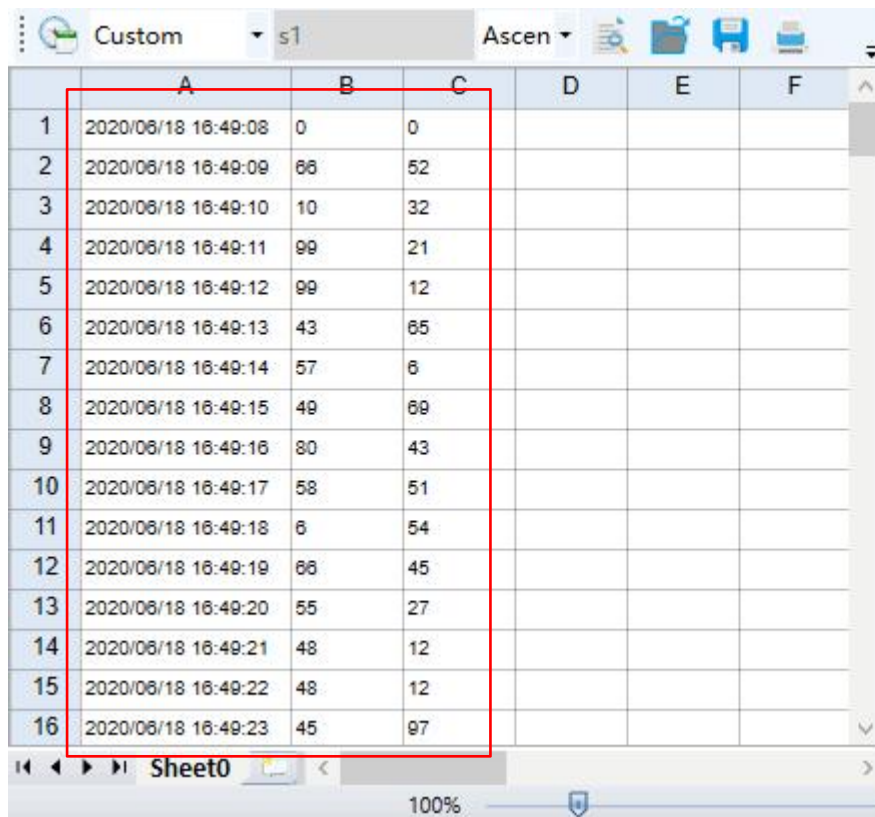
	A	B	C	D	E	F
1	2020/06/18 16:49:08	0	0			
2	2020/06/18 16:49:09	66	52			
3	2020/06/18 16:49:10	10	32			
4	2020/06/18 16:49:11	99	21			
5	2020/06/18 16:49:12	99	12			
6	2020/06/18 16:49:13	43	65			
7	2020/06/18 16:49:14	57	6			
8	2020/06/18 16:49:15	49	69			
9	2020/06/18 16:49:16	80	43			
10	2020/06/18 16:49:17	58	51			
11	2020/06/18 16:49:18	6	54			
12	2020/06/18 16:49:19	66	45			
13	2020/06/18 16:49:20	55	27			
14	2020/06/18 16:49:21	48	12			
15	2020/06/18 16:49:22	48	12			
16	2020/06/18 16:49:23	45	97			



(10) Натисніть кнопку « ImportDataToControlByDialog » , відкрийте діалогове вікно імпорту, щоб вибрати файл Excel (D:\Hisreport2.xls) для імпорту в Report0, як показано НИЖЧЕ



(11) Зовнішній файл Excel (D:\Hisreport2.xls) успішно імпортовано до Report0



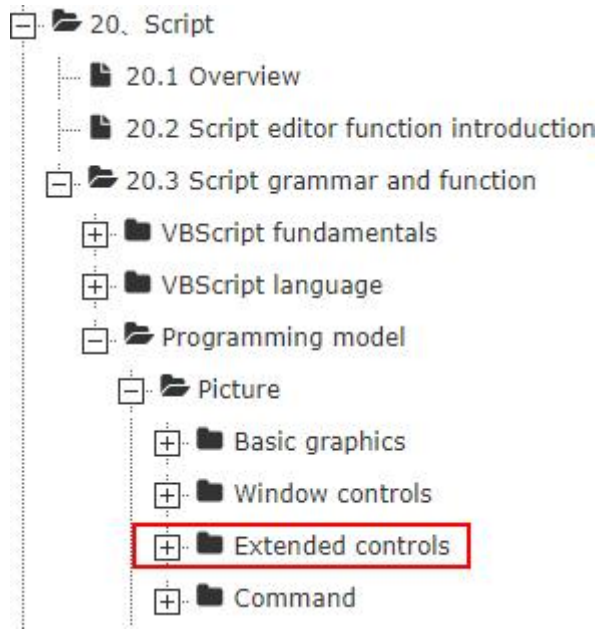
	A	B	C	D	E	F
1	2020/06/18 16:49:08	0	0			
2	2020/06/18 16:49:09	66	52			
3	2020/06/18 16:49:10	10	32			
4	2020/06/18 16:49:11	99	21			
5	2020/06/18 16:49:12	99	12			
6	2020/06/18 16:49:13	43	65			
7	2020/06/18 16:49:14	57	6			
8	2020/06/18 16:49:15	49	69			
9	2020/06/18 16:49:16	80	43			
10	2020/06/18 16:49:17	58	51			
11	2020/06/18 16:49:18	6	54			
12	2020/06/18 16:49:19	66	45			
13	2020/06/18 16:49:20	55	27			
14	2020/06/18 16:49:21	48	12			
15	2020/06/18 16:49:22	48	12			
16	2020/06/18 16:49:23	45	97			

ImportDataToControl

ImportDataToControlByDialog

Сценарії розширення елементів керування

Щоб отримати докладніші відомості про використання скриптів для розширених елементів керування, зверніться до розділу « 20.3 Граматика та функції сценаріїв » посібника користувача. Як показано на малюнку нижче:

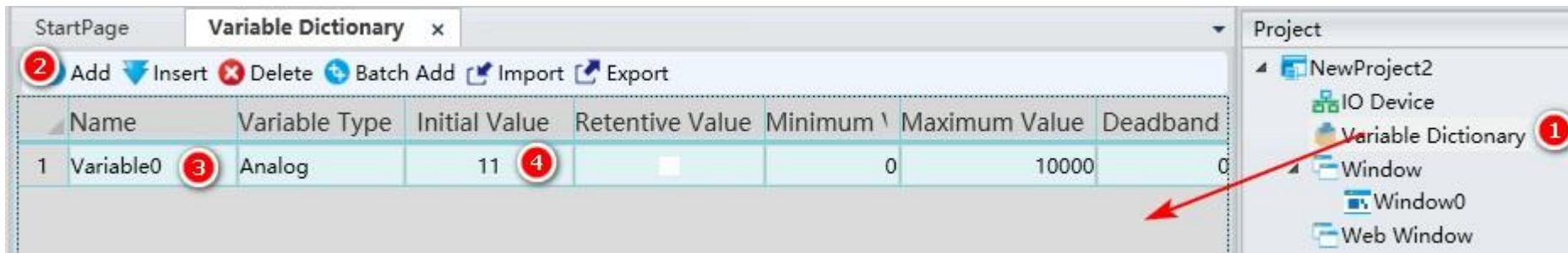


- Поняття сценаріїв
- Скрипти базової графіки
- Сценарії елементів керування вікном
- Сценарії Extend Controls
- Скрипти дій
- Сценарії вікна
- Кольорові сценарії _

➤ AnalogValueInput приклад :

Змінити значення змінної (1)

Створити змінну : Змінна0

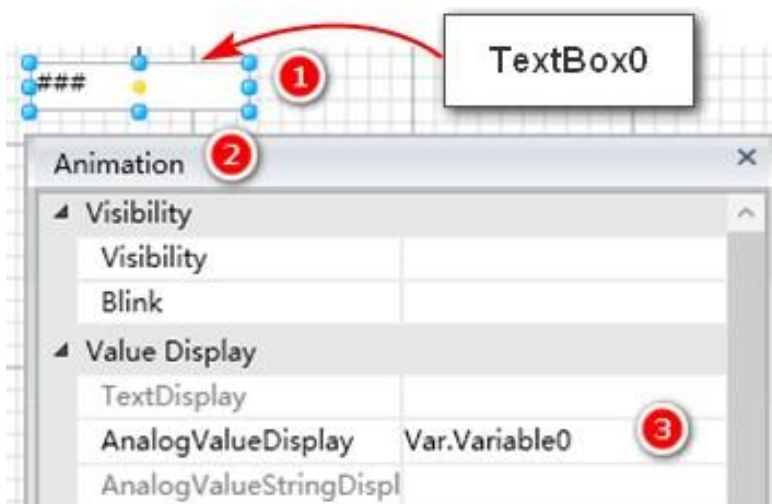


	Name	Variable Type	Initial Value	Retentive Value	Minimum Value	Maximum Value	Deadband
1	Variable0	Analog	11		0	10000	0

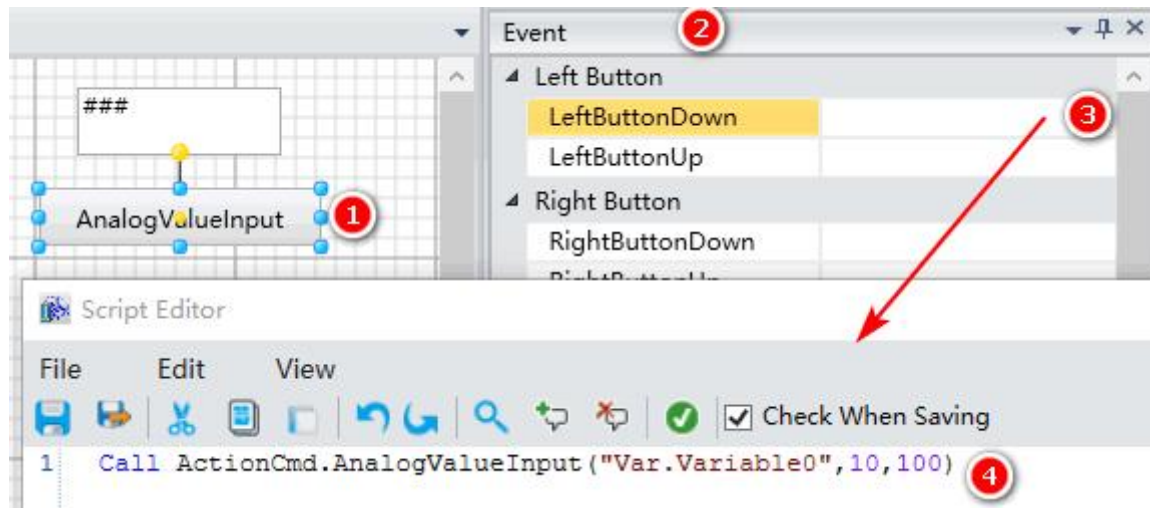
※Зверніться до розділу «6.3 Змінні» в посібнику користувача.

(2) Створіть текстове поле (Textbox0) у Window0 , анімацію відображення аналогового значення

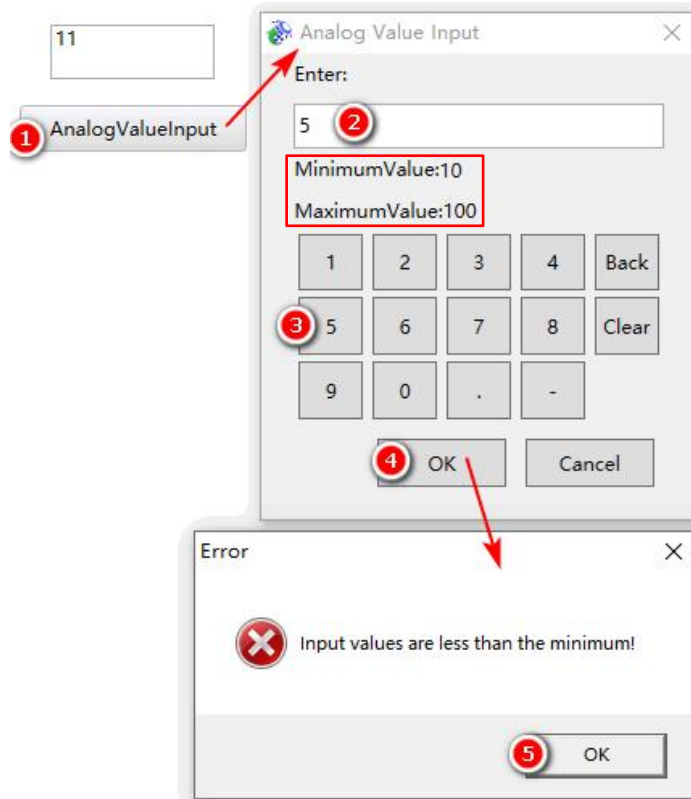
TextBox0 асоціюється зі змінною0



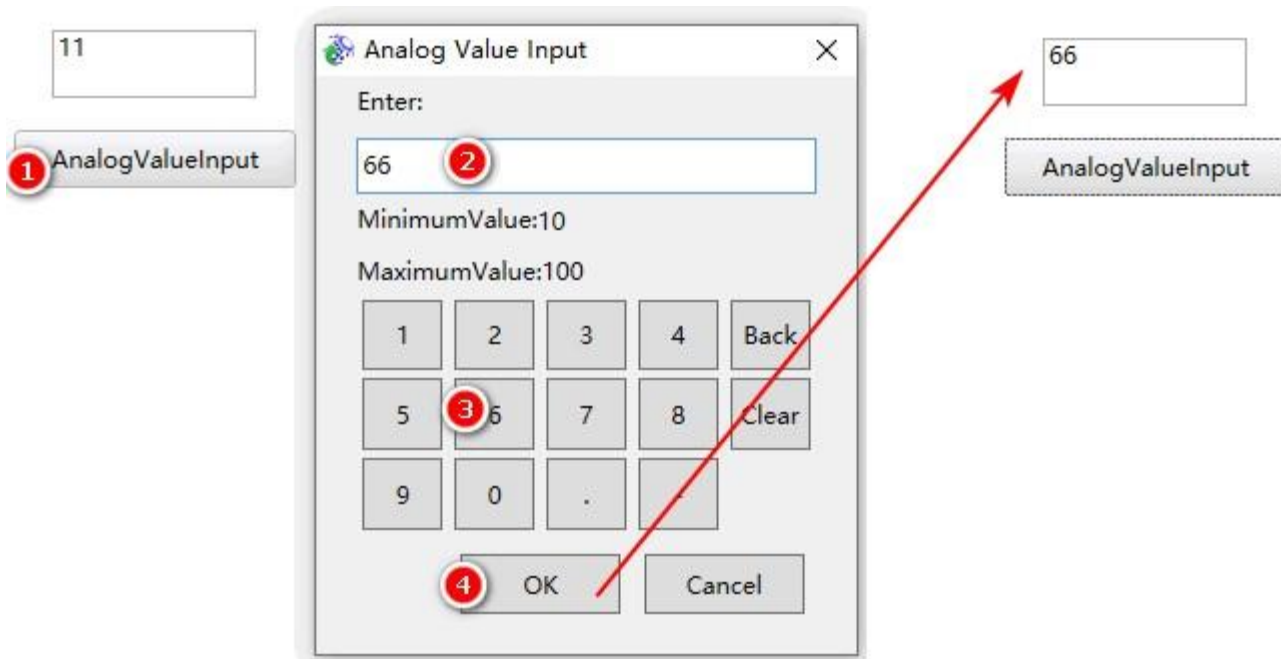
(3) Створіть кнопку (AnalogValueInput) у Window0 , налаштуйте подію LeftButtonDown кнопки



(4) Запустіть Window0 , натисніть кнопку « AnalogValueInput », відкрийте діалогове вікно введення аналогового значення, введіть число менше 10 або більше 100, помилка введення



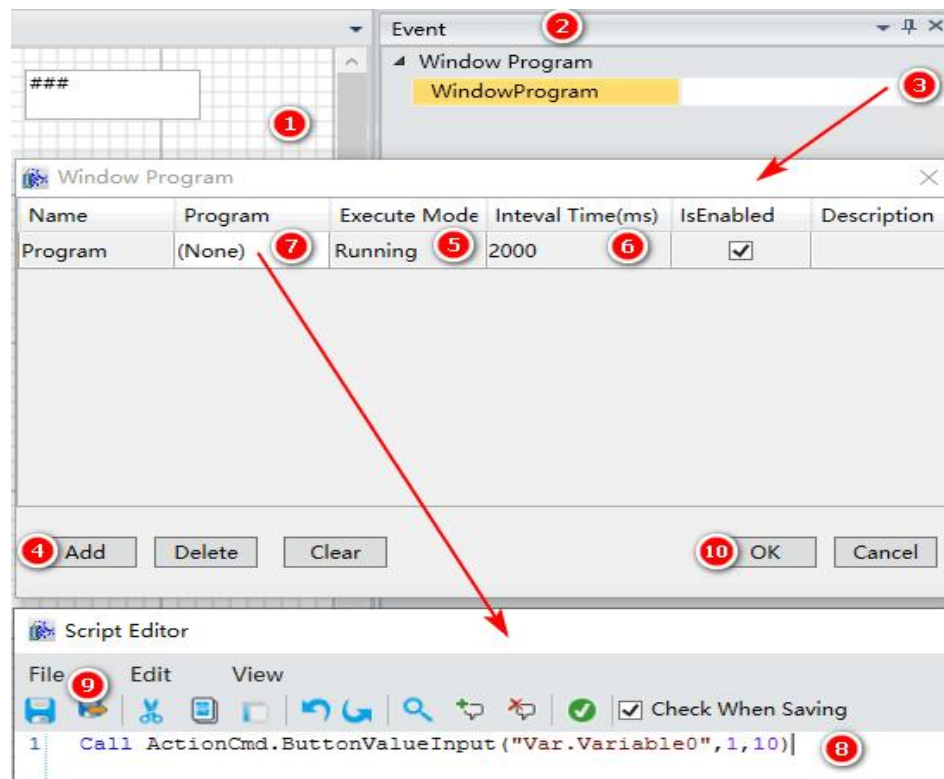
(5) Запустіть Window0, натисніть кнопку « AnalogValueInput », відкрийте діалогове вікно введення аналогового значення, введіть число від 10 до 100, введіть успішно



➤ **ButtonValueInput** приклад :

Змінна збільшується на 10 кожні 2 секунди

Перші 2 кроки такі самі, як перші 2 кроки **AnalogValueInput** приклад (3) Створіть віконну програму в вікно0 _



① Клацніть будь-який порожній пункт у Window0

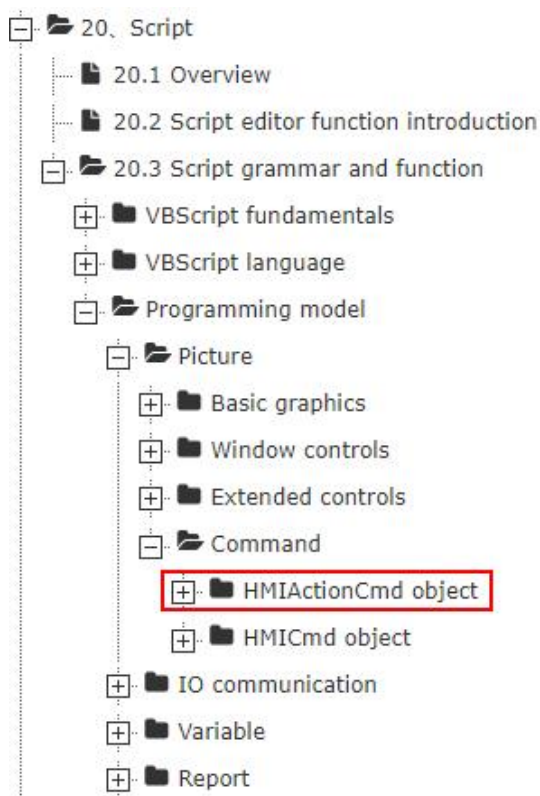
⑤ Встановіть цю програму виконується під час виконання

⑥ Встановіть цю програму на виконання кожні 2 секунди

(4) Запустіть Window0 , на початку Textbox0 відображає 11, а потім значення відображення TextBox0 збільшується на 10 кожні 2 секунди

A rectangular text box with a red border, containing the number "61".

Щоб отримати докладніші відомості про використання сценаріїв дій, зверніться до розділу « 20.3 Граматика та функції сценаріїв » посібника користувача. Як показано на малюнку нижче:



- Поняття сценаріїв
- Скрипти базової графіки
- Сценарії елементів керування вікном
- Сценарії Extend Controls
- Скрипти дій
- Сценарії вікна
- Кольорові сценарії _

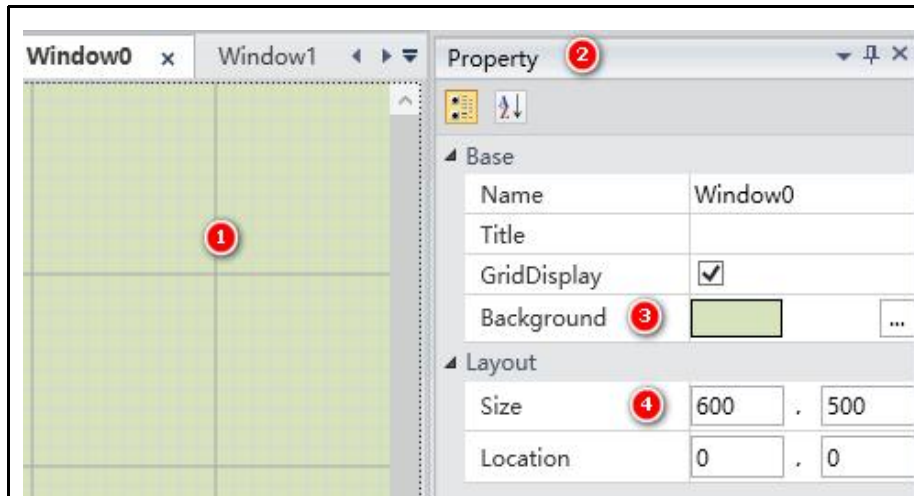
➤ OpenDialogWindow, CloseDialogWindow приклад :

Відкрийте діалогове вікно, а потім закрийте його
(1) Створіть 2 вікна (Window0, Window1) у проекті.

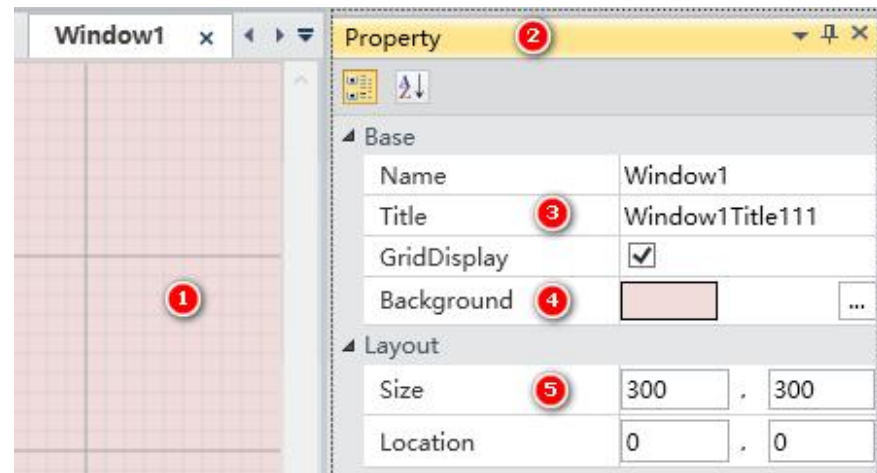


※Див. розділ «7.2.1.1 Додати вікно» в посібнику користувача.

(2) Встановити властивості Window0 , Window1

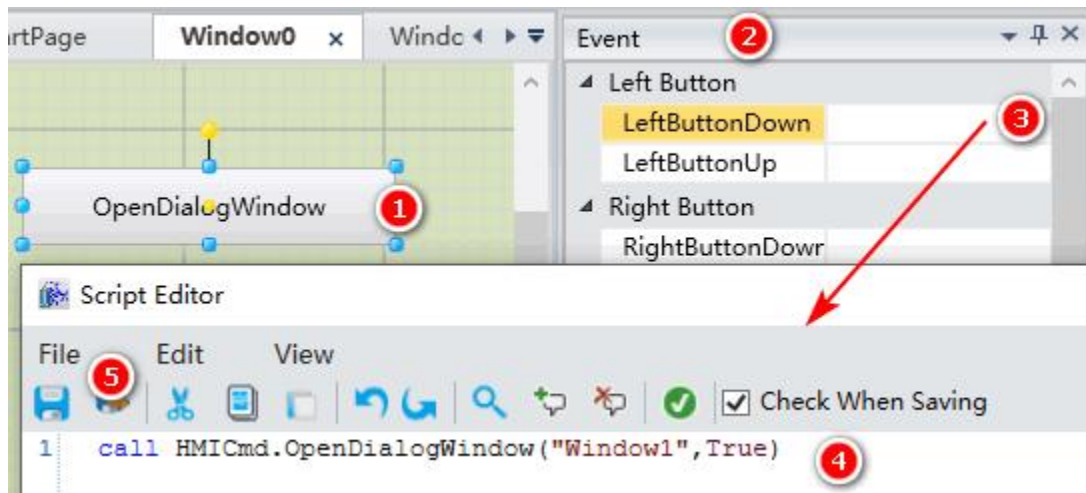


Вікно0

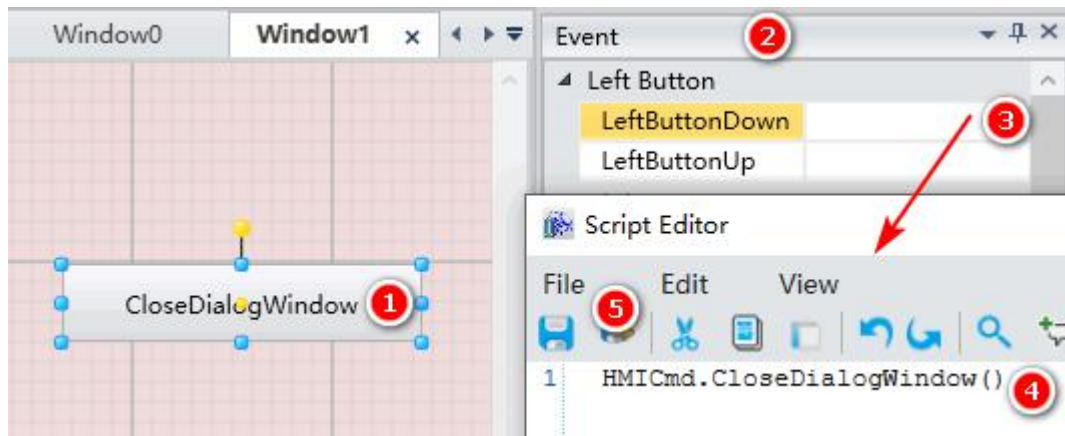


Вікно1

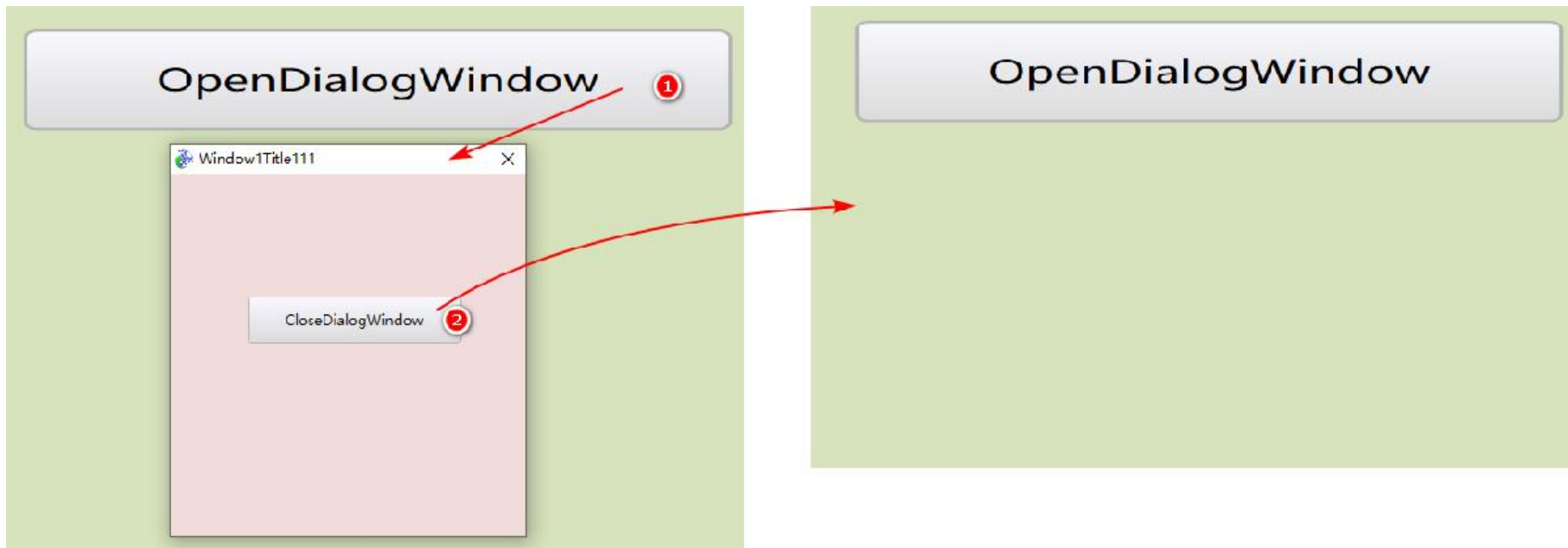
(3) Створіть кнопку (OpenDialogWindow) у Window0 , налаштуйте подію LeftButtonDown кнопки



(4) Створіть кнопку (CloseDialogWindow) у Window1 , налаштуйте подію LeftButtonDown кнопки



(5) Запустіть Window0



- ① Натисніть кнопку « OpenDialogWindow » у вікні Window0 , відкриється діалогове вікно Window1 із назвою «Window1Title111»
- ② Натисніть кнопку « CloseDialogWindow » у діалоговому вікні Window1 , тоді діалогове вікно Window1 закриється

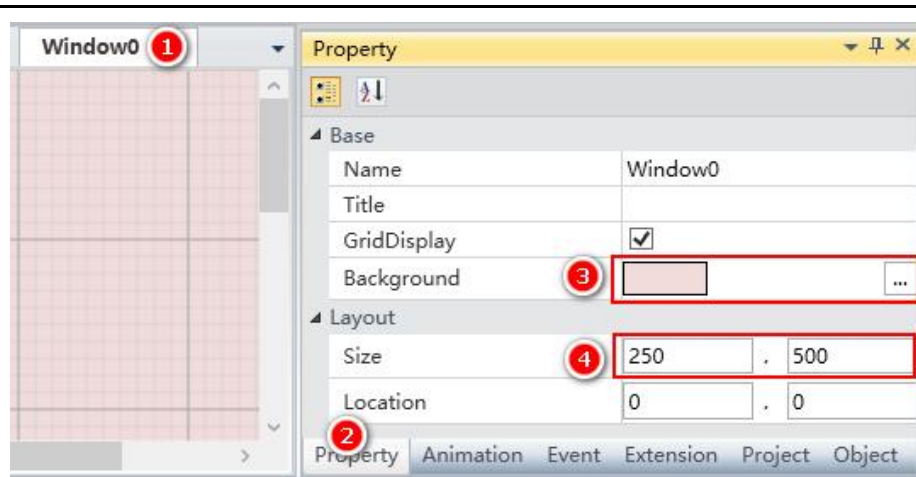
➤ OpenWindowAndCloseOther приклад :

(1) Створіть два вікна (Window0 , Window1) у проекті

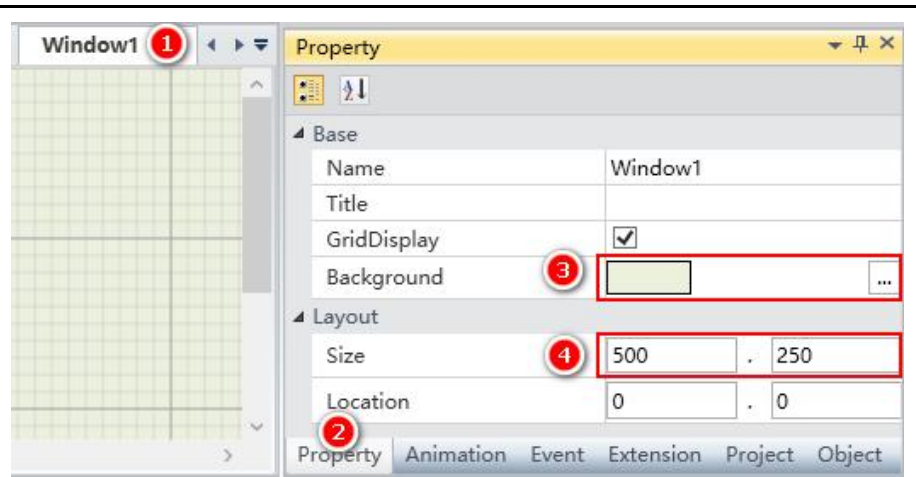


※Див. розділ «7.2.1.1 Додати вікно» в посібнику користувача.

(2) Встановіть властивості Window0 , Window1



Вікно0



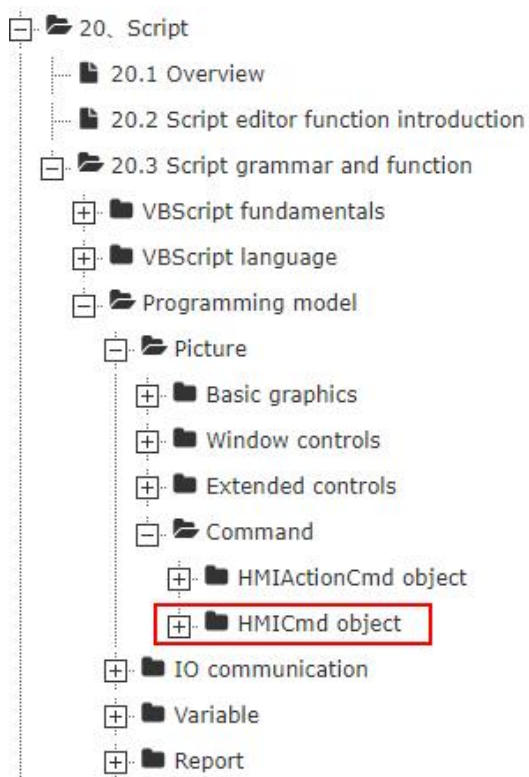
Вікно1

(3) Створіть кнопку (OpenWindowAndCloseOther) у Window0 ,
налаштуйте подію LeftButtonDown кнопки



(4) Запускайте лише Window0 за замовчуванням. Потім натисніть кнопку «
OpenWindowAndCloseOther»,
Вікно1 відкрито, Вікно0 закрито . Запущено лише Window1

Щоб отримати докладніші відомості про використання віконних сценаріїв, зверніться до розділу « 20.3 Граматика та функції сценаріїв » посібника користувача. Як показано на малюнку нижче:

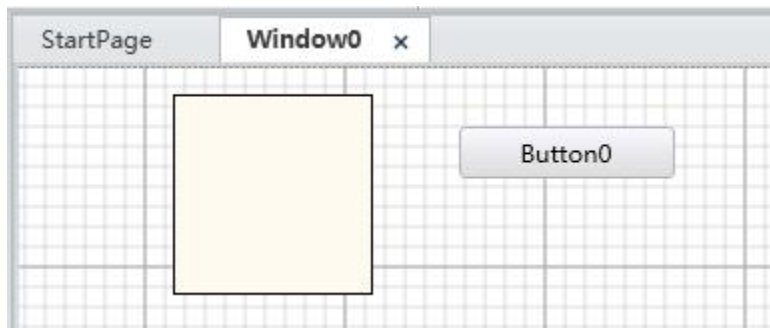


- Поняття сценаріїв
- Скрипти базової графіки
- Сценарії елементів керування вікном
- Сценарії Extend Controls
- Скрипти дій
- Сценарії вікна
- Кольорові сценарії _

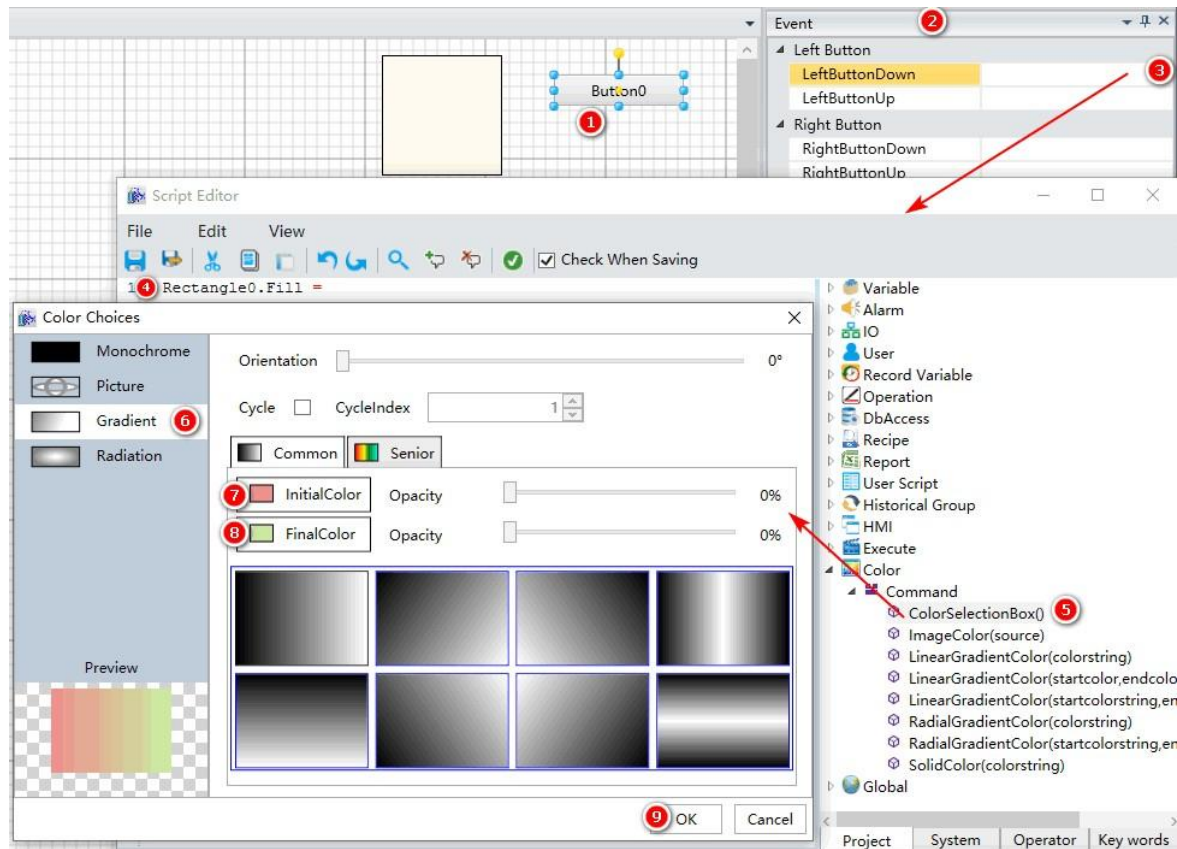
- **ColorSelectionBox** приклад (цей скрипт викликає редактор кольорів) :

Прямокутник заливається кольором лінійного градієнта

(1) Створіть Rectangle0 і Button0 у Window0, налаштуйте подію натискання лівої кнопки миші на кнопку button0



(2) Налаштуйте подію LeftMouseDown для Button0



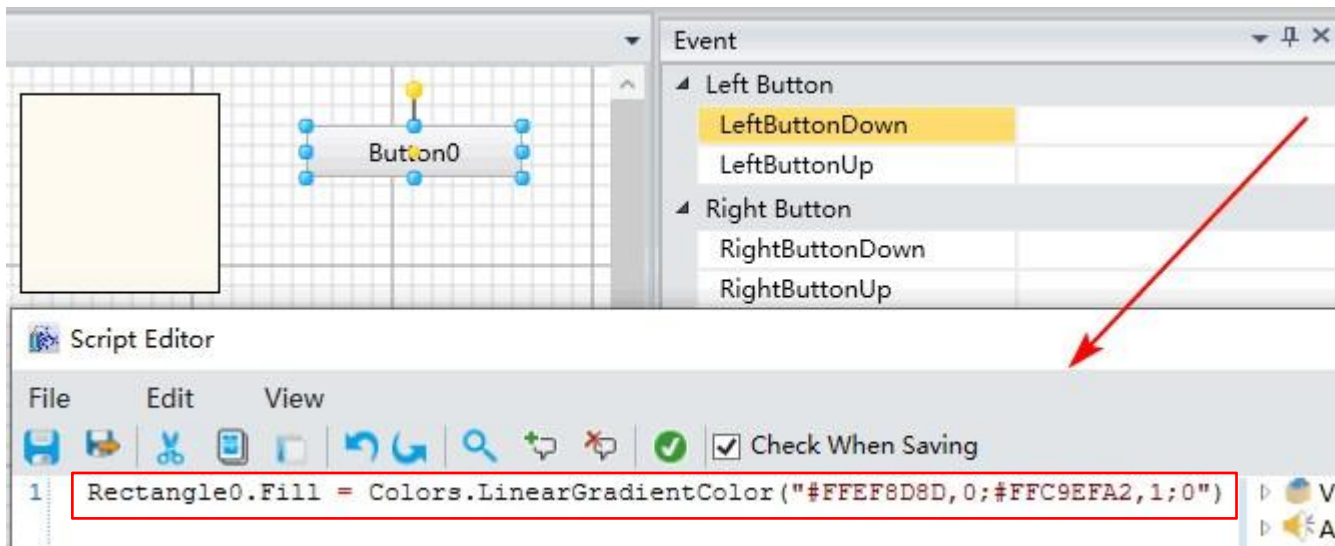
The screenshot displays the Delta HMI software interface with several windows open for configuring a color scenario:

- Event Window (Top Right):** Shows the event list for **Button0**. The **LeftButtonDown** event is selected (highlighted in yellow).
- Script Editor (Middle):** Displays the script for the selected event: `1 Rectangle0.Fill =`.
- Color Choices Window (Bottom Left):** Shows the configuration for the color scenario. The **Gradient** option is selected. The **Initial Color** is set to red and the **Final Color** is set to green. The **Orientation** is set to 0° . The **Preview** section shows a gradient bar transitioning from red to green.
- Object Tree (Bottom Right):** Shows the hierarchy of objects. The **Color** object is expanded, and the **ColorSelectionBox()** command is selected.

Red arrows and numbers (1-9) highlight specific steps in the configuration process:

1. Select the **Button0** object in the main workspace.
2. Select the **LeftButtonDown** event in the Event list.
3. Select the **LeftButtonDown** event in the Event list.
4. Enter the script `Rectangle0.Fill =` in the Script Editor.
5. Select the **ColorSelectionBox()** command in the Object Tree.
6. Select the **Gradient** color type in the Color Choices window.
7. Select the **Initial Color** (red) in the Color Choices window.
8. Select the **Final Color** (green) in the Color Choices window.
9. Click the **OK** button in the Color Choices window.

(3) Результати конфігурації такі



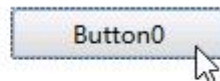
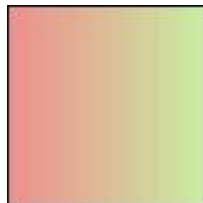
The screenshot displays the Delta software interface. On the left, a yellow rectangle is shown on a grid. In the center, a button labeled "Button0" is visible. On the right, the "Event" panel lists the following events:

- Left Button
 - LeftButtonDown (highlighted)
 - LeftButtonUp
- Right Button
 - RightButtonDown
 - RightButtonUp

A red arrow points from the "LeftButtonDown" event to the "Script Editor" window below. The "Script Editor" window has a menu bar with "File", "Edit", and "View". Below the menu bar is a toolbar with various icons. The script area contains the following code:

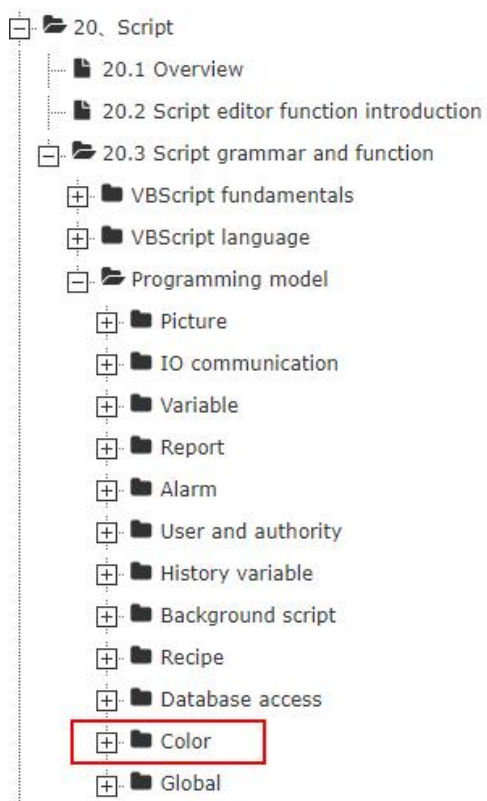
```
1 Rectangle0.Fill = Colors.LinearGradientColor("#FFEF8D8D,0;#FFC9EFA2,1;0")
```

(4) Запустіть Window0 , клацніть Button0 , потім Rectangle0
стає червоно-зеленою лінійною градієнтною заливкою



Щоб отримати докладніші відомості про використання кольорових сценаріїв, зверніться до розділу « 20.3 Граматика та функції сценаріїв » посібника користувача.

Як показано на малюнку нижче:



Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta,
відвідайте www.deltaww.com.



DIAView

Загальний приклад сценарію

Венді
2020/04



- Приклад пакетного призначення словника змінних
- Приклад мови глобальної системи комутації
- Приклад запиту на запис змінної
- Приклад запису групи історії запитів
- Приклад доступу до бази даних

призначення

Після цього розділу ви дізнаєтесь...

... докладніше про типові приклади сценаріїв

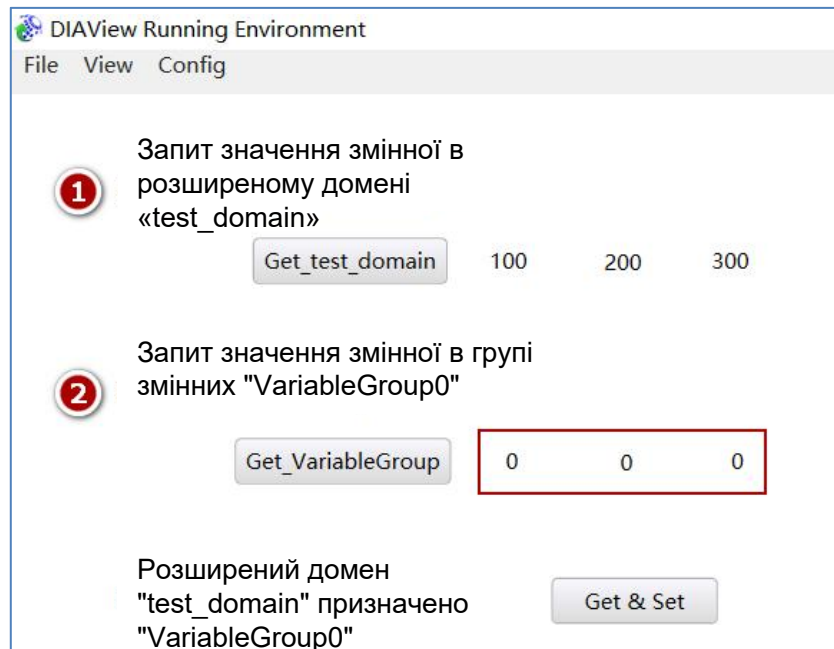
Контур

- Приклад пакетного призначення словника змінних
- Приклад мови глобальної системи комутації
- Приклад запиту на запис змінної
- Приклад запису групи історії запитів
- Приклад доступу до бази даних

➤ приклад 1:

- Призначте значення змінної в розширеному домені test_domain змінній у VariableGroup0

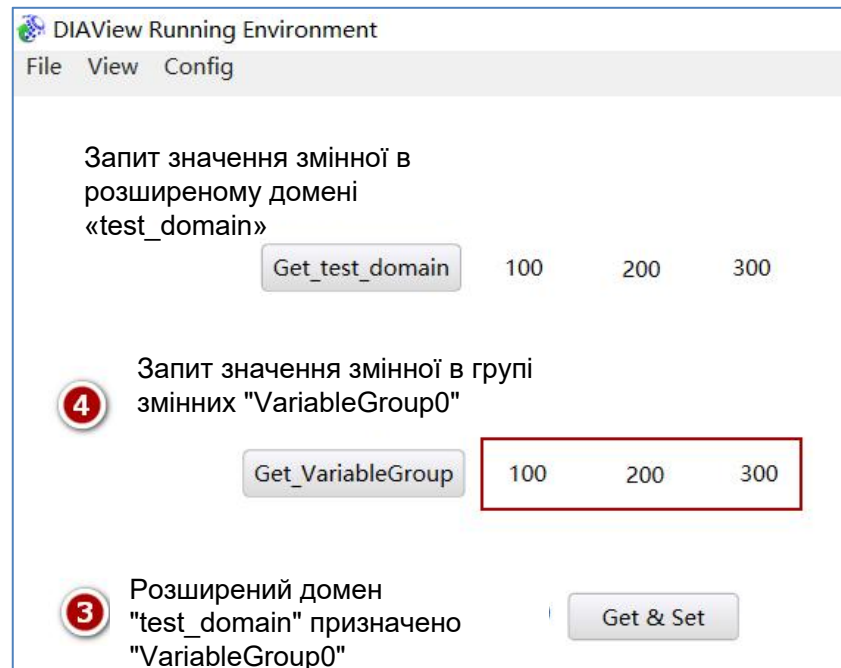
- ① Запитуйте значення трьох змінних у розширеному домені test_domain як 100, 200 і 300 відповідно
- ② Значення трьох змінних у VariableGroup0 дорівнюють 0



➤ приклад 1:

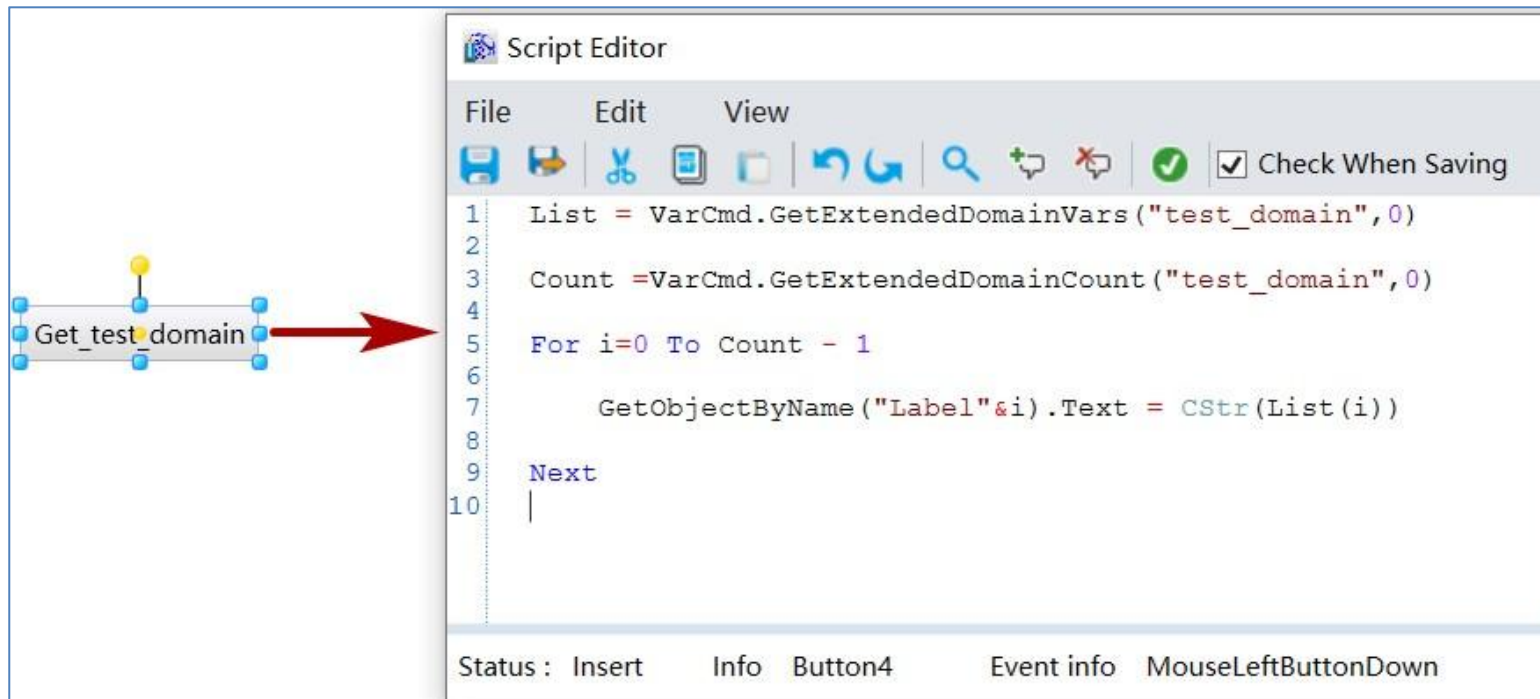
- Призначте значення змінної в розширеному домені test_domain змінній у VariableGroup0

- ③ Виконайте сценарій події кнопки, щоб призначити значення змінної в розширеному домені test_domain змінній у VariableGroup0
- ④ Знову запитайте значення трьох змінних у VariableGroup0: 100, 200 і 300, і призначення буде успішним



➤ Приклад 1: сценарій кнопки `Get_test_domain`

- Запитувати значення кожної змінної в розширеному домені `test_domain` і відобразити його в поточному вікні з міткою



The diagram illustrates the implementation of a button click event. On the left, a button labeled `Get_test_domain` is shown with a yellow lightbulb icon above it, indicating a script is attached. A red arrow points from the button to the 'Script Editor' window on the right.

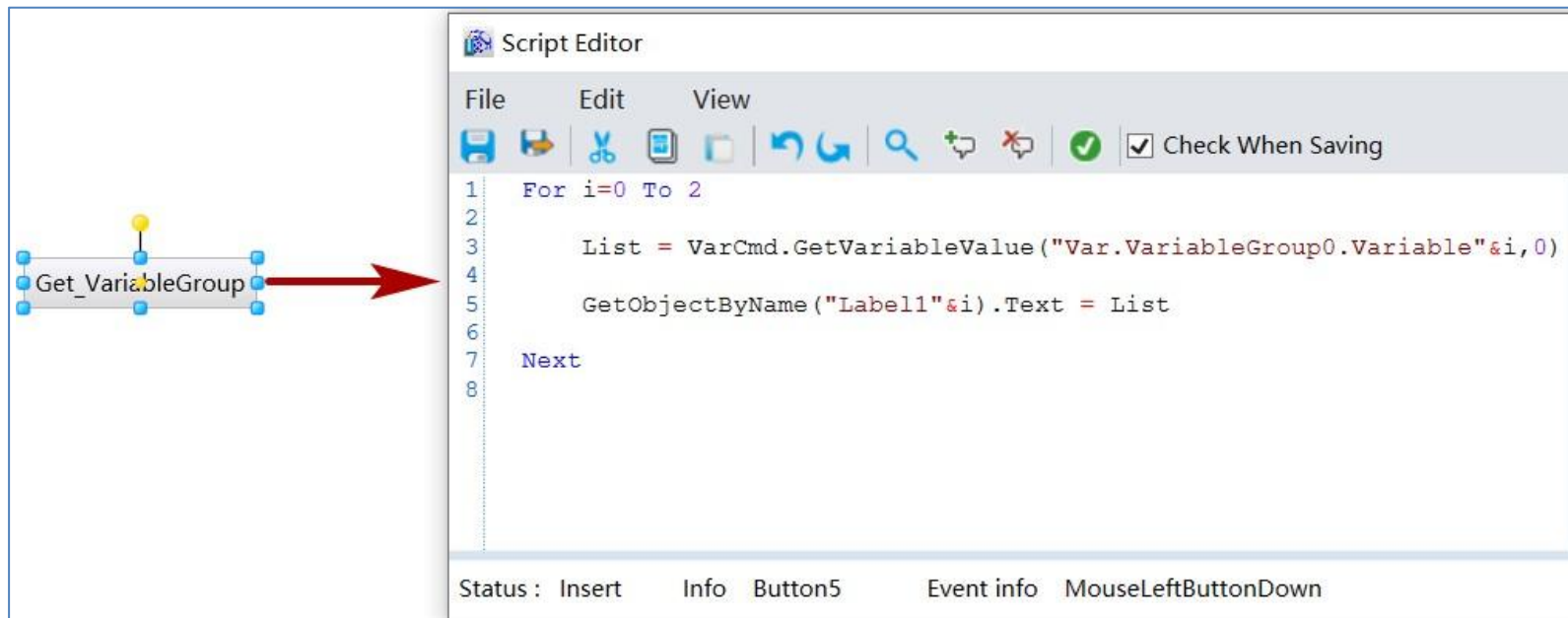
Script Editor

```
1 List = VarCmd.GetExtendedDomainVars("test_domain",0)
2
3 Count =VarCmd.GetExtendedDomainCount("test_domain",0)
4
5 For i=0 To Count - 1
6
7     GetObjectByName("Label"&i).Text = CStr(List(i))
8
9 Next
10 |
```

Status : Insert Info Button4 Event info MouseLeftButtonDown

➤ Приклад 1: сценарій кнопки **Get_variableGroup0**

- Запитуйте значення кожної змінної у VariableGroup0 і відображайте його у вікні, що працює, із міткою



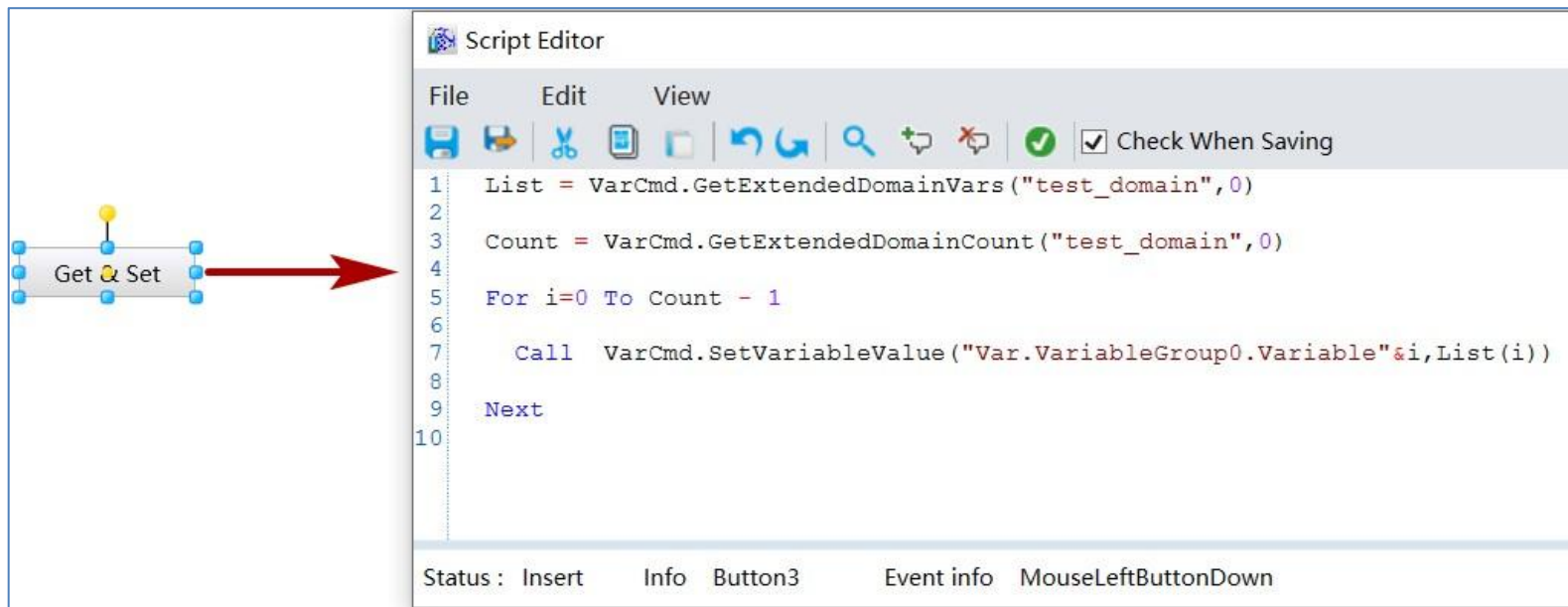
The diagram illustrates the implementation of a button's click event. On the left, a button labeled "Get_VariableGroup" is shown with a yellow lightbulb icon above it, indicating a script is attached. A red arrow points from the button to a "Script Editor" window on the right. The script editor contains the following code:

```
1 For i=0 To 2
2
3     List = VarCmd.GetVariableValue("Var.VariableGroup0.Variable"&i,0)
4     GetObjectByName("Label1"&i).Text = List
5
6
7 Next
8
```

The script editor window has a menu bar with "File", "Edit", and "View". Below the menu is a toolbar with icons for saving, opening, undo, redo, find, and other standard editing functions. A checkbox labeled "Check When Saving" is also present. The status bar at the bottom of the script editor shows "Status : Insert", "Info", "Button5", "Event info", and "MouseLeftButtonDown".

➤ Приклад 1: сценарій кнопки «Отримати та встановити».

- Призначте значення кожної змінної в розширеному домені test_domain групі змінних VariableGroup0



※Для отримання детального опису методів і властивостей змінних зверніться до користувача посібник "CH20.3 Синтаксис сценарію та функції – Змінні моделі програмування"

- Приклад пакетного призначення словника змінних
- Приклад мови глобальної системи комутації
- Приклад запиту на запис змінної
- Приклад запису групи історії запитів
- Приклад доступу до бази даних

➤ приклад 2:

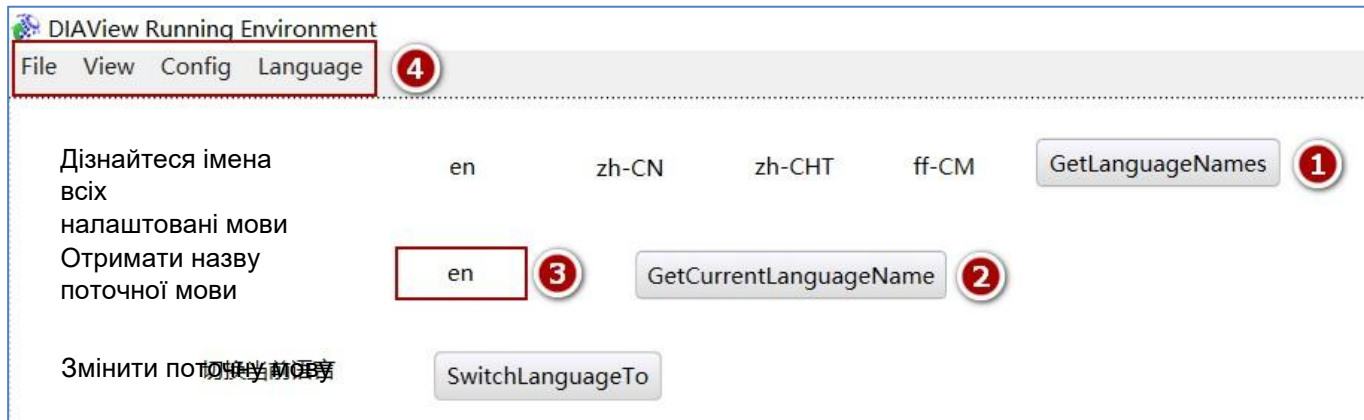
- Запитайте назви всіх налаштованих мов і переключіть поточну мову

① Отримайте назви всіх налаштованих мов як en, zh-CN, zh-CHT, ff-CM

② Отримати назву поточної мови

③ Вміст мітки показує, що назва поточної мови – en

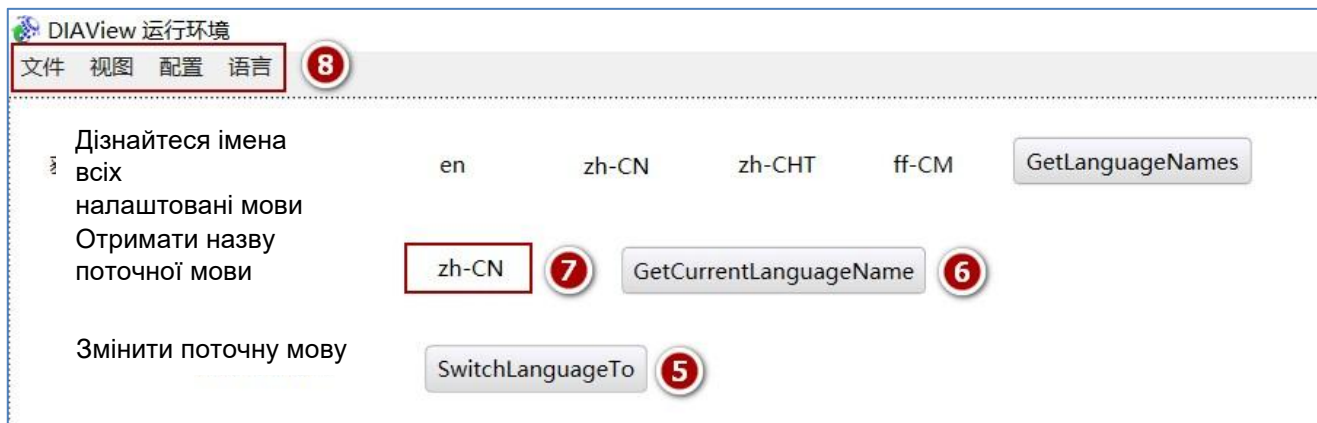
④ Переконайтеся, що мовою системи є en



➤ приклад 2:

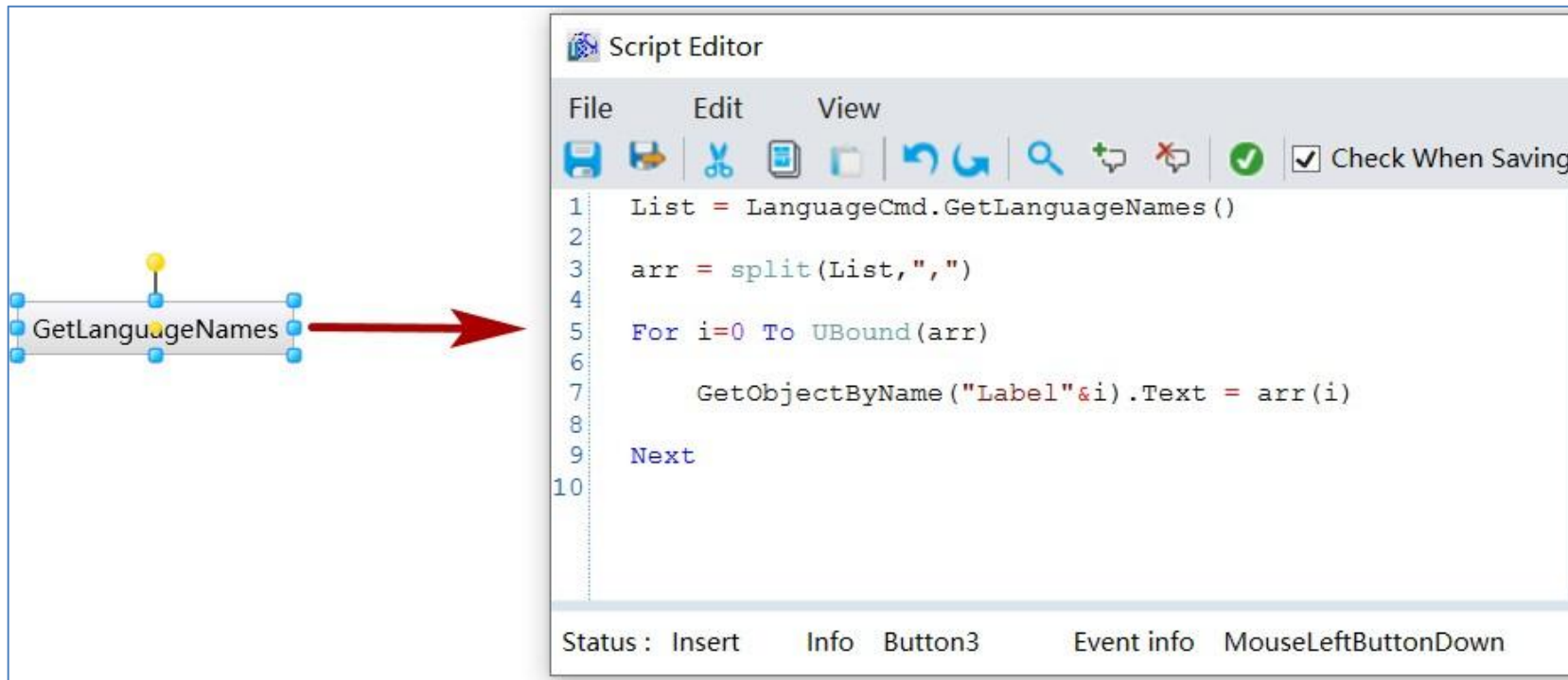
- Запитайте назви всіх налаштованих мов і переключіть поточну мову

- ⑤ Змінити поточну мову
- ⑥ Знову отримати назву поточної мови
- ⑦ Вміст мітки показує, що назва поточної мови – zh-CN
- ⑧ Переконайтеся, що мовою системи є zh-CN



➤ Приклад 2: сценарій кнопки **GetLanguageNames**

- Запитуйте назви всіх налаштованих мов і відображайте їх у поточному вікні з тегами.



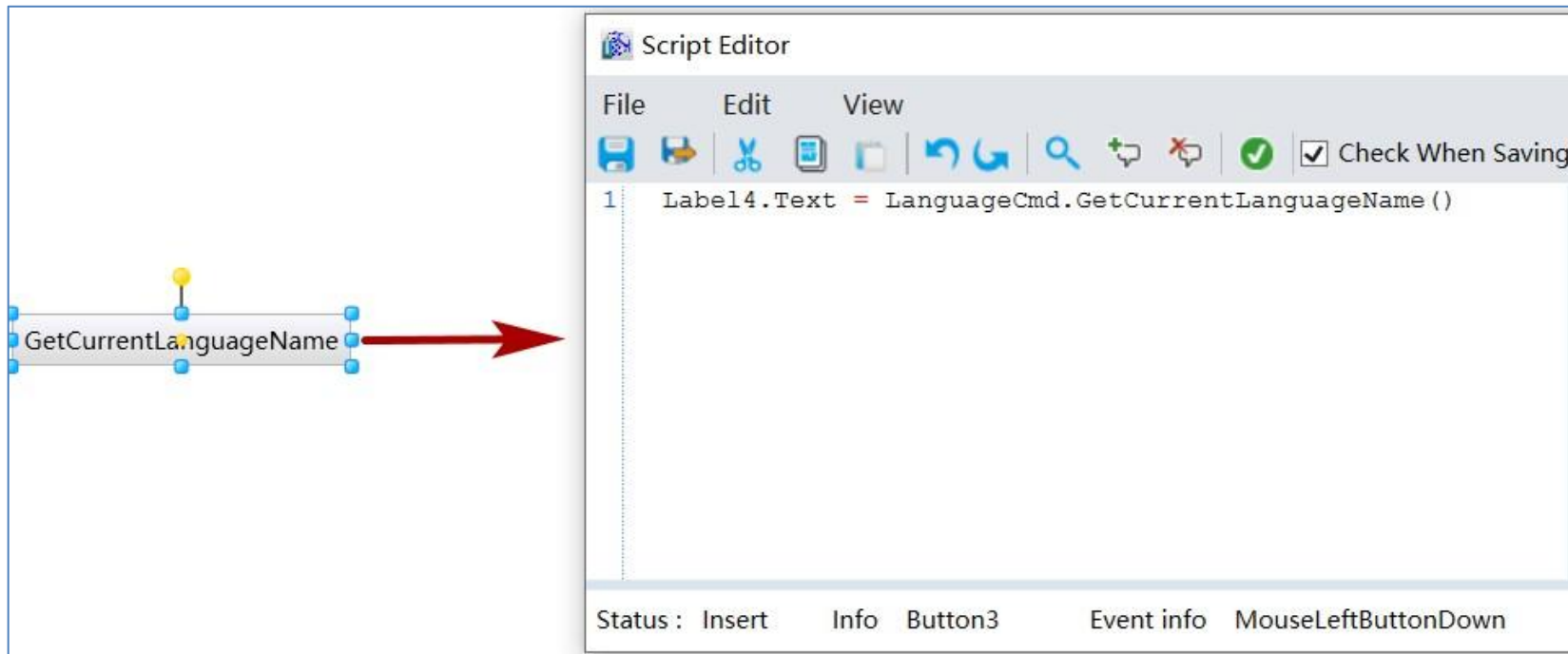
The diagram illustrates the implementation of a button click event. On the left, a button labeled "GetLanguageNames" is shown with a red arrow pointing to a "Script Editor" window on the right. The script editor contains the following code:

```
1 List = LanguageCmd.GetLanguageNames ()
2
3 arr = split(List, ",")
4
5 For i=0 To UBound(arr)
6
7     GetObjectByName ("Label"&i) .Text = arr(i)
8
9 Next
10
```

The status bar at the bottom of the script editor shows: Status : Insert Info Button3 Event info MouseLeftButtonDown

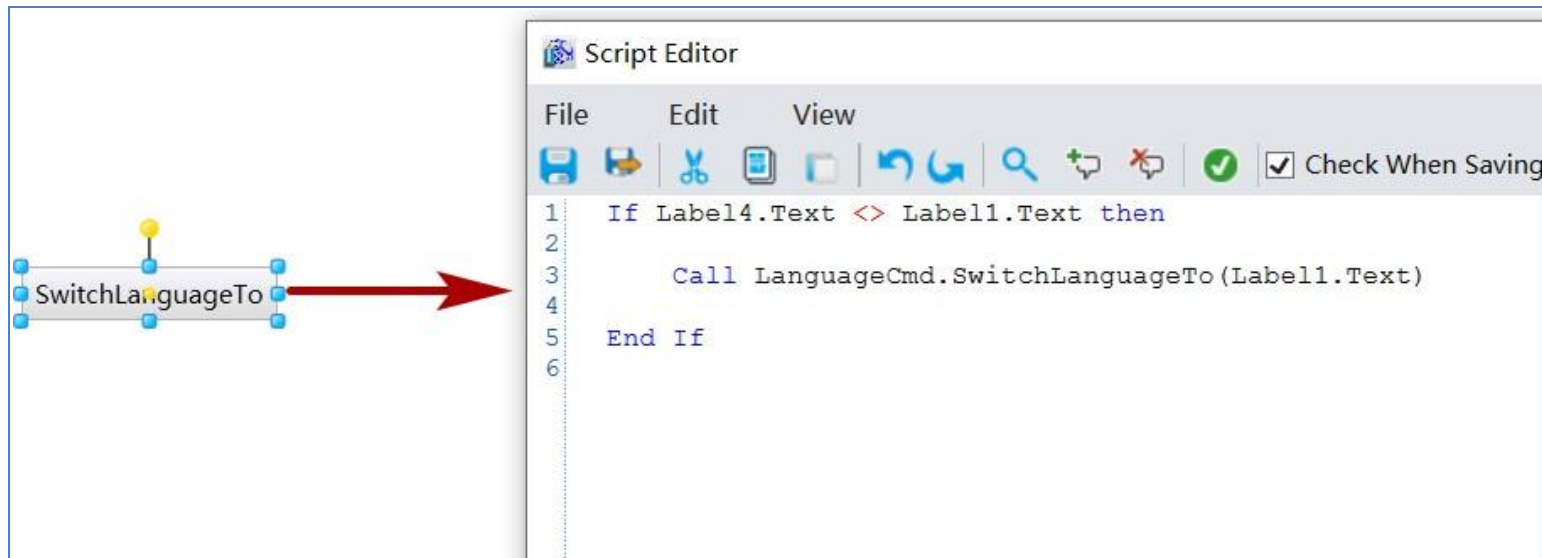
➤ Приклад 2: сценарій кнопки `GetCurrentLanguageName`

- Запитайте назву поточної мови та відобразіть її у вікні, що працює, із міткою.



➤ Приклад 2: сценарій кнопки SwitchLanguageTo

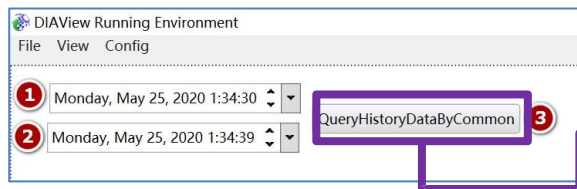
- Змінити поточну мову.



※Примітка. Детальний опис методів і атрибутів глобалізації див [до посібник користувача "СН 20.3 Синтаксис сценарію та функції – Модель програмування – Глобалізація"](#)

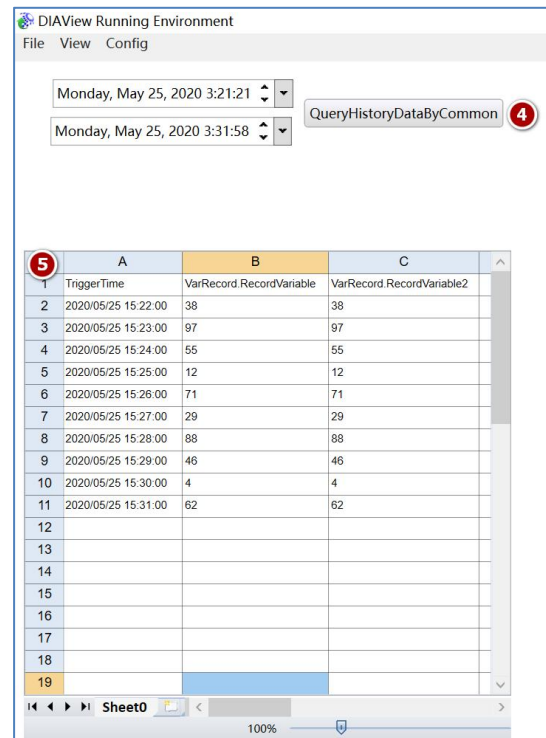
- Приклад пакетного призначення словника змінних
- Приклад мови глобальної системи комутації
- Приклад запиту на запис змінної
- Приклад запису групи історії запитів
- Приклад доступу до бази даних

- **Приклад 3: сценарій запиту змінного запису**
 Запис змінної зчитується через сценарій і відображається у звіті.
- Кнопка QueryHistoryDataByCommon (sheetIndex, умови, типи):
 запит змінної записи
 1. sheetIndex: індекс аркуша
 2. умови: колекція змінних записів, розділених комами
 3. типи: TriggeringTime, Value, також можна замінити на 0 і 1 відповідно



```
DateTimePicker0.Value = Sys.StartTime
DateTimePicker1.Value = Sys.Now

Звіт про
виклик0.SetWorkSheetStartTime(0,DateTime
Picker0.ValueTime)
Виклик Report0.SetWorkSheetEndTime(0,DateTime
Picker1.ValueTime)
Виклик Report0.SetVarRecordRulerName(0,"m1")
Виклик Report0.QueryHistoryDataByCommon(0,"Var
Record.RecordVariable,VarRecord.RecordVariabl
e,VarRecord.RecordVariable2","0,1,1")
```

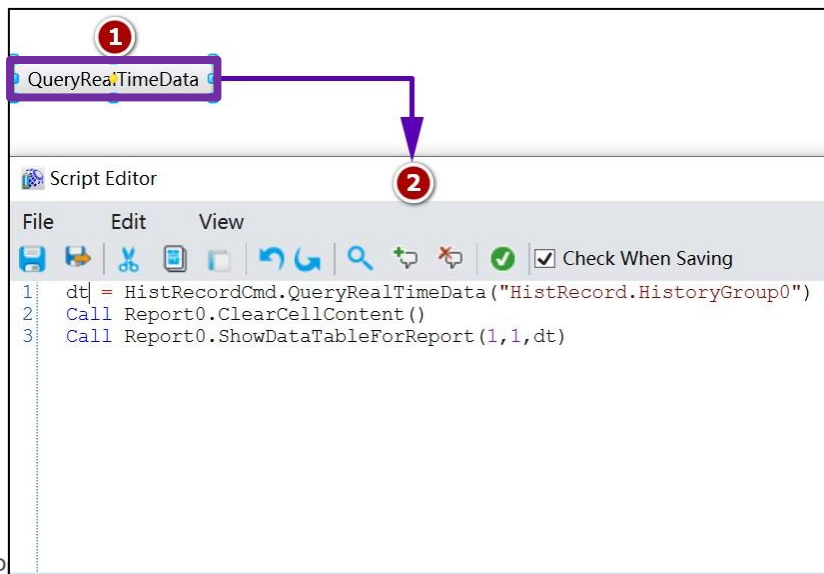


	A	B	C
1	TriggerTime	VarRecord.RecordVariable	VarRecord.RecordVariable2
2	2020/05/25 15:22:00	38	38
3	2020/05/25 15:23:00	97	97
4	2020/05/25 15:24:00	55	55
5	2020/05/25 15:25:00	12	12
6	2020/05/25 15:26:00	71	71
7	2020/05/25 15:27:00	29	29
8	2020/05/25 15:28:00	88	88
9	2020/05/25 15:29:00	46	46
10	2020/05/25 15:30:00	4	4
11	2020/05/25 15:31:00	62	62
12			
13			
14			
15			
16			
17			
18			
19			

- Приклад пакетного призначення словника змінних
- Приклад мови глобальної системи комутації
- Приклад запиту на запис змінної
- **Приклад запису групи історії запитів**
- Приклад доступу до бази даних

➤ Приклад 4: сценарій запису групи журналу запитів

- Прочитайте дані в реальному часі групи історичних записів за допомогою сценарію і відобразити його разом зі звітом.
- Кнопка QueryRealTimeData : запит даних у реальному часі історичного запису група;

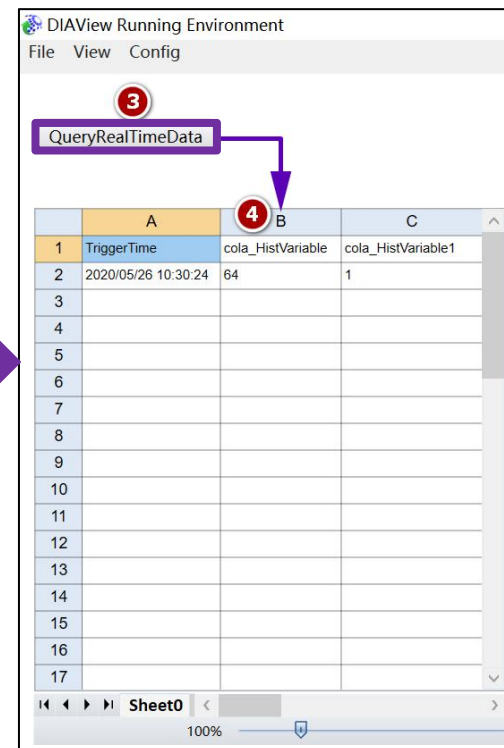


1. QueryRealTimeData

2. Script Editor

```

1 dt = HistRecordCmd.QueryRealTimeData("HistRecord.HistoryGroup0")
2 Call Report0.ClearCellContent()
3 Call Report0.ShowDataTableForReport(1,1,dt)
  
```

3. QueryRealTimeData

4. B

	A	B	C
1	TriggerTime	cola_HistVariable	cola_HistVariable1
2	2020/05/26 10:30:24	64	1
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			

Sheet0 100%

- Приклад пакетного призначення словника змінних
- Приклад мови глобальної системи комутації
- Приклад запиту на запис змінної
- Приклад запису групи історії запитів
- Приклад доступу до бази даних

➤ Приклад 5: сценарій запису групи журналу запитів

- Прочитайте дані групи історичних записів за допомогою сценарію та відобразити його зі звітом.
- Кнопка QueryWithStorageItemName : запитувати дані групи історичних записів, прикладом є запит даних за останню годину.
- Використовуйте функцію **DateAdd** , щоб встановити діапазон часу запити на останню годину;

1

QueryWithStorageItemName

2

Script Editor

File Edit View

Check When Saving

```
1 Dim StartTime
2 StartTime = DateAdd("h",-1,Now())
3 dt = HistRecordCmd.QueryWithStorageItemName("HistoryGroup0", "RecordVariable",RecordVariable1", StartTime, Sys.Now,-1)
4 Call Report1.ClearCellContent()
5 Call Report1.ShowDataTableForReport(1,1,dt)
```

3

QueryWithStorageItemName

4

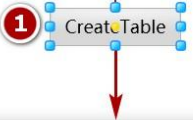
	A	B	C
1	TriggerTime	cola_HistVariable	cola_HistVariable1
2	2020/05/26 09:58:23	0	
3	2020/05/26 09:58:24	0	0
4	2020/05/26 09:58:25	6	6
5	2020/05/26 09:58:26	7	1
6	2020/05/26 09:58:27	8	1
7	2020/05/26 09:58:28	9	1
8	2020/05/26 09:58:29	10	1
9	2020/05/26 09:58:30	10	0
10	2020/05/26 09:58:31	11	1
11	2020/05/26 09:58:32	12	1
12	2020/05/26 09:58:33	13	1
13	2020/05/26 09:58:34	14	1
14	2020/05/26 09:58:35	15	1
15	2020/05/26 09:58:36	16	1
16	2020/05/26 09:58:37	17	1
17	2020/05/26 09:58:38	18	1

Sheet0 100%

- Приклад пакетного призначення словника змінних
- Приклад мови глобальної системи комутації
- Приклад запиту на запис змінної
- Приклад запису групи історії запитів
- Приклад доступу до бази даних

➤ Приклад 6: Доступ до бази даних

- ① Програмування сценарію події кнопки, створення таблиці **stu_table**
- ② Виконайте сценарій події кнопки, поверніть True, щоб створити таблицю **stu_table** успішно та відобразити таблицю **stu_table** у запущеному вікні;



Створити таблицю

Script Editor

File Edit View



```

1 dim dt
2 dt = DbAccessCmd.CreateTable("stu_table")
3 Call DbAccessCmd.AddColumn(dt,"ID","stu_id",1,500,false,true,true)
4 Call DbAccessCmd.AddColumn(dt,"NAME","stu_name",4,500,true,false,false)
5 Call DbAccessCmd.AddColumn(dt,"AGE","stu_age",4,500,true,false,false)
6 Call DbAccessCmd.AddColumn(dt,"CHINESE","chinese",4,500,true,false,false)
7 Call DbAccessCmd.AddColumn(dt,"MATH","math",4,500,true,false,false)
8 Call DbAccessCmd.AddColumn(dt,"ENGLISH","english",4,500,true,false,false)
9 Call DbAccessCmd.SetPrimary(dt,"ID")
10 call Report0.ClearCellContent()
11 Call Report0.ShowDataTableForReport(1,1,dt)
12 cc = DbAccessCmd.ExecuteCreateDatabaseTable("DbAccess.DatabaseAccess","mysql",dt)
13 MsgBox cc
        
```



Створено успішно

DIView Running Environment

File View Config

True

OK

	A	B	C	D	E	F
1	ID	NAME	AGE	CHINESE	MATH	ENGLISH
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						

Sheet0 100%

- The screenshot shows the VBA Script Editor with the following code:

```

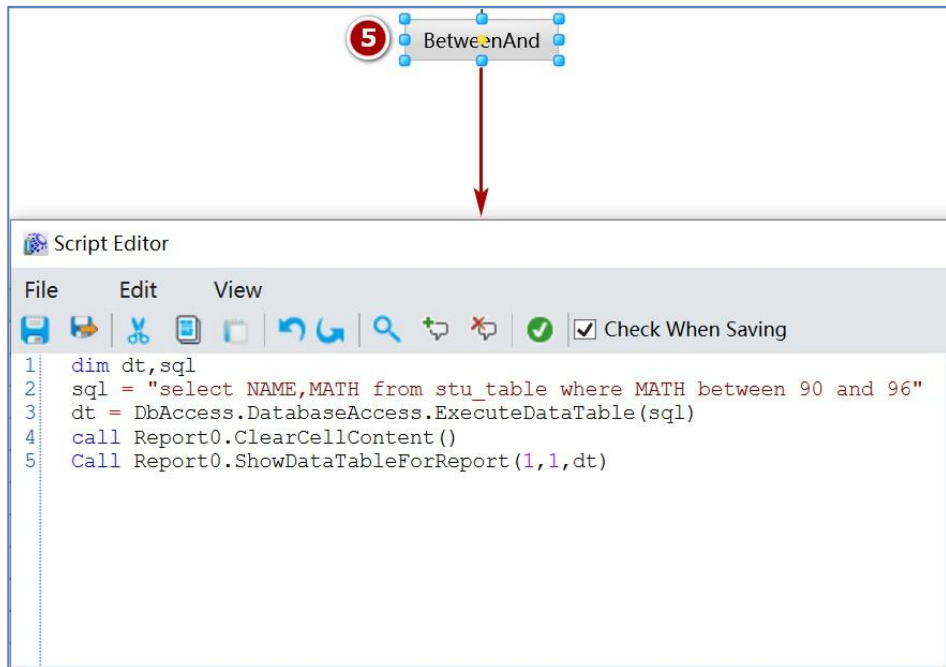
dim dt,sql
For i = 1 To 2
    sql1 = "insert into stu_table values("&i&","&"name"&i&","&"10','95','89','99')"

```

The screenshot shows the DIAView Running Environment. The VBA editor on the left contains a macro named `BatchInsertData`, which is highlighted with a red circle and the number 4. A red arrow points from this macro to the 'ID' column of a data table. The data table has columns labeled A through F, with headers 'ID', 'NAME', 'AGE', 'CHINESE', 'MATH', and 'ENGLISH'. The table contains data for rows 1 through 6. A dialog box is open in the center, displaying the number '6' and an 'OK' button.

	A	B	C	D	E	F
1	ID	NAME	AGE	CHINESE	MATH	ENGLISH
2	1	name1	10	95	89	99
3	2	name2	10	95	89	99
4	3	test3	12	88	97	93
5	4	test4	12	88	97	93
6	5	xingming5	9	80	90	95
7	6	xingming6	9	80	90	95
8						
9						
10						
11						
12						
13						
14						

- ⑤ Скрипт подій кнопки програмування, таблиця запитів **stu_table** кваліфіковані дані;
- ⑥ Виконайте сценарій події кнопки для відображення запитуваних даних у вікні, що працює;



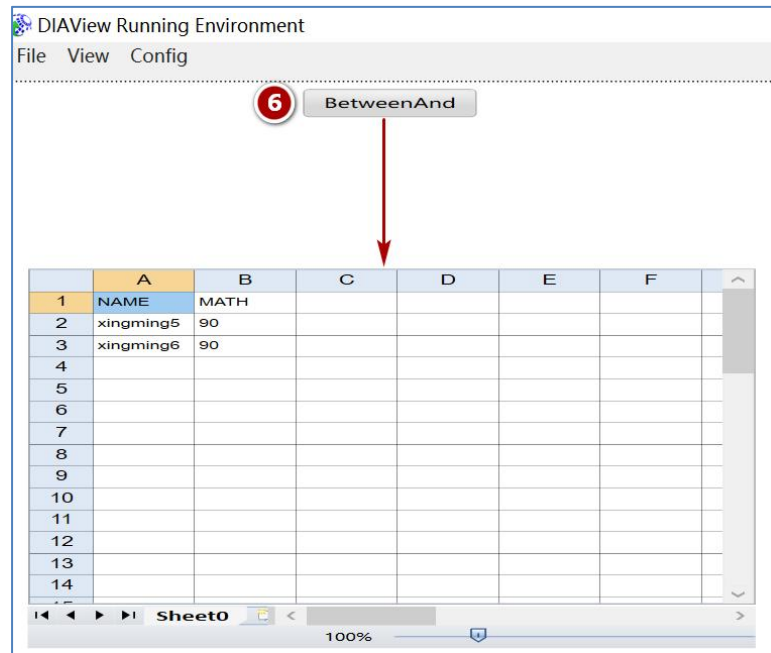
5 BetweenAnd

Script Editor

File Edit View

Check When Saving

```
1 dim dt,sql
2 sql = "select NAME,MATH from stu_table where MATH between 90 and 96"
3 dt = DbAccess.DatabaseAccess.ExecuteDataTable(sql)
4 call Report0.ClearCellContent()
5 Call Report0.ShowDataTableForReport(1,1,dt)
```



6 BetweenAnd

	A	B	C	D	E	F
1	NAME	MATH				
2	xingming5	90				
3	xingming6	90				
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						

Sheet0 100%

⑦ Програмування сценарію події кнопки для оновлення кваліфікованих даних у таблиці **stu_table** ;

⑧ Виконайте сценарій події кнопки та відобразіть виконану таблицю **stu_table** у запущеному вікні;



Script Editor

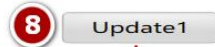
File Edit View

Check When Saving

```
1 dim dt,sql
2 sql = "update stu_table set NAME = 'new_name' where MATH between 90 and 96"
3 cc = DbAccess.DatabaseAccess.ExecuteNonQuery(sql)
4 MsgBox cc
5 dt = DbAccess.DatabaseAccess.GetTable()
6 call Report0.ClearCellContent()
7 Call Report0.ShowDataTableForReport(1,1,dt)
```

DIAView Running Environment

File View Config



	A	B	C	D	E	F	
1	ID	NAME	AGE	CHINESE	MATH	ENGLISH	
2	1	name1	10	95	89	99	
3	2	name2	10	95	89	99	
4	3	test3	12	88	97	93	
5	4	test4	12	88	97	93	
6	5	new_name	9	80	90	95	
7	6	new_name	9	80	90	95	
8							
9							
10							
11							
12							
13							
14							

- ⑨ Напишіть сценарій події кнопки для видалення кваліфікованих даних у таблиці **stu_table** ;
- ⑩ Виконайте сценарій події кнопки та відобразіть виконану таблицю **stu_table** у запущеному вікні;

9 Delete...Where

Script Editor

File Edit View

Check When Saving

```

1 dim dt,sql
2 sql1 = "delete from stu_table where id < 3;"
3 Call DbAccess.DatabaseAccess.ExecuteDataTable(sql1)
4 dt = DbAccess.DatabaseAccess.GetTable()
5 call Report0.ClearCellContent()
6 Call Report0.ShowDataTableForReport(1,1,dt)|
  
```

10 Delete...Where

DIAMView Running Environment

File View Config

	A	B	C	D	E	F
1	ID	NAME	AGE	CHINESE	MATH	ENGLISH
2	3	test3	12	88	97	93
3	4	test4	12	88	97	93
4	5	new_name	9	80	90	95
5	6	new_name	9	80	90	95
6						
7						
8						
9						
10						
11						
12						
13						
14						

Sheet0 100%

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Доступ до даних

Венді
2020/04



Контур

- Огляд доступу до бази даних
- База даних
- Конфігурація бази даних
- Операції з базами даних

призначення

Після цього розділу ви дізнаєтесь...

- ... База даних

- ... Конфігурація бази даних

- ... Операції з базами даних

Контур

- Огляд доступу до бази даних
- База даних
- Конфігурація бази даних
- Операції з базами даних



Доступ до бази даних

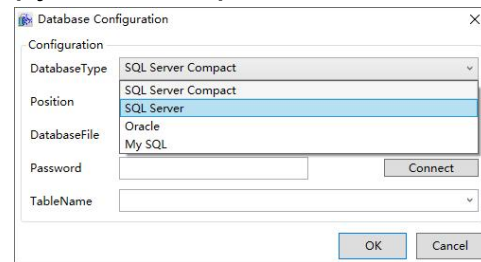
➤ DIAView надає методи та властивості для доступу, запитів і редагування даних у зовнішніх базах даних. Користувач налаштовує підключення до бази даних у доступі до бази даних і пише сценарій доступу до бази даних, щоб реалізувати роботу з даними у відповідній базі даних під час виконання. Забезпечити основні операції з базою даних, включаючи вставку, зміну, запит даних, створення та видалення таблиць.

- ① Підтримка різних баз даних, таких як My SQL, SQL Server, SQL Compact, Oracle тощо.
- ② Підтримка обміну виробничими даними через базу даних і програмне забезпечення сторонніх виробників
- ③ Підтримка файлу бази даних, назви таблиці даних у розкритому списку зручний вибір



База даних

Взаємодія з даними бази даних у реальному часі



**Третя сторона
Інтеграція програмного
забезпечення**



- Доступ до бази даних
- Конфігурація доступу до бази даних
- Конфігурація бази даних
- Операція з базою даних

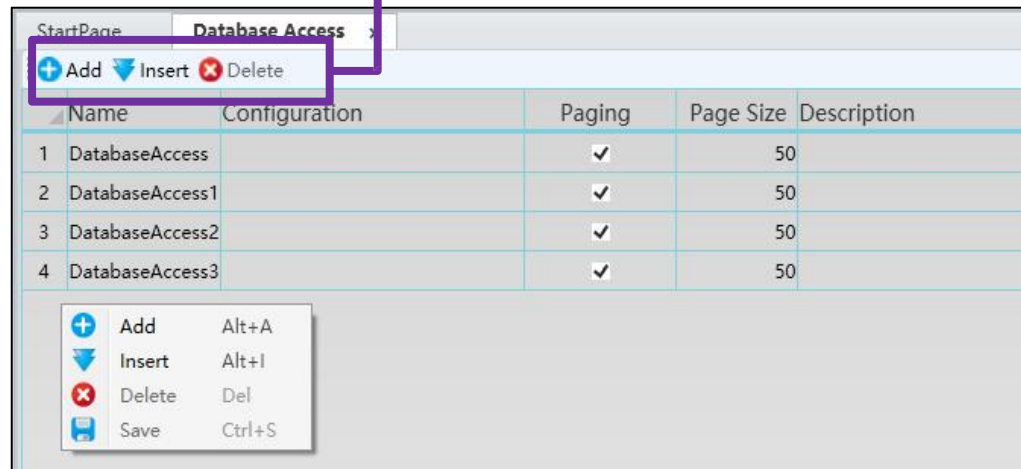
➤ Конфігурація доступу до бази даних :

- **Ім'я** : Ім'я доступу до бази даних, за замовчуванням – Доступ до бази даних, DatabaseAccess1 ... можна перейменувати;
- **Конфігурація** : інформація про конфігурацію бази даних;
- **Розбиття сторінок** : перевірте, чи розбита таблиця в базі даних на сторінки, щоб вказати розбиття сторінок, і перевірте за замовчуванням;
- **Розмір сторінки** : розмір сторінки, за замовчуванням 50 сторінок, які можна змінювати;
- **Опис** : опис або опис доступу до бази даних, який за замовчуванням порожній;

	Name	Configuration	Paging	Page Size	Description
1	DatabaseAccess	DIALinkHistoricalData_200519- ...	✓	50	
2	DatabaseAccess1		✓	50	
3	DatabaseAccess2		✓	50	

➤ Кнопки панелі інструментів:

- **Add / Alt + A** : додати доступ до бази даних;
- **Вставити / Alt + I** : вставити доступ до бази даних;
- **Delete / Del** : видалити доступ до бази даних;
- **Зберегти / Ctrl + S** : зберегти зміни;



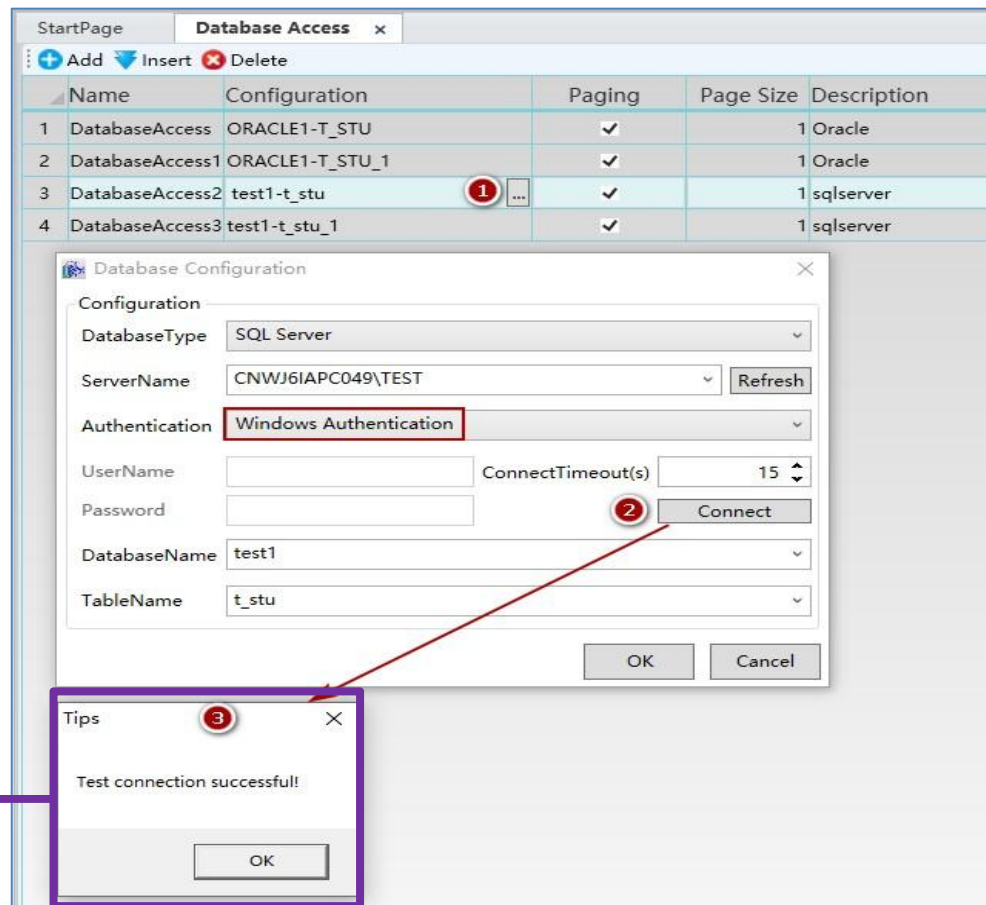
- Доступ до бази даних
- Конфігурація доступу до бази даних
- Конфігурація бази даних
- Операція з базою даних

➤ Конфігурація SQL Server:

Перед доступом до SQL Server

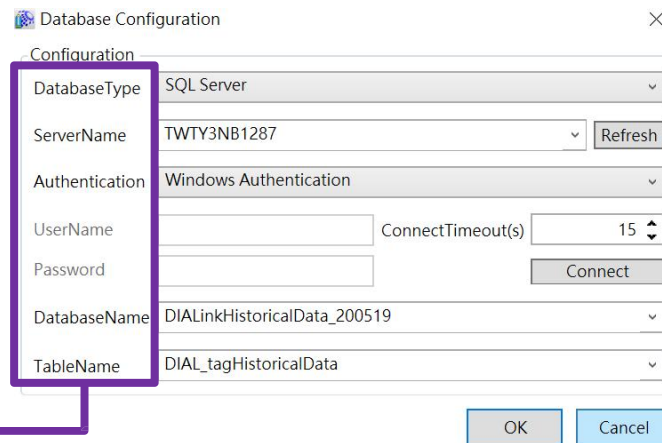
бази даних, будь ласка, переконайтеся,
що

з'єднання між DIAView і база даних
успішна



➤ Конфігурація бази даних:

- **ServerName** : ім'я або IP-адреса хоста, де розташована база даних SQL Server
- **Автентифікація** : метод автентифікації, який підтримує SQL Server:
 - ① **Автентифікація Windows** : не потрібно вводити ім'я користувача та пароль
 - ② **Автентифікація SQL** : введіть відповідне ім'я користувача та пароль, налаштовані на сервері SQL
- **Ім'я користувача** : ім'я користувача, що відповідає автентифікації SQL
- **Пароль**: пароль, що відповідає автентифікації SQL
- **ConnectTimeout** : за замовчуванням 15 секунд
- **DatabaseName** : Ім'я бази даних, до якої потрібно отримати доступ
- **TableName** : назва таблиці даних, до якої потрібно отримати доступ
- **Connect** : Перевірте з'єднання з базою даних і поверніть «Тестове з'єднання успішно»



Database Configuration

Configuration

DatabaseType: SQL Server

ServerName: TWTY3NB1287 [Refresh]

Authentication: Windows Authentication

Username: [] ConnectTimeout(s): 15 []

Password: [] [Connect]

DatabaseName: DIALinkHistoricalData_200519

TableName: DIAL_tagHistoricalData

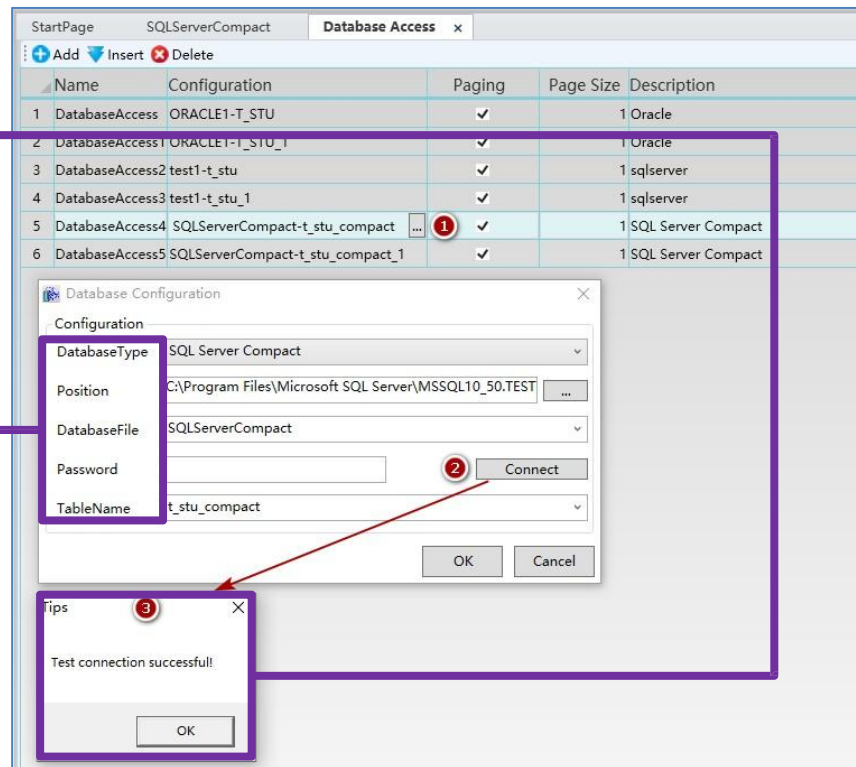
[OK] [Cancel]

➤ Перед доступом до SQL Server Compact

База даних, переконайтеся, що зв'язок між DIAView і база даних успішні.

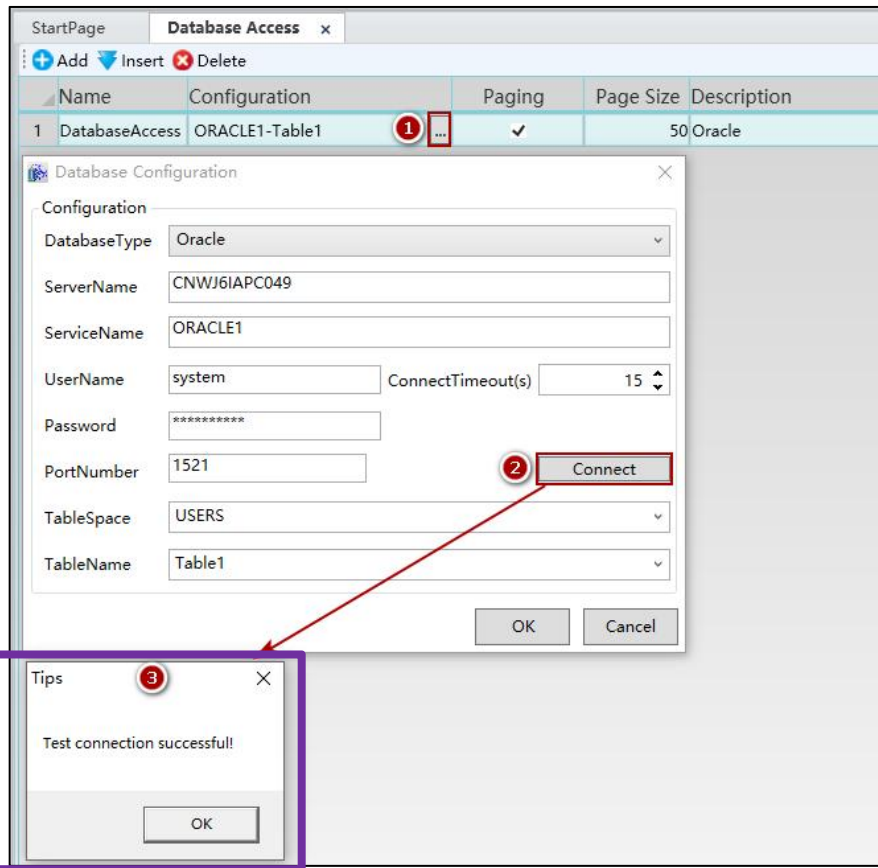
➤ Конфігурація бази даних

- Позиція: Шлях, де розташована компактна база даних SQL Server
- Файл бази даних: назва бази даних SQL Server Compact
- Пароль: пароль, що відповідає базі даних
- Назва таблиці: назва таблиці даних, до якої потрібно отримати доступ
- Підключитися: перевірте з'єднання з базою даних і поверніть «Тестове з'єднання успішно»



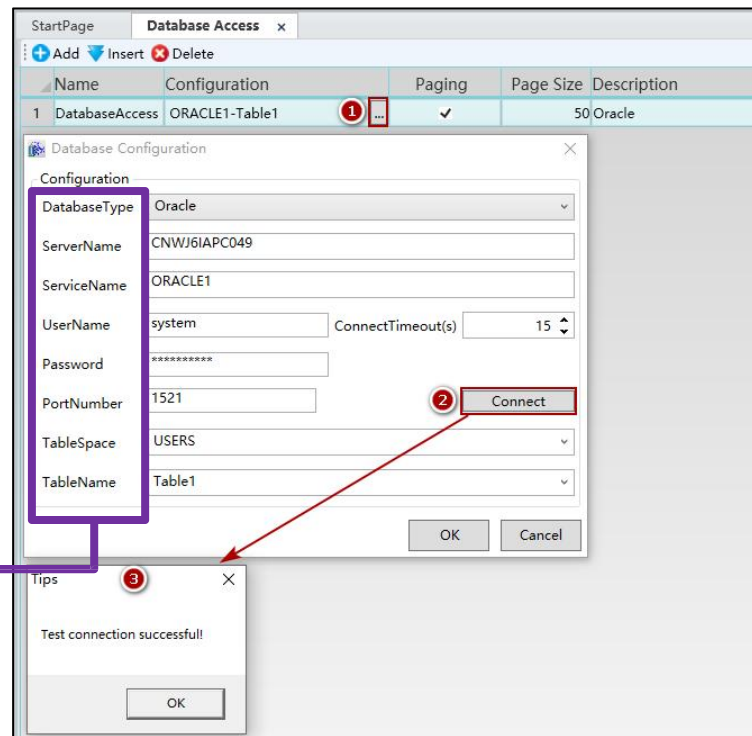
➤ Конфігурація бази даних Oracle:

Перш ніж отримати доступ до бази даних Oracle, забезпечити зв'язок між DIAView and Database успішно виконано



► Конфігурація бази даних

- **ServerName:** ім'я або IP-адреса хоста, на якому знаходиться OracleDatabase
- **ServiceName:** назва служби бази даних Oracle
- **UserName:** ім'я користувача облікового запису користувача в OracleDatabase
- **Пароль:** пароль облікового запису користувача в OracleDatabase
- **ConnectTimeout:** за замовчуванням 15 секунд
- **PortNumber:** номер порту, який використовується базою даних Oracle, за замовчуванням 1521
- **TableSpace:** табличний простір, до якого потрібно отримати доступ, якщо він порожній, виберіть табличний простір за замовчуванням
- **TableName:** назва таблиці даних, до якої потрібно отримати доступ
- **Підключитися:** перевірте з'єднання з базою даних і поверніть «Тестове з'єднання успішно»

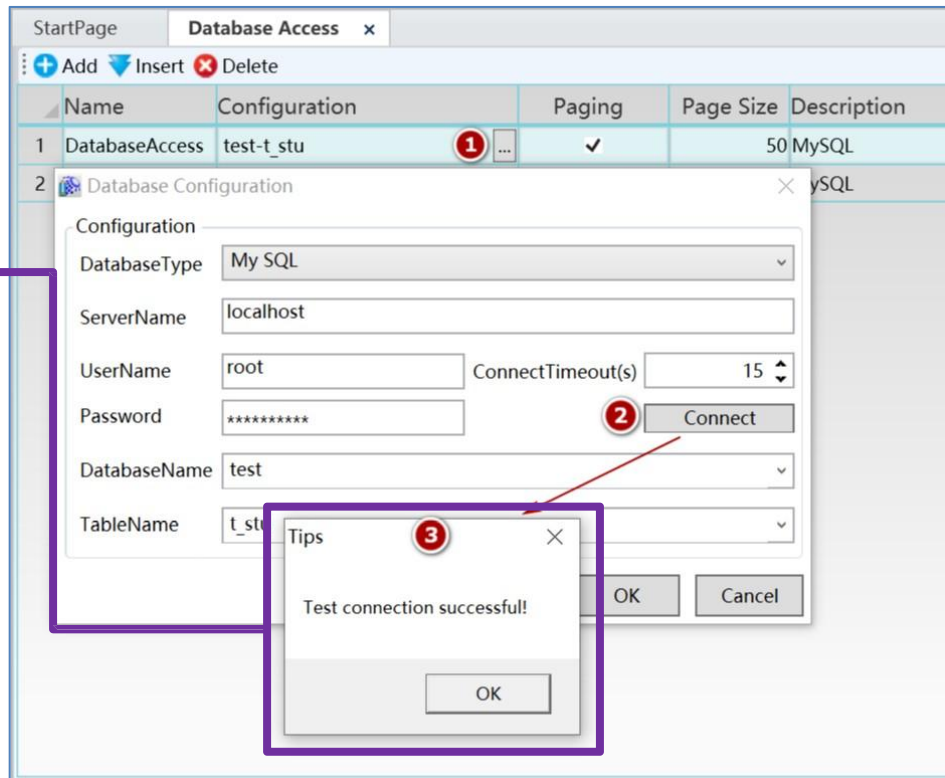


➤ Моя конфігурація бази даних SQL:

Перш ніж отримати доступ до My SQL Database,

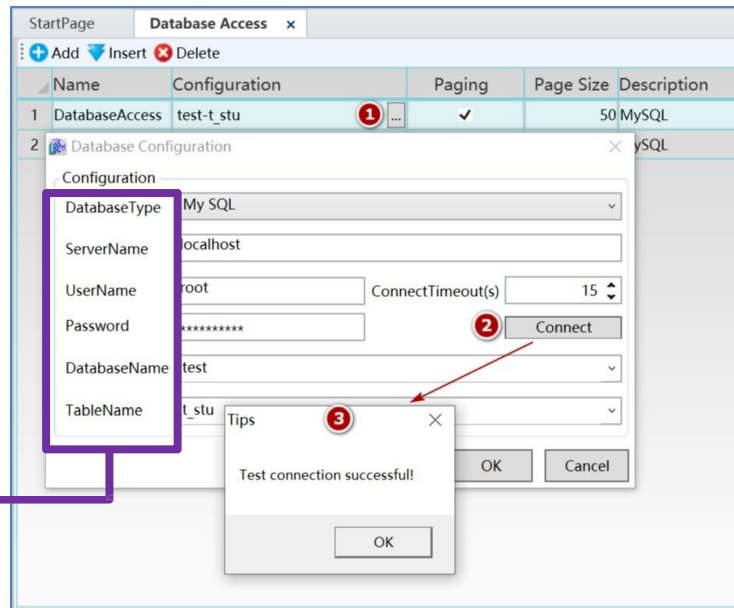
забезпечити зв'язок між

DIAMView and Database успішно
ВИКОНАНО



► Конфігурація бази даних

- **ServerName:** ім'я або IP-адреса хоста, де знаходиться Моя база даних SQL
знаходиться
- **UserName:** ім'я користувача облікового запису користувача в My SQL Database
- **Пароль:** пароль облікового запису користувача в My SQL Database
- **ConnectTimeout:** за замовчуванням 15 секунд
- **DatabaseName:** Ім'я бази даних, до якої потрібно отримати доступ
- **TableName:** назва таблиці даних, до якої потрібно отримати доступ
- **Підключитися:** перевірте з'єднання з базою даних і поверніть «Тестове з'єднання успішний»



- Доступ до бази даних
- Конфігурація доступу до бази даних
- Конфігурація бази даних
- Операція з базою даних



Доступ до бази даних – методи та властивості

- Доступ до бази даних надає методи й атрибути для доступу, запитів і редагування даних у зовнішній базі даних. Користувачі викликають метод доступу до бази даних у програмі сценаріїв, щоб реалізувати дані в операції Database під час виконання.

Будь ласка, зверніться до посібника користувача "20.3 Синтаксис і функції сценарію".

методи	
CheckTableIsExisted	Перевірте, чи існує таблиця даних із зазначеним ім'ям в базі даних
ExecuteDataTable	Виконати інструкцію SQL, повернути таблицю запиту
ExecuteNonQuery	Виконайте оператор SQL і поверніть число уражених рядків
ВиконатиScalar	Виконати оператор SQL, повернути перший рядок і перший стовпець
GetDataSourceRowCount	Отримайте кількість рядків у таблиці джерела даних
GetEmptyTable	Отримайте порожню таблицю даних
GetPageNum	Отримайте кількість сторінок
GetTable	Отримайте останню сторінку таблиці
GetTable	Прочитати таблицю за кількістю сторінок
Зберегти таблицю	Зберегти таблицю
Зберегти таблицю	Зберегти таблицю (зберегти вказаний стовпець)
TestConnection	Тест підключення до бази даних



Доступ до бази даних – методи та властивості

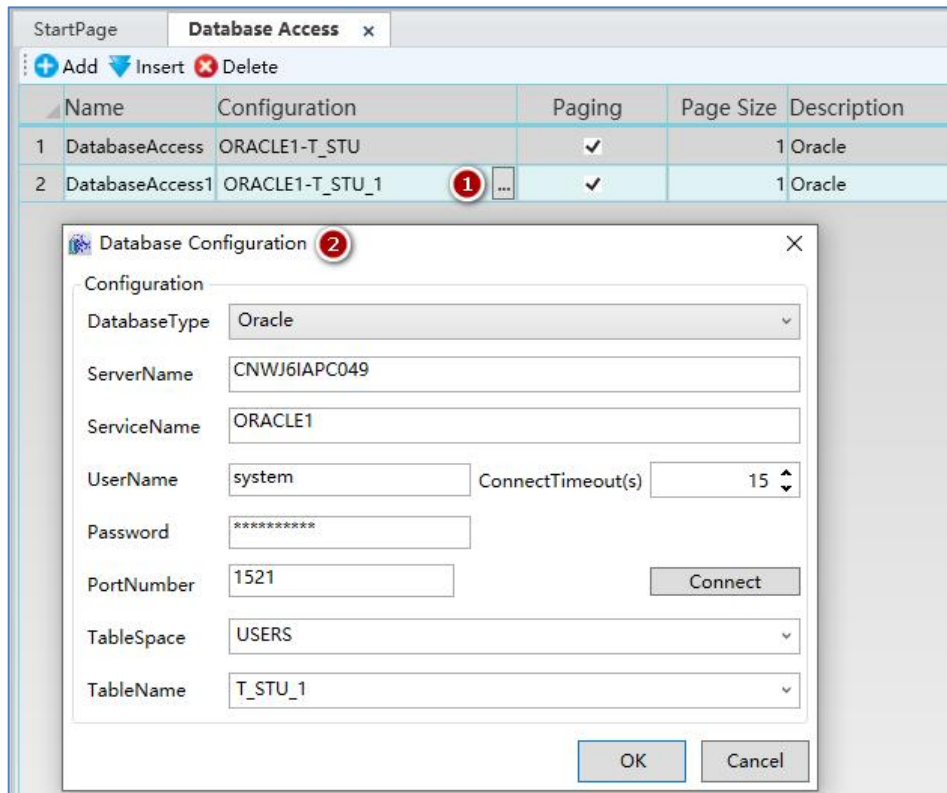
методи	
AddColumn	Додати стовпець
Створити таблицю	Створити таблицю
ExecuteBatchInsertData	Вставте таблицю даних
ExecuteCheckTableIsExisted	Перевірте, чи існує таблиця бази даних із вказаним іменем у базі даних
ExecuteCreateDatabaseTable	Створіть таблицю з такою ж структурою в базі даних відповідно до наданої таблиці даних
ExecuteCreateSave	Збережіть і поверніть кількість постраждалих рядків
ExecuteDataTable	Виконайте команду та поверніть запитану таблицю даних
ExecuteDropDatabaseTable	Видалити вказані дані в базі даних
ExecuteGetTable	Прочитайте дані перших рядків підрахунку
ExecuteNonQuery	Виконайте команду, щоб повернути кількість постраждалих рядків
ExecuteNonQueryDT	Відповідно до параметрів (позначених @), щоб отримати дані з DataTable для виконання операції
ВиконатиЗберегти	Виберіть стовпець, щоб зберегти та повернути кількість постраждалих рядків
ВиконатиScalar	Виконайте команду, щоб повернути кількість постраждалих рядків
Виберіть DT	Фільтрувати за вказаним порядком сортування
SetPrimary	Встановити первинний ключ

Доступ до бази даних – методи та властивості

Властивості	
ConnectionString	Рядок підключення до бази даних
опис	опис
IsPaging	Чи розбивати сторінки
MaxPagingCount	Максимальна кількість рядків у відображеній таблиці
Ім'я	Ім'я доступу до бази даних
Провайдер	Відображення постачальника підключення до бази даних, а саме типу бази даних
TableName	Назва таблиці даних
UniquelIdentifier	Унікальний ID

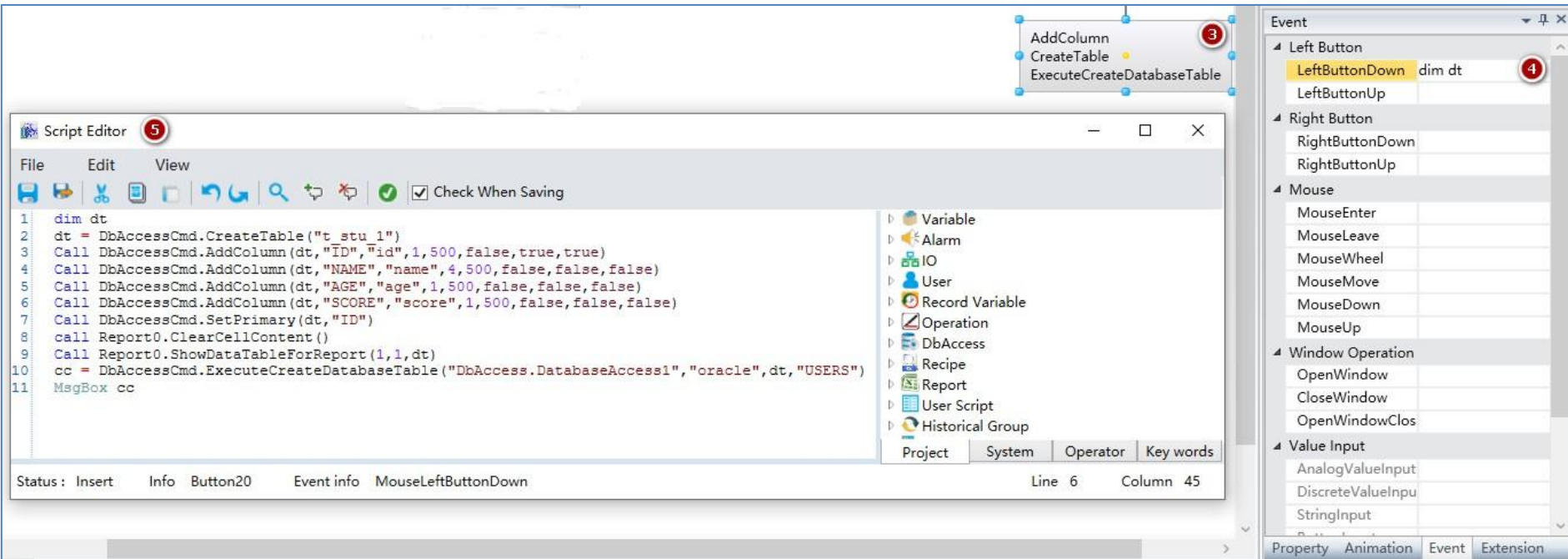
➤ Приклад. Сценарій викликає доступ до бази даних:

- ① Доступ до бази даних конфігурації
- ② Підключення до бази даних конфігурації



Доступ до бази даних - сценарій

- ③ Створіть кнопки керування.
- ④ Запустіть програму сценарію, натиснувши ліву кнопку на панелі керування.
- ⑤ Напишіть програму сценарію та викличте метод DBAccess, щоб реалізувати дані в операції Database під час виконання.



The screenshot displays the Delta IDE interface. On the left, the 'Script Editor' window (marked with a red circle 5) contains a Visual Basic script for a button click event. The script creates a database table named 't_stu_1' with columns for ID, NAME, AGE, and SCORE, and then displays the contents of the 'USERS' table in a report.

```
1 dim dt
2 dt = DbAccessCmd.CreateTable("t_stu_1")
3 Call DbAccessCmd.AddColumn(dt,"ID","id",1,500,false,true,true)
4 Call DbAccessCmd.AddColumn(dt,"NAME","name",4,500,false,false,false)
5 Call DbAccessCmd.AddColumn(dt,"AGE","age",1,500,false,false,false)
6 Call DbAccessCmd.AddColumn(dt,"SCORE","score",1,500,false,false,false)
7 Call DbAccessCmd.SetPrimary(dt,"ID")
8 call Report0.ClearCellContent ()
9 Call Report0.ShowDataTableForReport (1,1,dt)
10 cc = DbAccessCmd.ExecuteCreateDatabaseTable ("DbAccess.DatabaseAccess1","oracle",dt,"USERS")
11 MsgBox cc
```

On the right, the 'Event' panel (marked with a red circle 4) shows the 'Left Button' section with the 'LeftButtonDown' event selected. The 'Script Editor' window also shows a snippet of the event procedure (marked with a red circle 3):

```
AddColumn
CreateTable
ExecuteCreateDatabaseTable
```

The status bar at the bottom indicates the current state: 'Status: Insert', 'Info', 'Button20', 'Event info', 'MouseLeftButtonDown', 'Line 6', 'Column 45'.



Доступ до бази даних – типові приклади

➤ **Поширений приклад:**

- ① Програмування сценарію події кнопки, створення таблиці **stu_table**
- ② Виконайте сценарій події кнопки, поверніть True, щоб створити таблицю **stu_table** успішно та відобразити таблицю **stu_table** у запущеному вікні;

The screenshot shows the 'Script Editor' window with the 'CreateTable' step selected. The step is highlighted with a red arrow and a red circle containing the number '1'. The script editor shows the following code:

```

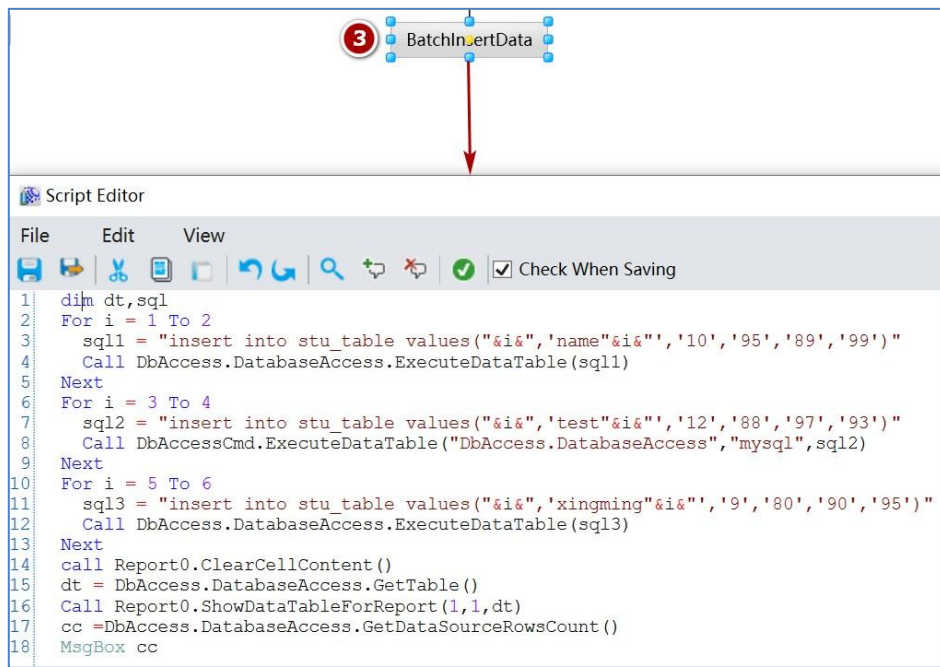
1 dim dt
2 dt = DbAccessCmd.CreateTable("stu_table")
3 Call DbAccessCmd.AddColumn(dt, "ID", "stu_id", 1, 500, false, true, true)
4 Call DbAccessCmd.AddColumn(dt, "NAME", "stu_name", 4, 500, true, false, false)
5 Call DbAccessCmd.AddColumn(dt, "AGE", "stu_age", 4, 500, true, false, false)
6 Call DbAccessCmd.AddColumn(dt, "CHINESE", "chinese", 4, 500, true, false, false)
7 Call DbAccessCmd.AddColumn(dt, "MATH", "math", 4, 500, true, false, false)
8 Call DbAccessCmd.AddColumn(dt, "ENGLISH", "english", 4, 500, true, false, false)
9 Call DbAccessCmd.SetPrimary(dt, "ID")
10 call Report0.ClearCellContent()
11 Call Report0.ShowDataTableForReport(1, 1, dt)
12 cc = DbAccessCmd.ExecuteCreateDatabaseTable("DbAccess.DatabaseAccess", "mysql", dt)
13 MsgBox cc
  
```

[illegible]

Доступ до бази даних – типові приклади

③ Програмування сценарію події кнопки для додавання кількох даних до таблиці **stu_table** ;

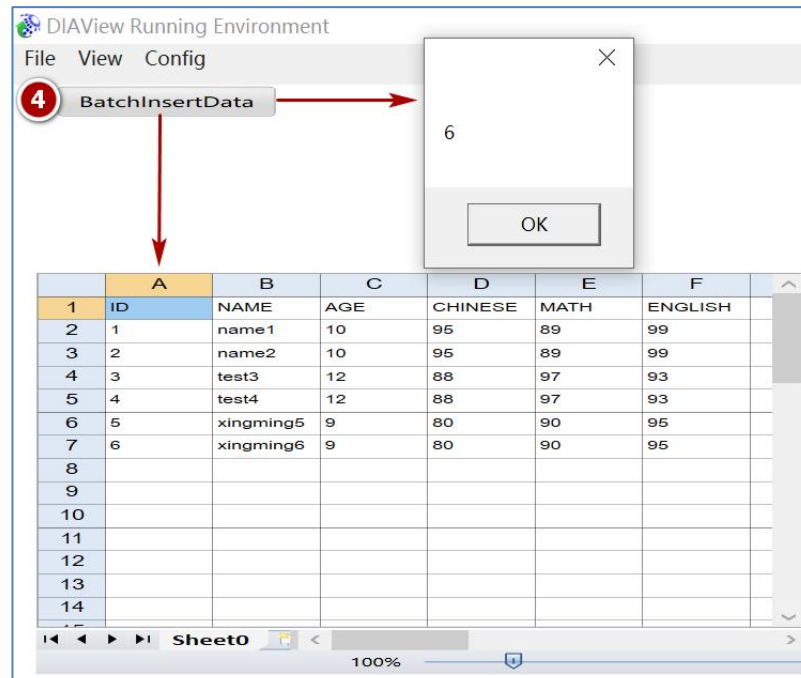
④ Виконайте сценарій події кнопки, поверніть кількість доданих даних в і відобразіть таблицю **stu_table** в ходове вікно;



Script Editor

```

1 dim dt,sql
2 For i = 1 To 2
3     sql1 = "insert into stu_table values("&i&","&name&i&","&10&","&95&","&89&","&99&")"
4     Call DbAccess.DatabaseAccess.ExecuteDataTable(sql1)
5 Next
6 For i = 3 To 4
7     sql2 = "insert into stu_table values("&i&","&test&i&","&12&","&88&","&97&","&93&")"
8     Call DbAccessCmd.ExecuteDataTable("DbAccess.DatabaseAccess","mysql",sql2)
9 Next
10 For i = 5 To 6
11     sql3 = "insert into stu_table values("&i&","&xingming&i&","&9&","&80&","&90&","&95&")"
12     Call DbAccess.DatabaseAccess.ExecuteDataTable(sql3)
13 Next
14 call Report0.ClearCellContent()
15 dt = DbAccess.DatabaseAccess.GetTable()
16 Call Report0.ShowDataTableForReport(1,1,dt)
17 cc =DbAccess.DatabaseAccess.GetDataSourceRowsCount()
18 MsgBox cc
    
```



DIAView Running Environment

File View Config

BatchInsertData

6

OK

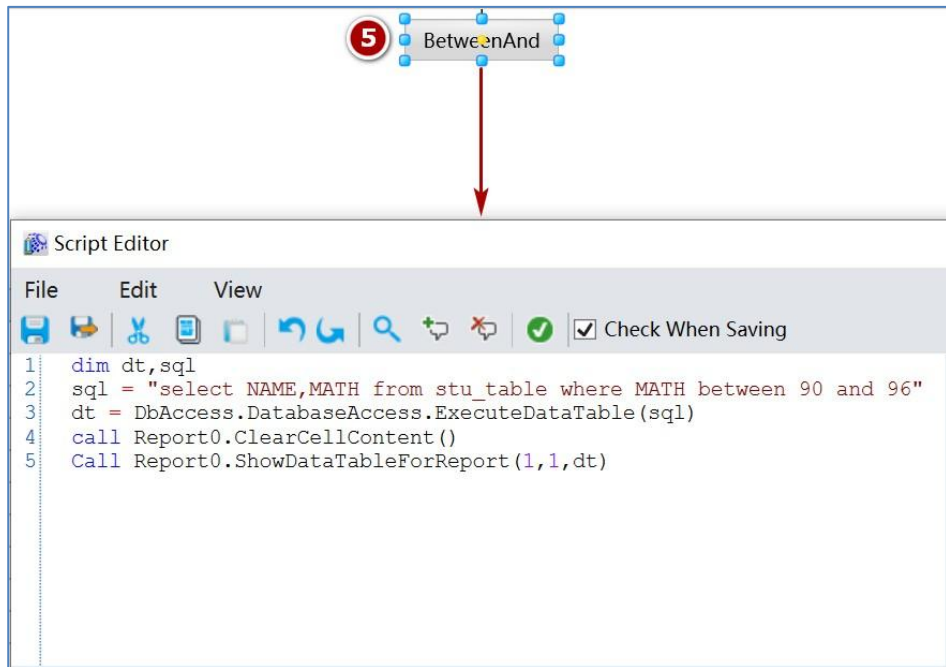
	A	B	C	D	E	F
1	ID	NAME	AGE	CHINESE	MATH	ENGLISH
2	1	name1	10	95	89	99
3	2	name2	10	95	89	99
4	3	test3	12	88	97	93
5	4	test4	12	88	97	93
6	5	xingming5	9	80	90	95
7	6	xingming6	9	80	90	95
8						
9						
10						
11						
12						
13						
14						

Sheet0

100%

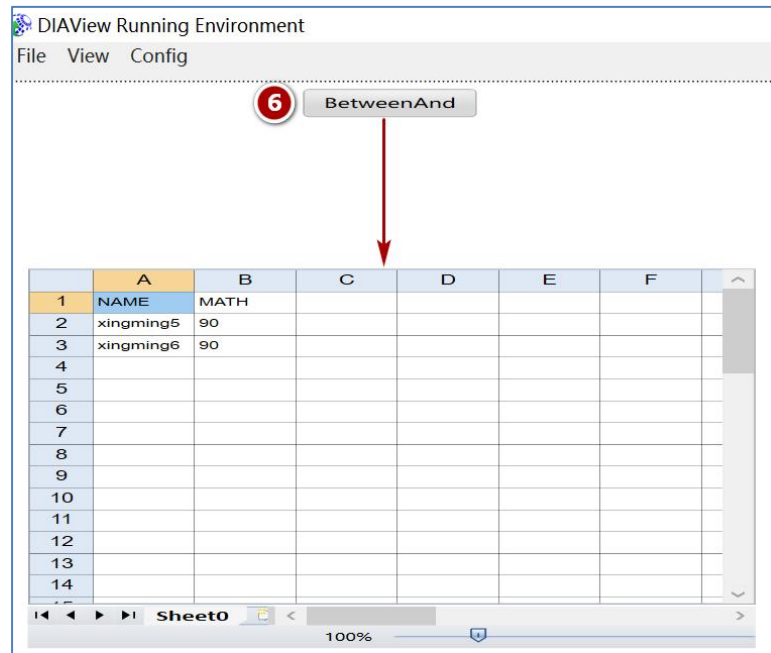
Доступ до бази даних – типові приклади

- ⑤ Скрипт подій кнопки програмування, таблиця запитів **stu_table** кваліфіковані дані;
- ⑥ Виконайте сценарій події кнопки для відображення запитуваних даних у вікні, що працює;



Script Editor

```
1 dim dt,sql
2 sql = "select NAME,MATH from stu_table where MATH between 90 and 96"
3 dt = DbAccess.DatabaseAccess.ExecuteDataTable(sql)
4 call Report0.ClearCellContent()
5 Call Report0.ShowDataTableForReport(1,1,dt)
```



DIAView Running Environment

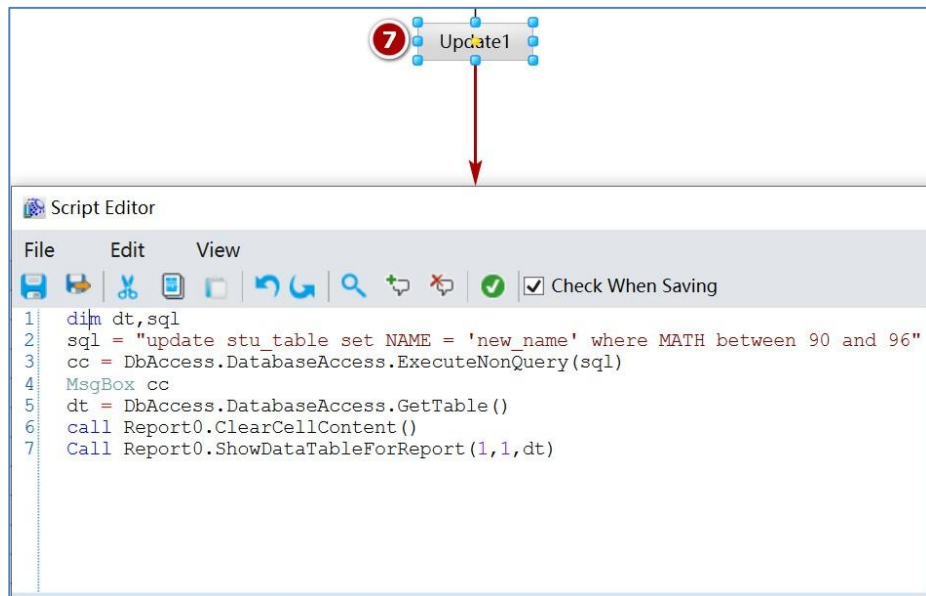
File View Config

	A	B	C	D	E	F
1	NAME	MATH				
2	xingming5	90				
3	xingming6	90				
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						

Sheet0 100%

Доступ до бази даних – типові приклади

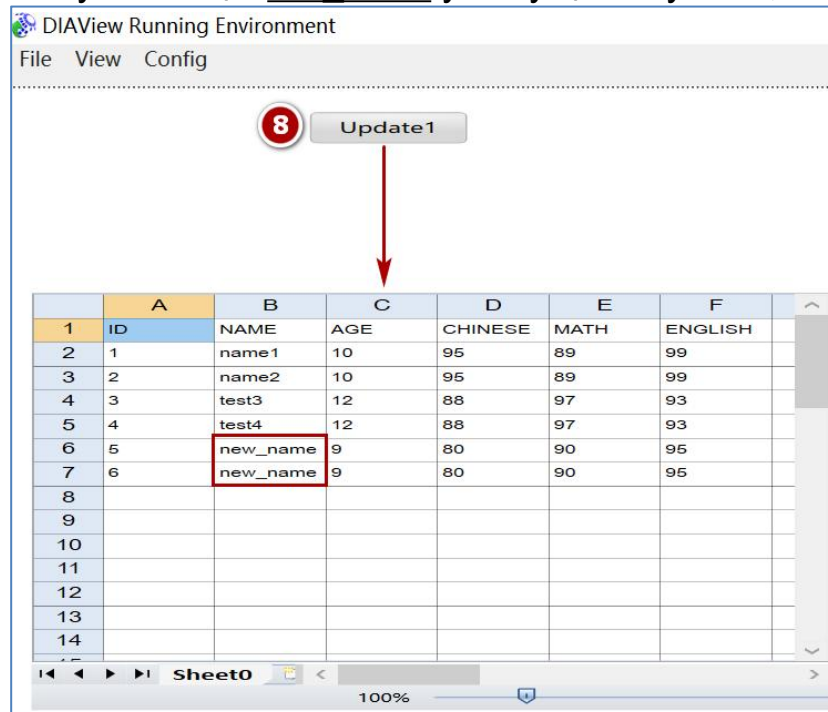
- ⑦ Програмування сценарію події кнопки для оновлення кваліфікованих даних у таблиці **stu_table** ;
- ⑧ Виконайте сценарій події кнопки та відобразіть виконану таблицю **stu_table** у запущеному вікні;



7 Update1

```

1 dim dt,sql
2 sql = "update stu_table set NAME = 'new_name' where MATH between 90 and 96"
3 cc = DbAccess.DatabaseAccess.ExecuteNonQuery(sql)
4 MsgBox cc
5 dt = DbAccess.DatabaseAccess.GetTable()
6 call Report0.ClearCellContent()
7 Call Report0.ShowDataTableForReport(1,1,dt)
    
```



8 Update1

	A	B	C	D	E	F
1	ID	NAME	AGE	CHINESE	MATH	ENGLISH
2	1	name1	10	95	89	99
3	2	name2	10	95	89	99
4	3	test3	12	88	97	93
5	4	test4	12	88	97	93
6	5	new_name	9	80	90	95
7	6	new_name	9	80	90	95
8						
9						
10						
11						
12						
13						
14						

Sheet0 100%

Доступ до бази даних – типові приклади

- ⑨ Напишіть сценарій події кнопки для видалення кваліфікованих даних у таблиці **stu_table** ;
- ⑩ Виконайте сценарій події кнопки та відобразіть виконану таблицю **stu_table** у запущеному вікні;

9 Delete...Where

Script Editor

File Edit View

Check When Saving

```

1 dim dt,sql
2 sql1 = "delete from stu_table where id < 3;"
3 Call DbAccess.DatabaseAccess.ExecuteDataTable(sql1)
4 dt = DbAccess.DatabaseAccess.GetTable()
5 call Report0.ClearCellContent()
6 Call Report0.ShowDataTableForReport(1,1,dt)|
    
```

DIAMView Running Environment

File View Config

10 Delete...Where

	A	B	C	D	E	F
1	ID	NAME	AGE	CHINESE	MATH	ENGLISH
2	3	test3	12	88	97	93
3	4	test4	12	88	97	93
4	5	new_name	9	80	90	95
5	6	new_name	9	80	90	95
6						
7						
8						
9						
10						
11						
12						
13						
14						

Sheet0 100%

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Багатомовні налаштування

朱明杰

2020/06/12



Контур

- Багатомовний огляд
- Багатомовна конфігурація
- Сценарій глобалізації

призначення

Після цього розділу ви дізнаєтеся...

- ... Основні операції та керування кількома мовами

- ... Використання глобальних скриптів

Контур

- Багатомовний огляд
- Багатомовна конфігурація
- Сценарій глобалізації

➤ DIAView Багатомовний огляд

- Глобалізація надає користувачам можливість витягувати вміст кожного функціонального модуля в багатомовний список ресурсів під час розробки та редагувати витягнуті ресурси для досягнення функції вільного перемикання між кількома мовами на екрані **виконання** .
- Наразі підтримувані типи багатомовних ресурсів: **рядок, зображення, звук** .

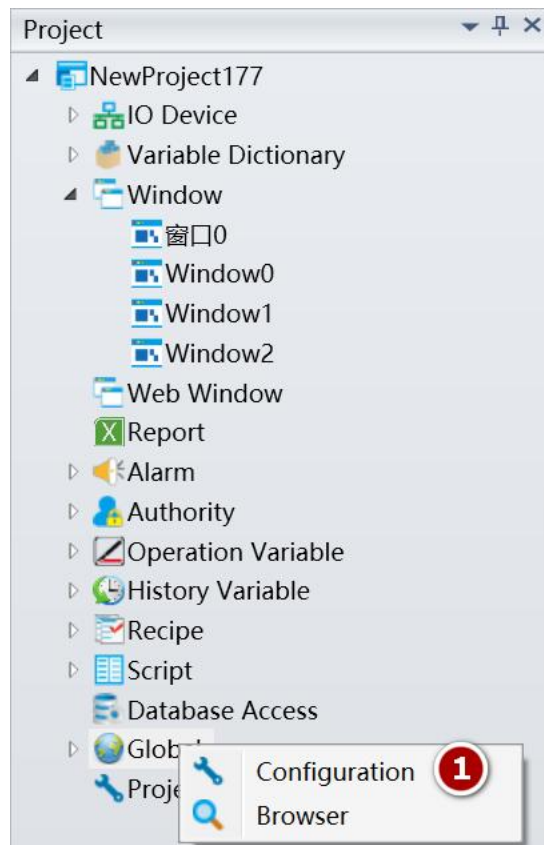
- Багатомовний огляд
- Багатомовна конфігурація
- Сценарій глобалізації

➤ **Приклад базової роботи багатомовної конфігурації:**

- ① Клацніть правою кнопкою миші значок «Глобальний», щоб відкрити

Інтерфейс налаштування «Конфігурація».

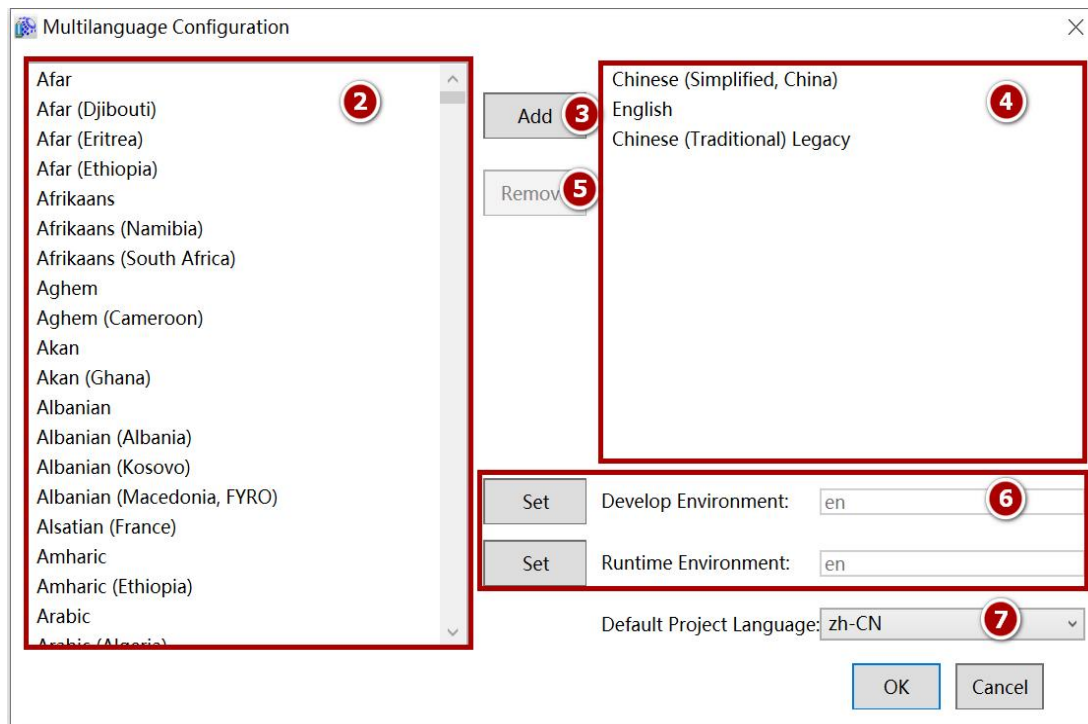
Багатомовна конфігурація



➤ Приклад базової роботи

багатомовної конфігурації:

- ② Виберіть мову, яку потрібно додати, на панелі вибору мови у вікні «Багатомовна конфігурація».
- ③ Натисніть кнопку «ДОДАТИ», щоб додати потрібну мову до глобальної мовної панелі.
- ④ Глобальний тип мови налаштування мовної панелі завершено.
- ⑤ Виберіть мову в полі глобальної мови та натисніть кнопку «Видалити», щоб видалити непотрібну мову.

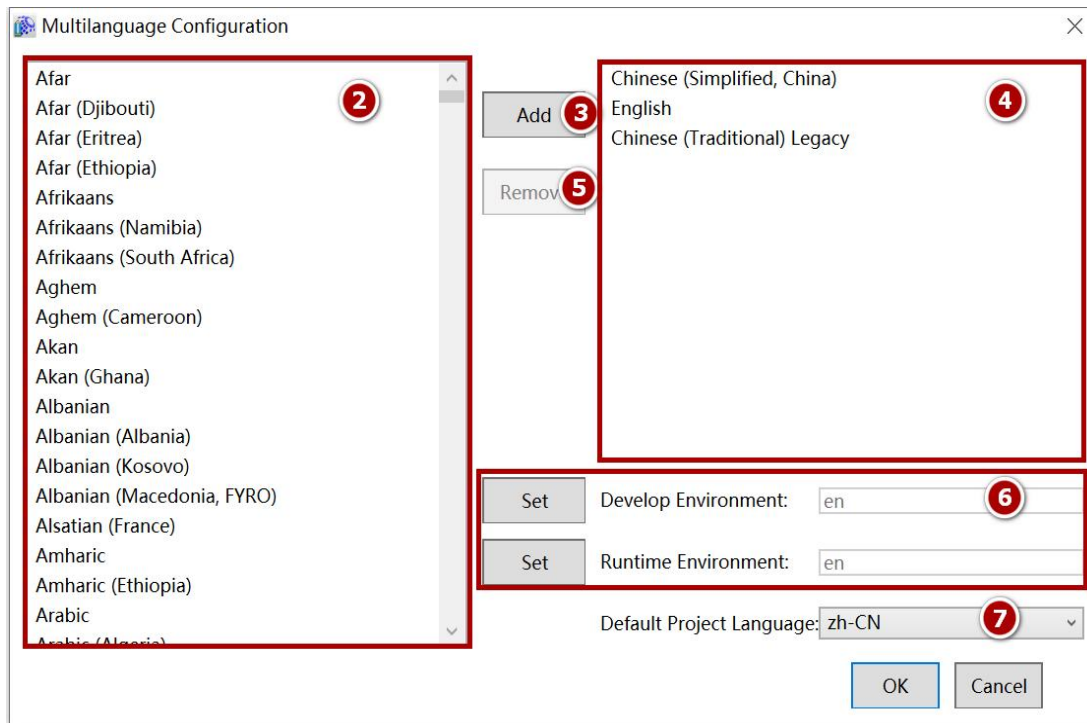


➤ Приклад базової роботи

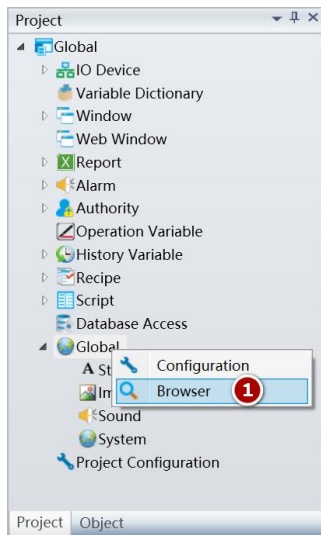
багатомовної конфігурації:

⑥ Виберіть мову в стовпці глобальної мови та натисніть «Установити» ліворуч серед середовища розробки та середовища виконання, щоб установити мову середовища за замовчуванням.

⑦ Виберіть мову в стовпці стандартної мови проекту, щоб установити мову проекту за замовчуванням.
Наразі попередньо встановлена мова підтримує китайську спрощену, китайську традиційну та англійську.



➤ Перегляд багатомовних цитат

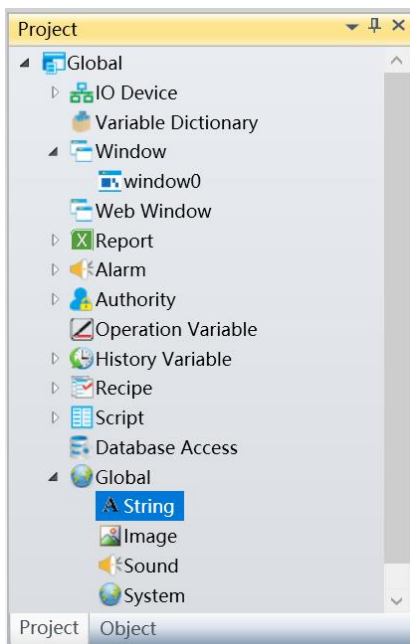


① Клацніть правою кнопкою миші піктограму «Глобальний», щоб відкрити вікно перегляду багатомовних довідкових матеріалів **«Браузер»**.

- Ліворуч від таблиці перегляду багатомовних довідкових матеріалів є багатомовний деревовидний каталог, а праворуч — вікно відображення багатомовних довідкових ресурсів.
- У деревоподібному каталозі ліворуч круглий символ перед ресурсом **«ЖОВТИЙ»**, що вказує на те, що на ресурс є посилання, і «сірий» означає, що на нього не було посилання.
- Виберіть ресурс, який потрібно переглянути, у каталозі дерева ліворуч, а вікно відображення багатомовного посилання праворуч відобразить усі об'єкти в проекті, які використовують ресурс.

➤ Багатомовна конфігурація рядків

- Рядок є бібліотекою ресурсів **текстового типу** у функції глобалізації, і користувач може налаштувати його для редагування.



StartPage String x					
+ Add ▼ Insert ✖ Delete ✓ Clean ↶ Import ↷ Export Filter:					
	Name	Chinese (Simplified, C	Chinese (Traditional)	English	
1	String1	按鈕0	按鈕0	Button0	
2	String2	按鈕1	按鈕1	Button1	
3	String3	橢圓0	橢圓0	Ellipse0	
4	String4	配方0	配方0	Recipe0	
5	String5				

Для базової роботи з багатомовними рядками символів, зверніться до розділу «18.4 Рядки символів» у посібнику користувача.

➤ Багатомовна конфігурація зображення

- Зображення є ресурсною бібліотекою шляху зображення в умовах глобалізації функцію, і користувач може налаштувати та редагувати її.

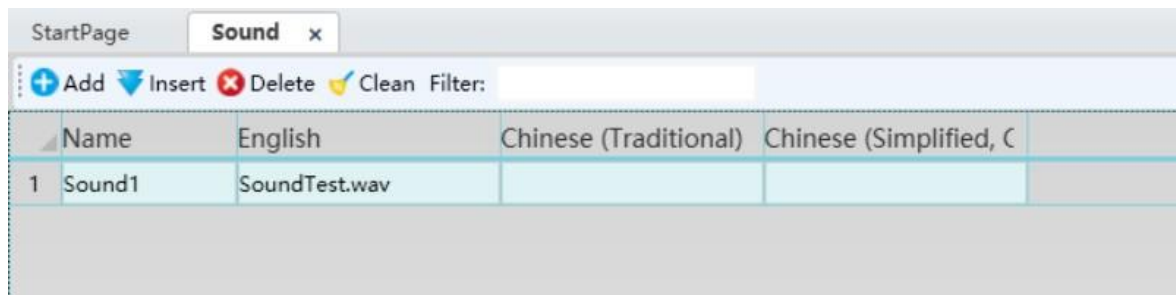


StartPage		Image x			
		+ Add + Insert ✖ Delete ✓ Clean Filter:			
	Name	Chinese (Traditional)	English	Chinese (Simplified, C	Akan
1	Image1	DefaultImage.jpg	桌面.jpg	57r58PICPiY_1024.jpg	timg1.jpg
2	Image2	DefaultGif.gif	timg.gif	timgBBK7H1YB.gif	timg1.gif

Основні операції багатомовного зображення див. у «18.5 Зображення» в посібнику користувача.

➤ Багатомовна конфігурація звуку

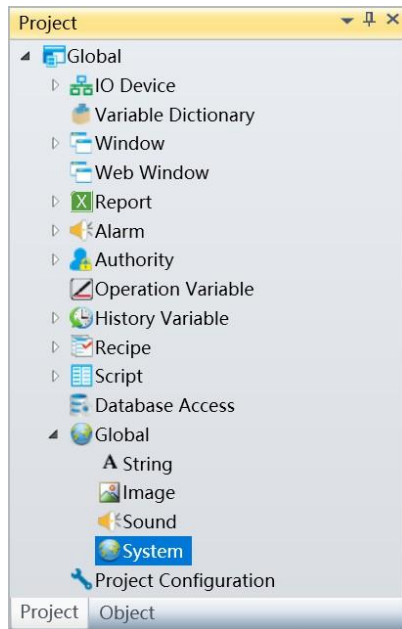
- **Звук** - це бібліотека звукових ресурсів у функції глобалізації та користувачів
можна налаштувати його.



Щоб дізнатися про базові операції багатомовного звуку, див на «18.6 Sound» у посібнику користувача.

➤ Багатомовна конфігурація роботи системи

- **Багатомовність системи** означає, що **системний текст** у вікні відображається вказаною мовою. Багатомовність системи включає вміст рядка меню та розкритого списку розширених елементів керування, а також рядок меню під час виконання.

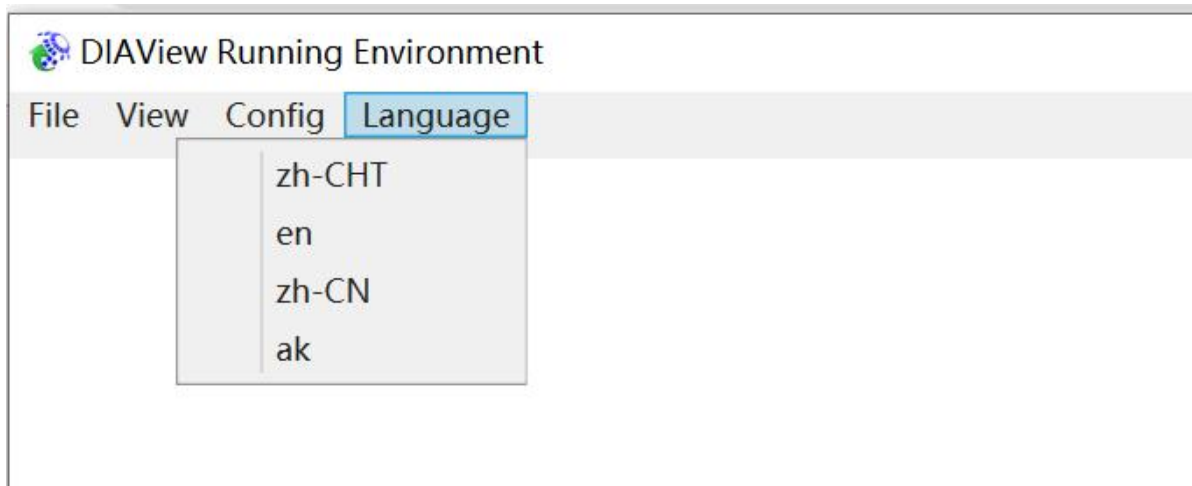


StartPage Image Sound System x				
Import Export Filter:				
	Name	Chinese (Traditional)	English	Chinese (Simplified, C Akan
1	HMI_WebBrowse	就緒	Be Ready	就绪
2	HMI_WebBrowse	完成	Finish	完成
3	HMI_WebBrowse	響應.....	Response.....	响应.....
4	HMI_WebBrowse	網址	Address	地址
5	HMI_WebBrowse	後退	GoBack	后退
6	HMI_WebBrowse	前進	GoForward	前进
7	HMI_WebBrowse	轉到	GoSearch	转到
8	HMI_WebBrowse	首頁	HomePage	主页

Для базової роботи з багатомовними звуками, будь ласка зверніться до посібника користувача "18.7 System Multi-language".

➤ Операція перемикавання мови під час виконання

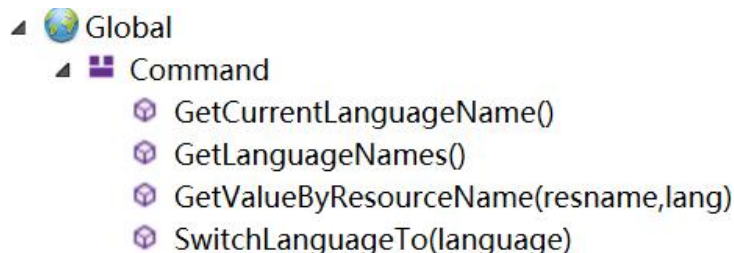
- Коли система працює, користувачі можуть перемикатися між кількома мовами в такими способами:



- Багатомовний огляд
- Багатомовна конфігурація
- Сценарій глобалізації

➤ Глобальний громадський підхід

- Коли користувачі налаштовують події віконних кнопок, екранні програми та сценарії програм користувача, DIAView надає прості та зручні у використанні властивості сценаріїв і методи для виконання деяких спеціальних завдань і функцій.



Основні інструкції щодо роботи сценарію глобалізації детально описано в посібнику користувача «20.3 Синтаксис і функції сценарію → Модель програмування → Глобалізація».

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.



DIAView

Конфігурація проекту

Венді
2020/05



Контур

- Огляд конфігурації проекту
- Час виконання
- Вікно та веб-вікно
- Мережа

призначення

У цьому розділі ви дізнаєтеся....

... докладніше про конфігурацію проекту

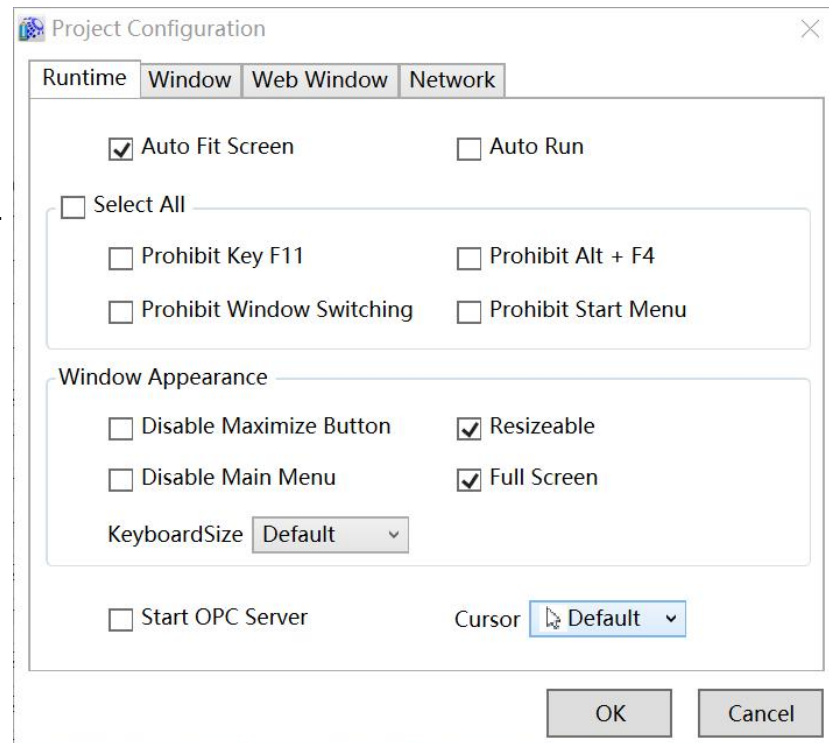
Контур

- Огляд конфігурації проекту
- Час виконання
- Вікно та веб-вікно
- Мережа

Огляд конфігурації проекту

➤ Огляд конфігурації проекту

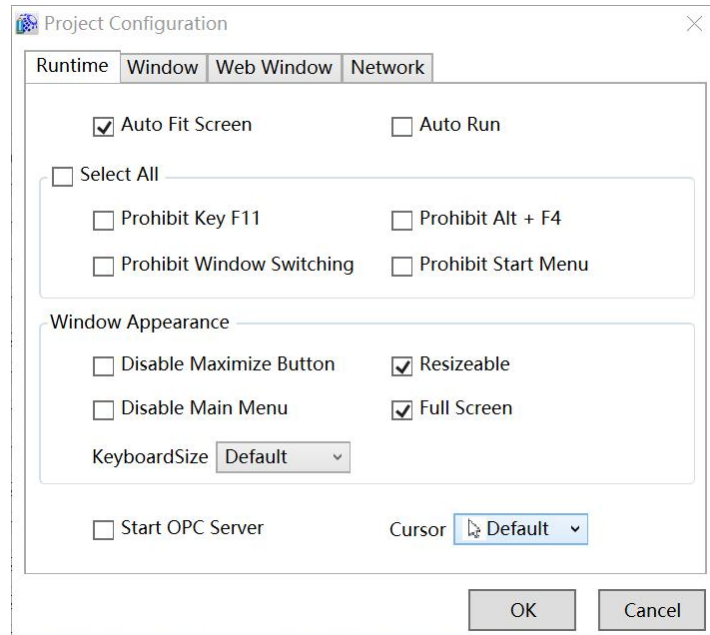
- Основною функцією конфігурації проекту є налаштування комбінацій клавіш у середовищі Runtime, екрана, відкритого під час Runtime, і конфігурації мережі.
- Гнучкий вибір екрана, відкритого під час виконання;
- Різноманітне відображення екрана, розмір клавіатури, стиль миші тощо під час впровадження Runtime;
- Підтримка комбінацій клавіш під час увімкнення/вимкнення Runtime;
- Підтримка конфігурації мережі, наприклад встановлення номера порту, пароля зв'язку, синхронізації часу обслуговування, запуску веб-сервісу тощо;



- Огляд конфігурації проекту
- Час виконання
- Вікно та веб-вікно
- Мережа

➤ Параметри виконання:

Автоматичний розмір екрана	Позначено, розмір вікна автоматично адаптується до повноекранного відображення під час виконання
Автозапуск	Перевірте, проект за замовчуванням Runtime DIAView автоматично під час запуску комп'ютера
Заборонити ключ F11	Поставте прапорець, щоб вимкнути клавішу F11 для перетворення режиму повноекранного перегляду під час виконання
Заборонити Alt + F4	Поставте прапорець і вимкніть «Alt + F4», щоб закрити проект під час виконання
Заборонити перемикання вікон	Позначено, заборонено використовувати клавіші «Alt + tab»/«Window + tab» для перемикання вікон під час виконання
Заборонити меню «Пуск».	Поставте прапорець, щоб вимкнути використання клавіші «Вікно» для відкриття меню «Пуск» під час Час виконання
Вимкнути кнопку розгортання	Позначено, кнопка розгортання недоступна під час виконання
Змінний розмір	Позначено, розмір вікна можна змінювати під час виконання
Вимкнути головне меню	Перевірте, системне меню не буде видно під час виконання
Повноекранний	Позначено, екран відображається на весь екран під час виконання
Розмір клавіатури	Стиль клавіатури, що з'являється під час виконання, Default — розмір за замовчуванням, Adapt — адаптивний екран, Full — повноекранний режим
Запустіть сервер OPC	Поставте прапорець, щоб запустити сервер OPC DA під час виконання
Курсор	Стиль миші в Runtime, Default і Hand



Project Configuration

Runtime Window Web Window Network

☒ Auto Fit Screen ☐ Auto Run

☐ Select All

☐ Prohibit Key F11 ☐ Prohibit Alt + F4

☐ Prohibit Window Switching ☐ Prohibit Start Menu

Window Appearance

☐ Disable Maximize Button ☒ Resizeable

☐ Disable Main Menu ☒ Full Screen

KeyboardSize Default

☐ Start OPC Server Cursor Default

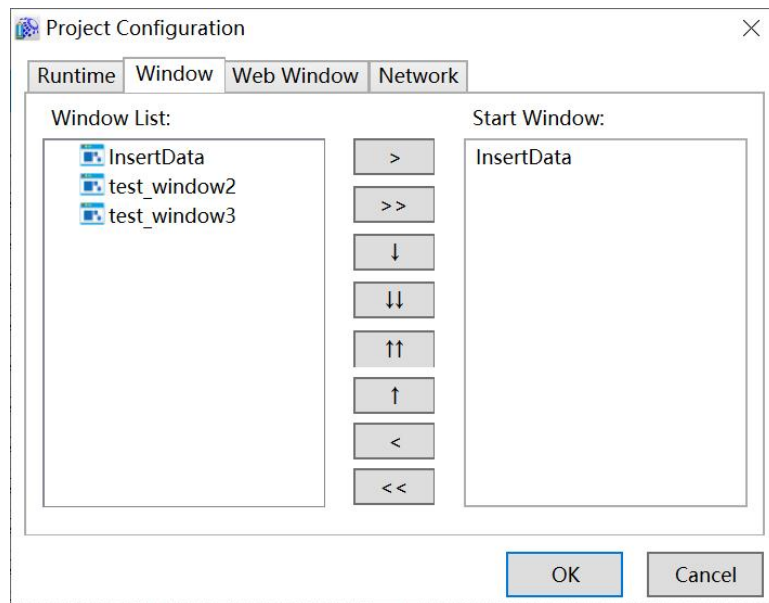
OK Cancel

- Огляд конфігурації проекту
- Час виконання
- Вікно та веб-вікно
- Мережа

➤ Вікно та веб-вікно

- Встановіть порядок відображення вікна (вікна веб-сторінки) і вікна (вікна веб-сторінки), спочатку відкритого в Runtime:

Список вікон	Список вікон, наявних у проекті
Почати вікно	Початкове вікно може бути одне або декілька, і вони відкриваються в порядку зверху вниз, а останнє відкрите вікно (нижнє вікно в списку) відображається спереду
>	Додати вибране вікно до початкового вікна
>>	Додайте всі вікна до початкового вікна
↓	Перемістити порядок вікон вниз
⇓	Перемістити вікно на останнє у списку
⇓	Перемістити вікно на перше місце у списку
↑	Перемістити вікна вгору
<	Перемістити вибране вікно з початкового вікна
<<	Перемістити всі вікна з початкового вікна

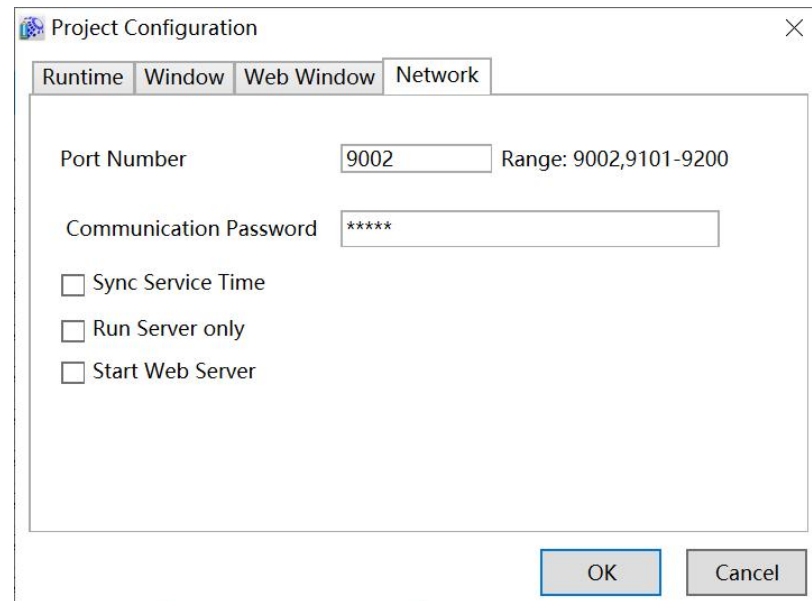


- Огляд конфігурації проекту
- Час виконання
- Вікно та веб-вікно
- Мережа

➤ Мережа

- В архітектурі C/S налаштуйте відповідну інформацію, таку як номер порту, пароль зв'язку, час синхронізації, лише службу Runtime і запустіть веб-службу.

Номер порту	Типове значення 9002, діапазон 9002, 9101-9200
Пароль зв'язку	Пароль, який використовується під час входу клієнта в службу, за замовчуванням - admin
Час служби синхронізації	Чи дозволяти клієнту встановлювати системний час локальної машини на основі часу обслуговування
Запустити лише сервер	Проект середовища виконання за замовчуванням запускає клієнт і сервер одночасно. Якщо позначено, під час виконання запускається лише сервер
Запустіть веб-сервер	Після встановлення галочки Runtime Project запуститься веб-сервіс



Project Configuration

Runtime Window Web Window Network

Port Number Range: 9002,9101-9200

Communication Password

☐ Sync Service Time

☐ Run Server only

☐ Start Web Server

OK Cancel

Розумніше. Екологічніше. Разом з нами



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"

www.delta-electronics.com.ua

вул. Святослава Хороброго, 29-А, 49001, м.Дніпро

Пошта: sales@rts.ua

Щоб дізнатися більше про Delta, відвідайте
www.deltaww.com.

