

# DIAView

Object

Ruby

2020/02/27



# Outline

- The concepts of Toolbox
- How to draw Basic Graphics
- How to draw Window Controls
- How to draw Extend Controls

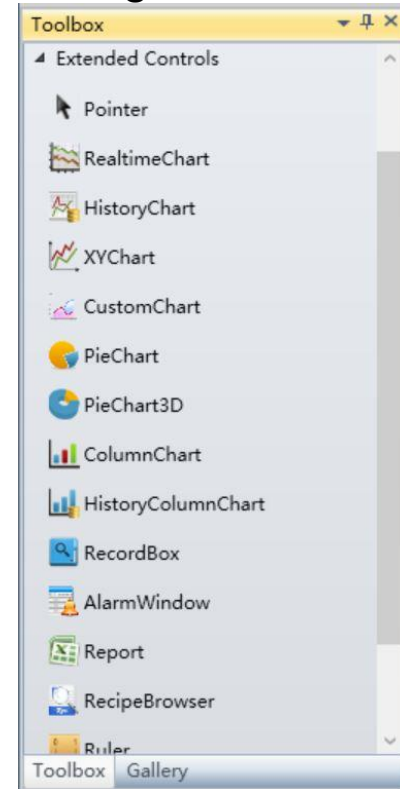
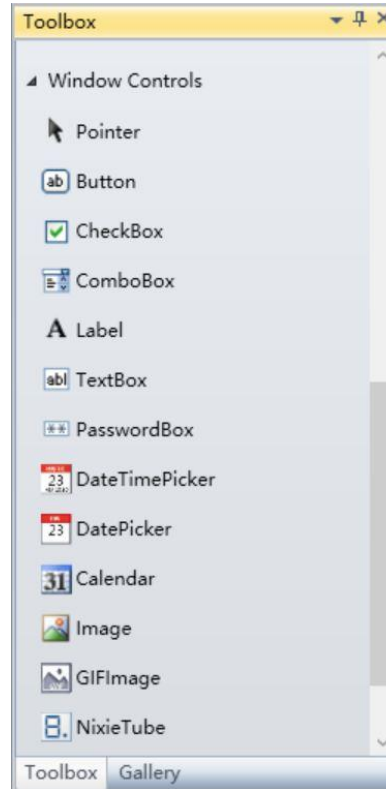
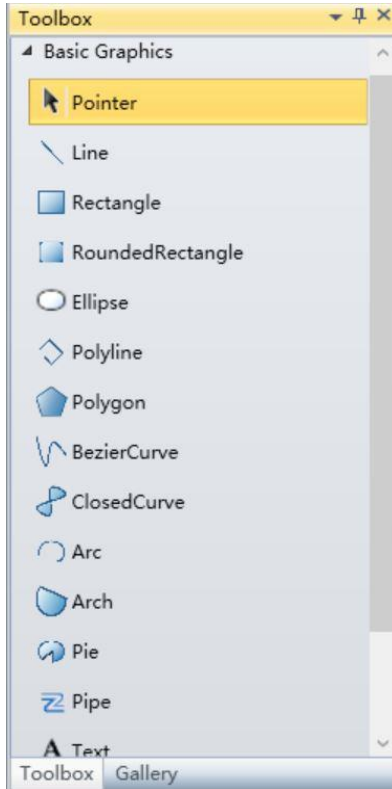
In this chapter, you will learn .....

- ... more about the Basic Graphics
- ... more about the Window Controls
- ... more about the Extend Controls

- The concepts of Toolbox
- How to draw Basic Graphics
- How to draw Window Controls
- How to draw Extend Controls

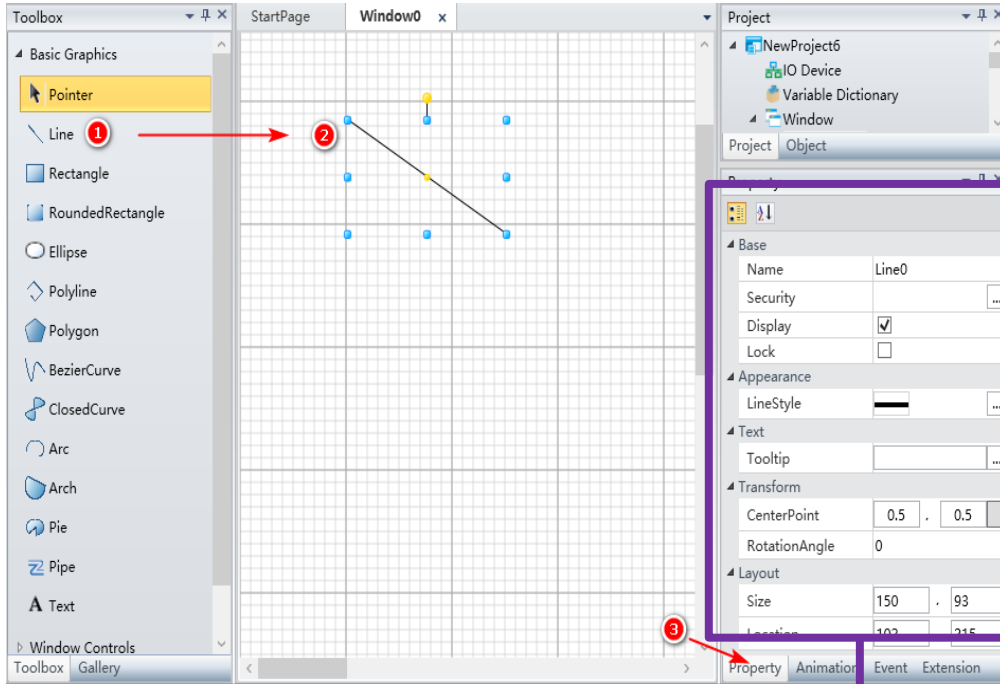
# Introduction to the Toolbox

- The Toolbox in the graphic development window contains basic graphics, window controls and extend controls. As shown in the figure below:



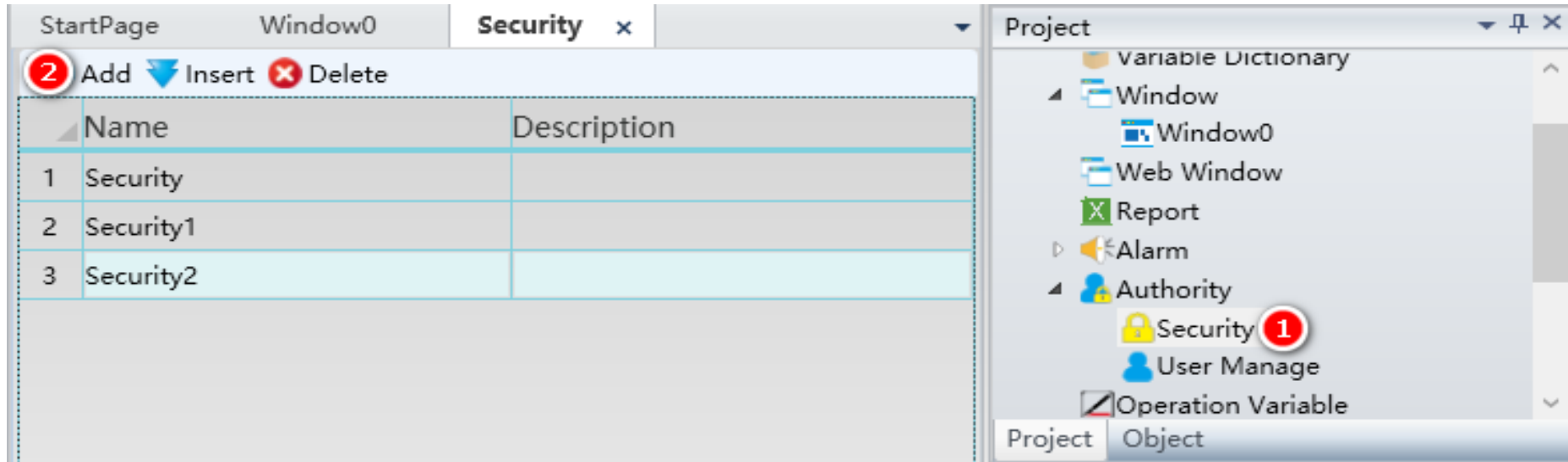
- The concepts of Toolbox
- How to draw Basic Graphics
- How to draw Window Controls
- How to draw Extend Controls

➤ Draw a Line0 , open its properties window



- **Name:** Name of the graphic in the window; the names of graphic objects in the same window cannot be same.
- **Security:** Set the operation authorities of the graphic, and make the graphic belong to one or more security zones for user authority management.
- **Display:** Whether to display the graphic during execution.
- **Lock:** Whether to lock the graphic; once locked, mouse operations will become invalid.
- **LineStyle:** Set line type, for example, style, thickness, color, etc.
- **Tootip:** Set the prompt content that the mouse stays on the drawing during execution
- **CenterPoint:** Set the location of the center point of the graphic
- **RotationAngle:** Set the rotation angle of the graphic.
- **Size:** Set size of the graphic.
- **Location:** The location of the graphic in the window.

- Security Zone example:  
(1) Create security zone



The screenshot displays the 'Security' configuration window. The 'Add' button is highlighted with a red circle containing the number 2. The table below lists the existing security zones. On the right, the 'Project' tree shows the 'Security' node under 'Authority', which is also highlighted with a red circle containing the number 1.

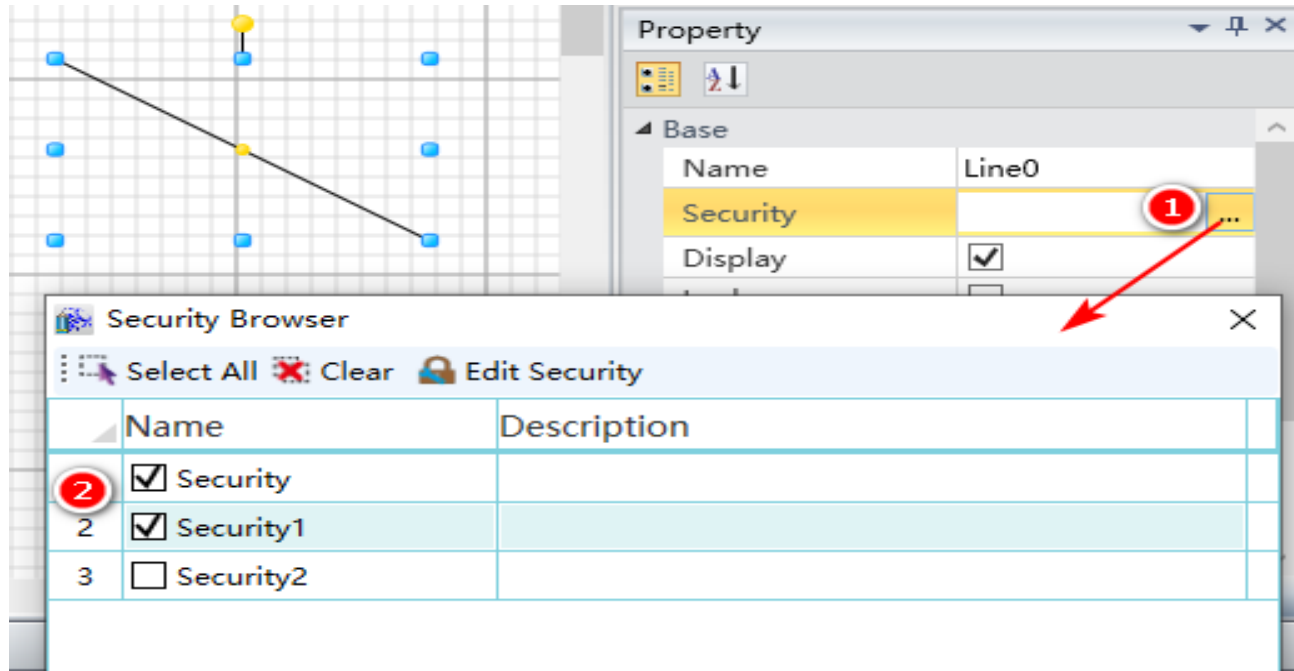
	Name	Description
1	Security	
2	Security1	
3	Security2	

Project Tree Structure:

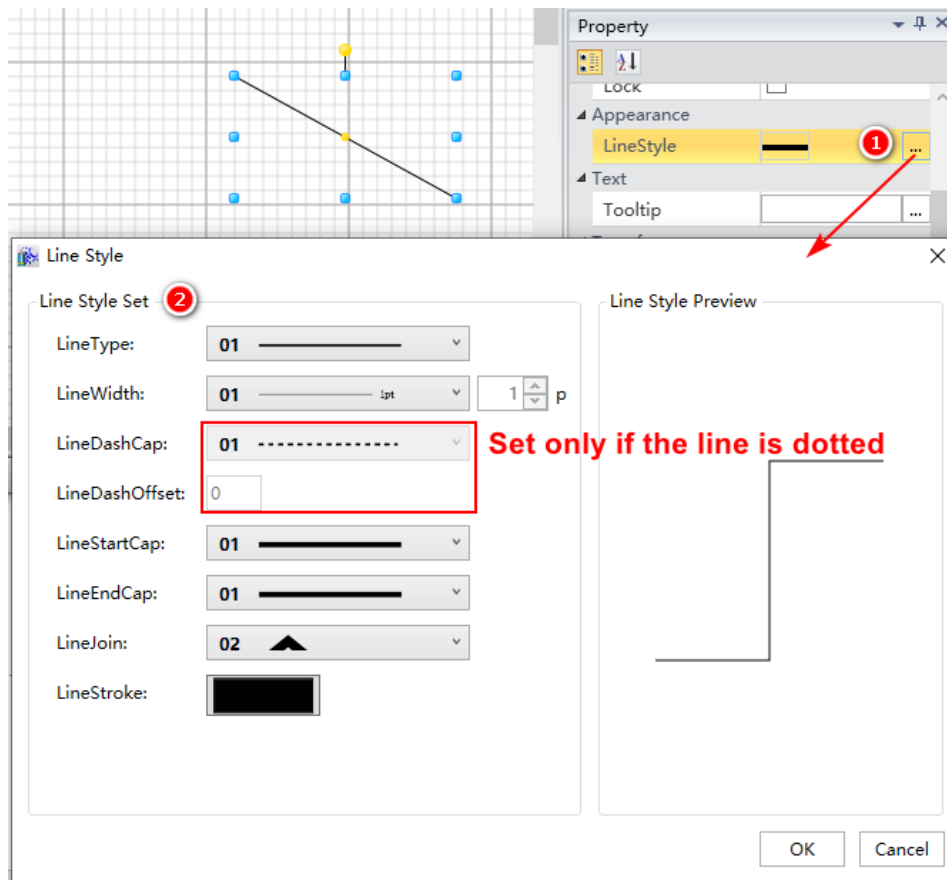
- Variable Dictionary
  - Window
    - Window0
  - Web Window
  - Report
  - Alarm
  - Authority
    - Security (1)
    - User Manage
  - Operation Variable



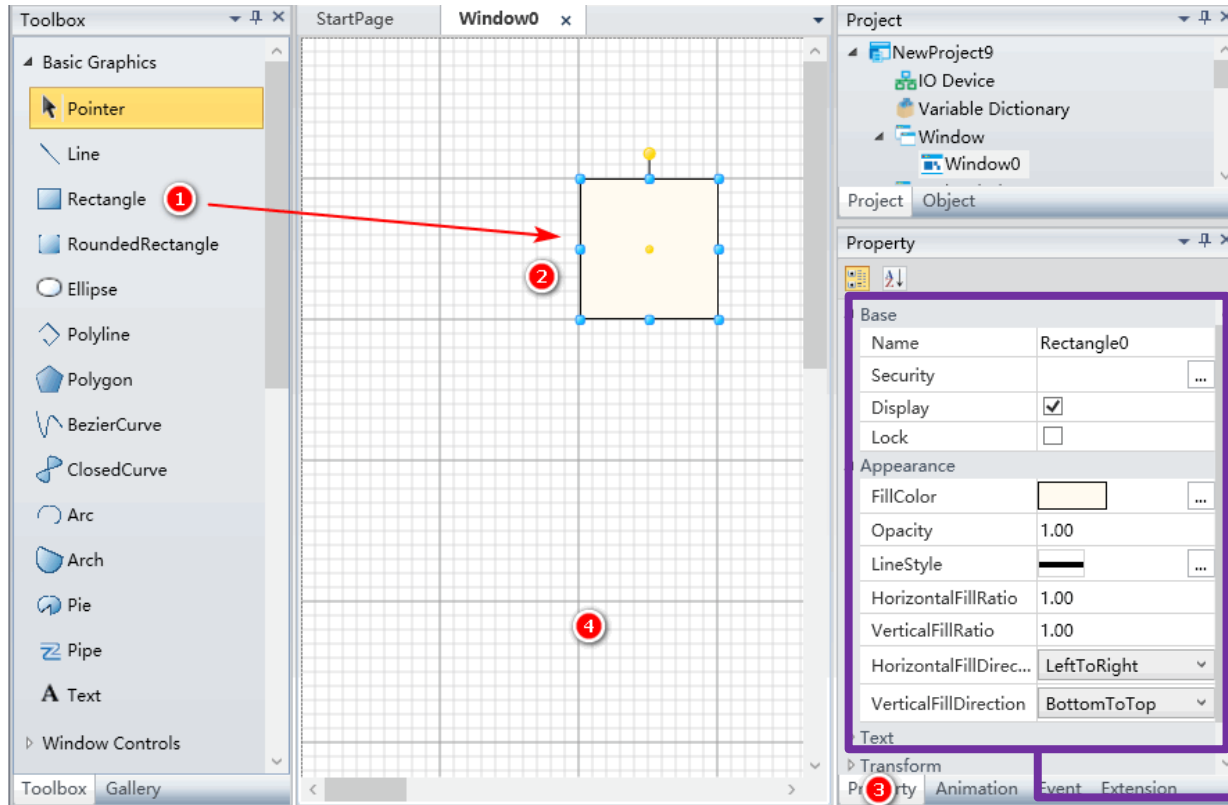
(2)The graphic associated security zone



- Set line style of the graphic



➤ Draw a Rectangle0 , open its properties window

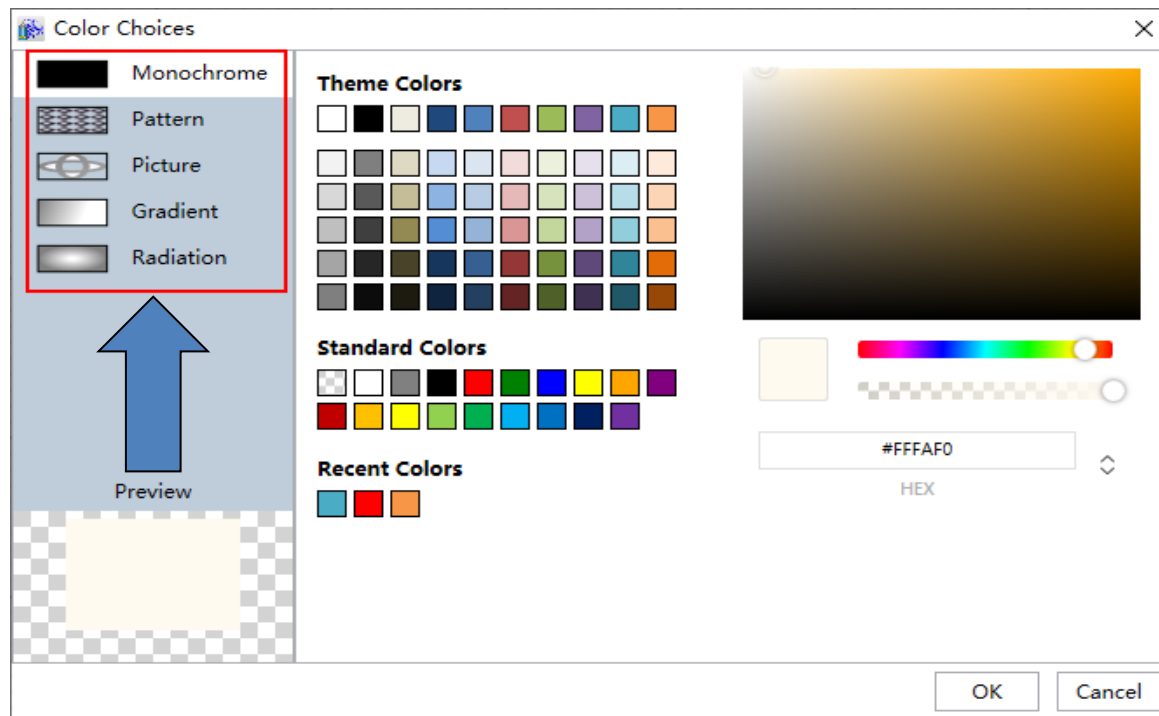


- **Fill color:** Set the fill color of rectangle
- **Opacity :** Set the transparency of the rectangle
- **LineStyle:** Set border line type, for example, style, thickness, color, etc
- **HorizontalFillRatio:** Set the horizontal fill ratio of rectangle(0~1)
- **HorizontalFillDirection:** Set the horizontal fill direction of rectangle
- **VerticalFillRatio:** Set the vertical fill ratio of rectangle(0~1)
- **VerticalFillDirection:** Set the vertical fill direction of rectangle

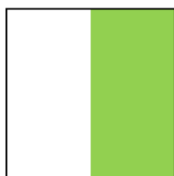
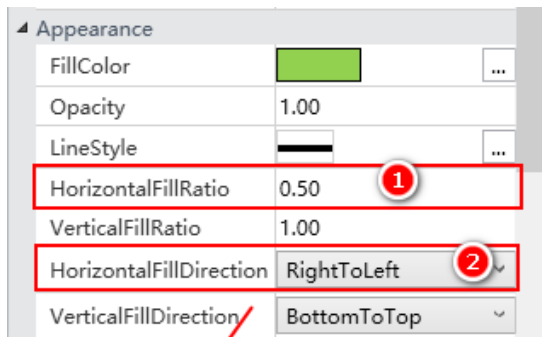
➤ There are five color fill modes in the color choices:

- ① Monochrome fill
- ② Pattern fill
- ③ Picture fill
- ④ Gradient fill
- ⑤ Radiation fill

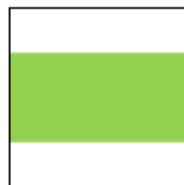
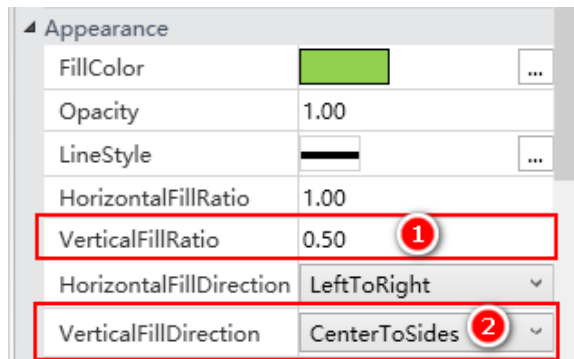
As shown in the figure below:



## ➤ Fill Ratio and Direction

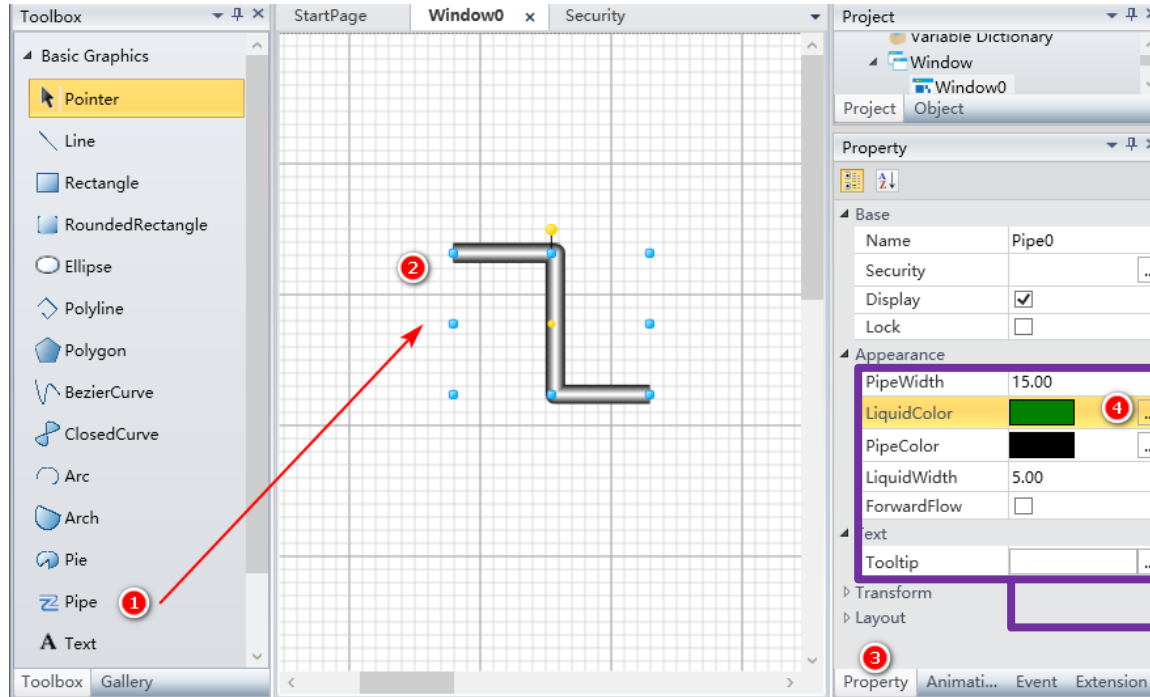


Horizontal fill ratio and direction



Vertical fill ratio and direction

- Draw a Pipe0 , open its properties window

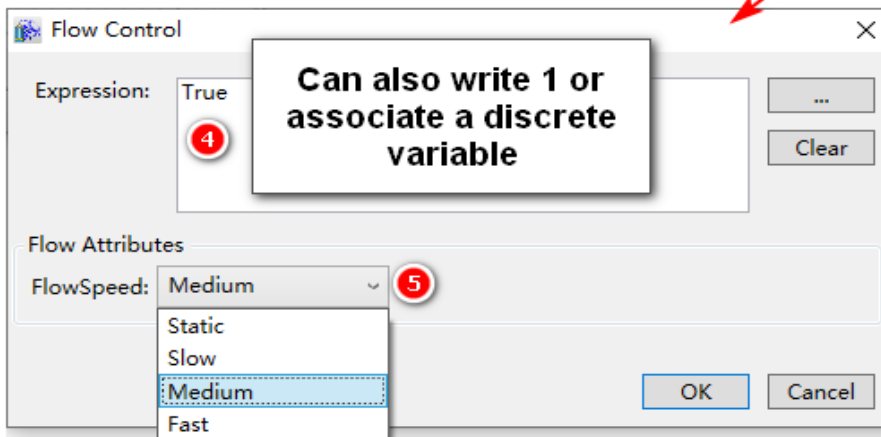
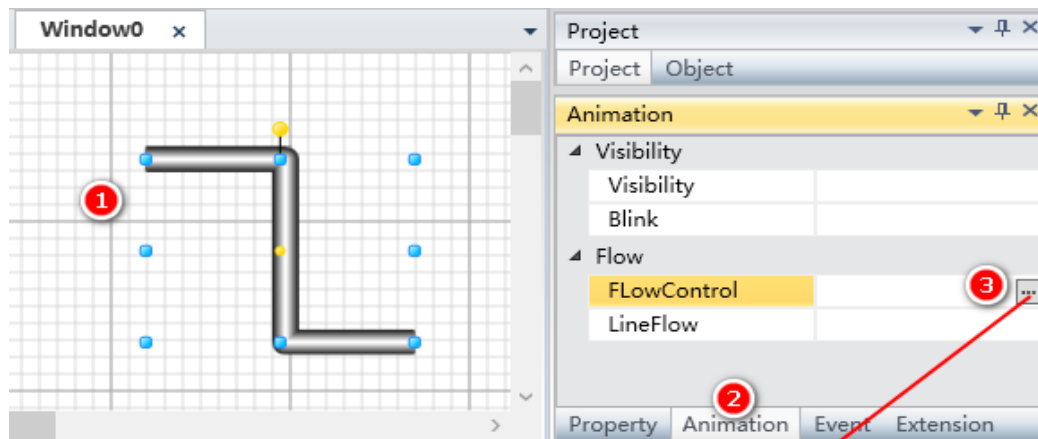


- **PipeWidth:** Set the width of the pipe.
- **LiquidColor:** Set the color of the liquid in the pipe.
- **PipeColor:** Set the color of the pipe
- **LiquidWidth:** Set the width of the liquid in the pipe
- **ForwardFlow:** Set the flow direction of the liquid in the pipe.

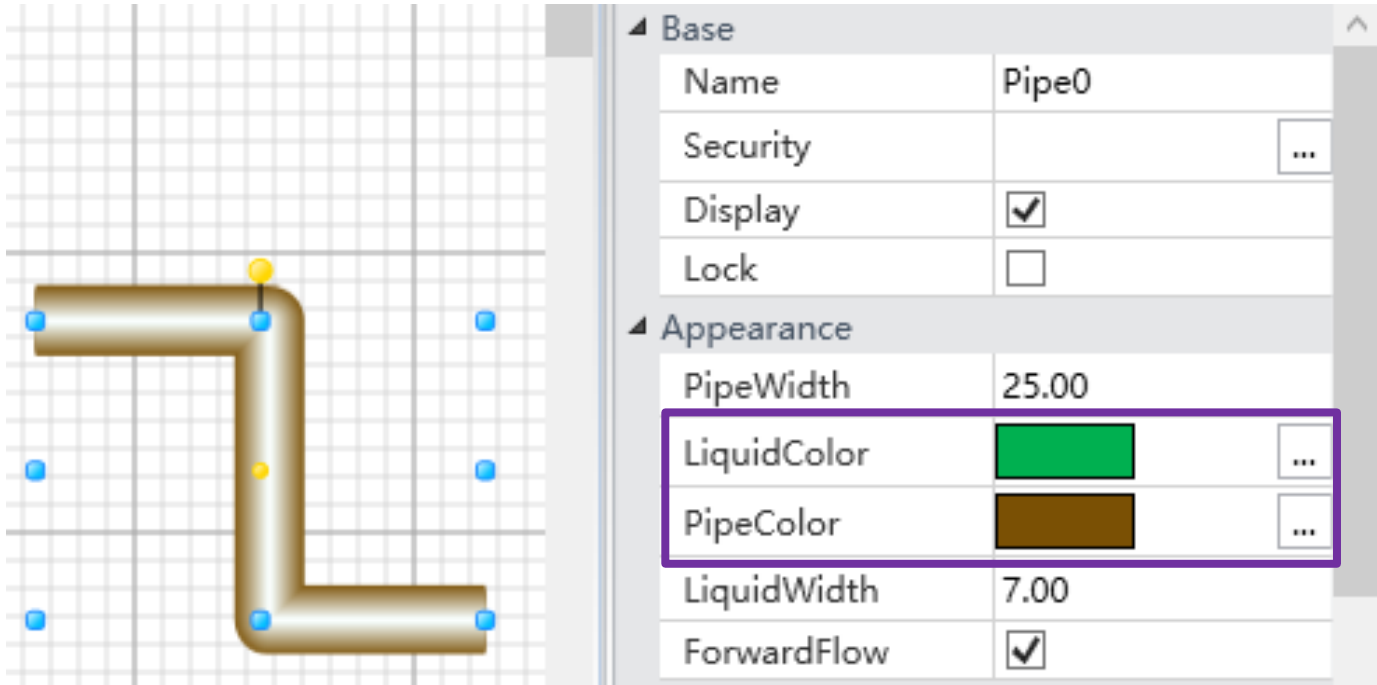
# Create a Flowing Pipe

## ➤ Flowing Pipe example:

(1) Configure the flow control animation of the pipe.

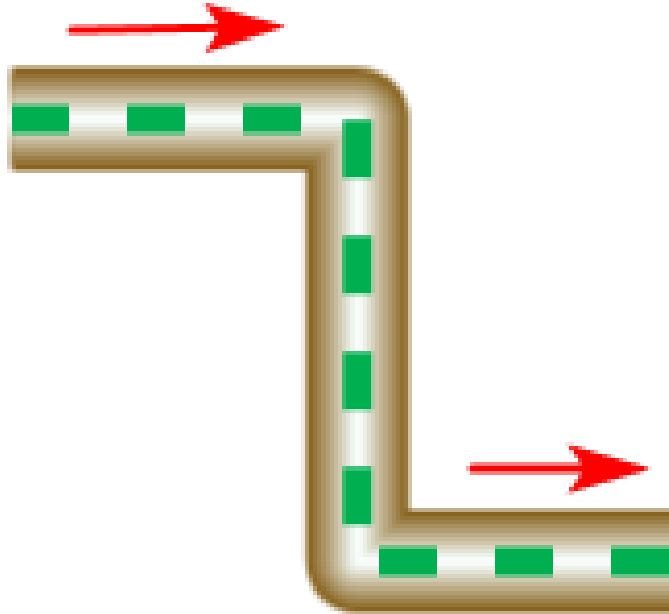


## (2) Set properties of the pipe

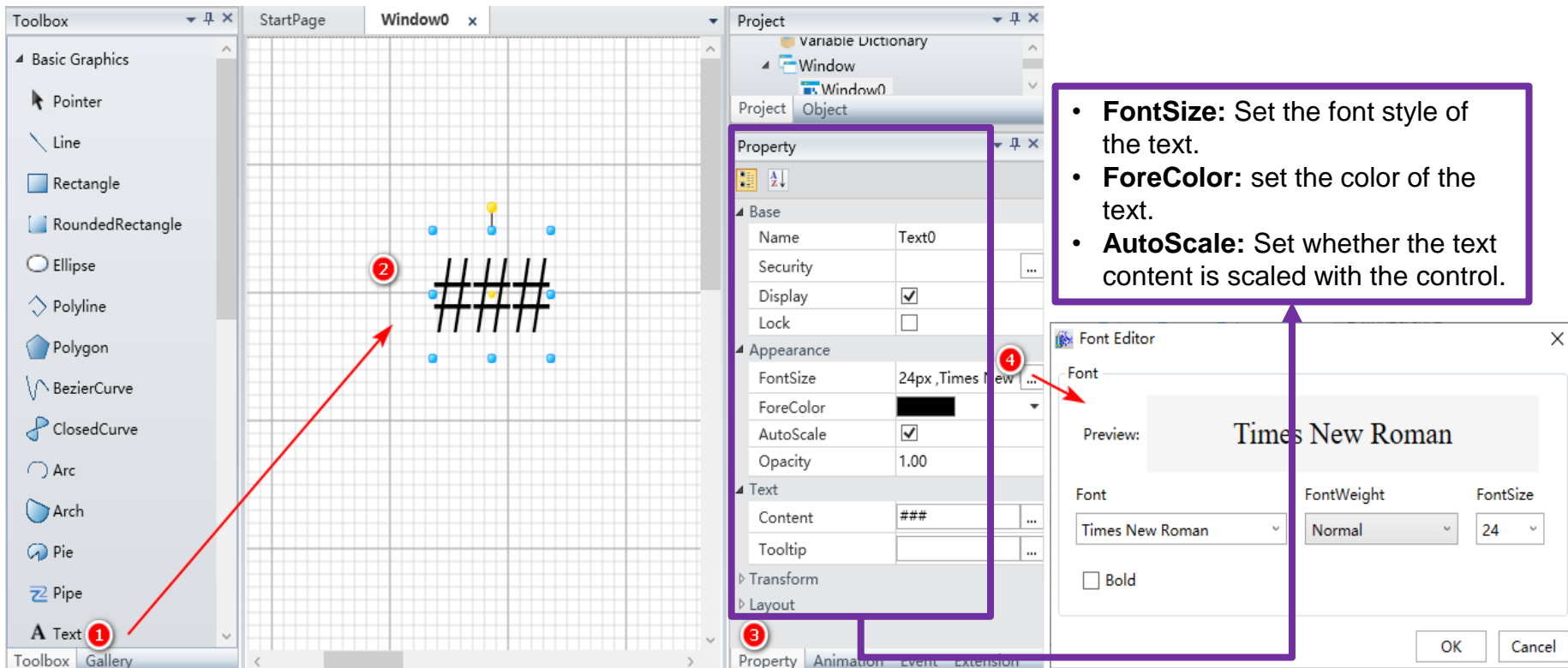




(3) Run the project.



➤ Draw a Text0 , open its properties window



The screenshot illustrates the steps to create and configure a text object in the Delta IDE:

- 1**: Select the **Text** tool from the **Basic Graphics** toolbox.
- 2**: Place the text object on the canvas.
- 3**: Open the **Property** window for the selected text object. The **Text** tab shows the **Content** as **###**.
- 4**: Open the **Font Editor** window to configure the font. The **Font** is set to **Times New Roman**, **FontWeight** is **Normal**, and **FontSize** is **24**.

Additional settings visible in the **Property** window:


- Base**: Name: Text0, Security: (empty), Display: ☒, Lock: ☐
- Appearance**: FontSize: 24px, Times New Roman, ForeColor: (black), AutoScale: ☒, Opacity: 1.00
- Text**: Content: ###, Tooltip: (empty)
- Transform**: (collapsed)
- Layout**: (collapsed)

Additional settings visible in the **Font Editor** window:

- Font**: Times New Roman
- FontWeight**: Normal
- FontSize**: 24
- Bold**: ☐

- **FontSize**: Set the font style of the text.
- **ForeColor**: set the color of the text.
- **AutoScale**: Set whether the text content is scaled with the control.


## ➤ Set properties of text

Appearance	
FontSize	36px, 华文琥珀 ...
ForeColor	
AutoScale	<input checked="" type="checkbox"/>
Opacity	1.00
Text	
Content	文本 ...
Tooltip	...



文本

Size 、 color and Scale

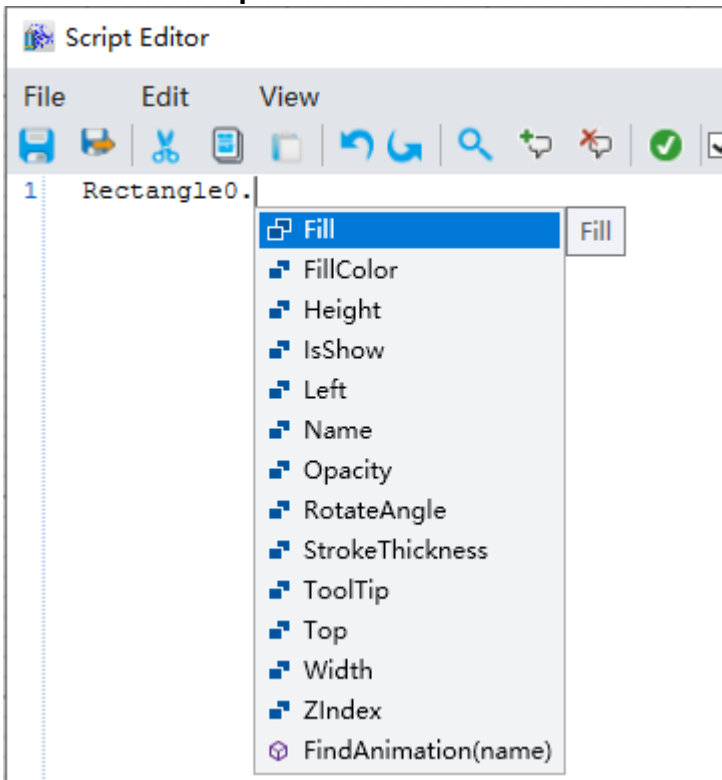
Appearance	
FontSize	36px, 微软雅黑 ...
ForeColor	
AutoScale	<input type="checkbox"/>
Opacity	1.00
Text	
Content	文本 ...
Tooltip	...



文本

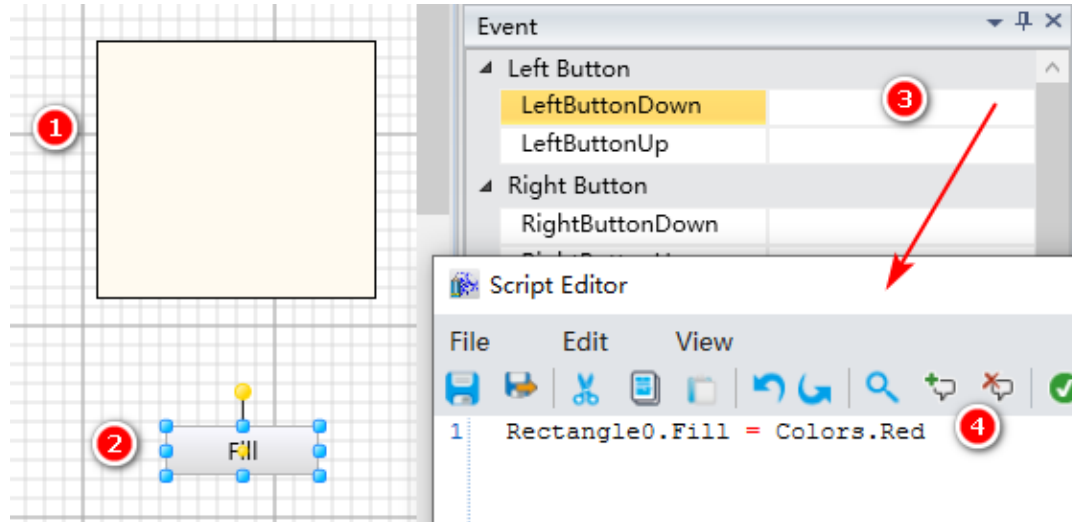
Size 、 color and Scale

- The script can be used to set the control properties



➤ Fill script example:

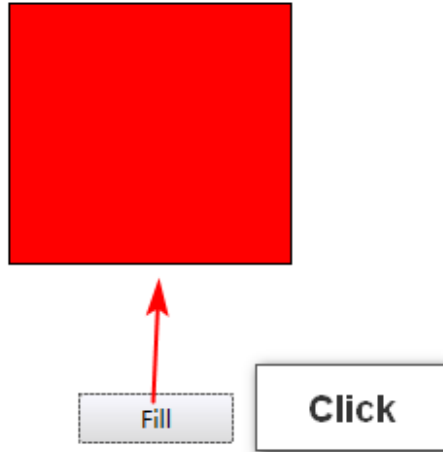
(1) Create Rectangle0 and Button0 in the Window0, configure the LeftButtonDown event of the button0





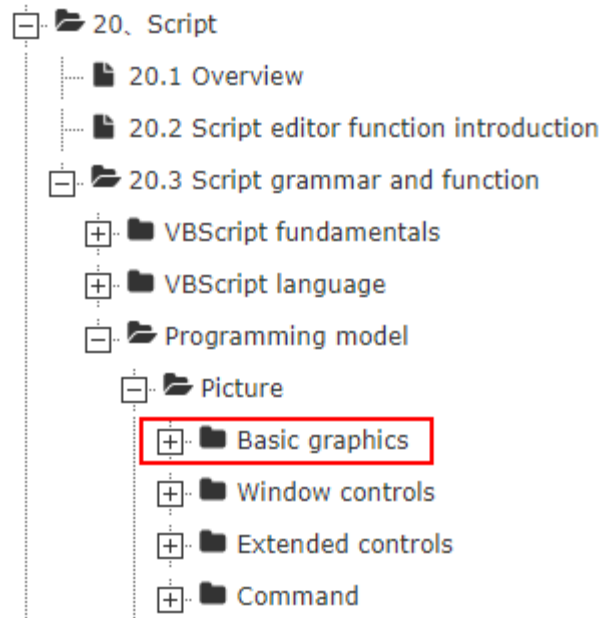
# The Script of Basic Graphics

(2)Run the project



# The Scripts of Basic Graphics

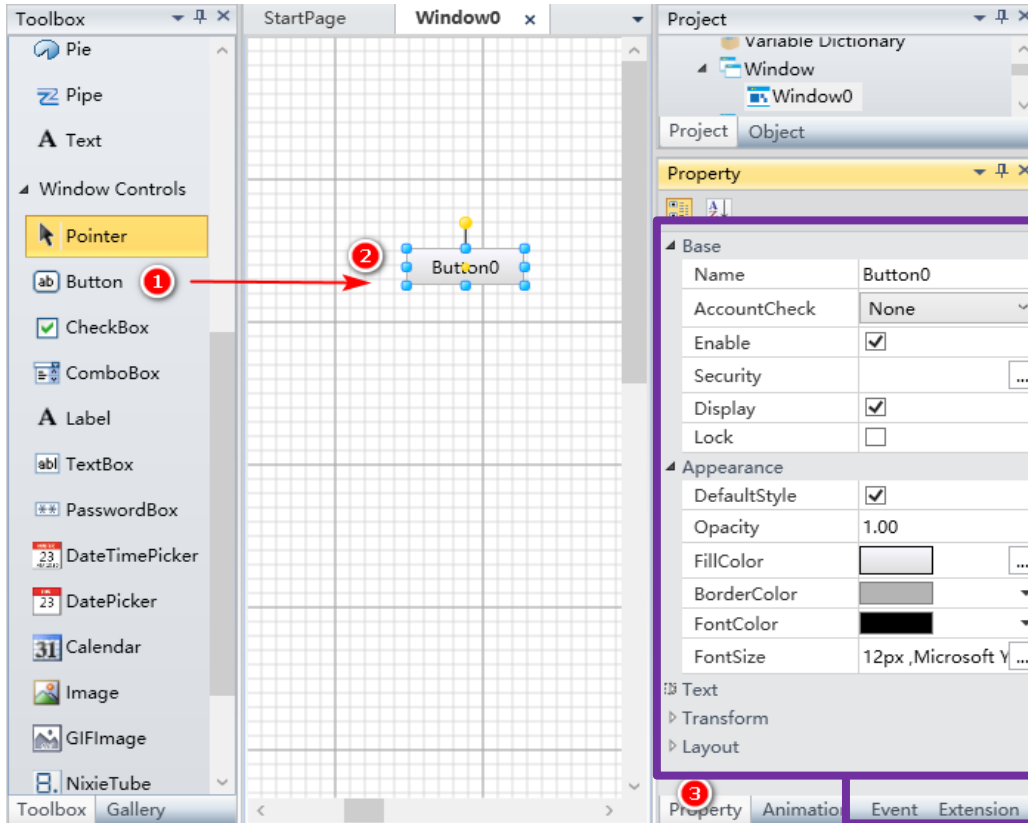
The scripts usage of basic graphics, please refer to the section “20.3 Script grammar and function” in the user manual. As shown in the figure below:



- The concepts of Toolbox
- How to draw Basic Graphics
- How to draw Window Controls
- How to draw Extend Controls



➤ Draw a Button0 , open its properties window

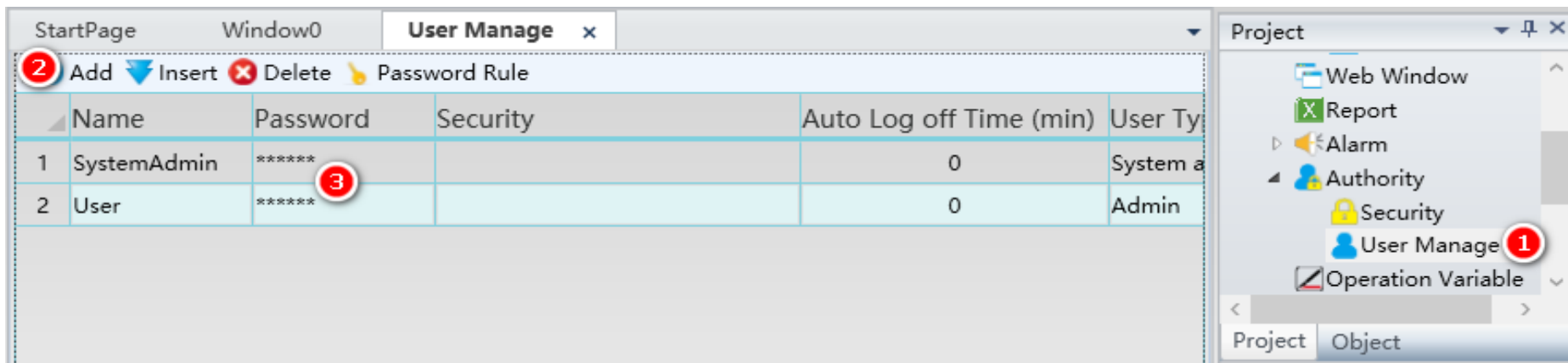


- **AccountCheck:** Sets the runtime account confirmation properties;  
 (1) None: User account password confirmation is not required;  
 (2) GeneralCheck: An account password confirmation is required to operate the control in the running environment  
 (3) DoubleCheck: In the running environment, each time you operate this control, you need to confirm the account and password
- **Enable:** Sets the availability of the button. Unchecked means the button operation is not available.
- **DefaultStyle:** Set whether to use the default style of button

# The AccountCheck Property of Button

➤ AccountCheck property example:

(1) Create user



The screenshot displays the 'User Manage' window. The table lists two users: 'SystemAdmin' and 'User'. The 'User' row is highlighted, and a red circle with the number '3' is placed over the password field. The 'Project' tree on the right shows the 'User Manage' node selected, with a red circle and the number '1' next to it. The 'Add' button in the toolbar is marked with a red circle and the number '2'.

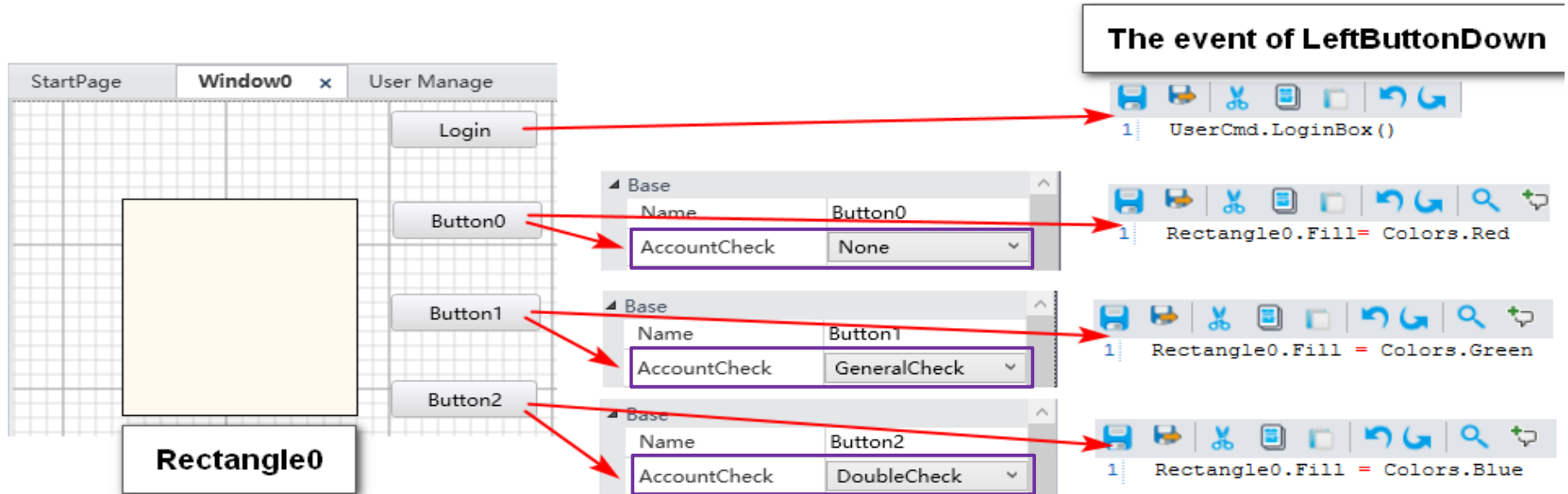
	Name	Password	Security	Auto Log off Time (min)	User Ty
1	SystemAdmin	*****		0	System a
2	User	*****		0	Admin

Project Tree:

- Web Window
- Report
- Alarm
- Authority
- Security
- User Manage (1)
- Operation Variable

# The AccountCheck Property of Button

(2) Create four buttons in the Window0 , configure the AccountCheck property and the LeftButtonDown event of the four buttons



The screenshot displays a software development environment with a window titled "Window0" containing a grid and four buttons: "Login", "Button0", "Button1", and "Button2". A yellow rectangle labeled "Rectangle0" is also present. Red arrows indicate the configuration of the "AccountCheck" property and the "LeftButtonDown" event for each button.

**The event of LeftButtonDown**

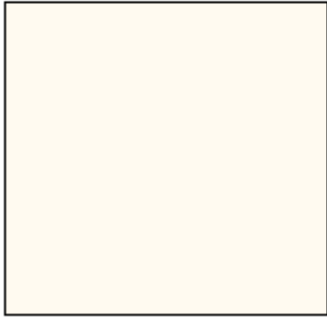
- Login:** UserCmd.LoginBox()
- Button0:** Rectangle0.Fill = Colors.Red
- Button1:** Rectangle0.Fill = Colors.Green
- Button2:** Rectangle0.Fill = Colors.Blue

Name	AccountCheck
Button0	None
Button1	GeneralCheck
Button2	DoubleCheck



(3)Run the project

# The AccountCheck Property of Button



Button0

Button1

Button2

Button3

# The AccountCheck Property of Button

(4) AccountCheck : None

① Click "Button0", the rectangle0 turn red

(5) AccountCheck : GeneralCheck

① Click "Button1", Pop-up Prompt: Please login first!

As shown in Figure 1,

② SystemAdmin or User login, then Click "Button1",  
the rectangle0 turn green



Figure 1

(6) AccountCheck : DoubleCheck

① Click "Button2", Pop-up Prompt: please login first!

As shown in Figure 1,

② SystemAdmin or User login, then Click "Button2",  
Pop-up Prompt: Please conform the password again.

As shown in Figure 2,

③ Enter the password again and click OK, then the rectangle0  
turn blue

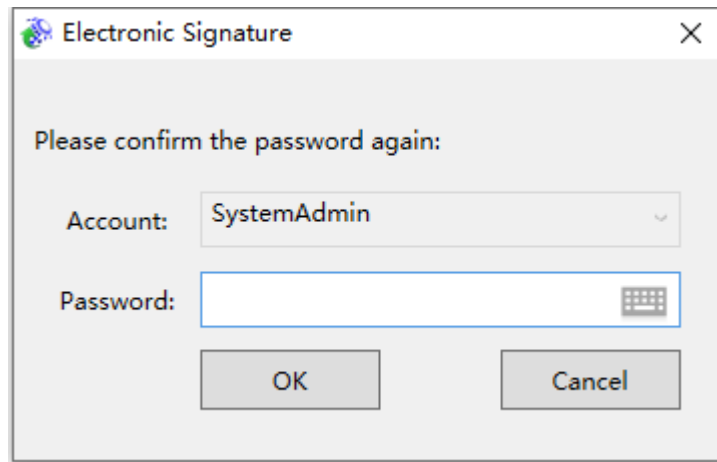
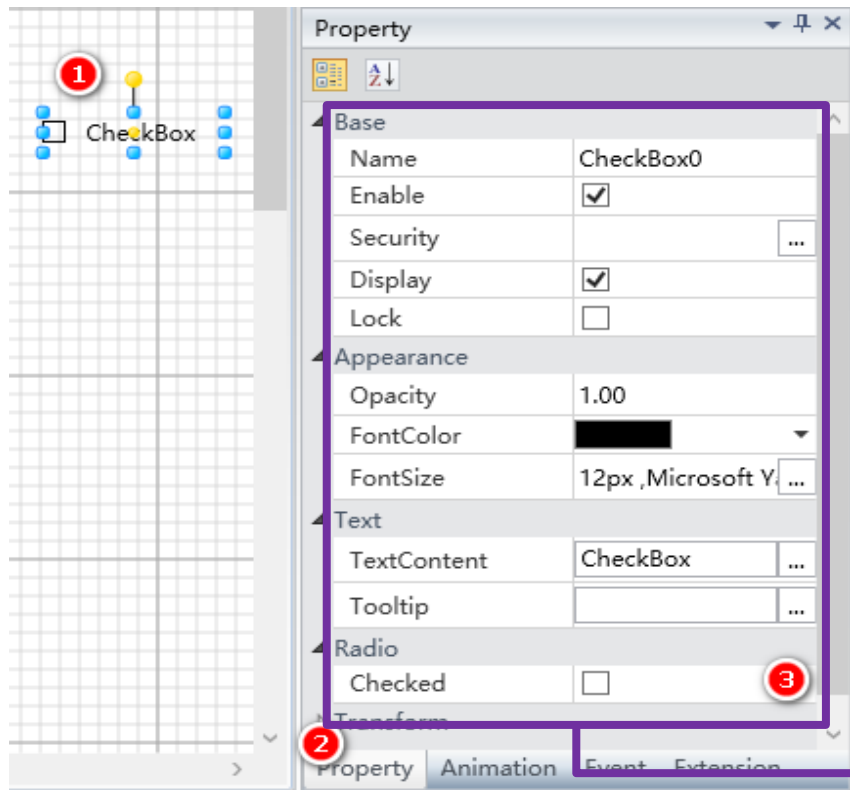


Figure 2

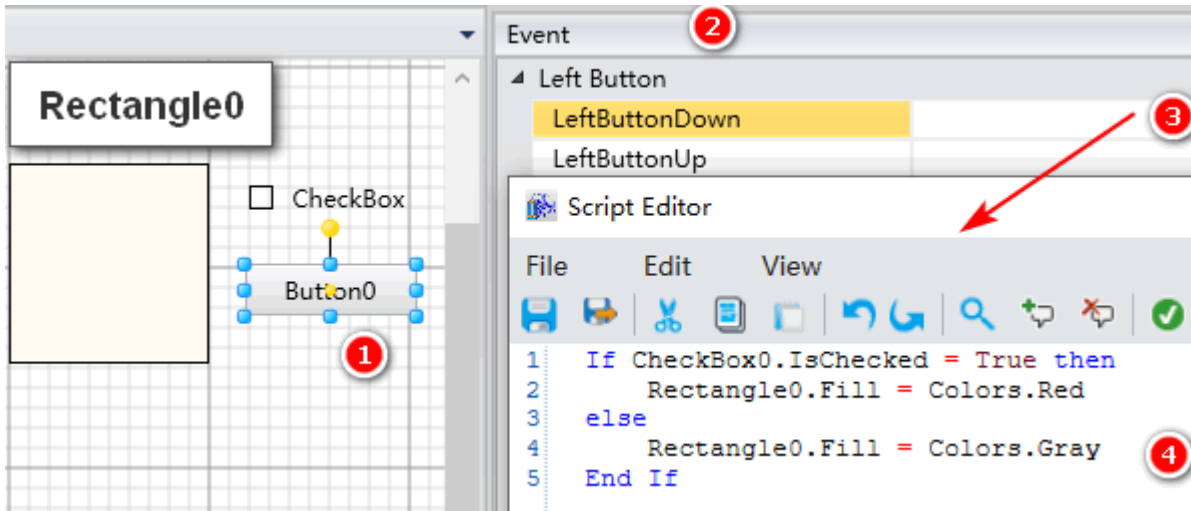
- Draw a CheckBox0, open its properties window



- **FontSize:** Set the font style of the text.
- **ForeColor:** set the color of the text.
- **Checked:** Set whether the check box is checked

➤ Checked property example

(1) Create a Checkbox0 , Rectangle0 , Button0 in the Window0 , configure the LeftButtonDown event of the Button0

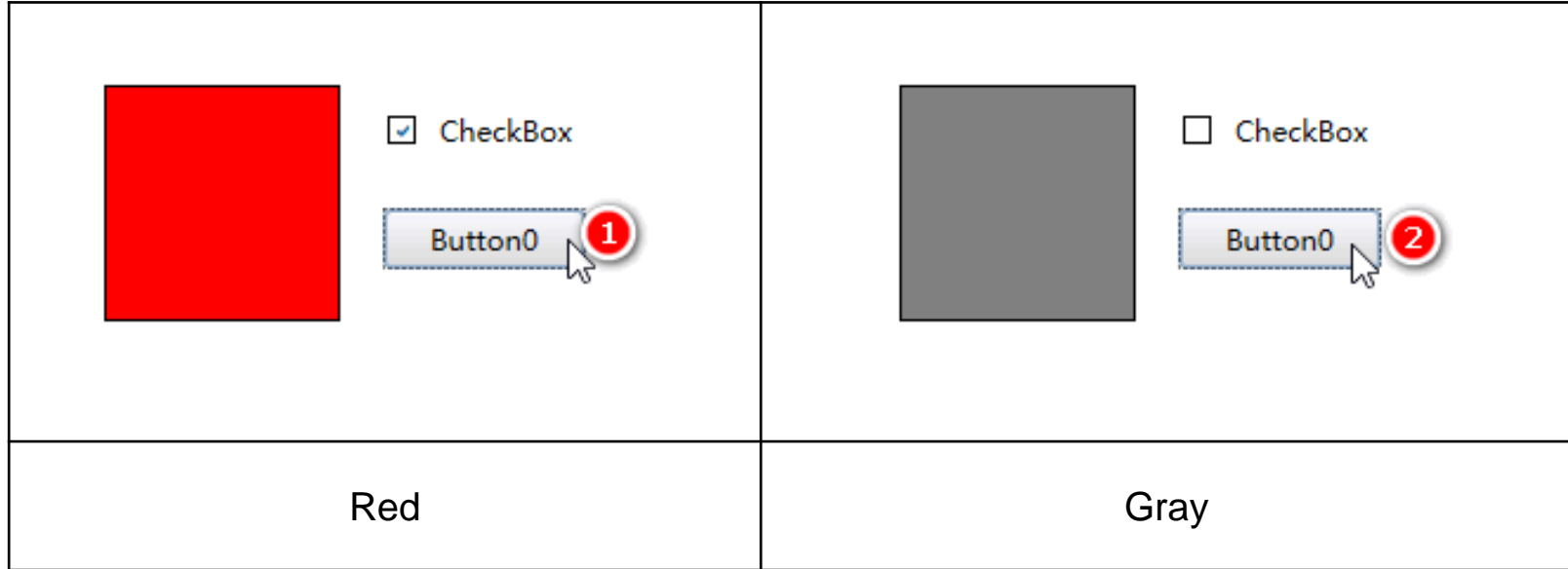


The screenshot displays a software development environment with a grid-based workspace. In the workspace, there is a yellow rectangle labeled "Rectangle0", a checkbox labeled "CheckBox", and a button labeled "Button0". A red circle with the number "1" is placed over the "Button0" label. To the right, the "Event" window shows a list of events for the "Left Button", with "LeftButtonDown" selected. A red circle with the number "2" is placed over the "Event" window title bar. A red arrow points from the "LeftButtonDown" event to the "Script Editor" window. The "Script Editor" window shows 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 circle with the number "3" is placed over the "LeftButtonDown" event in the "Event" window, and a red circle with the number "4" is placed over the "End If" statement in the "Script Editor" window.

## (2)Run the project

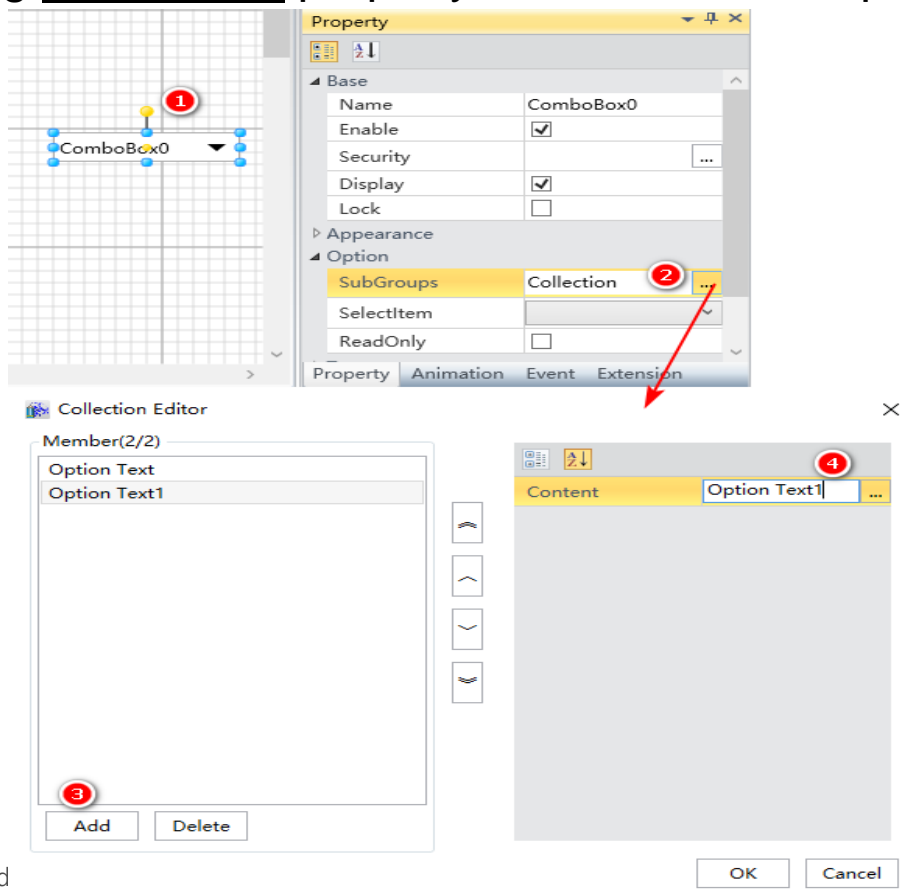


①When the Checkbox0 is checked, the Rectangle0 turns red

②When the Checkbox0 is unchecked, the rectangle0 turns gray



- Using SubGroups property to add or delete options of the ComboBox



The screenshot illustrates the process of managing a ComboBox's options in the Delta IDE. It is divided into two main parts: the main IDE window and a floating 'Collection Editor' dialog.

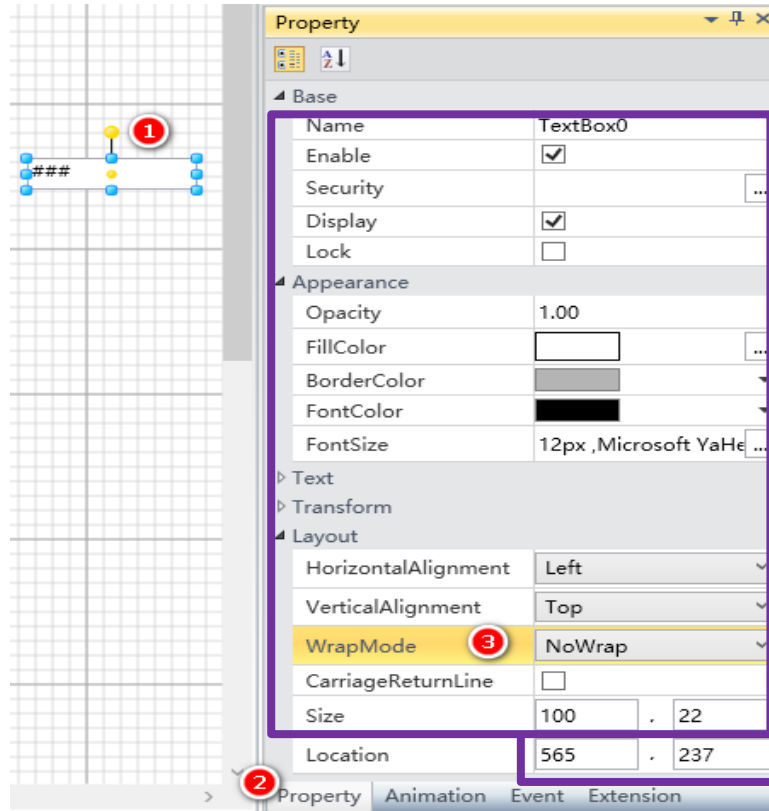
**Main IDE Window:**

- Property Window:** Shows the properties for 'ComboBox0'. The 'SubGroups' property is set to 'Collection' and is highlighted with a red circle labeled '2'. A red arrow points from this property to the 'Collection Editor' dialog.
- Collection Editor Dialog:** This dialog is used to manage the items in the 'SubGroups' collection.
  - Member(2/2):** The left pane shows the current collection of options, which includes 'Option Text1'.
  - Buttons:** At the bottom of the left pane are 'Add' and 'Delete' buttons. The 'Add' button is highlighted with a red circle labeled '3'.
  - Content Area:** The right pane shows the 'Content' of the selected option, 'Option Text1', which is highlighted with a red circle labeled '4'.
  - Navigation:** Between the two panes are four navigation buttons: a left arrow, an up arrow, a down arrow, and a right arrow.
  - Footer:** At the bottom of the dialog are 'OK' and 'Cancel' buttons.

- Using scripts to add or delete options of the ComboBox

```
ComboBox0.AddItem("AA")  
//Add option AA  
  
ComboBox0.AddItems("BB,CC,DD")  
//Add 3 options: BB, CC, DD  
  
ComboBox0.ClearItems()  
//Clears all options  
  
ComboBox0.RemoveAtItem(1)  
//Remove option with index 1  
  
ComboBox0.RemoveItem("AA")  
//Remove option AA
```

- Draw a TextBox0, open its properties window

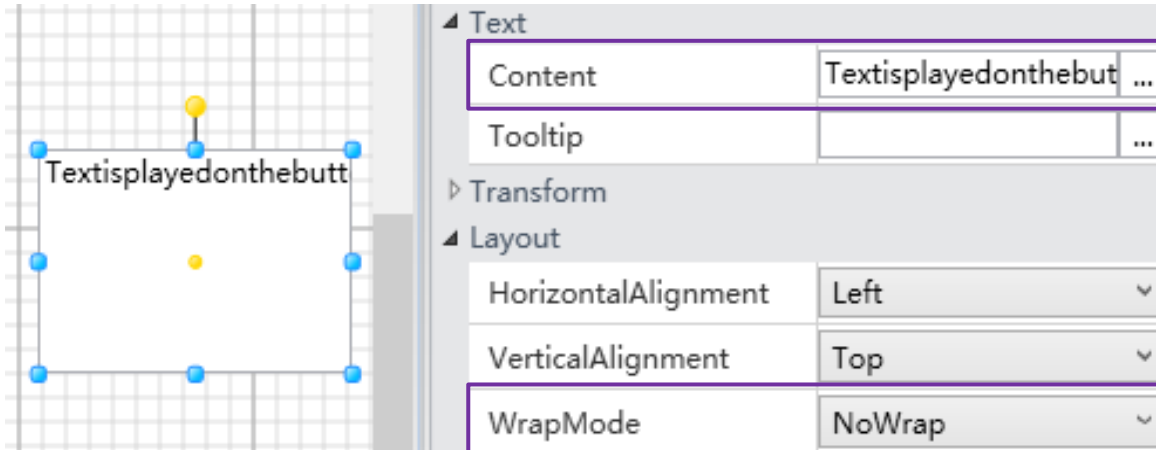


- **CarriageReturnLine:** Press enter key to a new line
- **HorizontalAlignment:** Left, Center, Right
- **VerticalAlignment:** Top, Center, Bottom
- **WrapMode:** NoWrap, WrapWithOverflow, Wrap

## ➤ WrapMode example1 :

Input letters in the Textbox0 : Textisplayedonthebutton support multilanguage

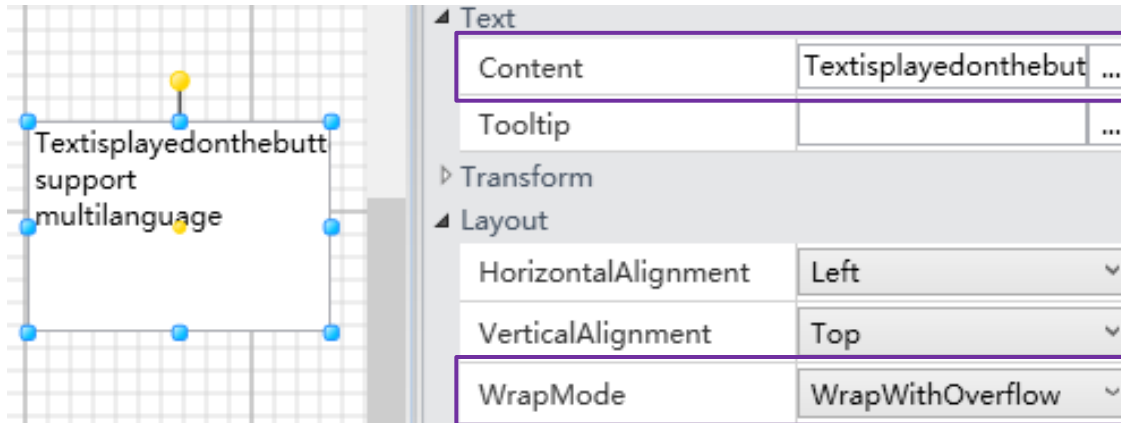
① The WrapMode property : NoWrap



➤ WrapMode example2 :

Input letters in the Textbox0 : Textisplayedonthebutton support multilanguage

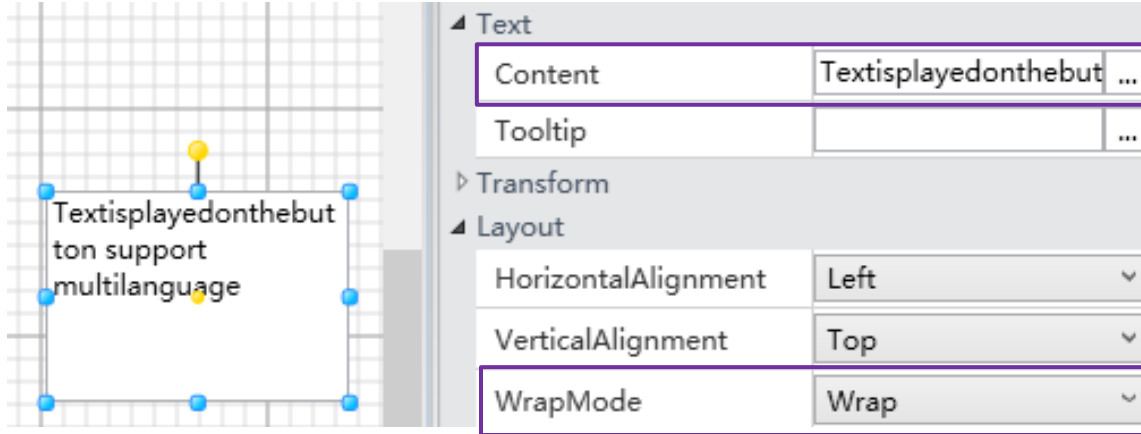
②The WrapMode property : WrapWithOverflow



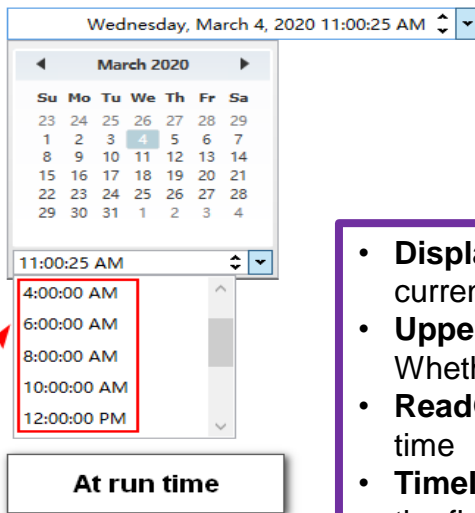
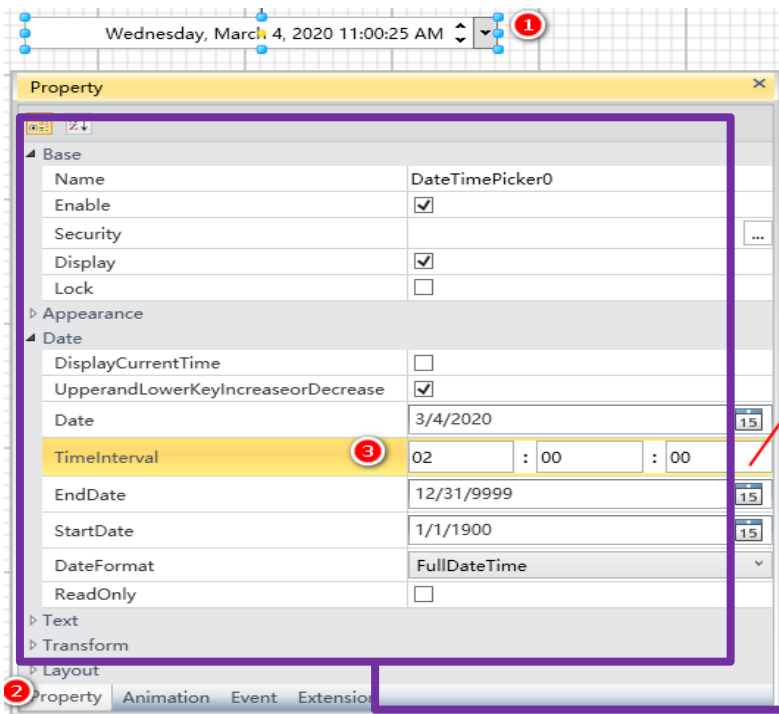
➤ WrapMode example3 :

Input letters in the Textbox0 : Textisplayedonthebutton support multilanguage

③The WrapMode property : Wrap



- Draw a DateTimePicker0, open its properties window

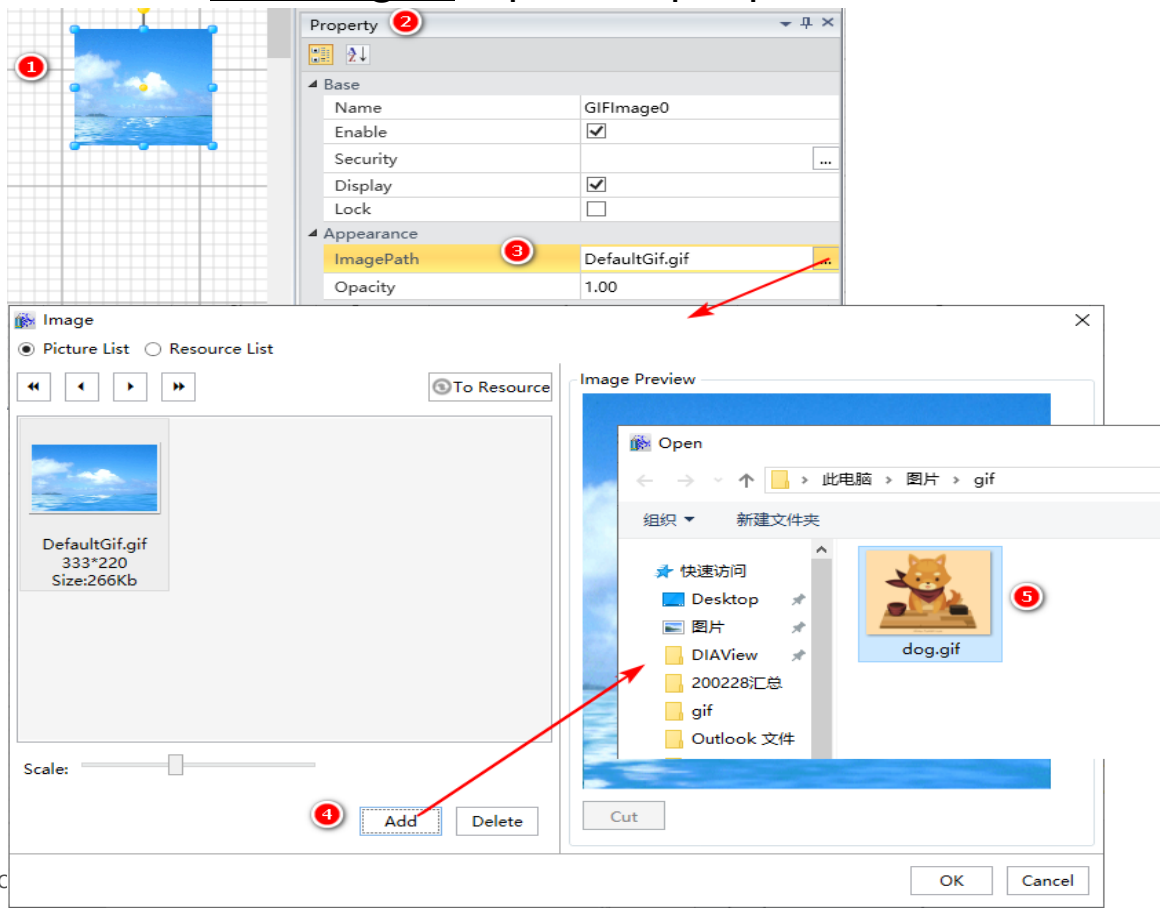


- **DisplayCurrentTime:** Whether to display current time
- **UpperandLowerKeyIncreaseorDecrease:** Whether the up and low key are available
- **ReadOnly:** The value can only be read at run time
- **TimeInterval:** Set time interval ,as shown on the figure.



# Window Controls-GIFImage

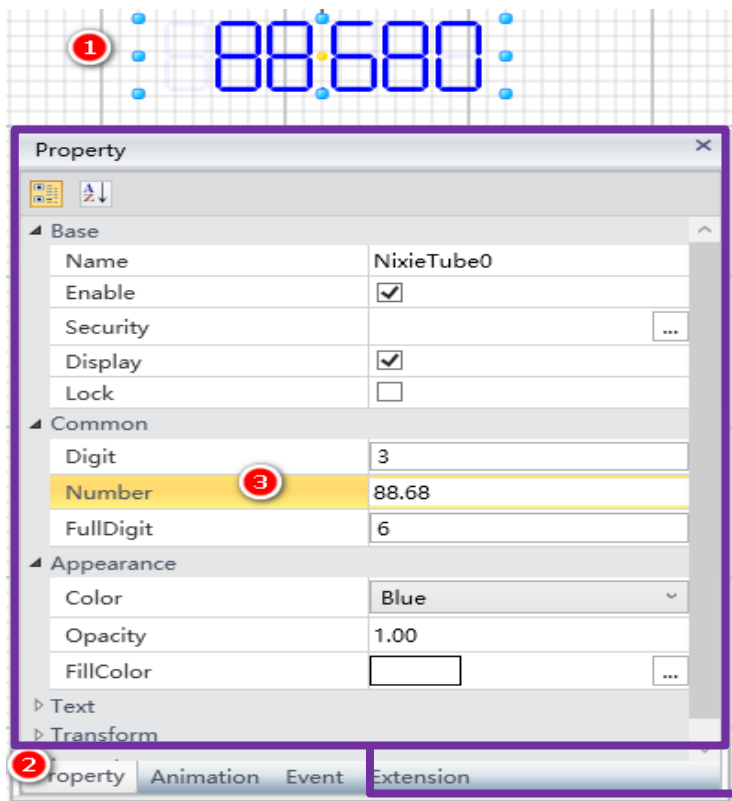
➤ Draw a GIFImage0, open its properties window



※ For more details, please refer to the section “7.6.11 GIFImage” in user manual.



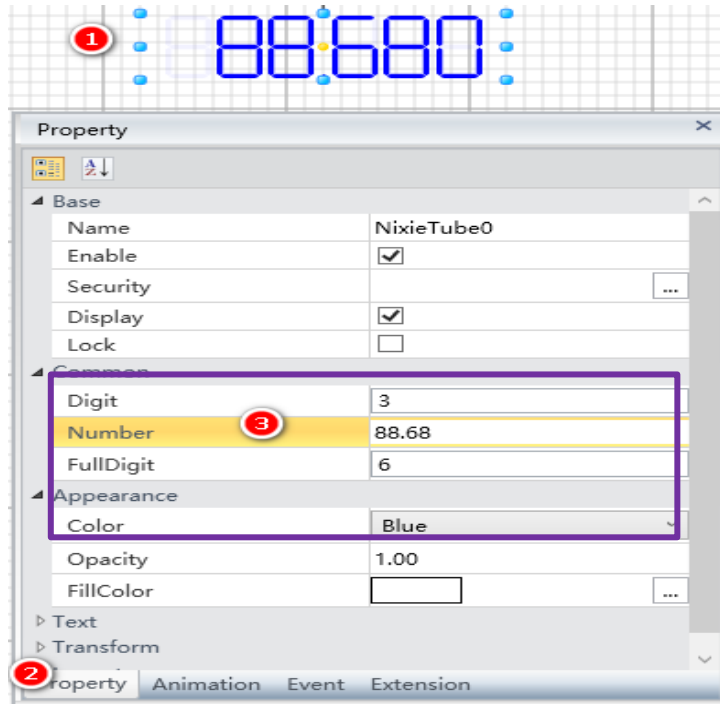
- Draw a NixieTube0, open its properties window



- **Digit:** Number of digits in the integer part
- **FullDigit:** Total number of digit in the NixieTube
- **Number:** The number to display
- **Color:** Set the font color of NixieTube
- **FillColor:** Set the background color of NixieTube

➤ NixieTube display value

(1) Create a NixieTube0 in the Window0, set its properties





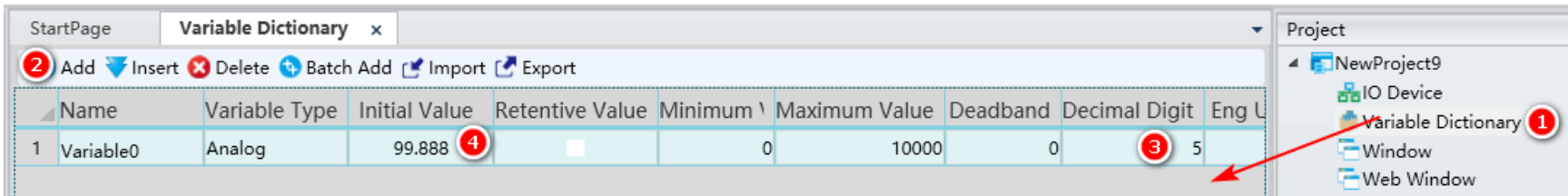
# Window Controls-NixieTube

(2)Run the project

88.680

➤ NixieTube display variable

(1) Create a Variable0, set its initial value to 99.888



StartPage Variable Dictionary x

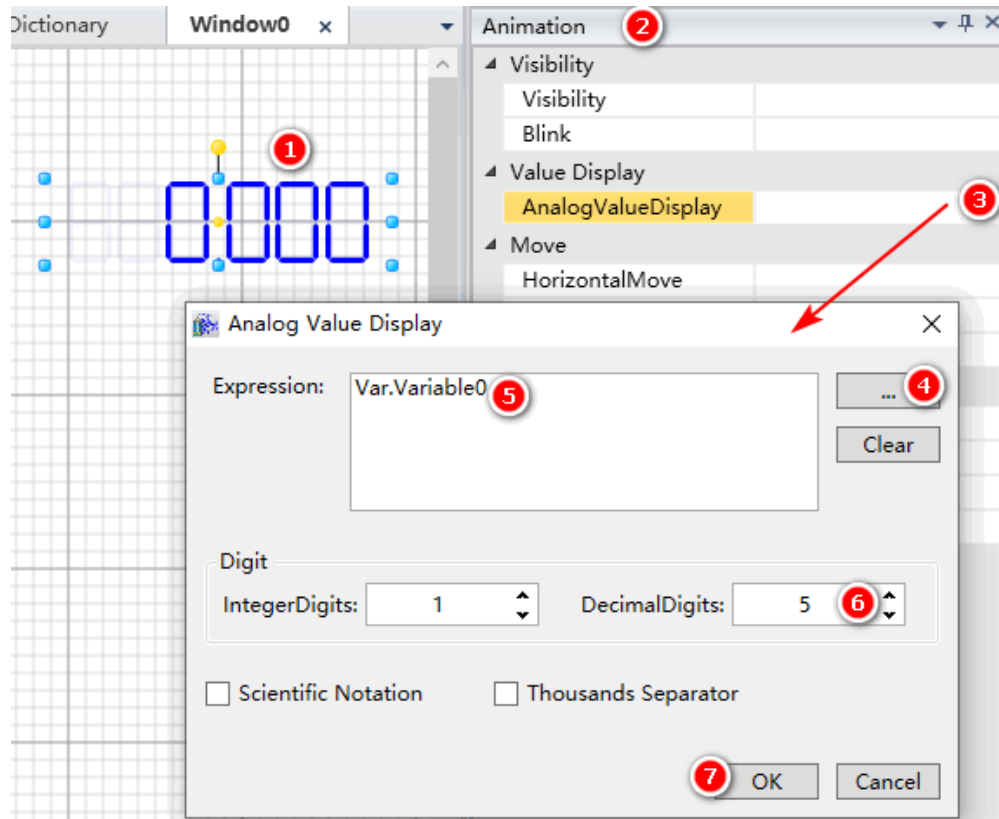
2 Add Insert Delete Batch Add Import Export

	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

(2) Create a NixieTube0 in the Window0, the analog value display animation associated Variable0



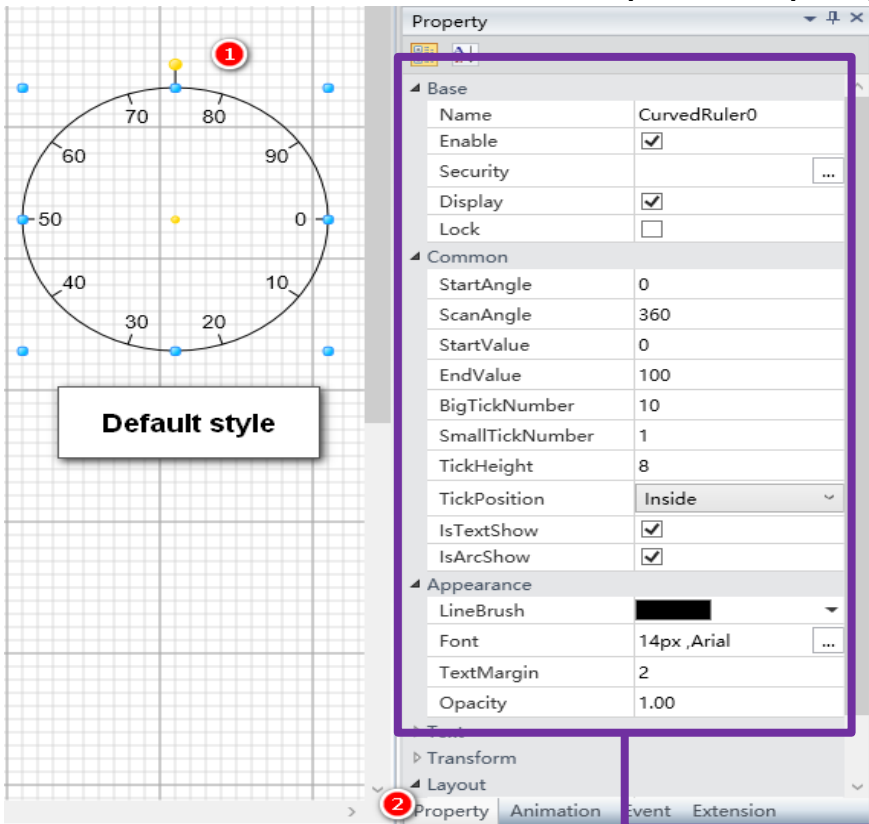


# Window Controls-NixieTube

(3)Run the project

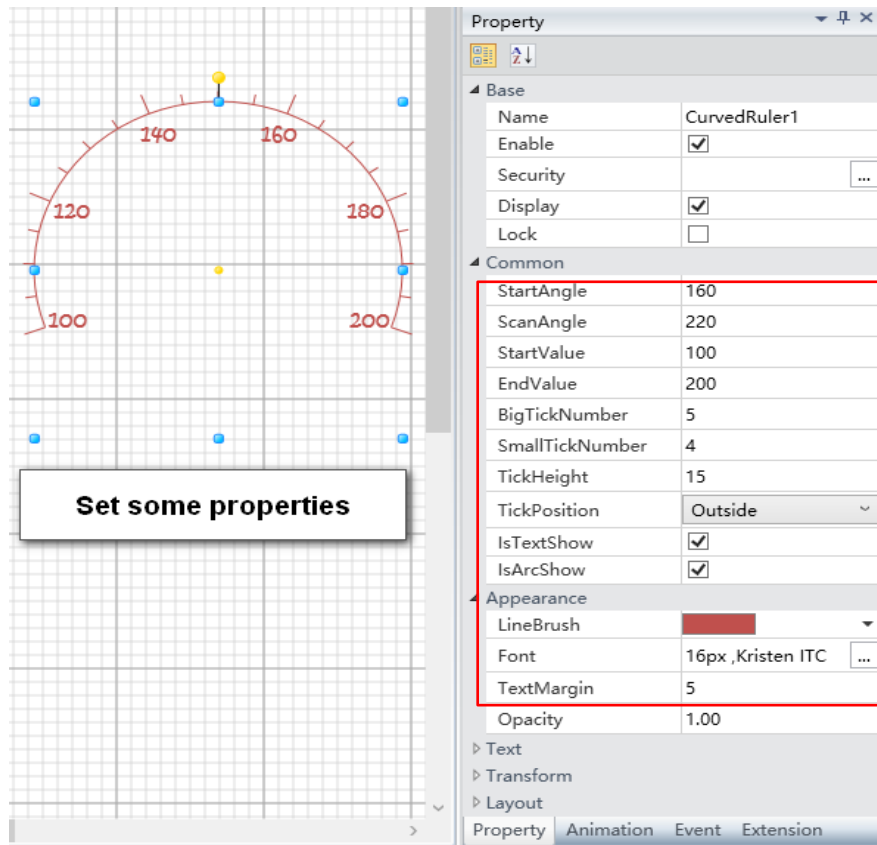
899.888

➤ Draw a CurvedRuler0, open its properties window



- **StartAngle:** Set the starting angle of the CurvedRuler (unit: degrees).
- **ScanAngle:** Set the scanning angle of the CurvedRuler (unit: degrees).
- **StartValue:** Set the starting value of the CurvedRuler.
- **EndValue:** Set the ending value of the CurvedRuler.
- **BigTickNumber:** Set the number of big tick of the CurvedRuler.
- **SmallTickNumber:** Set the number of small tick of the CurvedRuler.
- **TickHeight:** Set the tick height of the CurvedRuler.
- **TickPosition:** Set the tick position of the CurvedRuler.
- **IsTextShow:** Set whether the text of CurvedRuler is displayed
- **IsArcShow:** Set whether the border line of CurvedRuler is displayed
- **LineBrush:** Set the current color of CurvedRuler.
- **Font:** Set the font format of the text displayed on the CurvedRuler.
- **TextMargin:** Set the distance between text and border lines

- Draw a CurvedRuler0, set its properties



The screenshot shows a curved ruler drawn on a grid. The ruler is red and has major ticks labeled 100, 120, 140, 160, 180, and 200. A yellow dot is placed on the ruler at the 160 mark. A white box with the text "Set some properties" is overlaid on the grid.

The Property window on the right shows the properties for 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 properties:

Property	Value
StartAngle	160
ScanAngle	220
StartValue	100
EndValue	200
BigTickNumber	5
SmallTickNumber	4
TickHeight	15
TickPosition	Outside
IsTextShow	<input checked="" type="checkbox"/>
IsArcShow	<input checked="" type="checkbox"/>

The Appearance section shows the LineBrush set to a red color, the Font set to 16px, Kristen ITC, and the TextMargin set to 5. The Opacity is set to 1.00.





# Window Controls-Table

➤ Draw a Table0, open its properties window

Property

Base

Name	Table1
Security	...
Display	<input checked="" type="checkbox"/>
Lock	<input type="checkbox"/>

Appearance

Opacity	1.00
FirstRowBackground	
FirstColumnBackground	
DataBackground	
BorderStyle	...
GridStyle	...
TableDesign	Collection
ObliqueLine	<input type="checkbox"/>
RowCrossDisplay	<input checked="" type="checkbox"/>

Text

Transform

Property Animation Event Extension

Grid Editor

Grid Style

LineStyle	01
LineThickness	02 2pt
LineDashCapStyle	01
Color	

View

OK Cancel

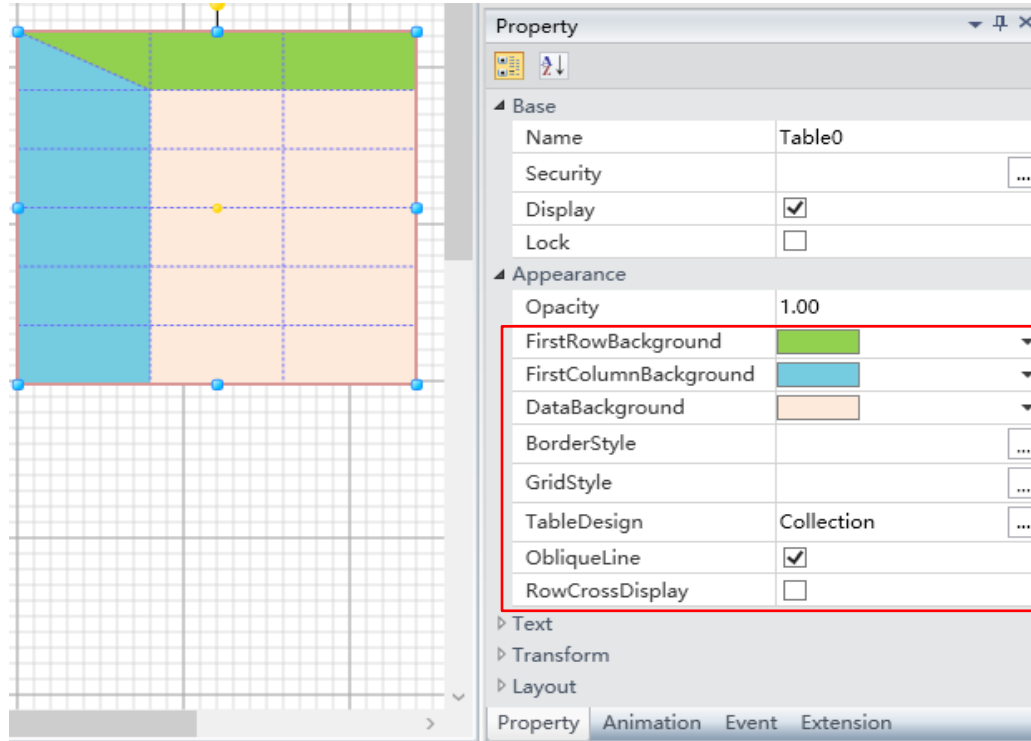
Table Design

Column	Row
Index	Width
1	80.00
2	80.00
3	80.00
4	80.00

Add Delete OK Cancel

- **FirstRowBackground:** Set background of the first row.
- **FirstColumnBackground:** Set background of the first column, It only works if the ObliqueLine property is checked.
- **DataBackground:** Set background of data area, It only works if the RowCrossDisplay property is unchecked.
- **TableDesign:** Set the rows and columns of the Table.
- **ObliqueLine:** Set whether to display the oblique line in the upper left corner.
- **RowCrossDisplay:** Set whether to enable row cross display.

- Create a table with 6 rows and 3 columns
  - (1) Create a Table0 in the Window0, and set its some properties



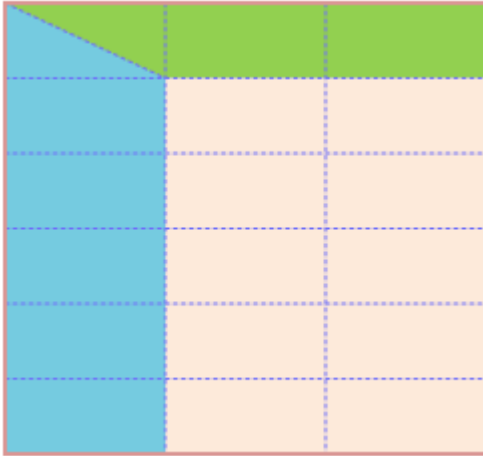
The screenshot displays a software interface with a grid-based workspace on the left and a 'Property' window on the right. The workspace contains a table with 6 rows and 3 columns. The first row is green, the first column is blue, and the remaining cells are orange. The 'Property' window is titled 'Table0' and shows various settings. A red box highlights the 'Appearance' section, which includes the following properties:

Property	Value
FirstRowBackground	Green
FirstColumnBackground	Blue
DataBackground	Orange
BorderStyle	...
GridStyle	...
TableDesign	Collection
ObliqueLine	<input checked="" type="checkbox"/>
RowCrossDisplay	<input type="checkbox"/>



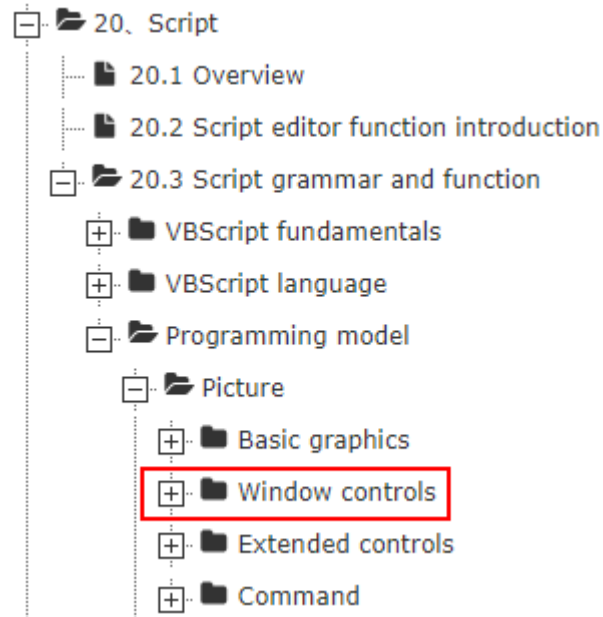
# Window Controls-Table

(2)Run the project



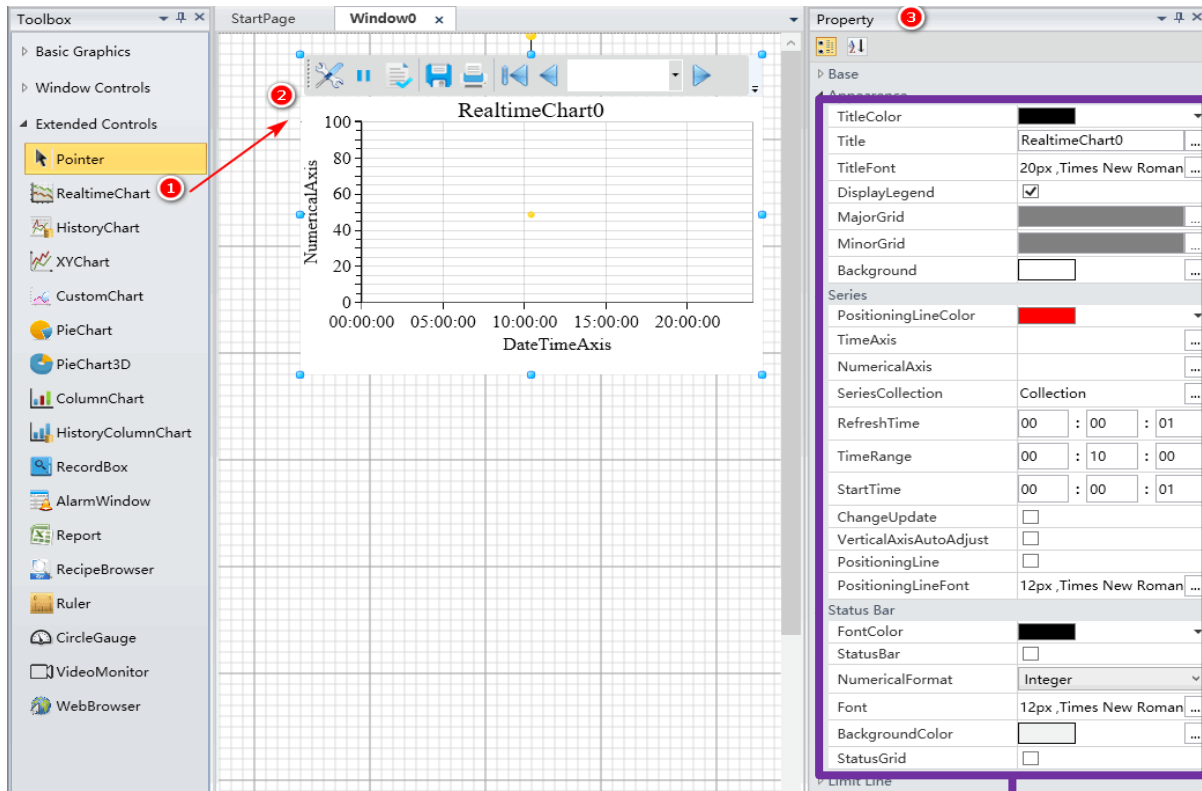
# The Scripts of Window Controls

The scripts usage of window controls, please refer to the section “20.3 Script grammar and function” in the user manual. As shown in the figure below:



- The concepts of Toolbox
- How to draw Basic Graphics
- How to draw Window Controls
- How to draw Extend Controls

➤ Create a RealTimeChart0, open its properties window



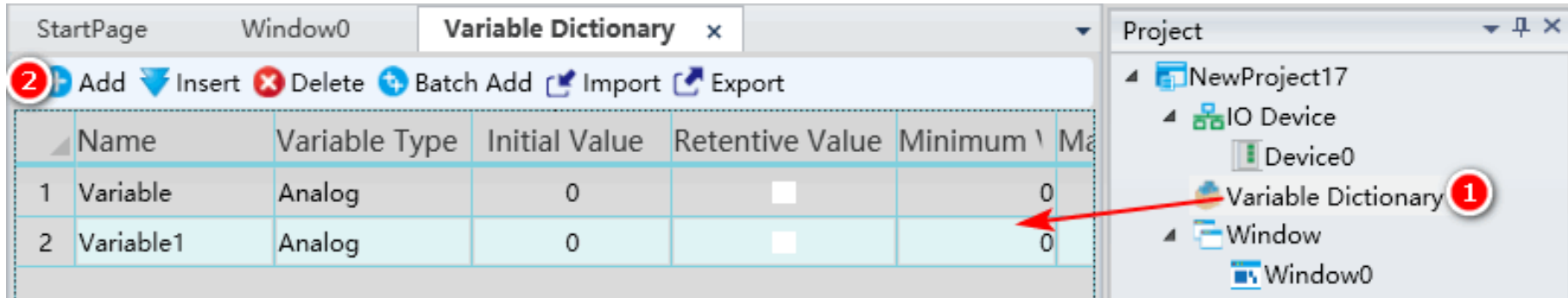
- **TimesAxis:** Set properties of TimeAxis
- **NumericalAxis:** Set properties of NumericalAxis
- **SeriesCollection:** Add or delete series, set properties of series
- **ChangeUpdate:** Set whether to update data only when the value changes
- **VerticalAxisAutoChange:** Set whether to automatically adjust the maximum value of the vertical axis.
- **PositionLine:** Whether to display position line
- **PositionLineColor:** Set the color of position line
- **PositionLineFont:** Set the font format of the data on the position line
- **DisplayLegend:** Whether display legend of a chart.
- **MajorGrid:** Set major grid style
- **MinorGrid:** Set minor grid style

※ For more details, please refer to the section "7.7.1 RealTimeChart" in the user manual

The properties of HistoryChart · XYChart · CustomChart · ColumnChart · HistoryColumnChart are similar to RealTimeChart, so no explanation is given

- Create a RealTimeChart0 that refreshes the real-time data:

(1) Create 2 Variables: Variable , Variable1



StartPage Window0 Variable Dictionary x

2 Add Insert Delete Batch Add Import Export

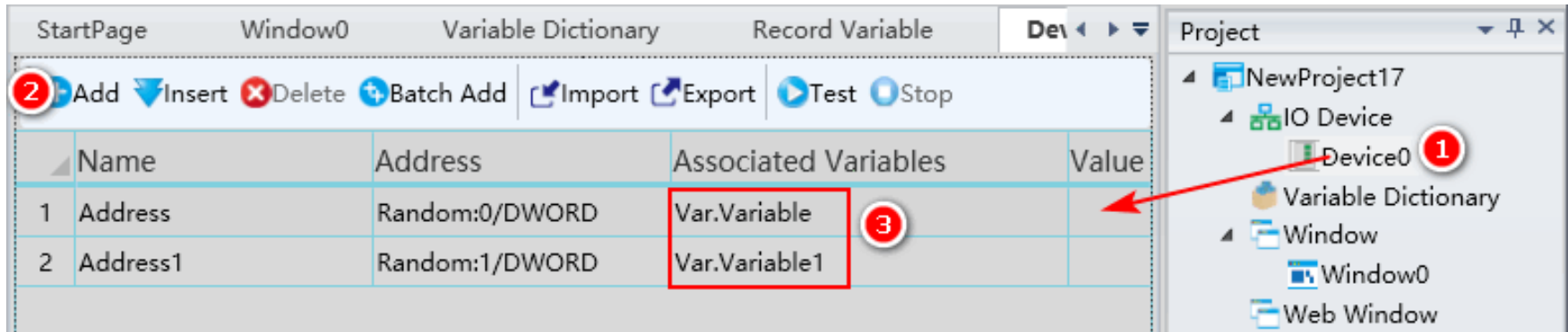
	Name	Variable Type	Initial Value	Retentive Value	Minimum \	Ma
1	Variable	Analog	0	<input type="checkbox"/>	0	
2	Variable1	Analog	0	<input type="checkbox"/>	0	

Project

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

※Refer to the section "6.3 Variables" in user manual.

(2) Create a simulation device-Device0, add two address in Device0 that associated Variable, Variable1 respectively



The screenshot displays the Delta RealTimeChart software interface. The main window is titled "Variable Dictionary" and contains a table with the following columns: Name, Address, Associated Variables, and Value. The table lists two entries:

	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 entries is highlighted with a red box, and a red circle with the number 3 is placed next to it. The "Project" panel on the right shows a tree view with the following structure:

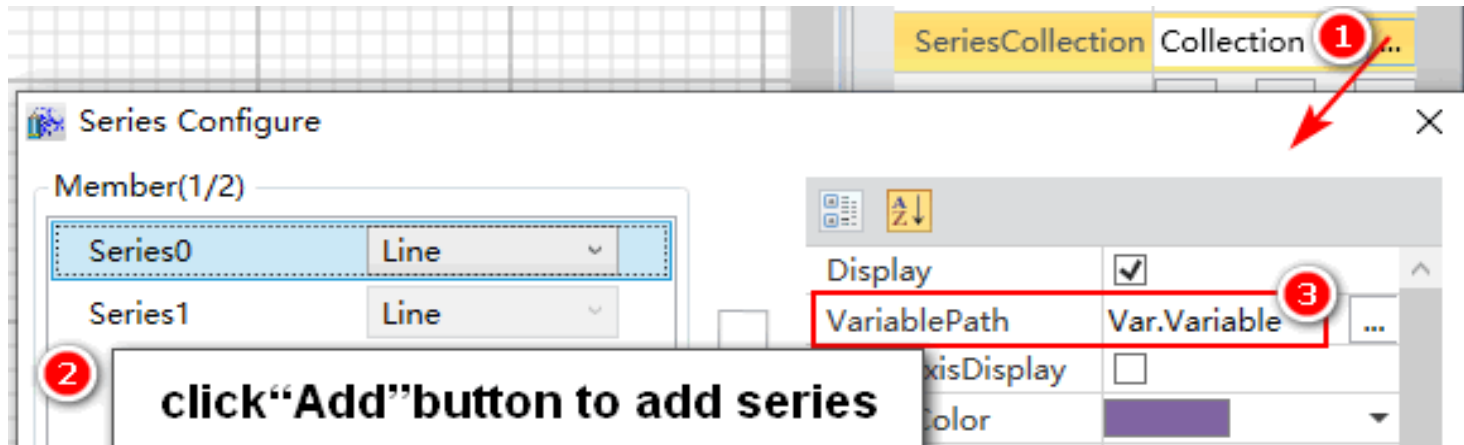
- NewProject17
  - IO Device
    - Device0
  - Variable Dictionary
  - Window
    - Window0
  - Web Window

A red arrow points from the "Device0" entry in the Project panel to the "Associated Variables" column of the table. A red circle with the number 1 is placed next to "Device0". A red circle with the number 2 is placed next to the "Add" button in the top toolbar.

※ The method of creating a simulation device, refer to section “5.10.1 Simulator” in the user manual



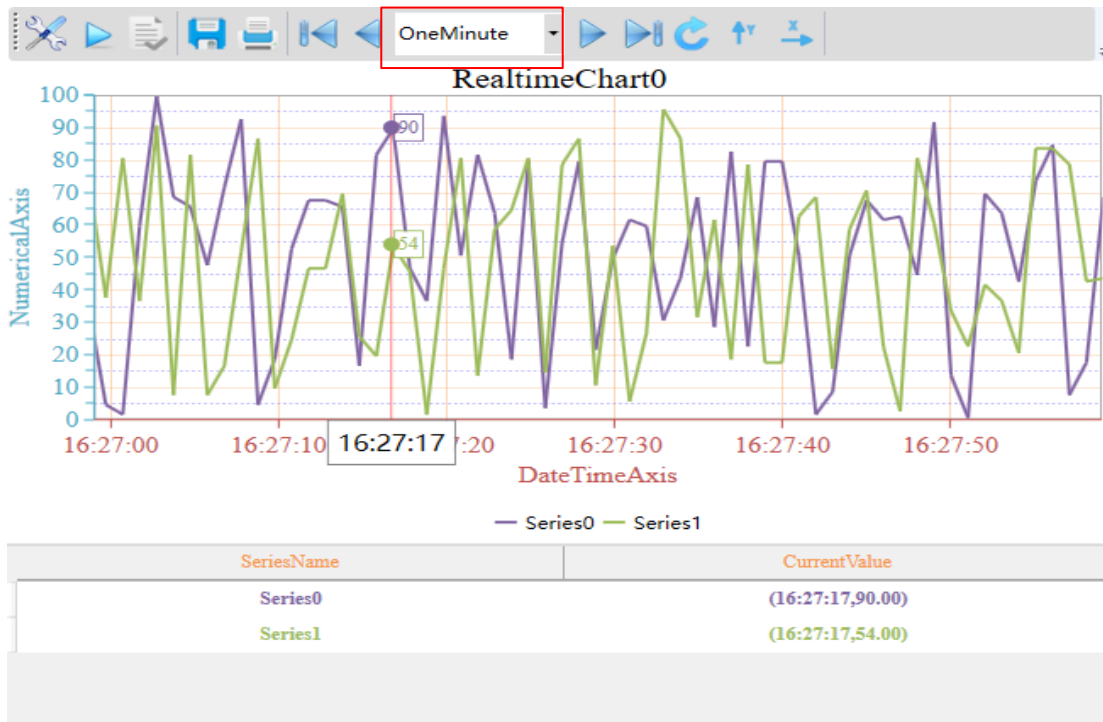
(3) Create a RealtimeChart0 in the Window0, add two curves(Series0,Series1) in RealtimeChart0 that associated Variable and Variable1 respectively



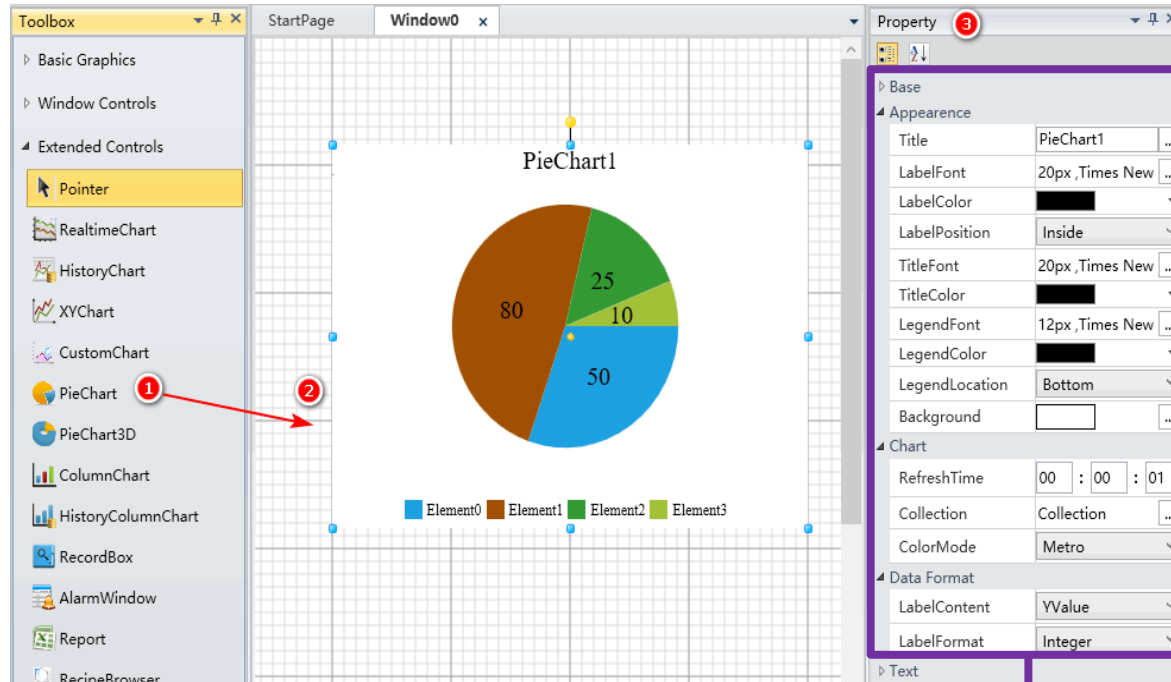
## (4) Set some properties of the RealtimeChart0

Appearance	
TitleColor	<div></div>
Title	RealtimeChart0
TitleFont	20px , Times New Roman
DisplayLegend	<input checked="" type="checkbox"/>
MajorGrid	<div></div>
MinorGrid	<div></div>
Background	<div></div>
Series	
PositioningLineColor	<div></div>
TimeAxis	
NumericalAxis	
SeriesCollection	Collection
RefreshTime	00 : 00 : 01
TimeRange	00 : 10 : 00
StartTime	00 : 00 : 01
ChangeUpdate	<input type="checkbox"/>
VerticalAxisAutoAdjust	<input type="checkbox"/>
PositioningLine	<input checked="" type="checkbox"/>
PositioningLineFont	12px , Times New Roman
Status Bar	
FontColor	<div></div>
StatusBar	<input checked="" type="checkbox"/>
NumericalFormat	Double2
Font	12px , Times New Roman
BackgroundColor	<div></div>
StatusGrid	<input type="checkbox"/>

## (5)Run the project



➤ Create a PieChart0, open its properties window

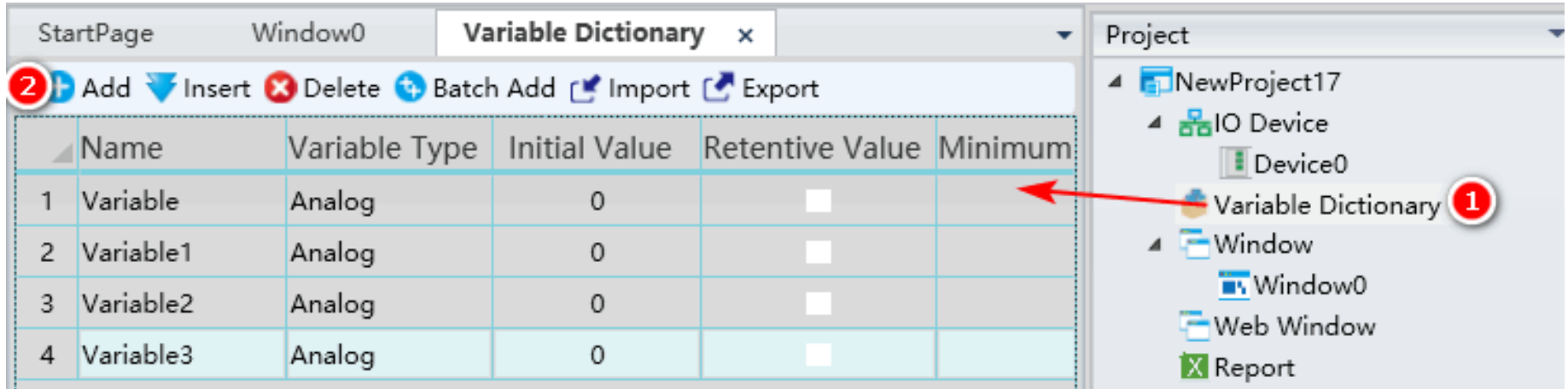


- **Title:** Set the title
- **TitleFont:** Set the title font style
- **TitleColor:** Set the title color
- **LabelFont:** Set label font style
- **LabelColor:** Set label color
- **LabelPosition:** Set label position (inside, Outside, OutsideExtended)
- **LegendFont:** Set legend font
- **LegendColor:** Set legend color
- **LegendLocation:** Set legend position
- **RefreshTime:** Set the refresh time
- **ColorMode:** Set the color mode (Metro, Custom)
- **LabelContent:** Set the label content (YValue, XValue, Percentage)
- **LabelFormat:** Set the label format (Int, Double1, Double2, Scientific, Percent)
- **Collection:** Configures the elements and associated with variables of the pie chart

※ For more details, please refer to the section "7.7.5 PieChart" in the user manual  
The properties of Pie3DChart are similar to PieChart, so no explanation is given

➤ Create a Pie that refreshes the real-time data:

(1) Create four variables: Variable, Variable1, Variable2, Variable3



StartPage Window0 Variable Dictionary x

2 Add Insert Delete Batch Add Import Export

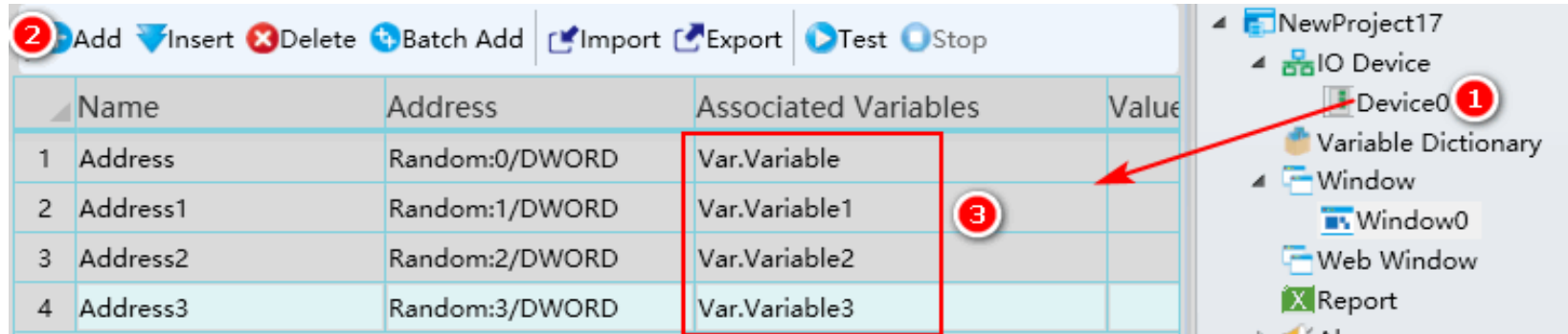
	Name	Variable Type	Initial Value	Retentive Value	Minimum
1	Variable	Analog	0	<input type="checkbox"/>	
2	Variable1	Analog	0	<input type="checkbox"/>	
3	Variable2	Analog	0	<input type="checkbox"/>	
4	Variable3	Analog	0	<input type="checkbox"/>	

Project

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

※Refer to the section "6.3 Variables" in user manual.

(2) Create a simulation device-Device0, add four address in Device0 that associated Variable, Variable1, Variable2, Variable3 respectively



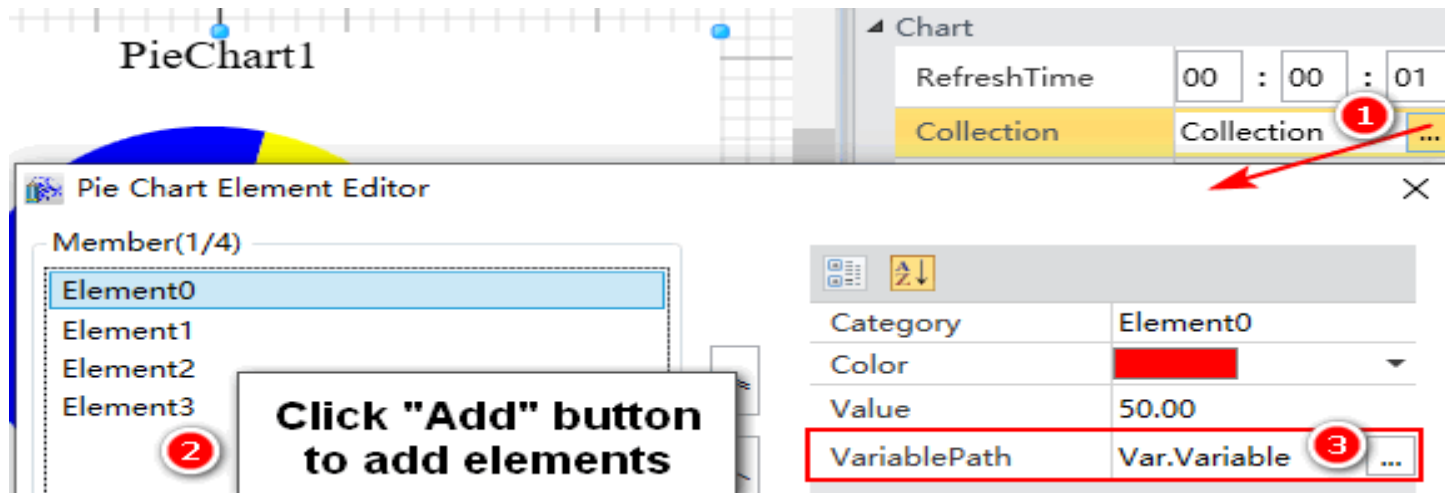
The screenshot shows the software interface for creating a simulation device. On the left, a table lists four addresses for 'Device0':

	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	

The 'Associated Variables' column is highlighted with a red box. A red arrow points from the 'Device0' entry in the project tree on the right to the table. The project tree on the right shows 'NewProject17' with sub-items: 'IO Device' (containing 'Device0'), 'Variable Dictionary', 'Window' (containing 'Window0'), 'Web Window', and 'Report'.

✘ The method of creating a simulation device, refer to section “5.10.1 Simulator” in the user manual

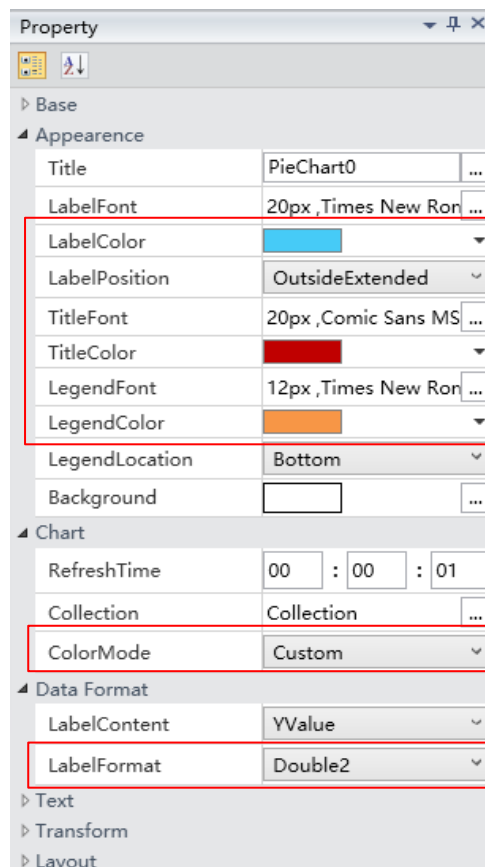
(3) Create a PieChart0 in the Window0, add four elements(Element0 , Element1, Element2 , Element3) in PieChart0 that associated Variable, Variable1, Variable2, Variable3 respectively



The screenshot shows the Delta software interface. At the top, a timeline displays a pie chart labeled "PieChart1". Below it, the "Pie Chart Element Editor" window is open, showing a list of elements: "Element0", "Element1", "Element2", and "Element3". A red circle with the number "2" is next to "Element0", and a text box says "Click 'Add' button to add elements". To the right, the "Chart" control properties are visible. A red circle with the number "1" is next to the "Collection" property, which has a dropdown arrow. Below that, the "VariablePath" property is highlighted with a red box, showing "Var.Variable" and a red circle with the number "3" next to the dropdown arrow.

Property	Value
RefreshTime	00 : 00 : 01
Collection	Collection ...
Category	Element0
Color	[Red]
Value	50.00
VariablePath	Var.Variable ...

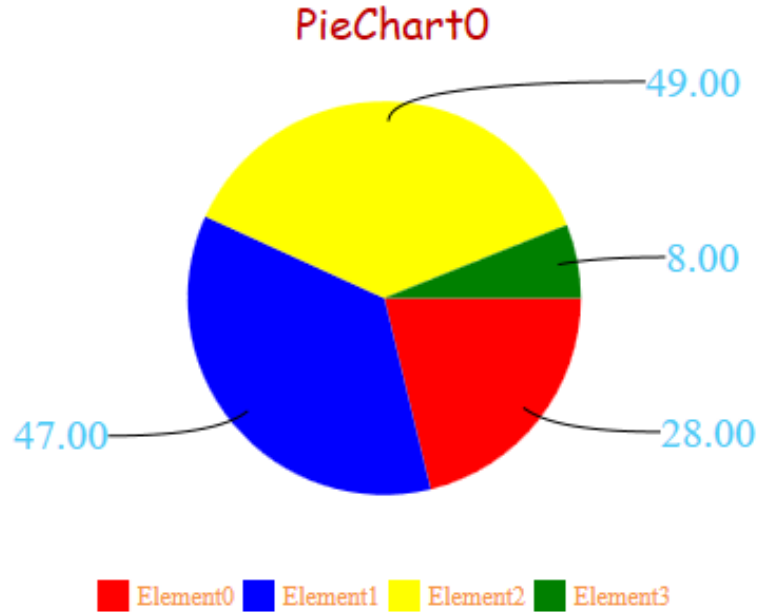
## (4)Set some properties

A screenshot of a software property window for a 'PieChart' control. The window is titled 'Property' and has a close button. It is organized into sections: 'Base', 'Appearance', 'Chart', 'Data Format', 'Text', 'Transform', and 'Layout'. The 'Appearance' section is expanded, showing properties like Title, LabelFont, LabelColor, LabelPosition, TitleFont, TitleColor, LegendFont, LegendColor, LegendLocation, and Background. The 'Chart' section is also expanded, showing RefreshTime, Collection, ColorMode, and LabelFormat. The 'Data Format' section is expanded, showing LabelContent and LabelFormat. Several properties are highlighted with red boxes: LabelColor (set to a blue color swatch), LabelPosition (set to 'OutsideExtended'), TitleColor (set to a red color swatch), LegendColor (set to an orange color swatch), ColorMode (set to 'Custom'), and LabelFormat (set to 'Double2').

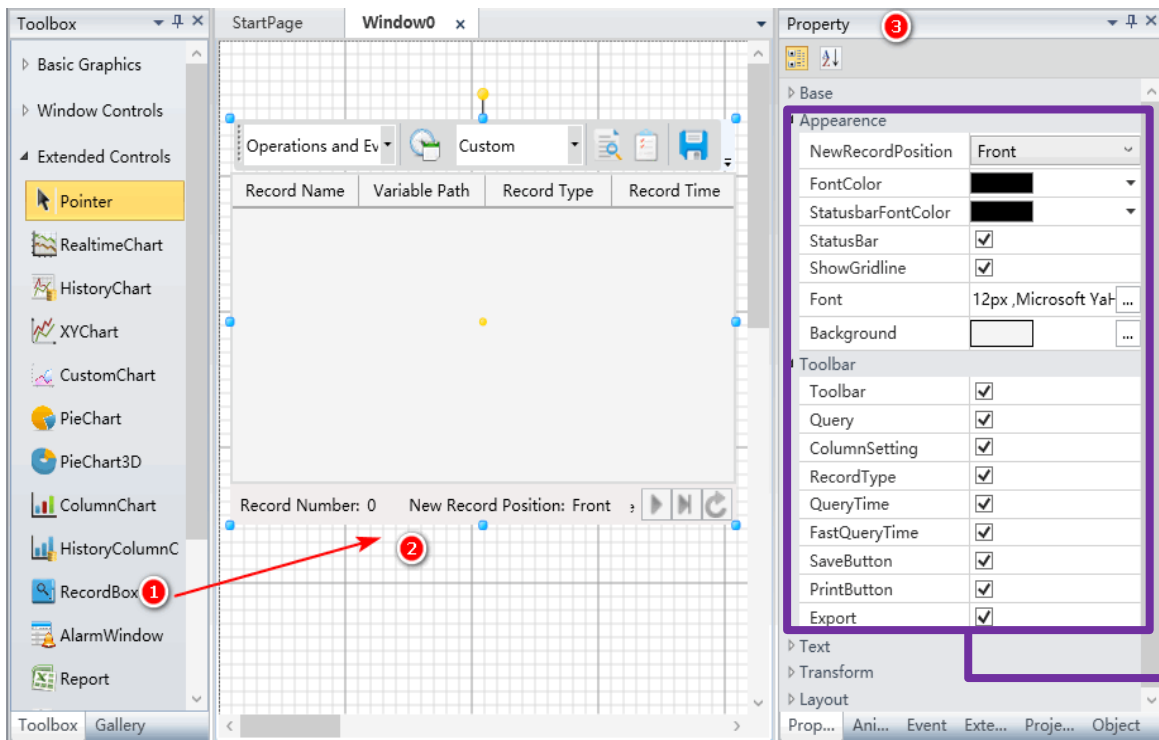
Property	
Base	
Appearance	
Title	PieChart0
LabelFont	20px, Times New Roman
LabelColor	[Blue Swatch]
LabelPosition	OutsideExtended
TitleFont	20px, Comic Sans MS
TitleColor	[Red Swatch]
LegendFont	12px, Times New Roman
LegendColor	[Orange Swatch]
LegendLocation	Bottom
Background	[White Swatch]
Chart	
RefreshTime	00 : 00 : 01
Collection	Collection
ColorMode	Custom
Data Format	
LabelContent	YValue
LabelFormat	Double2
Text	
Transform	
Layout	



(5)Run the project



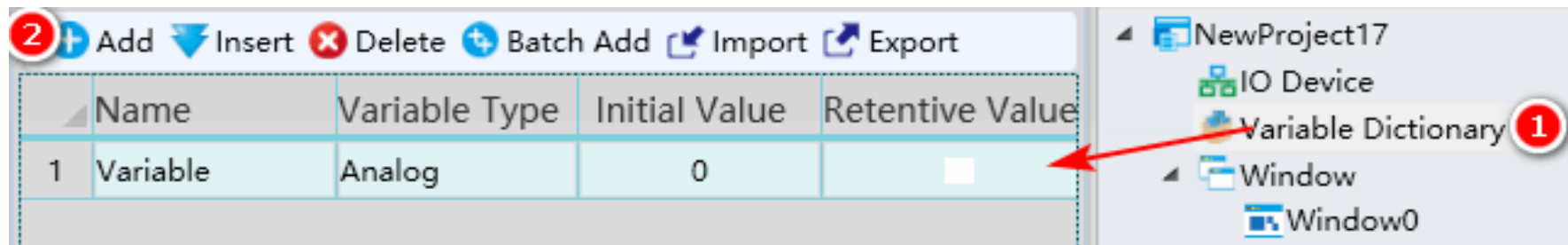
➤ Create a RecordBox0, open its properties window



- **Font:** Set font style
- **FontColor:** Set font color
- **StatusBar:** Set whether to display the status bar.
- **StatusbarFontColor:** Set the color of the status bar font.
- **ShowGridline:** Set whether to display grid.
- **NewRecordPosition:** Set the location of the latest record.
- **Background:** Set control background
- **Toolbar:** Set whether to display the toolbar

➤ RecordBox record operation variables:

(1) Create a Variable: Variable



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 'Retentive Value' column in the table. A red circle with the number '1' is next to the 'Variable Dictionary' in the tree view.

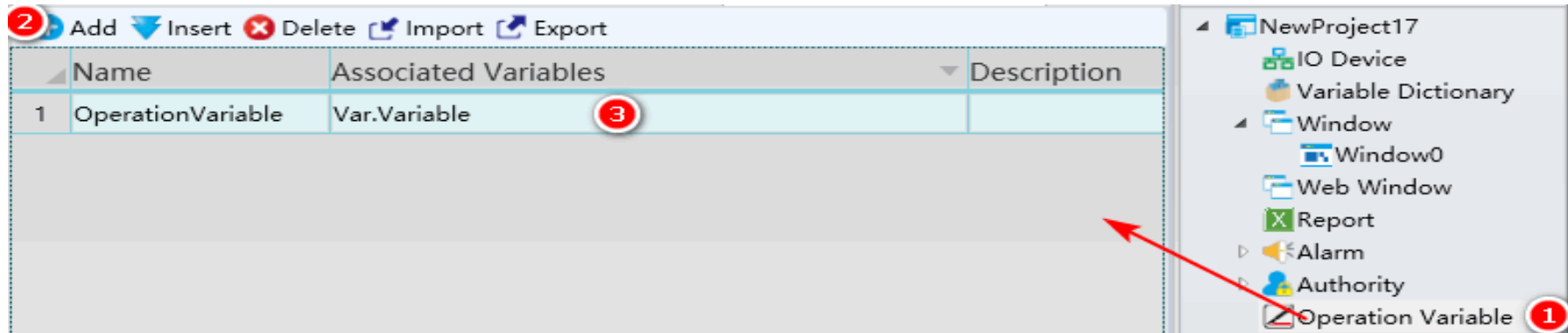
	Name	Variable Type	Initial Value	Retentive Value
1	Variable	Analog	0	

Tree View Structure:

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

※Refer to the section "6.3 Variables" in user manual.

(2) Create a OperationVariable that associated Variable



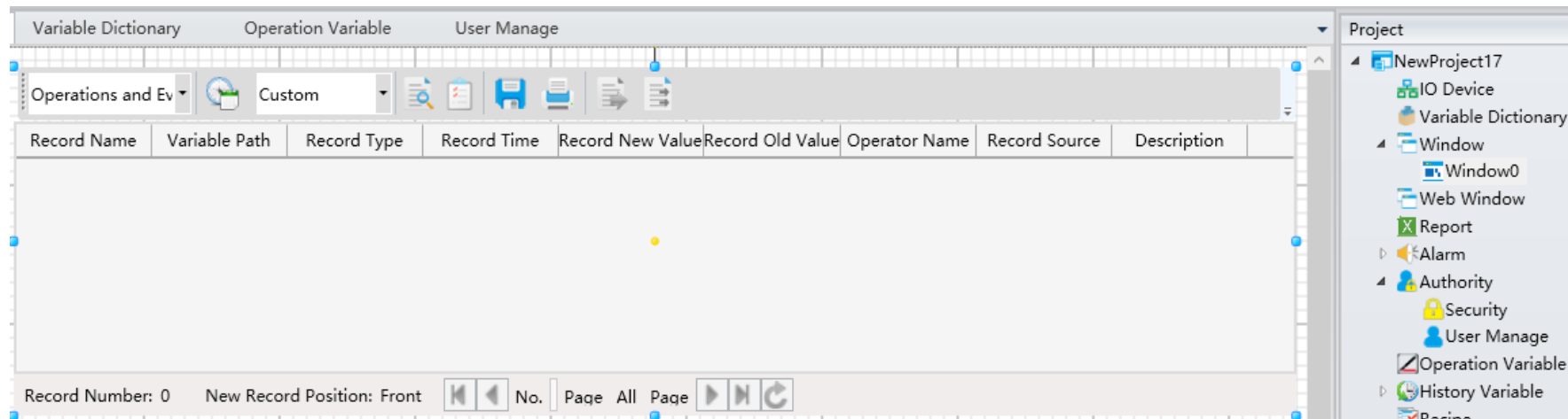
The screenshot shows the Delta software interface. On the left, a menu bar includes 'Add', 'Insert', 'Delete', 'Import', and 'Export'. Below this is a table with three columns: 'Name', 'Associated Variables', and 'Description'. The table contains one row with the following data:

	Name	Associated Variables	Description
1	OperationVariable	Var.Variable	

On the right side of the interface, there is a tree view showing the project structure. The tree includes 'NewProject17' with sub-items: 'IO Device', 'Variable Dictionary', 'Window', 'Window0', 'Web Window', 'Report', 'Alarm', 'Authority', and 'Operation Variable'. A red arrow points from the 'Operation Variable' item in the tree to the 'Var.Variable' cell in the table. A red circle with the number '1' is next to the 'Operation Variable' item in the tree, and a red circle with the number '3' is next to the 'Var.Variable' cell in the table.

※Refer to the section "13 Operation variable" in user manual.

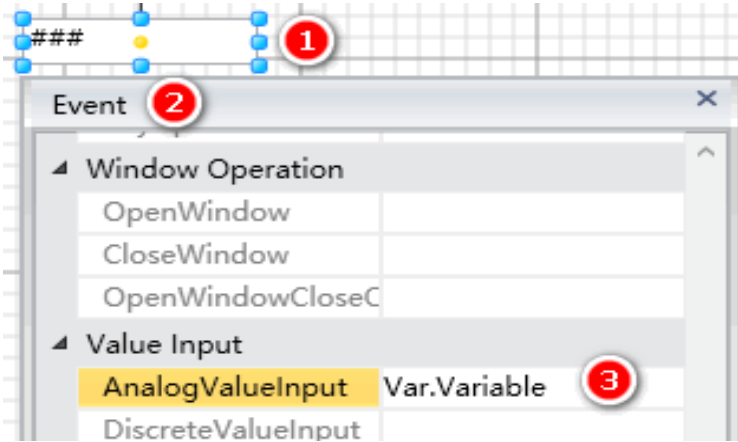
## (3) Create a RecordBox0 in the Window0



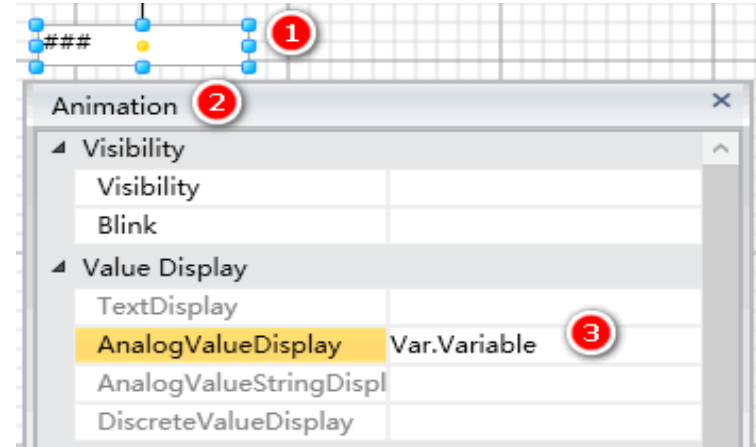
The screenshot displays the Delta software interface for configuring a RecordBox. The main window is titled "RecordBox0" and features a toolbar with icons for operations and events, a "Custom" dropdown, and a "Record Name" field. Below the toolbar is a table with columns: Record Name, Variable Path, Record Type, Record Time, Record New Value, Record Old Value, Operator Name, Record Source, and Description. The table is currently empty. At the bottom, there is a status bar showing "Record Number: 0", "New Record Position: Front", and navigation buttons. On the right side, a "Project" tree view shows the hierarchy: NewProject17 > IO Device > Variable Dictionary > Window > Window0 > Web Window > Report > Alarm > Authority > Security > User Manage > Operation Variable > History Variable > RecordBox0.

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

(4) Create a Textbox0 in Window0, the analog value input event and analog value display animation of TextBox0 are associated with Variable

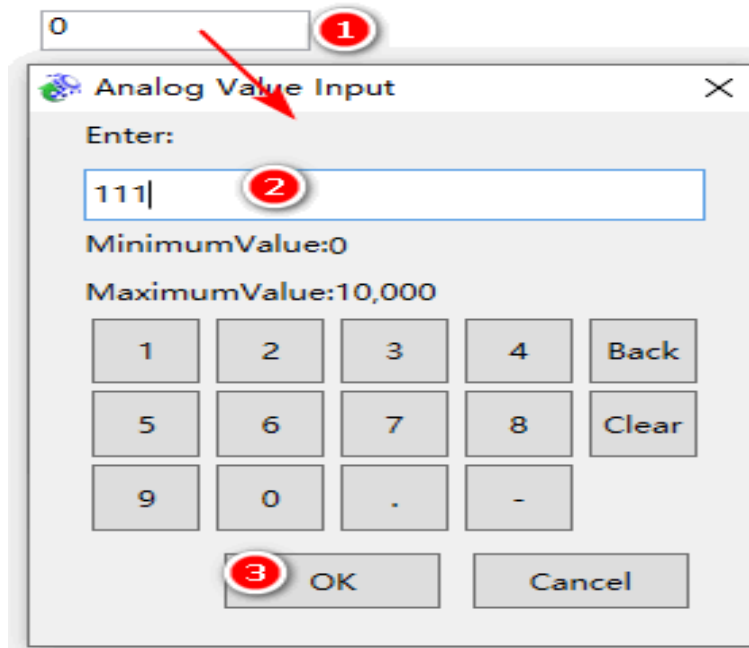


Analog Value Input



Analog Value Display

(5)Run the project. In Textbox0, input 111, then input 222



0

1

Analog Value Input

Enter:

111

2

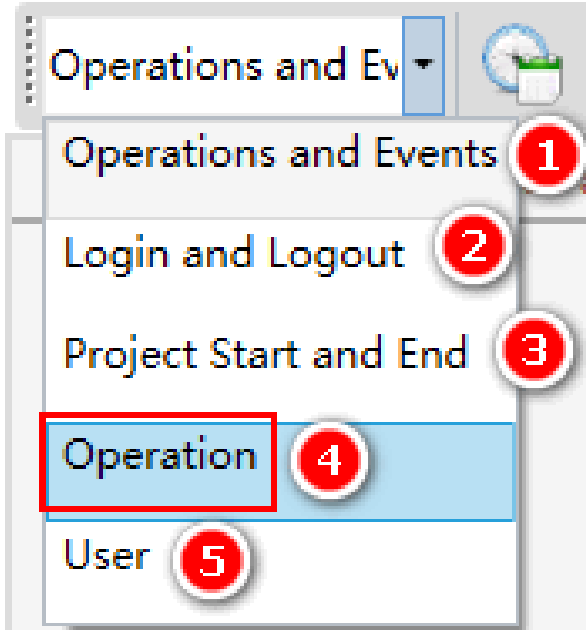
MinimumValue:0

MaximumValue:10,000

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

3 OK Cancel

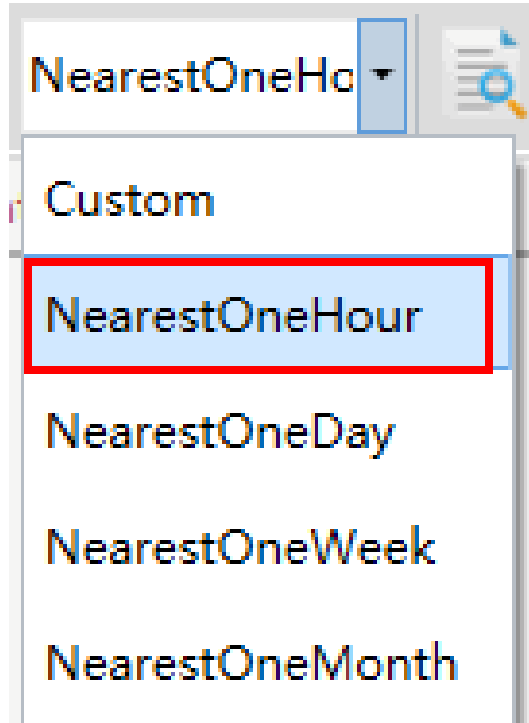
(6) Set query type: select the Operation



- ① Record all operations and events
- ② Only user login and logout
- ③ Only project start and project end
- ④ Only operations
- ⑤ Only user parameters modification









(7)Set Query time: select NearestOnehour



(8) Click the query button to query the records of operation variables

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		CNWI6IAPC015	
OperationVariable	Var.Variable		3/30/2020 1:35:29 PM	111	0		CNWI6IAPC015	




<

>

Record Number: 2
New Record Position: Front

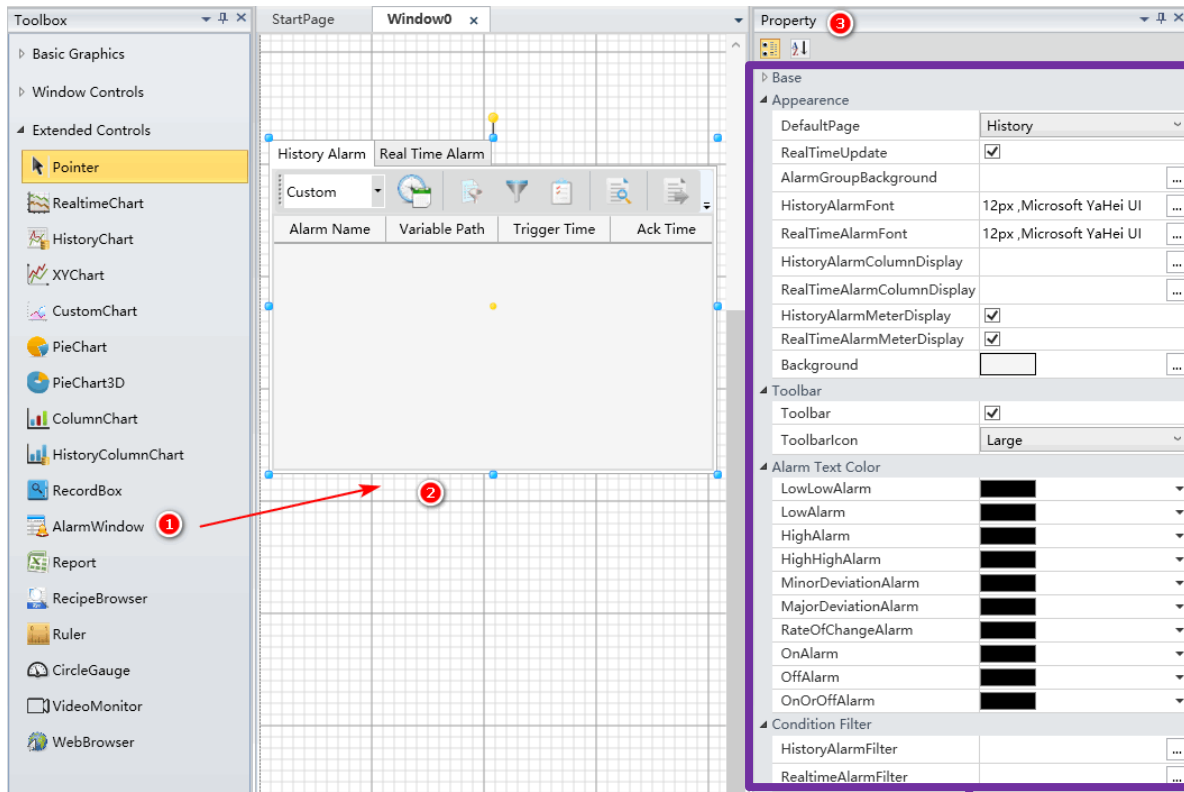
No. 
Page All 1 Page



# Extend Controls-AlarmWindow

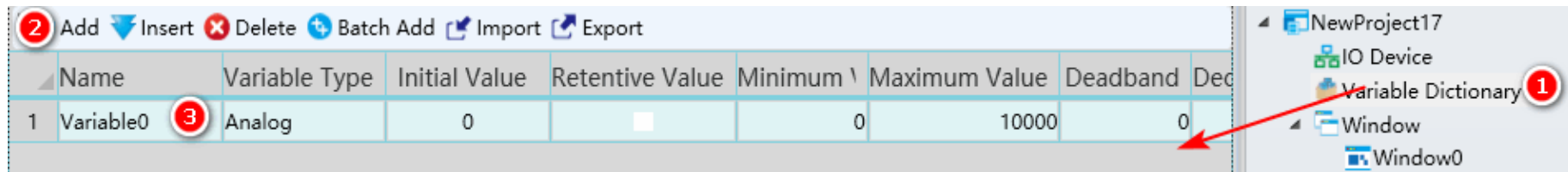
➤ Create a AlarmWindow0, open its properties



- **DefaultPage:** Set default page.
- **RealTimeUpdate:** Real time alarm window refresh data automatically
- **AlarmGroupBackground:** Set the background color for the alarm group.
- **HistoryAlarmColumnDisplay:** Set history alarm window column display.
- **RealTimeAlarmColumnDisplay:** Set real-time alarm window column display.
- **HistoryAlarmMeterDisplay:** Set whether to display the history alarm header.
- **RealTimeAlarmMeterDisplay:** Set whether to display the real-time alarm header.
- **Toolbar:** Set whether to display toolbar
- **ToolbarIcon:** Set the size of button in toolbar
- **Alarm Text Color:** Set alarm text color
- **HistoryAlarmFilter:** Set the filter conditions of historical alarm window
- **RealtimeAlarmFilter:** Set the filter conditions of real-time alarm window

➤ AlarmWindow show real-time alarms and query history alarms

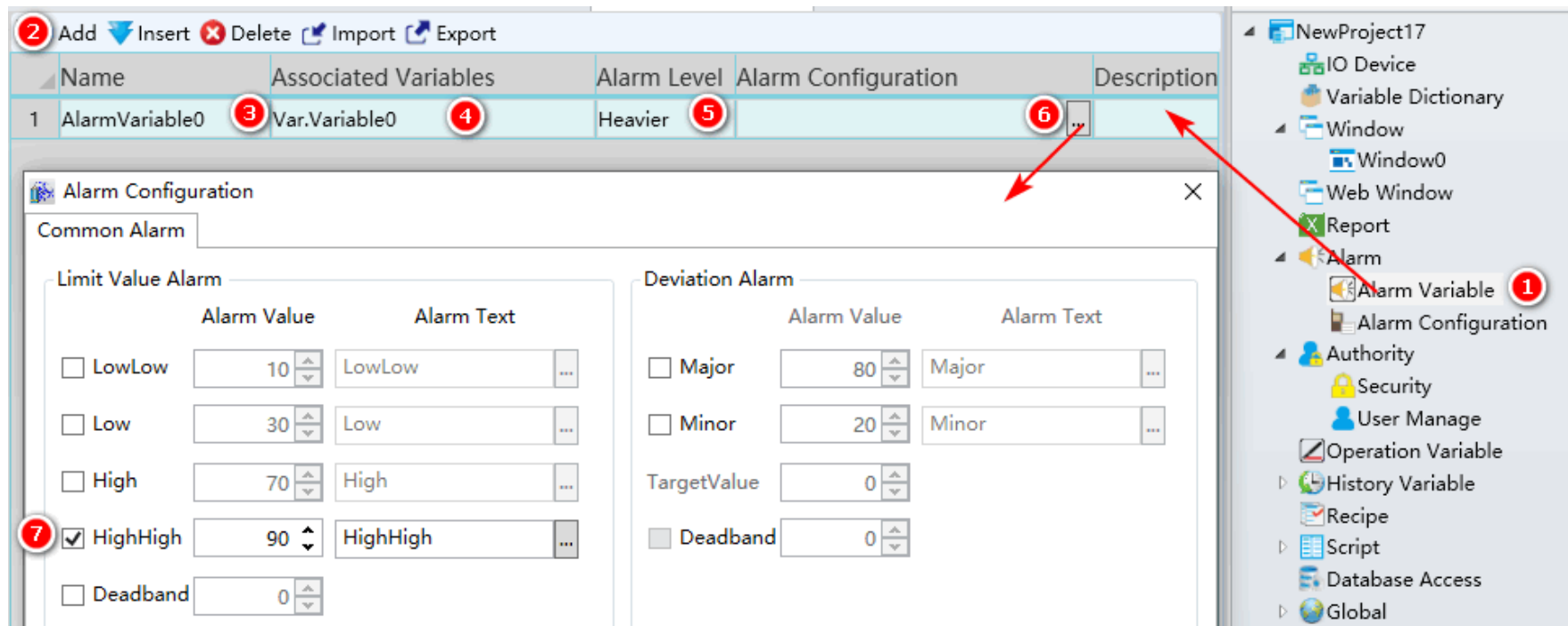
(1) Create a variable : Variable0



	Name	Variable Type	Initial Value	Retentive Value	Minimum \	Maximum Value	Deadband	Dec
1	Variable0	Analog	0	<input type="checkbox"/>	0	10000	0	

※Refer to the section "6.3 Variables" in user manual.

(2) Create a AlarmVariable0 that associated with the Variable0, set some parameters of AlarmVariable0(HighHigh alarm)



**Alarm Configuration Table**

Name	Associated Variables	Alarm Level	Alarm Configuration	Description
1 AlarmVariable0	Var.Variable0	Heavier		

**Alarm Configuration Window**

**Common Alarm**

**Limit Value Alarm**

	Alarm Value	Alarm Text
<input type="checkbox"/> LowLow	10	LowLow
<input type="checkbox"/> Low	30	Low
<input type="checkbox"/> High	70	High
<input checked="" type="checkbox"/> HighHigh	90	HighHigh
<input type="checkbox"/> Deadband	0	

**Deviation Alarm**

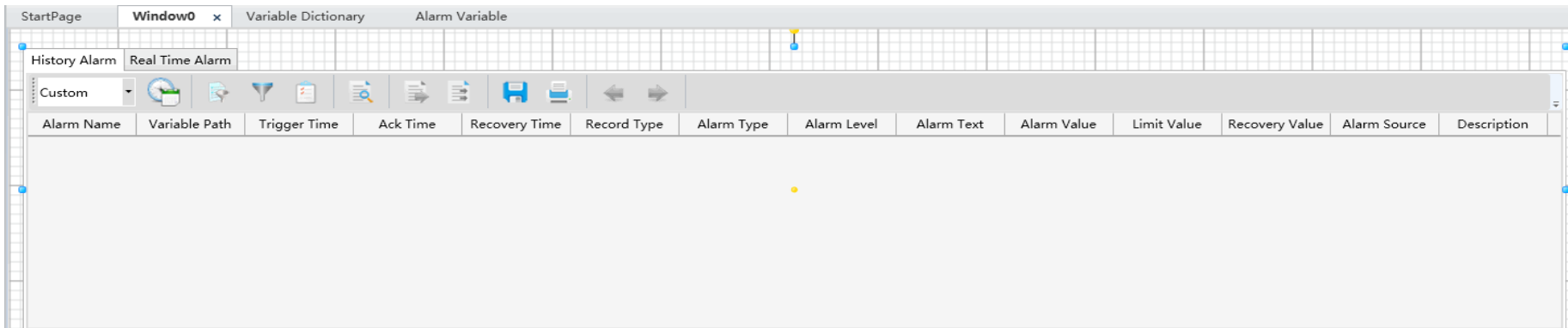
	Alarm Value	Alarm Text
<input type="checkbox"/> Major	80	Major
<input type="checkbox"/> Minor	20	Minor
TargetValue	0	
<input type="checkbox"/> Deadband	0	

**Project Tree**

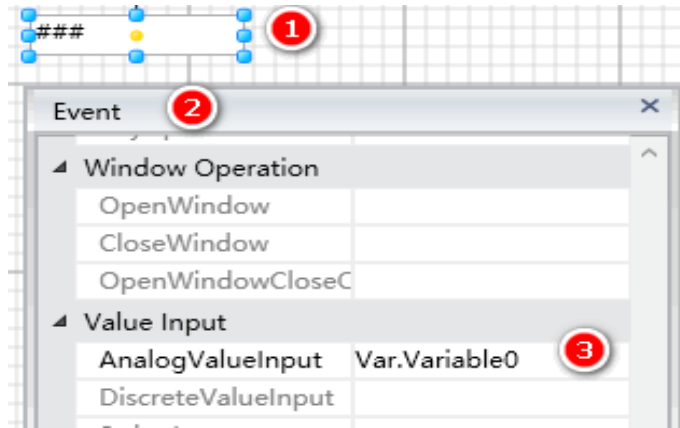
- NewProject17
  - IO Device
  - Variable Dictionary
  - Window
    - Window0
    - Web Window
  - Report
  - Alarm
    - Alarm Variable (1)
    - Alarm Configuration
  - Authority
    - Security
    - User Manage
  - Operation Variable
  - History Variable
  - Recipe
  - Script
  - Database Access
  - Global

※Refer to the section "11.3 Alarm variable" in user manual.

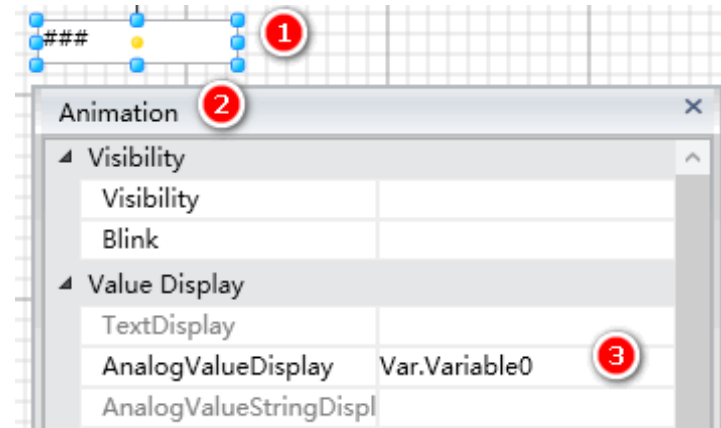
## (3) Create a AlarmWindow0 in Window0



(4) Create a Textbox0 in Window0, the analog value input event and analog value display animation of TextBox0 are associated with Variable

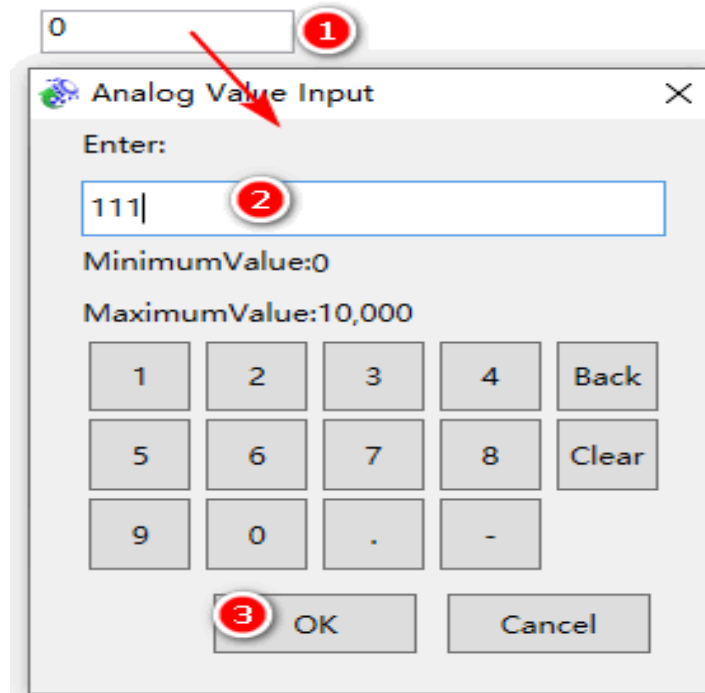


Analog Value Input



Analog Value Display

(5)Run the project. In Textbox0, input 111(Greater than the limit value of HighHigh alarm—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



(6)The Real Time Alarm window refresh realtime alarm at run time

History Alarm

Real Time Alarm1

2

⏸

Ack Selected

</



## Extend Controls-AlarmWindow

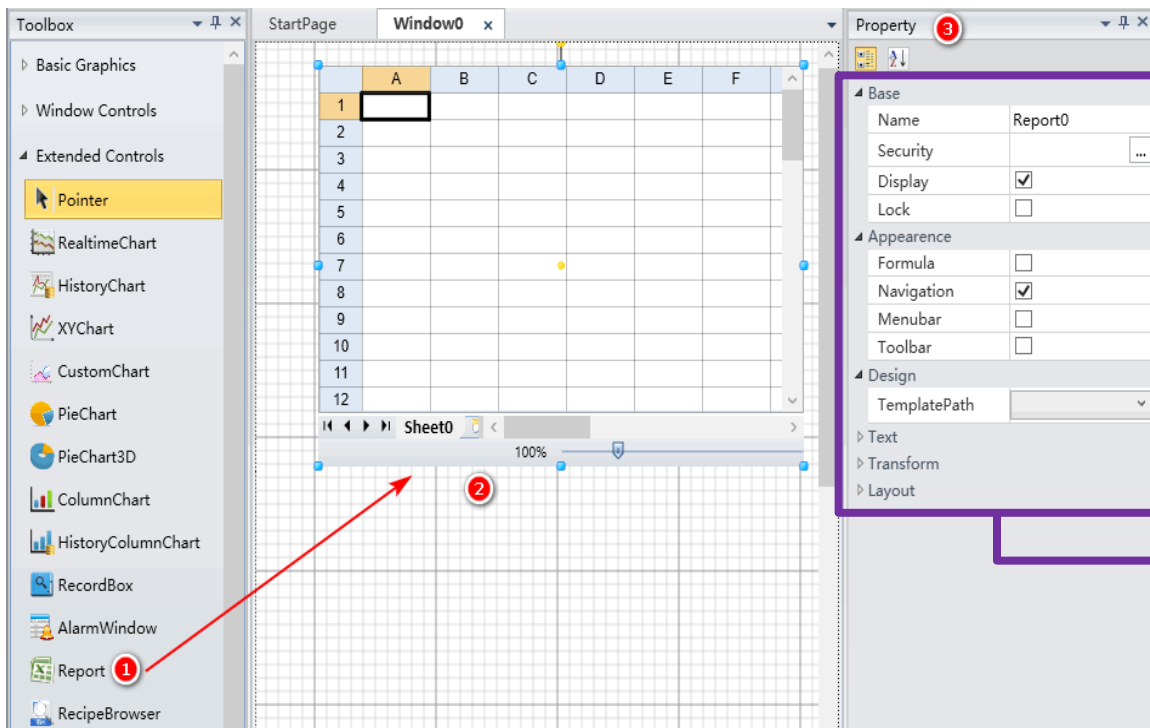
(7) The History Alarm window query history alarm at run time

**History Alarm**    **Real Time Alarm**

NearestOne | [Icons] | 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
1 Alarm.AlarmVariable0	Var.Variable0	3/31/2020 9:22:19 AM			Alarm	HighHighAlarm	Heavier	HighHigh	111	90		CNWJ6IAPC015	

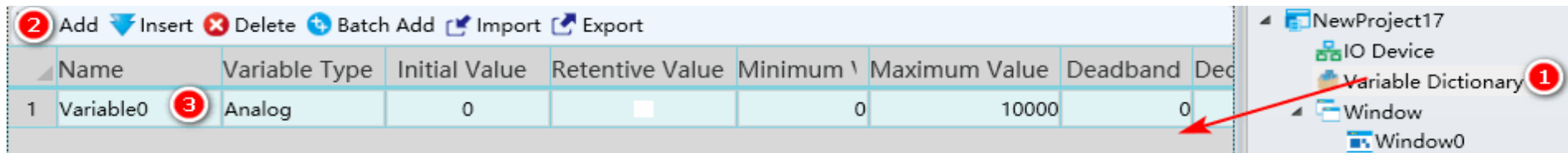
➤ Create a Report0, open its properties window



- **Formula:** Set whether to display formula bar
- **Navigation:** Set whether to display navigation bar
- **Menubar:** Set whether to display menu bar, check valid only when the current template path is null
- **ToolBar:** Set whether to display toolbar
- **TemplatePath:** Set current report template

➤ Report query history data:

(1) Create a variable: Variable0

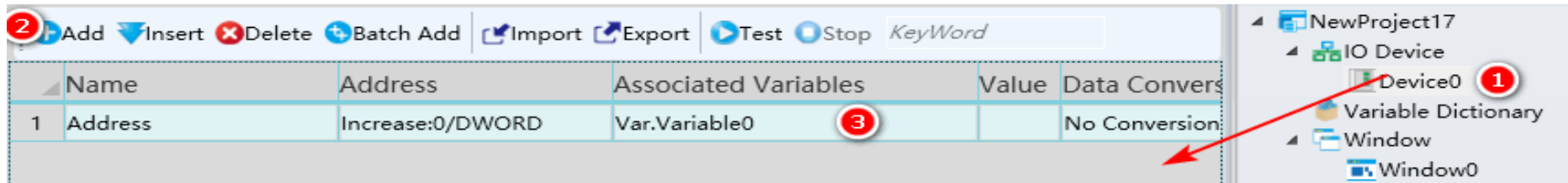


The screenshot displays the Delta Controls software interface. On the left, a table titled 'Variable Dictionary' lists variables. The first row shows 'Variable0' with an 'Analog' type, an initial value of 0, and a deadband of 0. A red circle with the number 3 is next to the 'Analog' type. Above the table, a toolbar contains buttons for 'Add', 'Insert', 'Delete', 'Batch Add', 'Import', and 'Export'. A red circle with the number 2 is next to the 'Add' button. On the right, a project tree shows the hierarchy: 'NewProject17' > 'IO Device' > 'Variable Dictionary' > 'Window' > 'Window0'. A red circle with the number 1 is next to 'Variable Dictionary'. A red arrow points from the 'Variable Dictionary' entry in the tree to the 'Variable0' row in the table.

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

※Refer to the section "6.3 Variables" in user manual.

(2) Create a simulation device-Device0, add a address in Device0 that associated Variable0



The screenshot shows the Delta simulation software interface. On the left, a toolbar contains buttons for 'Add' (2), 'Insert', 'Delete', 'Batch Add', 'Import', 'Export', 'Test', and 'Stop'. Below the toolbar is a table with the following columns: 'Name', 'Address', 'Associated Variables', 'Value', and 'Data Conversion'. The table has one row with the following data: 'Address', 'Increase:0/DWORD', 'Var.Variable0' (3), and 'No Conversion'. On the right, a tree view shows the project structure: 'NewProject17' > 'IO Device' > 'Device0' (1). A red arrow points from the 'Device0' entry in the tree view to the 'Var.Variable0' entry in the table.

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

✘ The method of creating a simulation device, refer to section “5.10.1 Simulator” in the user manual

(3) Create a history variable(RecordVariable) that associated Variable0

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	<input checked="" type="checkbox"/>	s1	Second	1	0001-01-01 00:00:00	Triggers every second
2	<input type="checkbox"/>	s10	Second	10	0001-01-01 00:00:00	Triggers every 10 seconds
3	<input type="checkbox"/>	s30	Second	30	0001-01-01 00:00:00	Triggers every 30 seconds
4	<input type="checkbox"/>	m1	Minute	1	0001-01-01 00:00:00	Triggers every minute
5	<input type="checkbox"/>	m30	Minute	30	0001-01-01 00:00:00	Triggers every 30 minutes
6	<input type="checkbox"/>	h1	Hour	1	0001-01-01 00:00:00	Triggers every hour

6

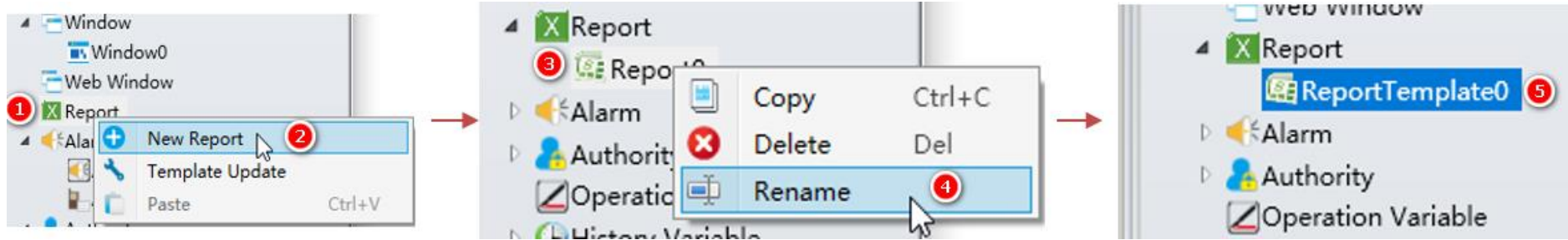
Project Tree:

- 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
    - Historical Group

1

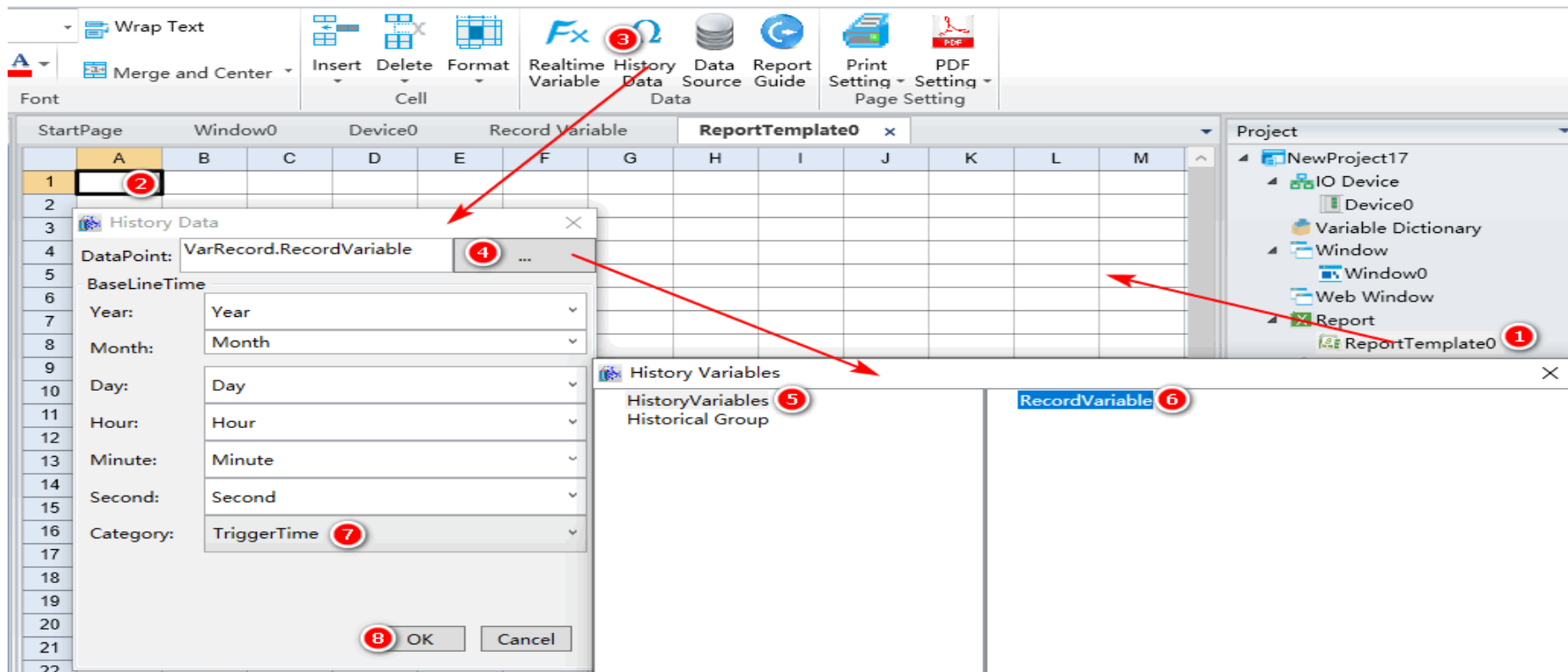
※Refer to the section "14.2 Setting history record variable" in user manual.

(4) Create a report template and name it ReportTemplate0



- ① Right click the Report node to open the menu
- ② Click New Report to add a report template: Report0
- ③ Right click the Report0 node to open the menu
- ④ Click the Rename to rename the report template
- ⑤ The report template is renamed as ReportTemplate0

(5) The cell in the first row , column A in the ReportTemplate0 is associated with the trigger time of RecordVariable



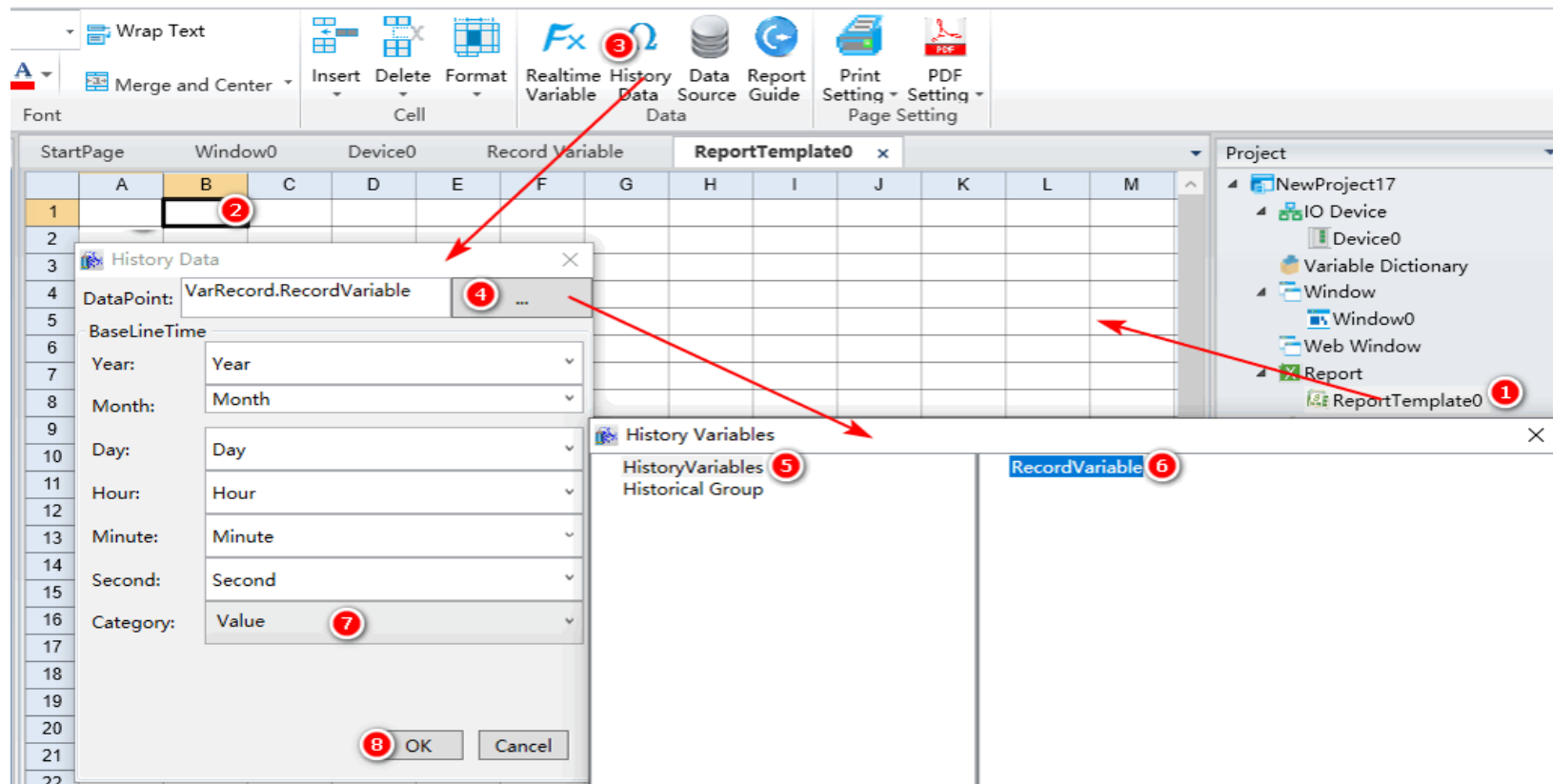
The screenshot displays the Delta4 software interface with the following components and annotations:

- Menu Bar:** Includes options like Wrap Text, Merge and Center, Font, Insert, Delete, Format, Realtime Variable, History Data, Data Source, Report Guide, Print Setting, and PDF Setting.
- Spreadsheet Grid:** The active sheet is **ReportTemplate0**. The first row, column A (cell A1) is highlighted with a red circle (2). The grid has columns labeled A through M and rows numbered 1 through 22.
- Project Tree:** Located on the right, it shows a hierarchy: NewProject17 > IO Device > Device0 > Variable Dictionary > Window > Window0 > Web Window > Report > ReportTemplate0 (1).
- History Data Dialog:** Opened over the spreadsheet, it shows:
  - DataPoint: VarRecord.RecordVariable (4)
  - BaseLineTime: Year (7)
  - Month: Month (7)
  - Day: Day (7)
  - Hour: Hour (7)
  - Minute: Minute (7)
  - Second: Second (7)
  - Category: TriggerTime (7)
  - Buttons: OK (8) and Cancel.
- History Variables Dialog:** Opened below the History Data dialog, it shows:
  - HistoryVariables Historical Group (5)
  - RecordVariable (6)

Red arrows indicate the flow of configuration: from the Project Tree (1) to the History Variables dialog (5), then to the History Data dialog (4), and finally to the cell A1 (2).

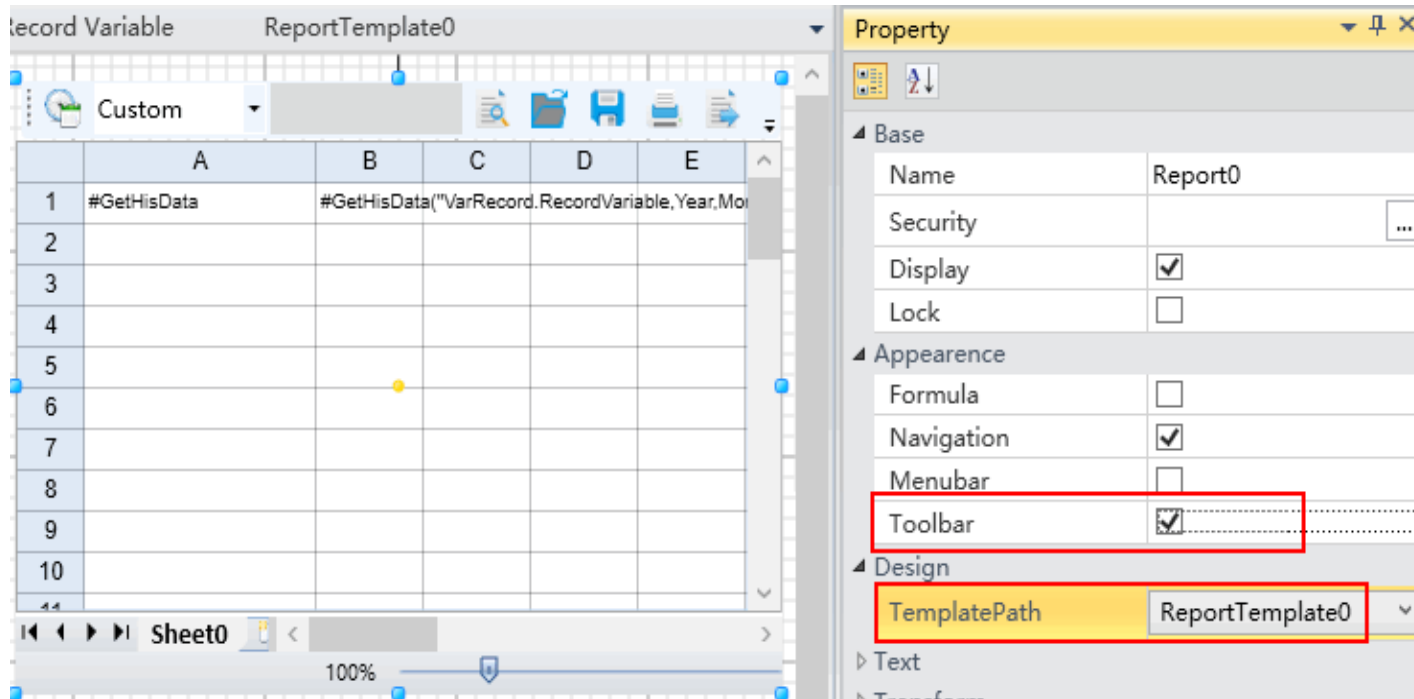


(6) The cell in the first row , column B in the ReportTemplate0 is associated with the value of the RecordVariable



The screenshot displays the Delta software interface. The main window is titled "ReportTemplate0" and shows a grid with columns A through M and rows 1 through 22. The cell in the first row, column B (B1) is highlighted with a red circle labeled 2. A red arrow points from this cell to the "History Data" dialog box, which is open over the grid. The dialog box has a "DataPoint:" field containing "VarRecord.RecordVariable" (labeled 4) and a "Category:" field containing "Value" (labeled 7). The "History Variables" section of the dialog box is also visible, showing "HistoryVariables" (labeled 5) and "Historical Group" (labeled 6). The "Project" pane on the right shows the project structure, including "NewProject17", "IO Device", "Device0", "Variable Dictionary", "Window", "Web Window", and "Report" (labeled 1). The "Report" folder is expanded, showing "ReportTemplate0".

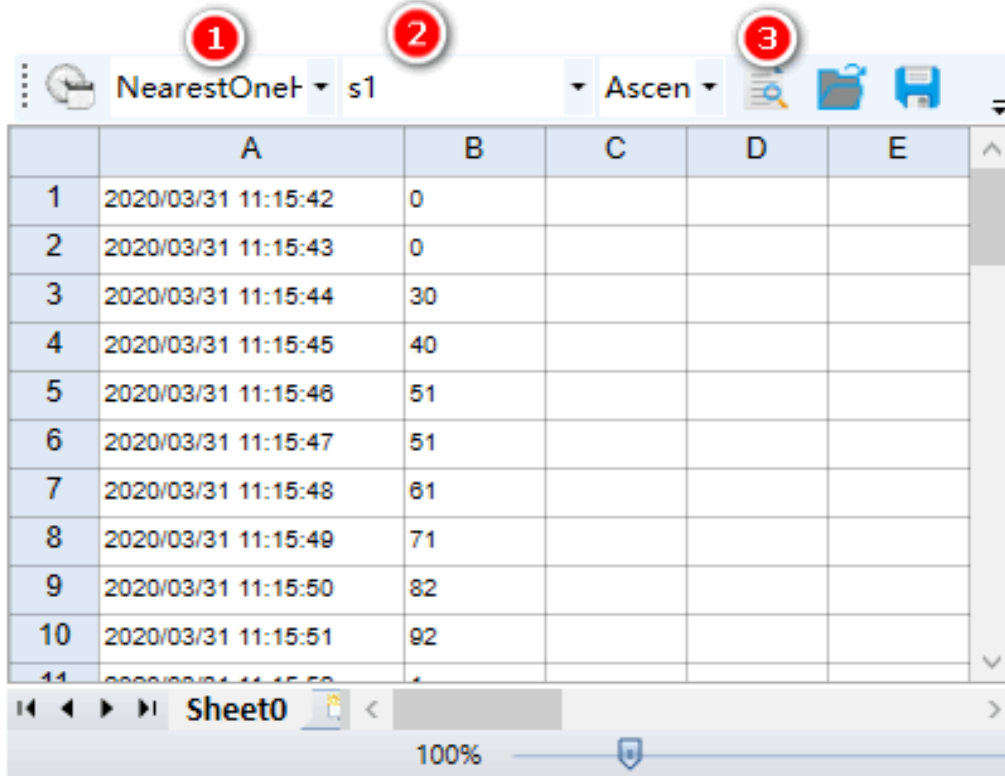
(7) Create a Report0 in Window0 and associate the ReportTemplate0, check the Toolbar property



The screenshot displays the Delta software interface. On the left, a spreadsheet titled 'ReportTemplate0' is visible, with columns A through E and rows 1 through 10. The 'Custom' toolbar is active. On the right, the 'Property' window is open, showing the configuration for 'Report0'. The 'Base' section includes properties for Name, Security, Display (checked), and Lock. The 'Appearance' section includes Formula, Navigation (checked), Menubar, and Toolbar (checked). The 'Design' section includes TemplatePath, which is set to 'ReportTemplate0'. The 'Toolbar' property in the 'Appearance' section and the 'TemplatePath' property in the 'Design' section are highlighted with red rectangles.

Property	Value
Name	Report0
Security	
Display	<input checked="" type="checkbox"/>
Lock	<input type="checkbox"/>
Formula	<input type="checkbox"/>
Navigation	<input checked="" type="checkbox"/>
Menubar	<input type="checkbox"/>
Toolbar	<input checked="" type="checkbox"/>
TemplatePath	ReportTemplate0

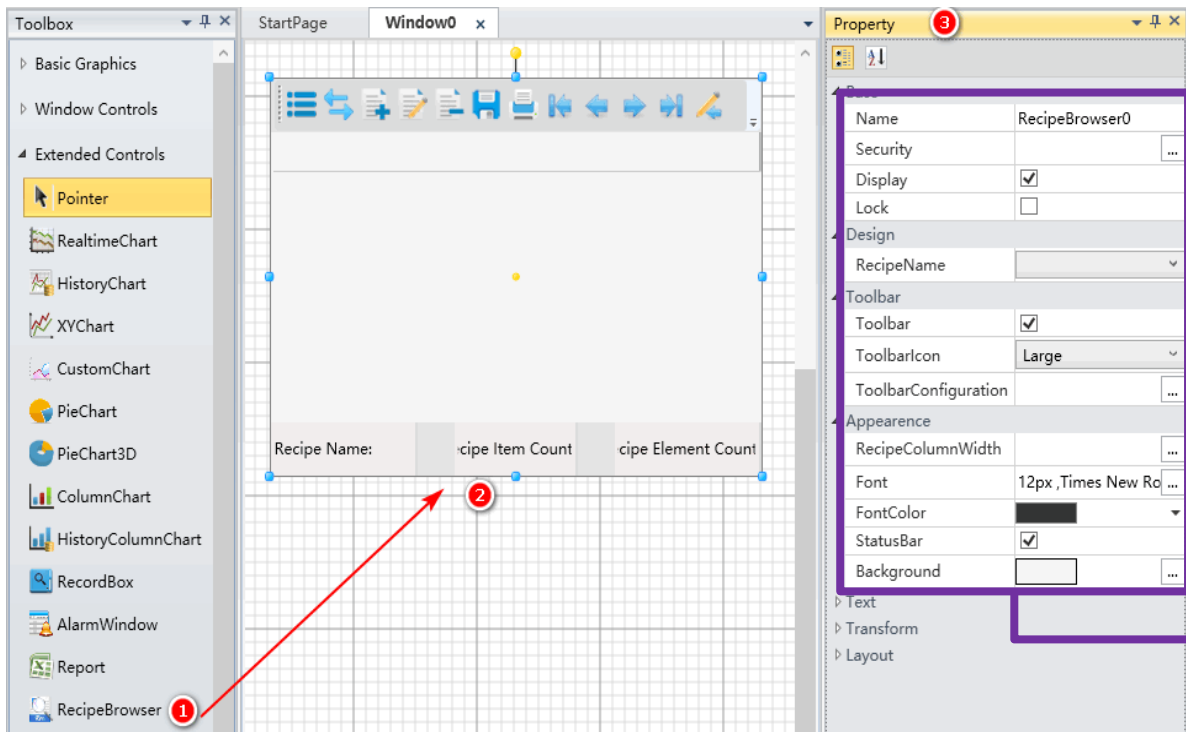
(8)Run the Window0 and query the last hour data at run time



	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	92			

# Extend Controls-RecipeBrowser

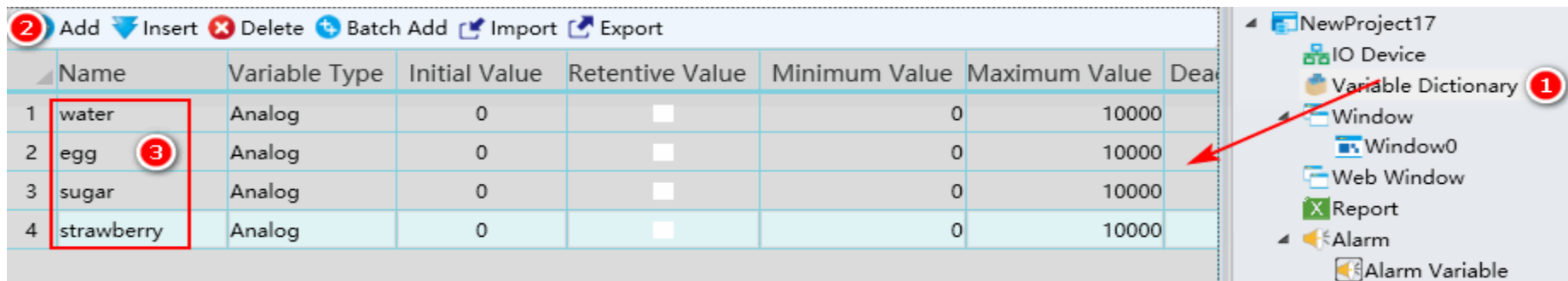
- Create a RecipeBrowser0, open its properties window



- **RecipeName:** Set the recipe bound by the recipe browser
- **Toolbar:** Set whether to display toolbar
- **ToolbarIcon:** Set the size of button in the toolbar
- **RecipeColumnWidth:** Set the recipe column width
- **Font:** Set the font of recipe browser
- **FontColor:** Set the font color of recipe browser
- **StatusBar:** Set whether to display the status bar

Use the recipe browser to display recipe:

(1) Create 4 Variables and name them water, egg, sugar, strawberry



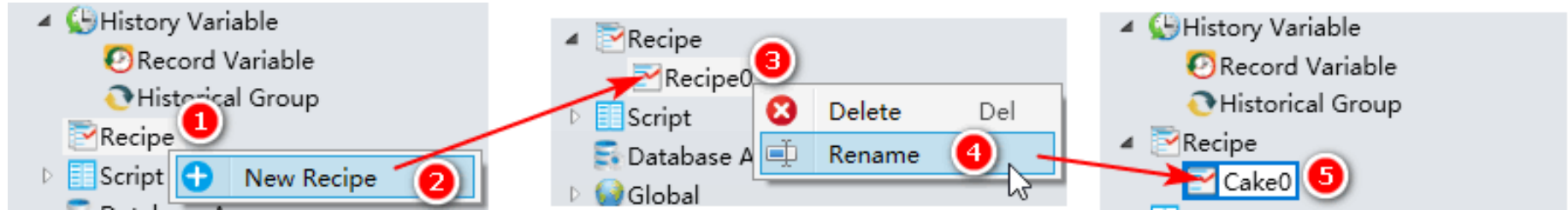
The screenshot displays the Delta Recipe Browser interface. On the left, a table lists four variables: 'water', 'egg', 'sugar', and 'strawberry'. Each variable is of type 'Analog' and has an initial value of 0, a retentive value checkbox, and a range from 0 to 10000. A red box highlights the first two rows, and a red circle with the number 3 is next to the 'egg' row. Above the table is a toolbar with buttons for Add, Insert, Delete, Batch Add, Import, and Export. On the right, a project tree shows the hierarchy: NewProject17 > IO Device > Variable Dictionary (circled with a red 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	<input type="checkbox"/>	0	10000	
2	egg	Analog	0	<input type="checkbox"/>	0	10000	
3	sugar	Analog	0	<input type="checkbox"/>	0	10000	
4	strawberry	Analog	0	<input type="checkbox"/>	0	10000	

- NewProject17
  - IO Device
    - Variable Dictionary (1)
    - Window
      - Window0
        - Web Window
          - Report
            - Alarm
              - Alarm Variable

※Refer to the section "6.3 Variables" in user manual.

(2) Create a recipe and name it Cake0



- ① Right click the Recipe node to open the menu
- ② Click New Recipe to add a recipe : Recipe0
- ③ Right click the Recipe0 node to open the menu
- ④ Click the Rename to name the recipe
- ⑤ The recipe is renamed as Cake0

(3) Create 4 recipe variables and name them Recipe\_water, Recipe\_egg, Recipe\_sugar, Recipe\_strawberry, which are associated with the water, egg, sugar and strawberry respectively

Name	Recipe Type	Associated Variables	Description	StrawberryCake1	StrawberryCake2
1 Recipe_water	Analog	Var.water		100	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 Click Cake0 node to open recipe interface

② Add 4 recipe elements

③ Name recipe elements

④ Recipe elements associated variables

⑤ Add recipe items

⑥⑦ Name recipe items and set the parameters of each recipe elements

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

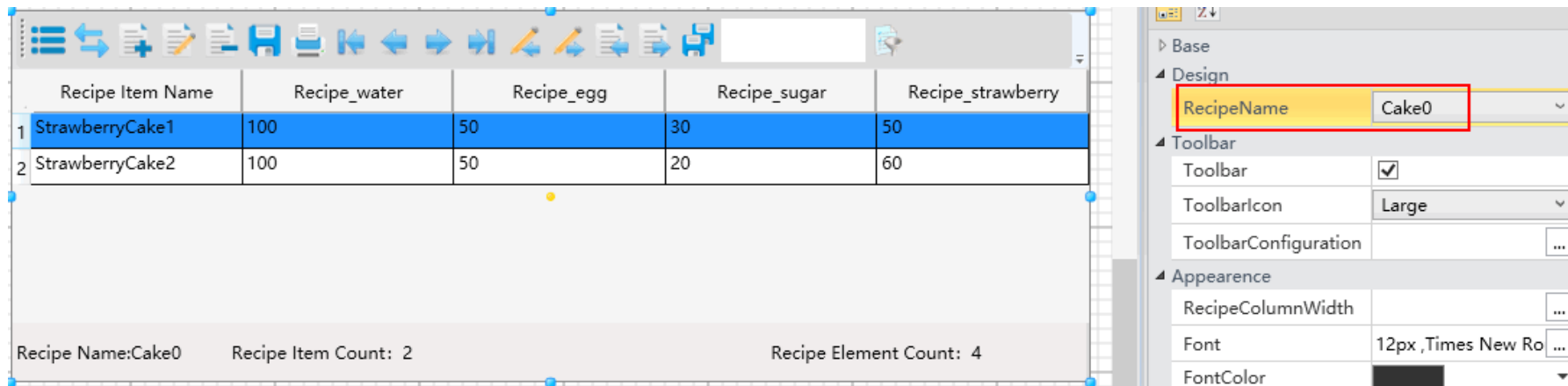
③   ④   ⑥   ⑦

①

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

## (4) Create RecipeBrowser0 and bind Cake0



The screenshot shows the RecipeBrowser control in a design tool. The control displays a table with recipe items and their ingredients. The 'RecipeName' property is highlighted in the Design tab of the Properties panel.

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

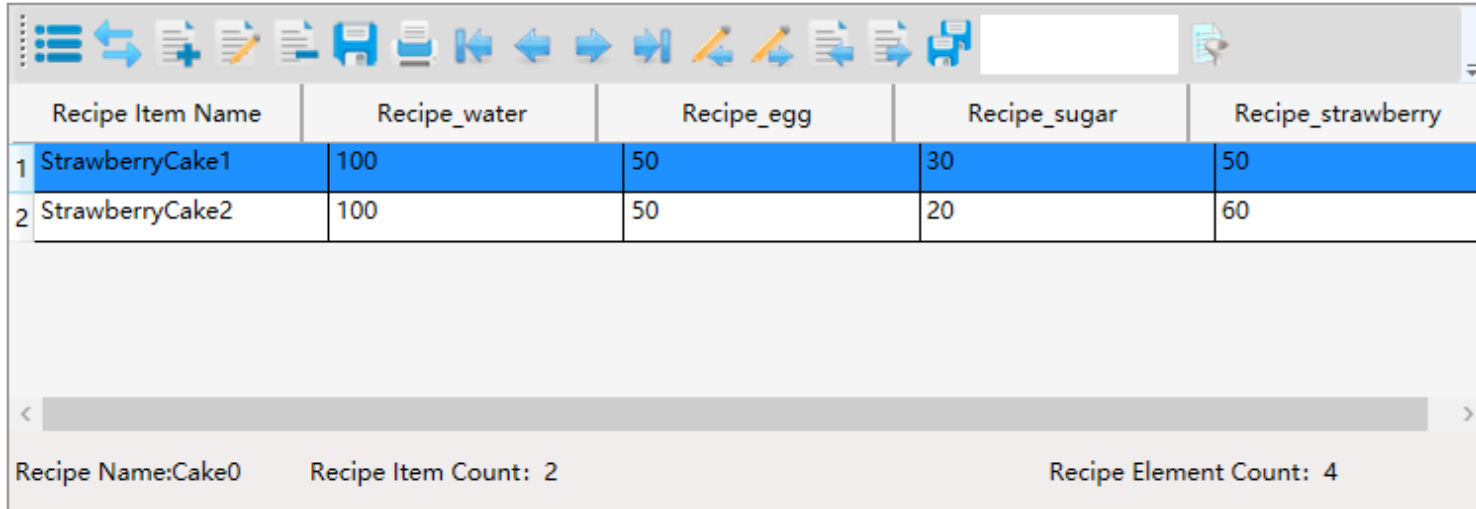
Recipe Name: Cake0      Recipe Item Count: 2      Recipe Element Count: 4

Properties Panel (Design Tab):

- Base
  - Design
    - RecipeName: Cake0
  - Toolbar
    - Toolbar: ☒
    - ToolbarIcon: Large
    - ToolbarConfiguration: ...
  - Appearance
    - RecipeColumnWidth: ...
    - Font: 12px, Times New Ro ...
    - FontColor: ...



## (5)Run the Window0



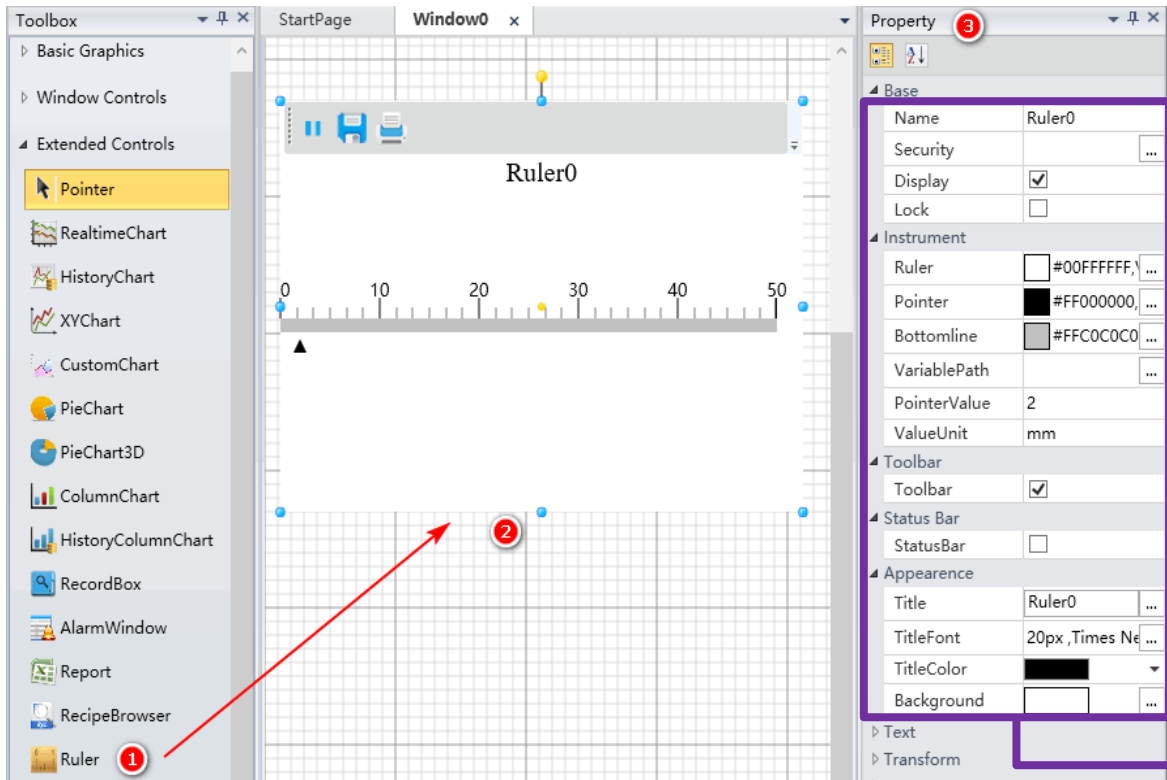
The screenshot shows a software window titled "RecipeBrowser". At the top is a toolbar with various icons for navigation and editing. Below the toolbar is a table with five columns: "Recipe Item Name", "Recipe\_water", "Recipe\_egg", "Recipe\_sugar", and "Recipe\_strawberry". The table contains two rows of data. The first row, "StrawberryCake1", has values 100, 50, 30, and 50. The second row, "StrawberryCake2", has values 100, 50, 20, and 60. Below the table is a horizontal scrollbar. At the bottom of the window, a status bar displays "Recipe Name: Cake0", "Recipe Item Count: 2", and "Recipe Element Count: 4".

	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

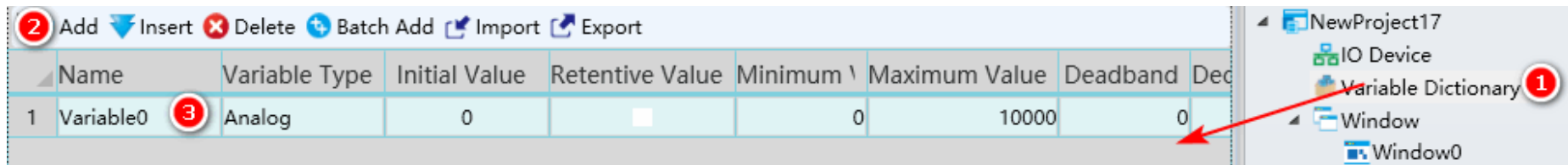
- Create a Ruler0, open its properties window



- **Ruler:** Set the parameters of ruler
- **Pointer:** Set the style of pointer
- **Bottomline:** Set the style of bottom line
- **VariablePath:** Set variable path associated with ruler
- **PointerValue:** Set the default value of the pointer
- **ValueUnit:** Set the value unit
- **Toolbar:** Set whether to display toolbar
- **StatusBar:** Set whether to display status bar
- **Title:** Set the content of title
- **TitleFont:** Set the font for the title
- **TitleColor:** Set the color of the title
- **Background:** Set the background color of ruler

➤ Ruler display real-time data:

(1) Create a variable : Variable0

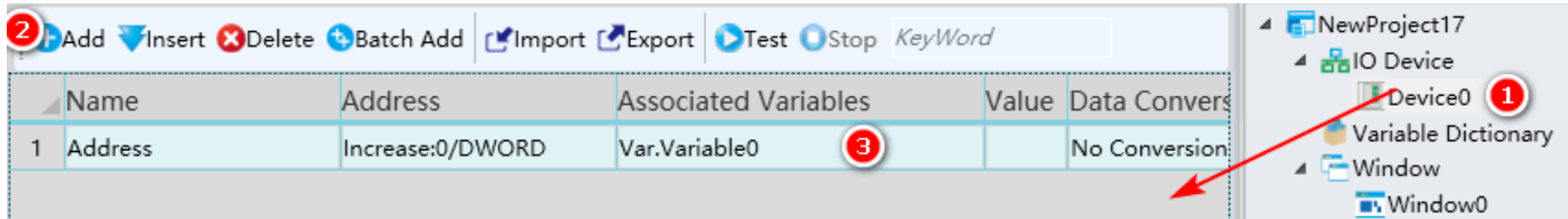


The screenshot shows the Delta Ruler software interface. On the left, a table lists variables. The first row is highlighted, showing 'Variable0' with an 'Analog' type, an initial value of '0', and a maximum value of '10000'. A red circle with the number '3' is next to the 'Variable0' entry. Above the table, a toolbar contains buttons for 'Add', 'Insert', 'Delete', 'Batch Add', 'Import', and 'Export'. A red circle with the number '2' is next to the 'Add' button. On the right, a project tree shows 'NewProject17' with sub-items 'IO Device', 'Variable Dictionary', 'Window', and 'Window0'. A red circle with the number '1' is next to 'Variable Dictionary'. A red arrow points from the 'Variable Dictionary' item to the 'Variable0' entry in the table.

	Name	Variable Type	Initial Value	Retentive Value	Minimum Value	Maximum Value	Deadband	Dec
1	Variable0	Analog	0	<input type="checkbox"/>	0	10000	0	

※Refer to the section "6.3 Variables" in user manual.

(2) Create a simulation device-Device0, add a address in Device0 that associated Variable0



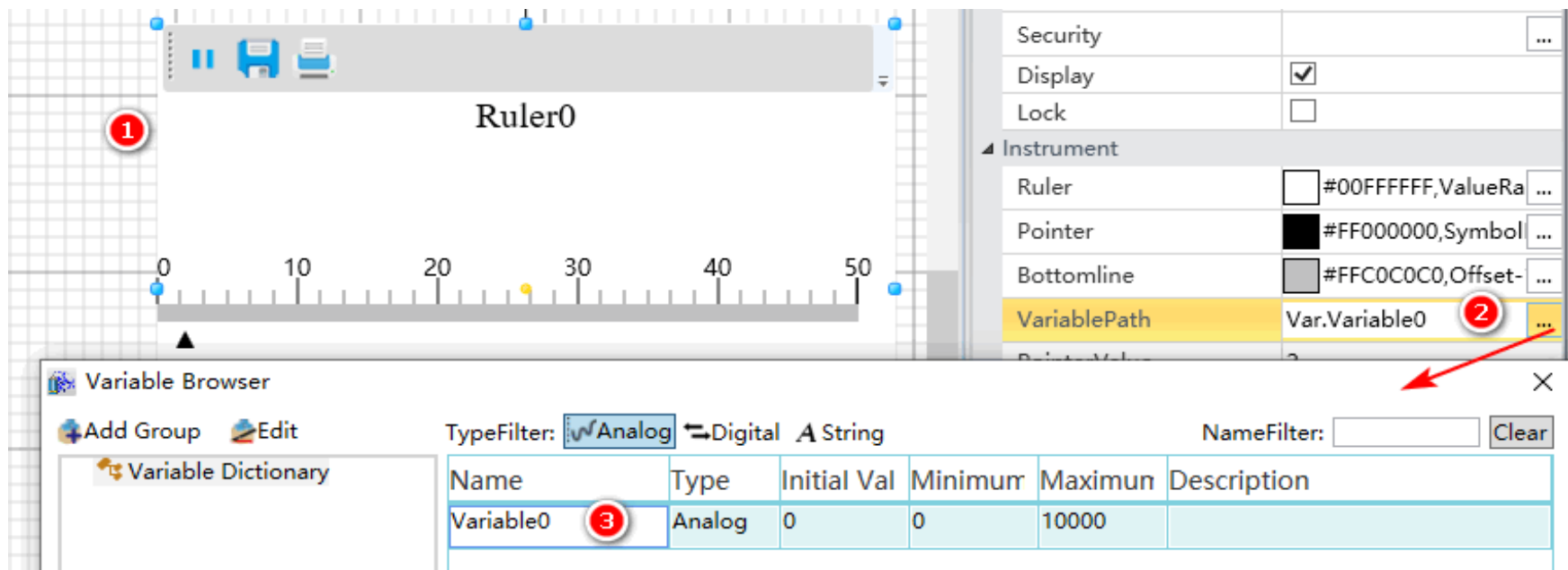
	Name	Address	Associated Variables	Value	Data Conversion
1	Address	Increase:0/DWORD	Var.Variable0		No Conversion

Toolbar: Add (2), Insert, Delete, Batch Add, Import, Export, Test, Stop, Keyword

Project Tree (Right): NewProject17  
├── IO Device  
│ └── Device0 (1)  
├── Variable Dictionary  
└── Window  
 └── Window0

✘ The method of creating a simulation device, refer to section “5.10.1 Simulator” in the user manual

(3) Create a Ruler0 in the Window0 that associated Variable0



**Ruler0**

0 10 20 30 40 50

**Instrument Properties**

Security	
Display	<input checked="" type="checkbox"/>
Lock	<input type="checkbox"/>
<b>Instrument</b>	
Ruler	#00FFFFFF, ValueRa ...
Pointer	#FF000000, Symbol ...
Bottomline	#FFC0C0C0, Offset- ...
VariablePath	Var.Variable0
Pointer Value	...

**Variable Browser**

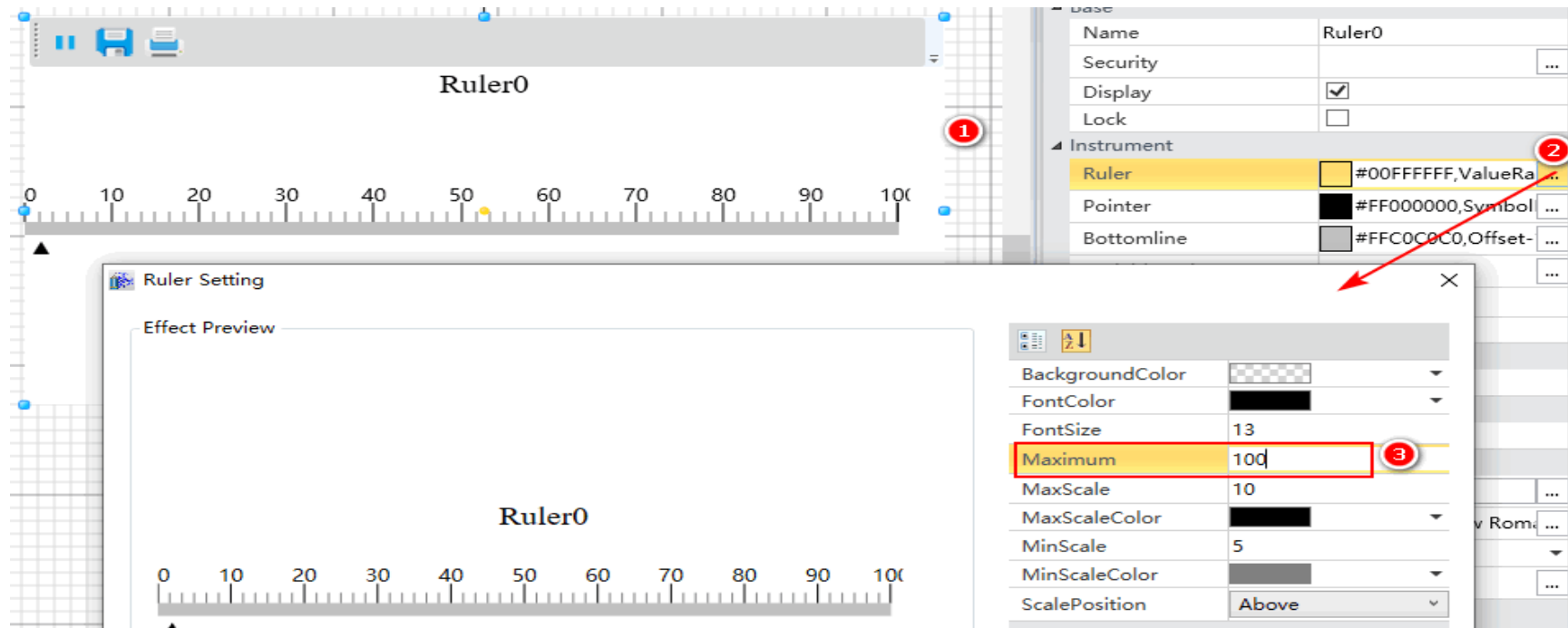
Add Group Edit

Variable Dictionary

TypeFilter: ☒ Analog ☐ Digital ☐ String NameFilter:  Clear

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

(4) Set Ruler0 maximum to 100

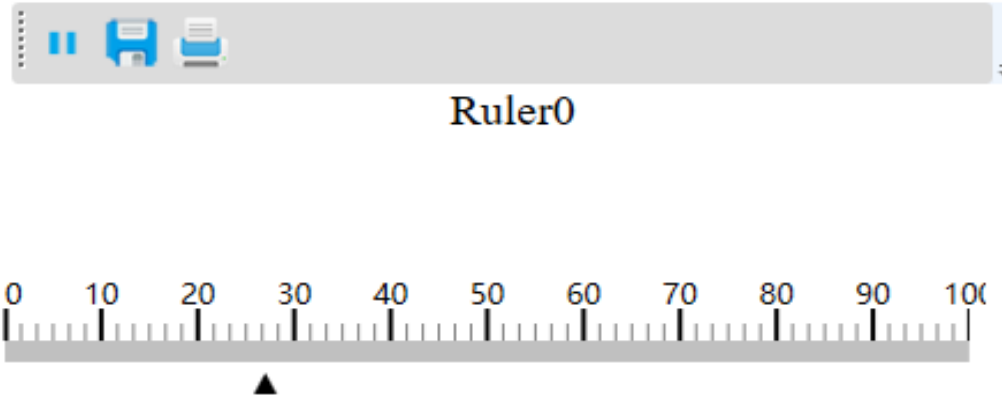


The image displays the Delta software interface for configuring a ruler control. The main window shows a ruler labeled "Ruler0" with a scale from 0 to 100. A red circle "1" is placed on the ruler's right edge. A "Ruler Setting" dialog is open, showing an "Effect Preview" of the ruler. A red circle "2" is placed on the "Ruler" property in the "Instrument" section of the settings panel. A red circle "3" is placed on the "Maximum" value of 100 in the "Ruler" settings dialog.

Base	
Name	Ruler0
Security	
Display	<input checked="" type="checkbox"/>
Lock	<input type="checkbox"/>
Instrument	
Ruler	#00FFFFFF,ValueRa...
Pointer	#FF000000,Symbol...
Bottomline	#FFC0C0C0,Offset-

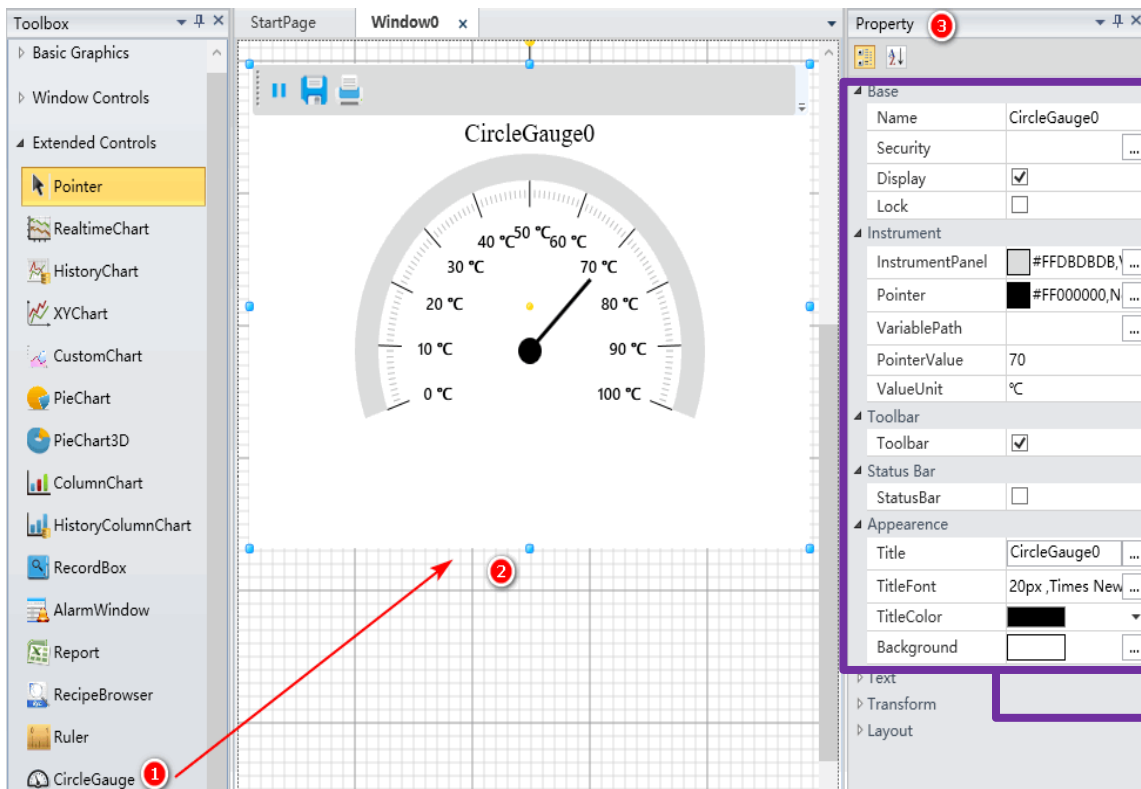
Ruler	
BackgroundColor	
FontColor	
FontSize	13
Maximum	100
MaxScale	10
MaxScaleColor	
MinScale	5
MinScaleColor	
ScalePosition	Above

(5) Run the project



# Extend Controls-CircleGauge

➤ Create a CircleGauge0, open its properties window

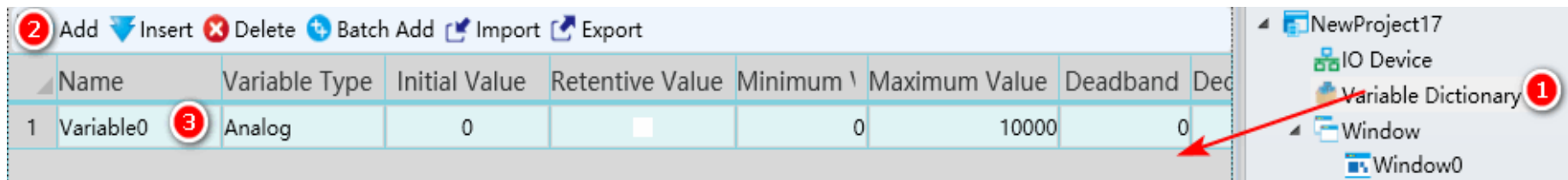


- **InstrumentPanel**: Set the parameters of instrument panel
- **Pointer**: Set the style of pointer
- **VariablePath**: Set variable path associated with CircleGauge
- **PointerValue**: Set the default value of the pointer
- **ValueUnit**: Set the value unit
- **Toolbar**: Set whether to display toolbar
- **StatusBar**: Set whether to display status bar
- **Title**: Set the content of title
- **TitleFont**: Set the font for the title
- **TitleColor**: Set the color of the title
- **Background**: Set the background color of CircleGauge



➤ CircleGauge display real-time data:

(1) Create a variable : Variable0

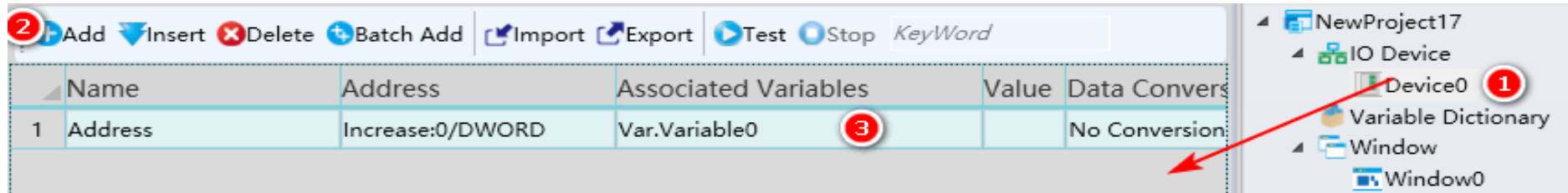


The screenshot shows the Delta software interface. On the right, the 'Variable Dictionary' is expanded, showing a tree structure with 'NewProject17' at the top, followed by 'IO Device', 'Variable Dictionary' (marked with a red circle 1), 'Window', and 'Window0'. A red arrow points from the 'Variable Dictionary' to the table below. The table has columns: Name, Variable Type, Initial Value, Retentive Value, Minimum Value, Maximum Value, Deadband, and Decimals. The first row is 'Variable0' (marked with a red circle 3), which is of type 'Analog', with an initial value of 0, a retentive value of 0, a minimum value of 0, a maximum value of 10000, a deadband of 0, and 0 decimals. The table is also marked with a red circle 2.

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

※Refer to the section "6.3 Variables" in user manual.

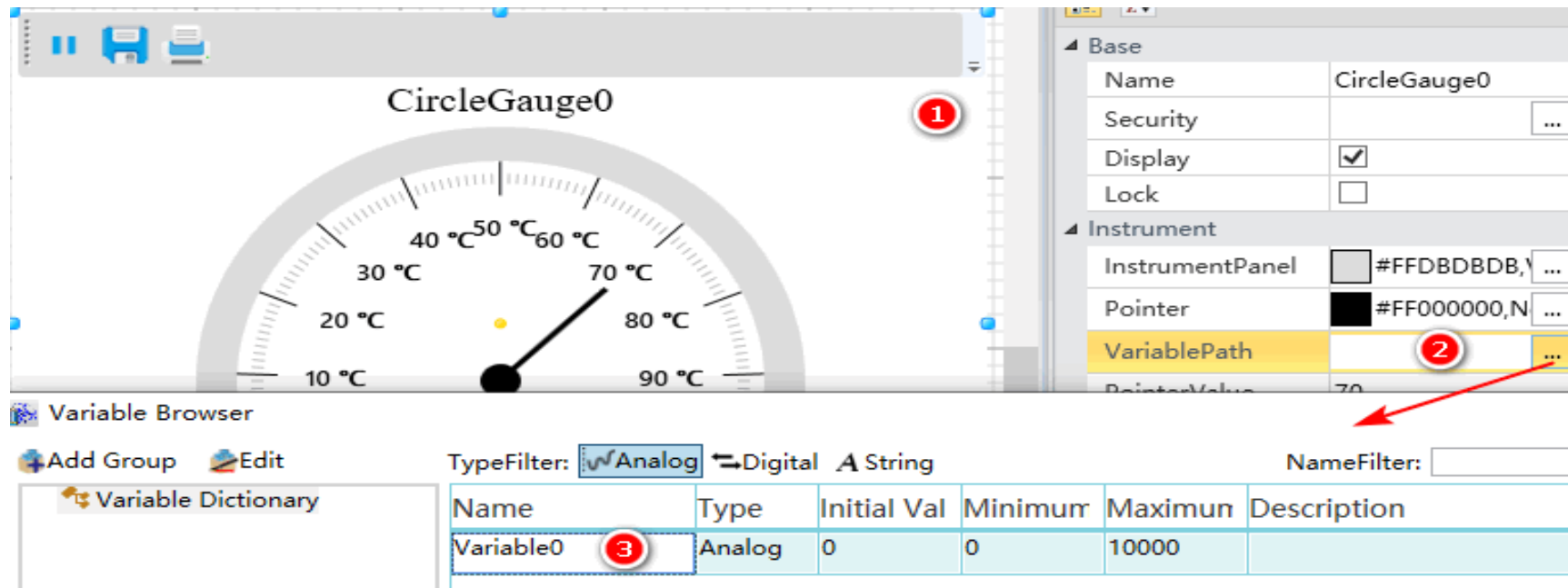
(2) Create a simulation device-Device0, add a address in Device0 that associated Variable0



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

※ The method of creating a simulation device, refer to section “5.10.1 Simulator” in the user manual

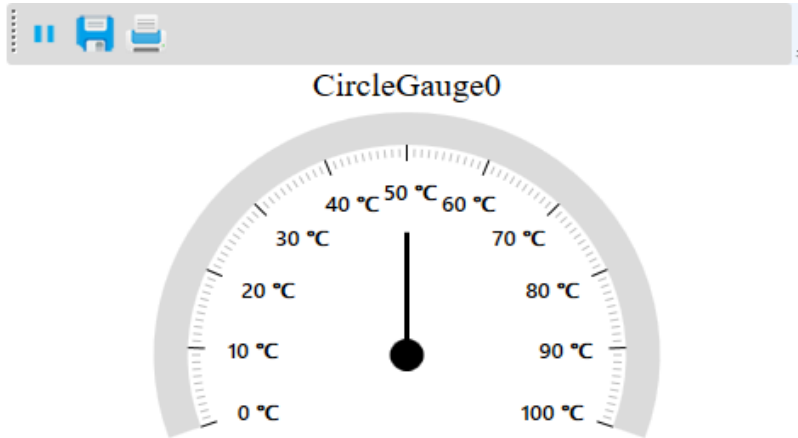
(3) Create a CircleGauge0 in the Window0 that associated Variable0



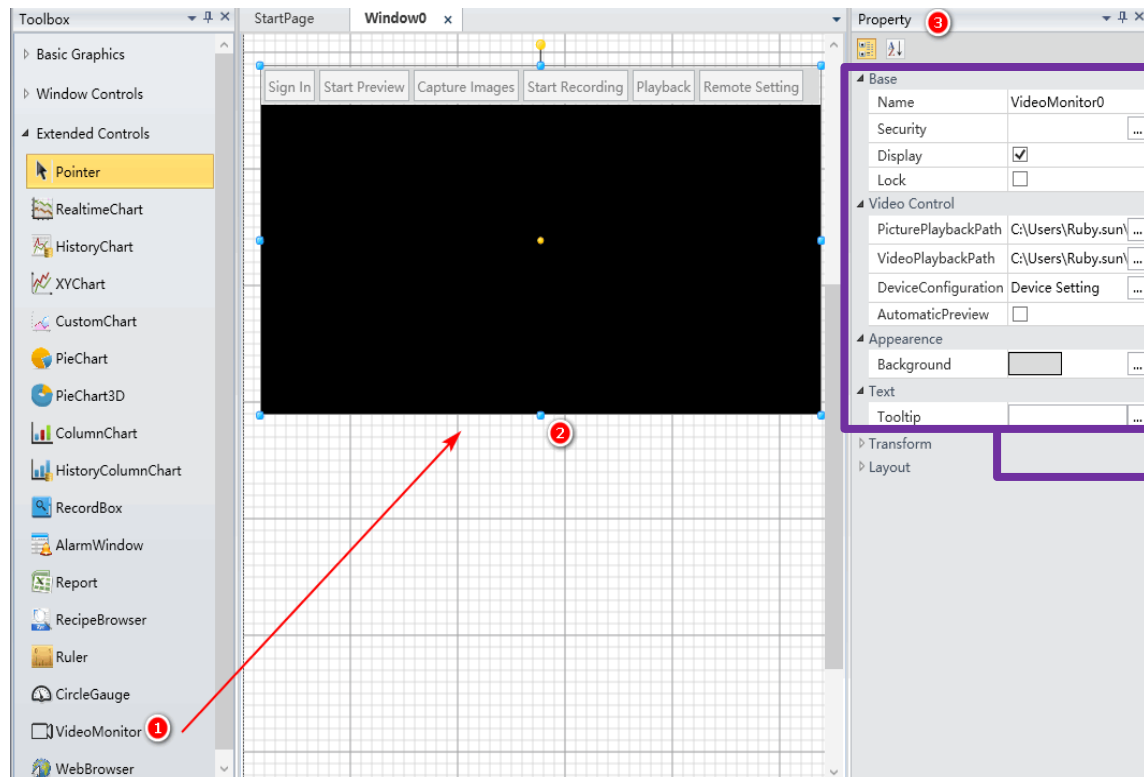
The screenshot displays the Delta Studio interface. On the left, a 'CircleGauge0' control is shown on a canvas, with a red circle '1' highlighting its top-right corner. The gauge is a semi-circular thermometer-style display with a scale from 10 °C to 90 °C. The needle points to approximately 70 °C. Below the canvas is a 'Variable Browser' section with 'Add Group' and 'Edit' buttons, and a 'Variable Dictionary' table. The table has a 'TypeFilter' set to 'Analog' and a 'NameFilter' set to an empty field. The table contains one entry, 'Variable0', which is highlighted with a red circle '3'. On the right, the 'Properties' panel for 'CircleGauge0' is open. It has two sections: 'Base' and 'Instrument'. The 'Base' section includes 'Name' (CircleGauge0), 'Security' (empty), 'Display' (checked), and 'Lock' (unchecked). The 'Instrument' section includes 'InstrumentPanel' (hex color #FFDBDBDB), 'Pointer' (hex color #FF000000), and 'VariablePath' (empty, highlighted with a red circle '2'). A red arrow points from the 'VariablePath' field to the 'Variable0' entry in the 'Variable Dictionary' table.

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

(4) Run the project



➤ Create a VideoMintor0, open its properties window



- **PicturePlaybackPath**: get or set picture playback path.
- **VideoPlaybackPath**: get or set video playback path.
- **DeviceConfiguration**: Device IP, Device port, and Username, password.
- **AutomaticPreview**: Set whether to preview the video automatically
- **Background**: Set the background color of VideoMintor



## Extend Controls-VideoMintor

➤ Record a video and play it back:

(1) The information for the camera (remote device) is as follows

Device Parameters

Device IP: 192.168.1.32

Device Port: 8000

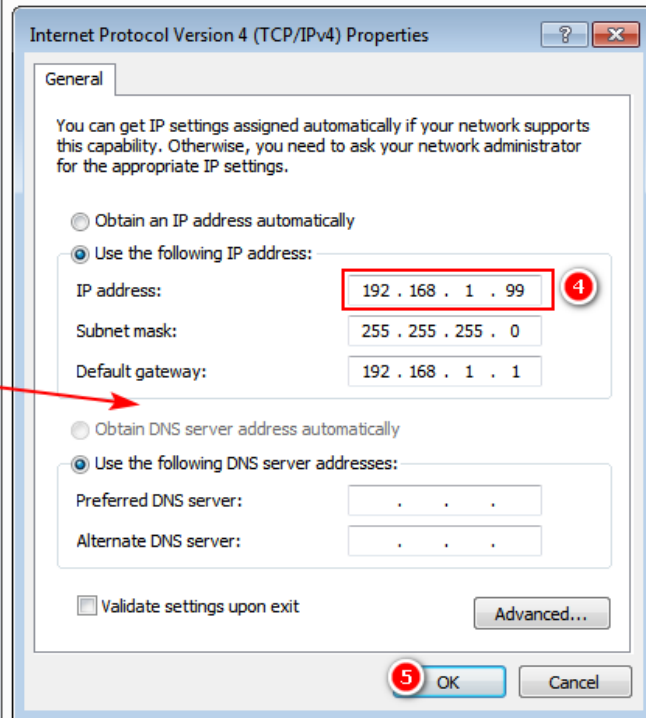
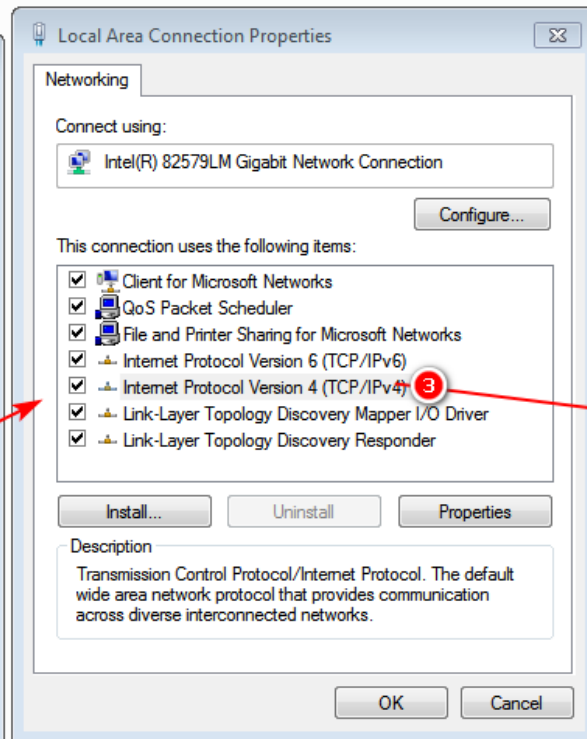
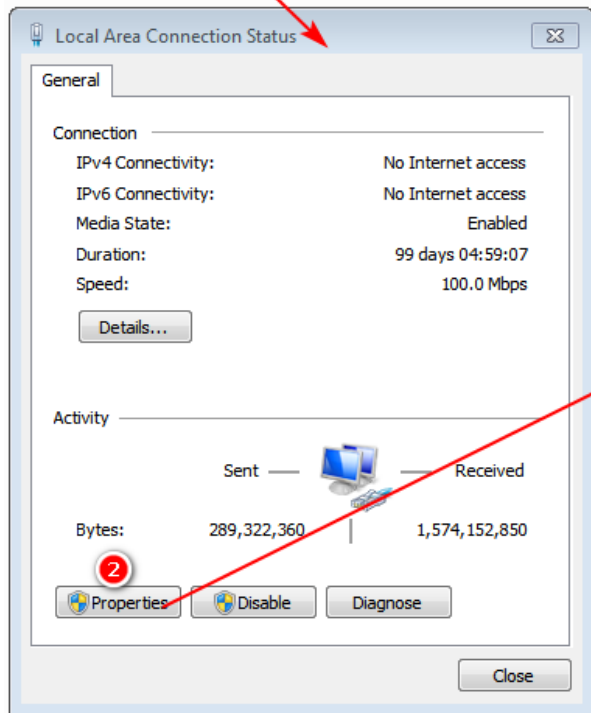
User Name: admin

User Password: ●●●●●

Logging Device

## (2) Computer connected to camera over Ethernet

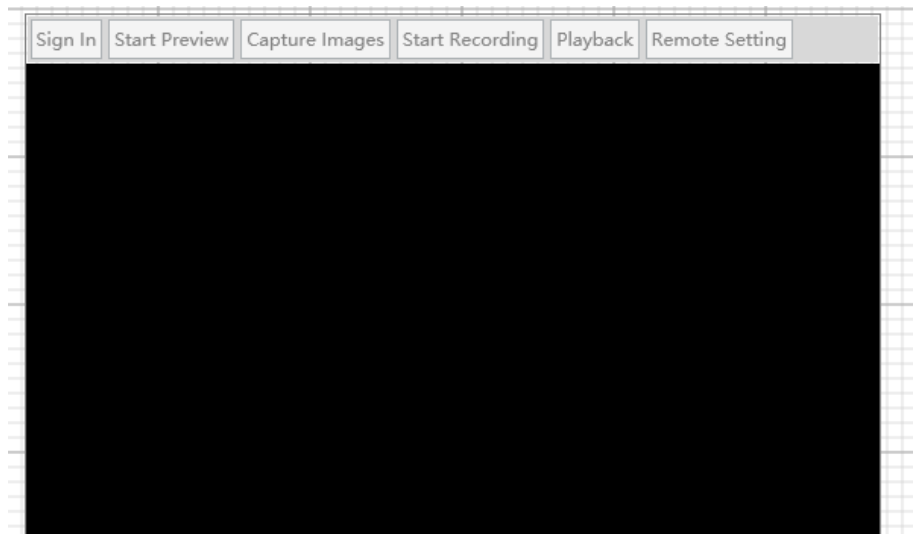
Local Area Connection  
Unidentified network  
Intel(R) 82579LM Gigabit Network...





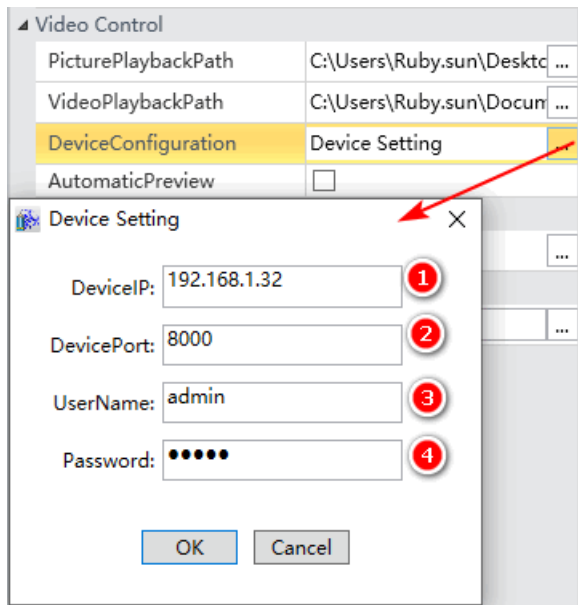
## Extend Controls-VideoMintor

(3) Create a VideoMintor0 in the Window0



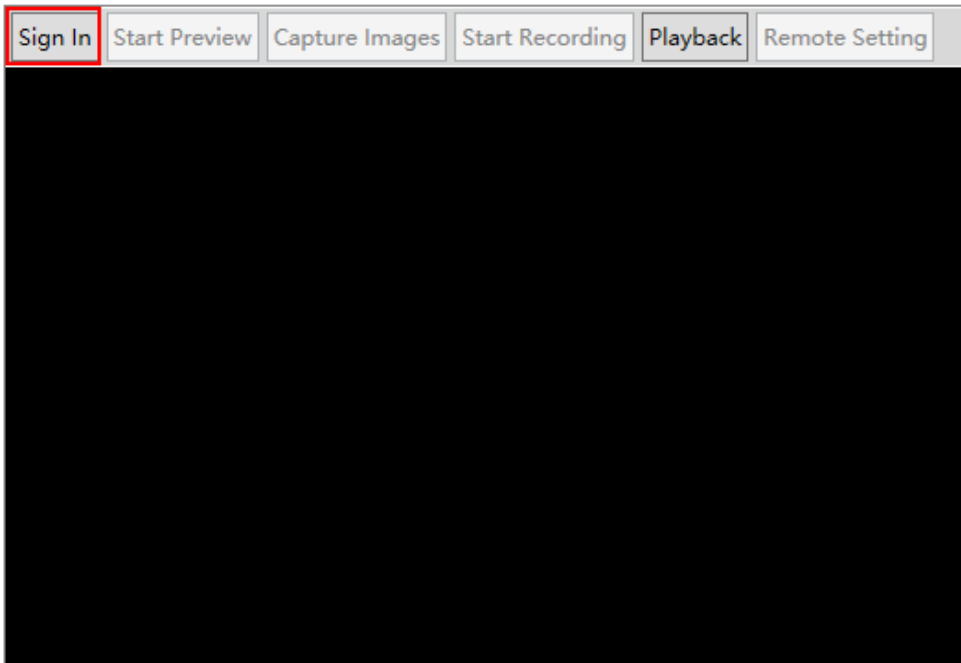


(4) Input the device parameters in the DeviceConfiguration window

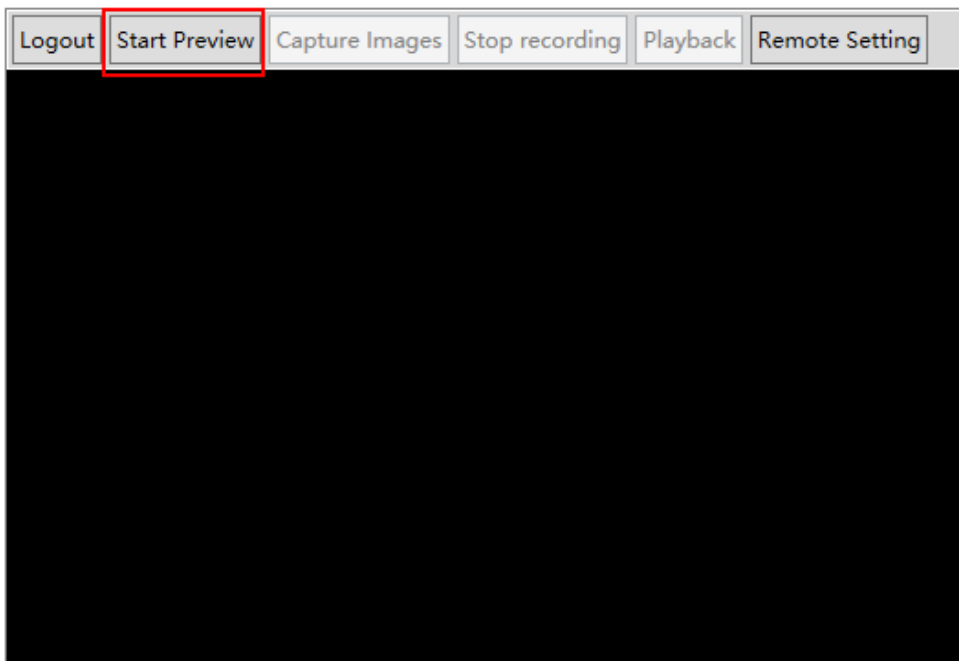


- ① Remote device IP
- ② Remote device port
- ③ Remote device user name
- ④ Remote device user password

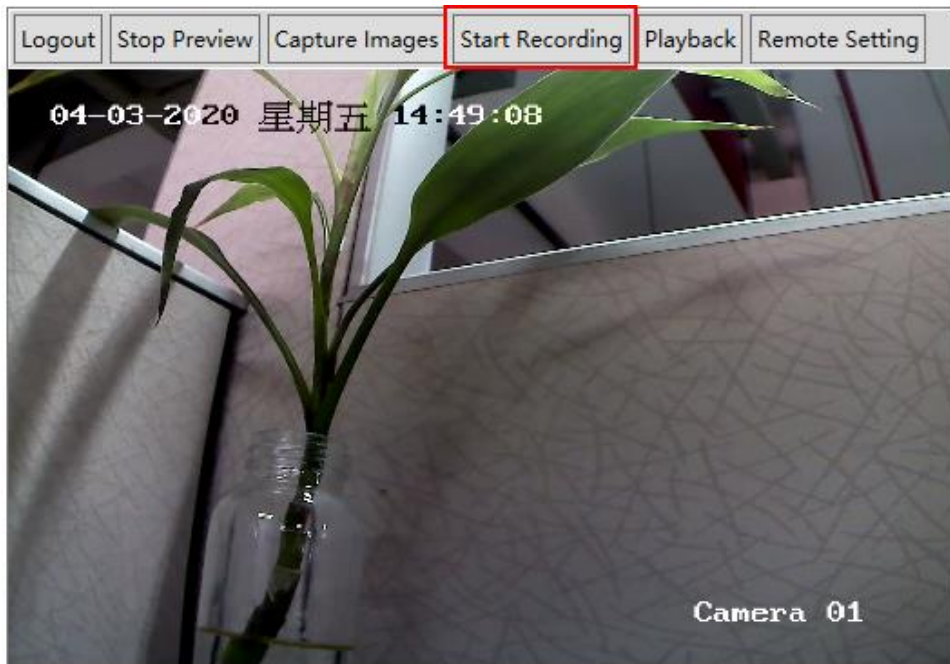
(5)Run the Project , login success



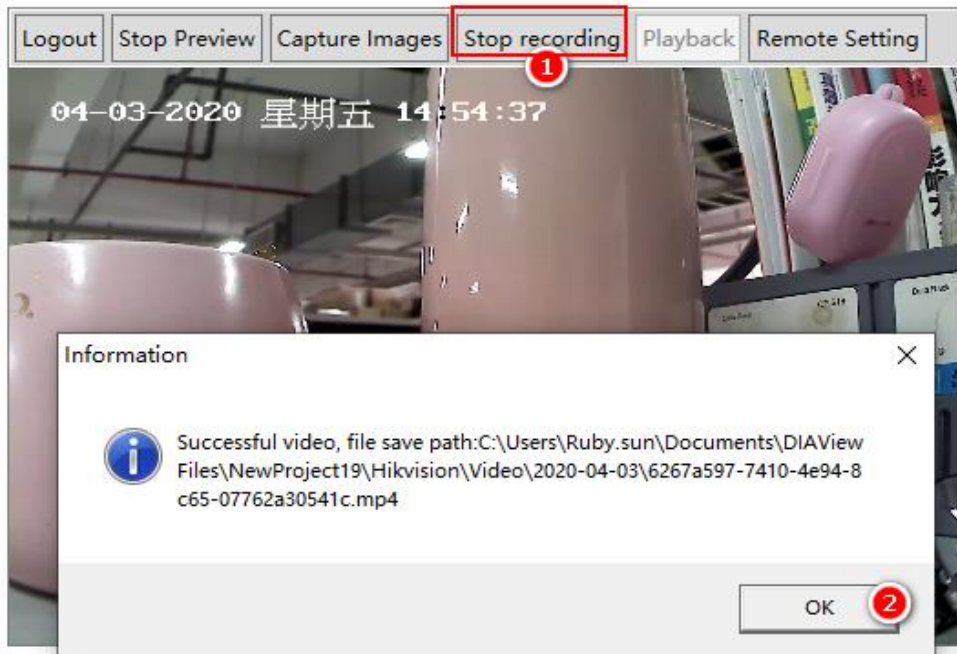
## (6)Start Preview



## (7)Start Recording



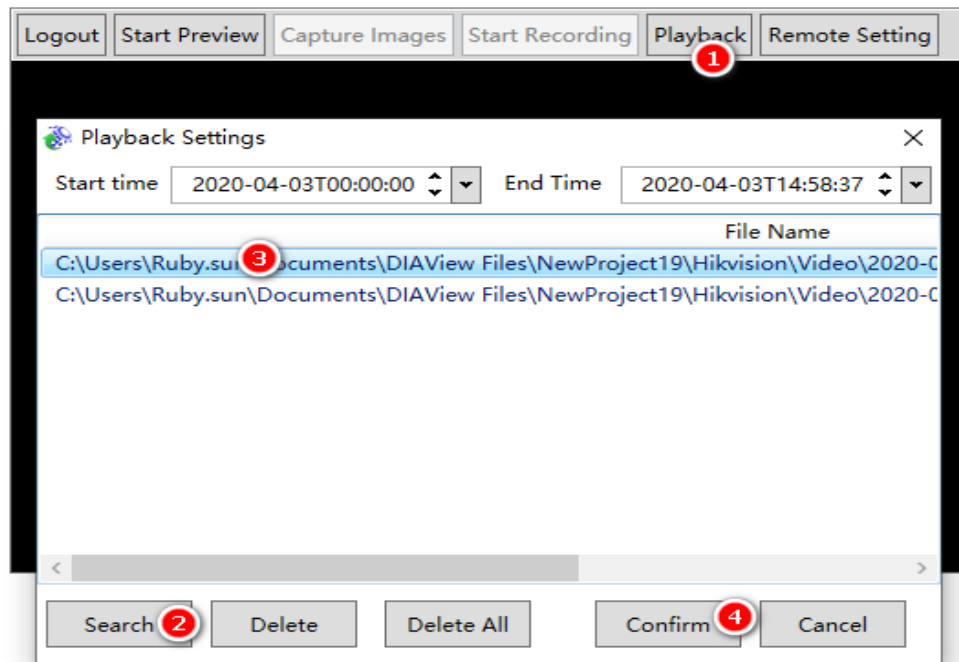
(8) Stop Recording to end record and save the video



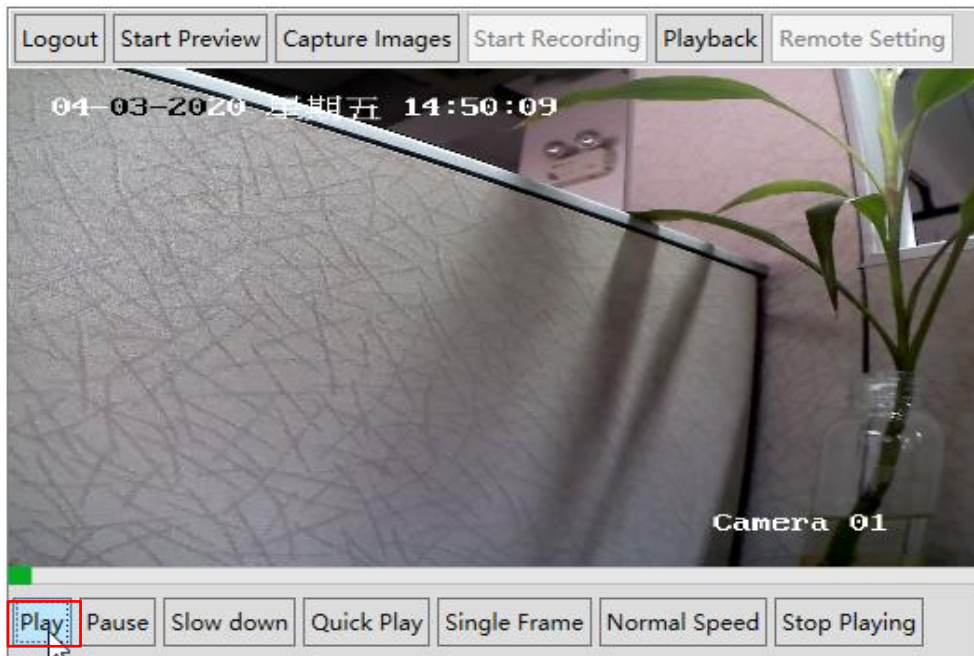
## (9)Stop Preview



## (10) Playback the video

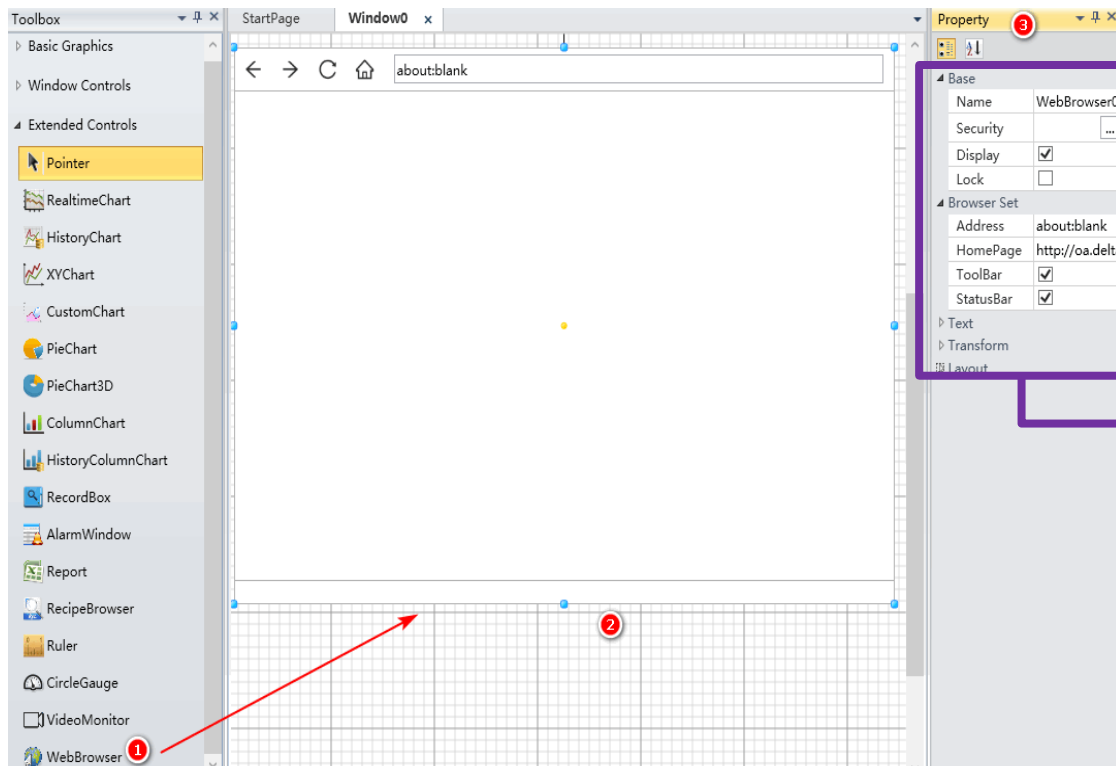


## (11)Start play





- Create a WebBrowser0, open its properties window



- **Address:** Set the current address of web browser
- **HomePage:** Set the home address of Web browser
- **ToolBar:** Set whether to display toolbar
- **StatusBar:** Set whether to display status bar

➤ The usage of Address property and Homepage property:

(1) Create a WebBrowser0 in the Window0, set its address and homepage, as shown in the figure below

Browser Set	
Address	<a href="https://translate.google.cn/">https://translate.google.cn/</a>
HomePage	<a href="https://www.baidu.com/">https://www.baidu.com/</a>

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

HomePage: <https://www.baidu.com/>

(2)Run the project, the initial page is as follows

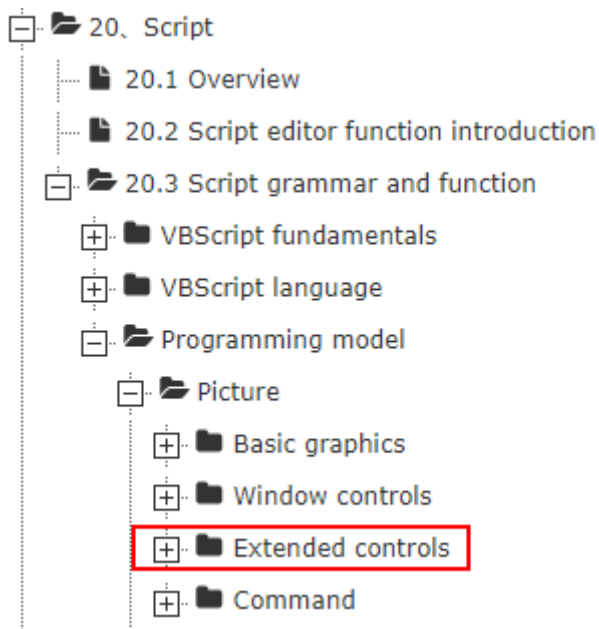


(3)Click HomePage button, and the following figure appears



# The Scripts of Extend Controls

The scripts usage of extend controls, please refer to the section “20.3 Script grammar and function” in the user manual. As shown in the figure below:



# Smarter. Greener. Together.

To learn more about Delta, please visit  
[www.deltaww.com](http://www.deltaww.com).

