

DIView

Common Script example

Wendy
2020/04



Outline

- Variable dictionary batch assignment example
- Global switching system language example
- Variable record query example
- Query history group record example
- Database Access example

Purpose

After this chapter, you will learn ...

...more about common script examples

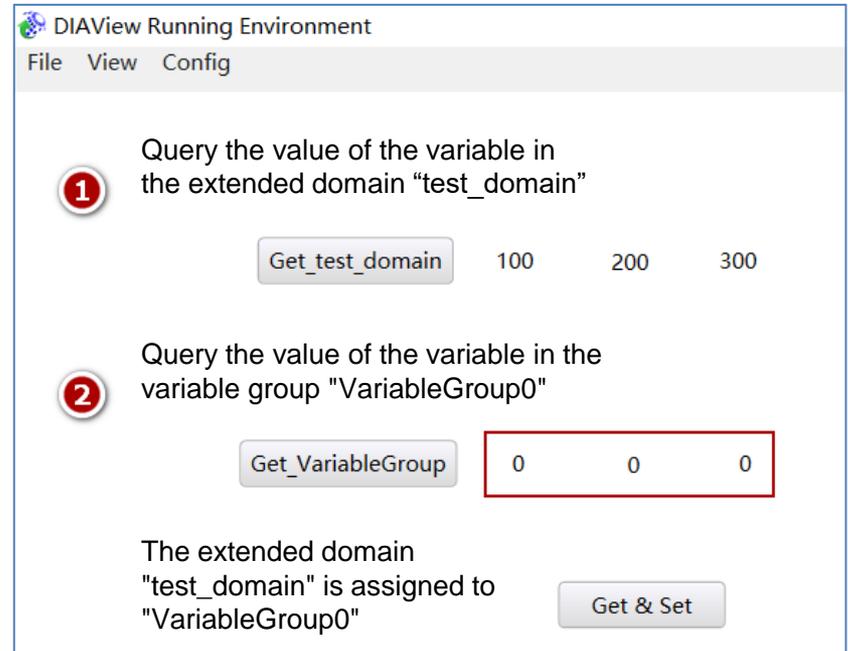
Outline

- Variable dictionary batch assignment example
- Global switching system language example
- Variable record query example
- Query history group record example
- Database Access example

➤ **Example 1:**

- **Assign the value of the variable in the extended domain test_domain to the variable in VariableGroup0**

- ① Query the values of the three variables in the extended domain test_domain as 100, 200, and 300 respectively
- ② Query the values of the three variables in VariableGroup0 are 0



The screenshot shows the DIAView Running Environment interface with a menu bar (File, View, Config). It displays two numbered steps:

1 Query the value of the variable in the extended domain "test_domain"

Get_test_domain 100 200 300

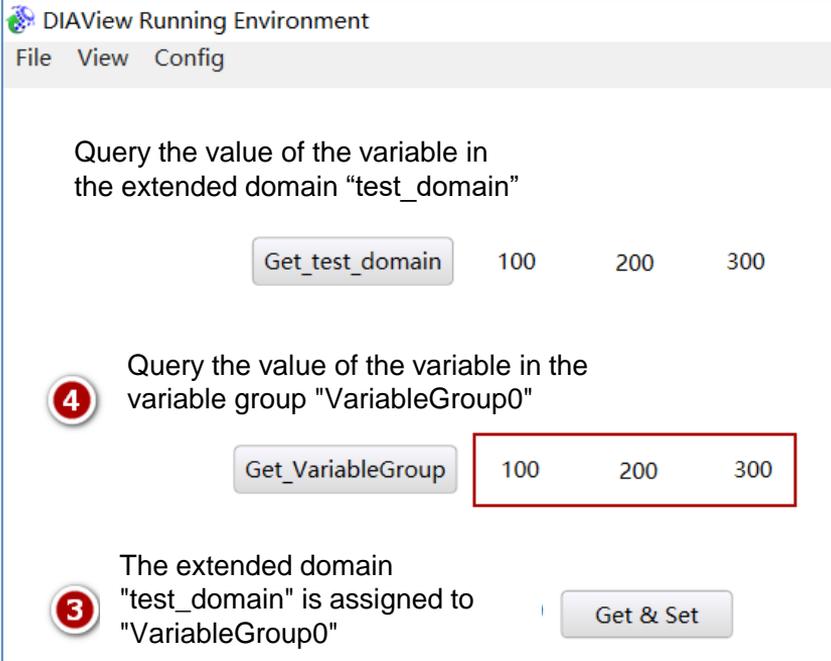
2 Query the value of the variable in the variable group "VariableGroup0"

Get_VariableGroup 0 0 0

The extended domain "test_domain" is assigned to "VariableGroup0"

Get & Set

- **Example 1:**
- **Assign the value of the variable in the extended domain test_domain to the variable in VariableGroup0**
 - ③ Execute the button event script to assign the value of the variable in the extended domain test_domain to the variable in VariableGroup0
 - ④ Query the values of the three variables in VariableGroup0 again are 100, 200, and 300, and the assignment is successful



DIAView Running Environment

File View Config

Query the value of the variable in the extended domain "test_domain"

Get_test_domain 100 200 300

4 Query the value of the variable in the variable group "VariableGroup0"

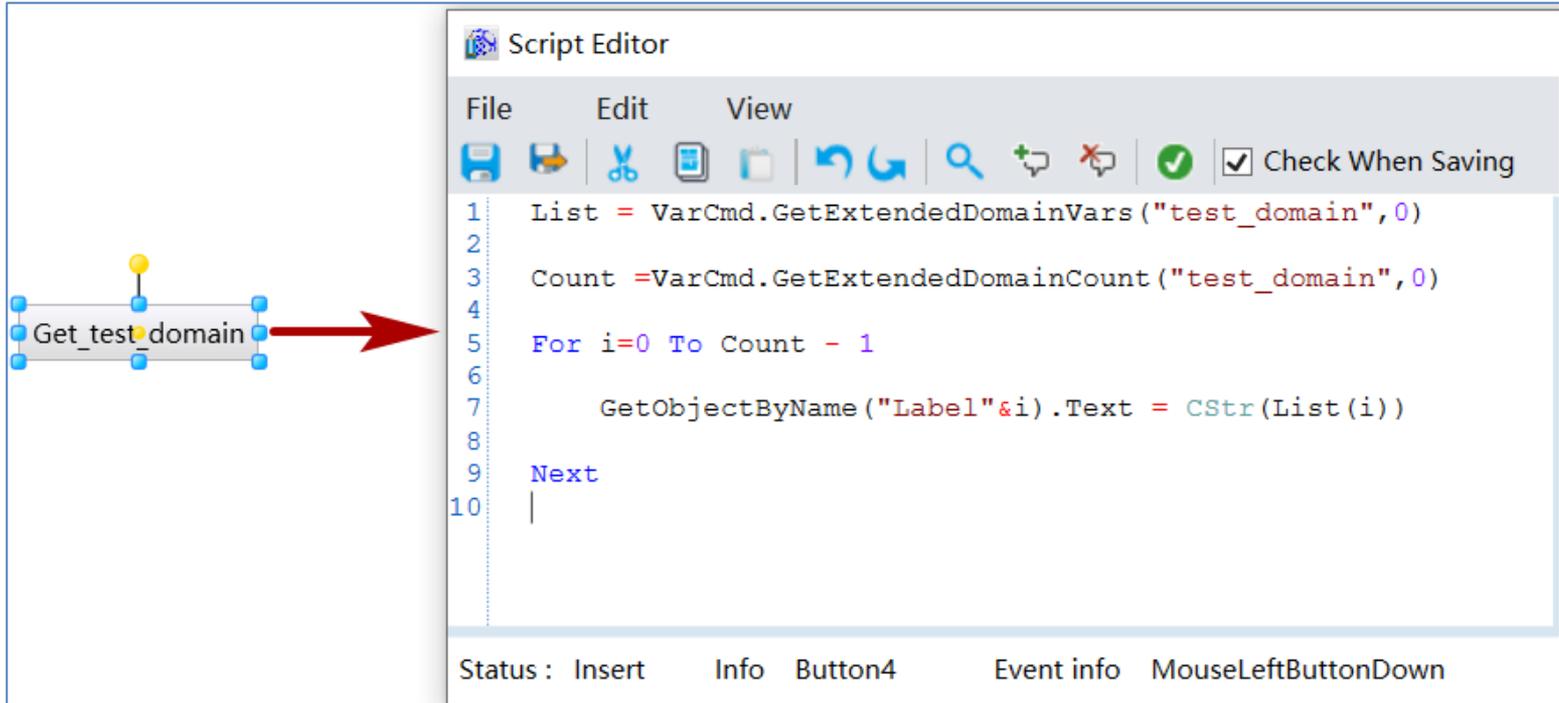
Get_VariableGroup 100 200 300

3 The extended domain "test_domain" is assigned to "VariableGroup0"

Get & Set

➤ **Example 1: Get_test_domain button script**

- Query the value of each variable in the extended domain test_domain and display it in the running window with a label



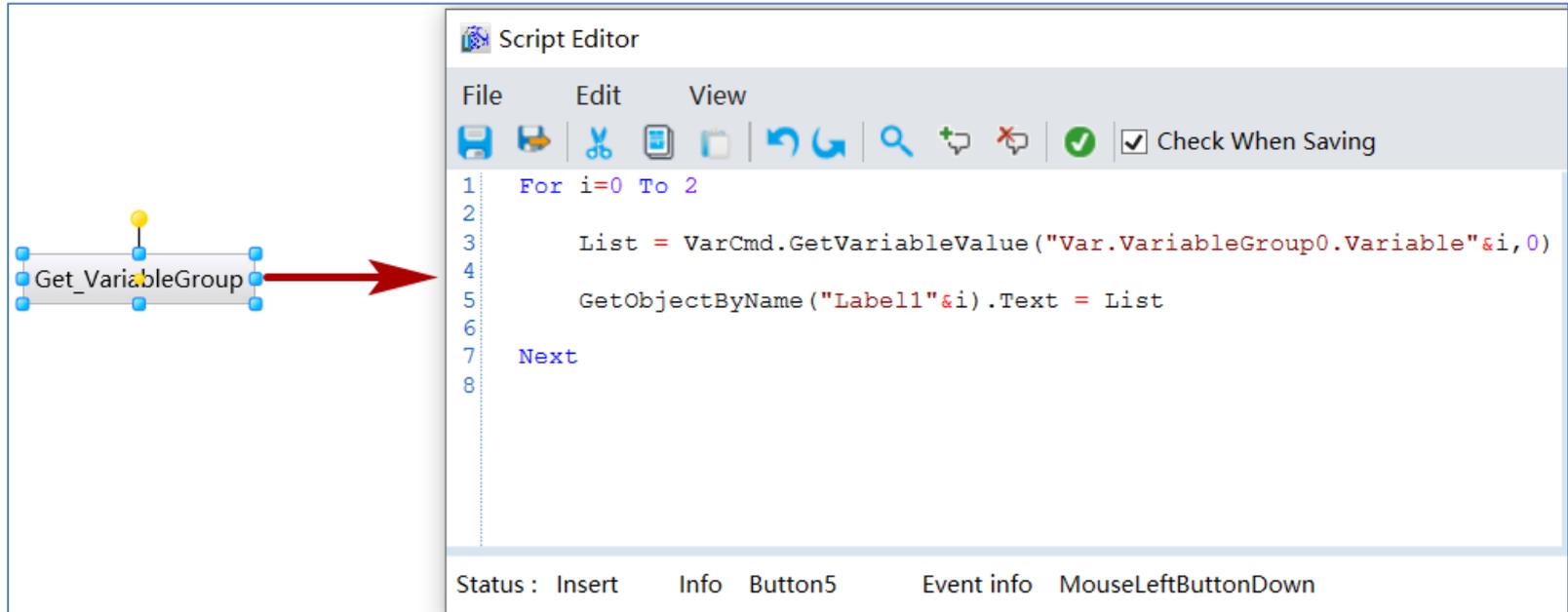
The screenshot shows a 'Script Editor' window with a menu bar (File, Edit, View) and a toolbar. The script content is as follows:

```
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 |
```

At the bottom of the editor, the status bar displays: Status : Insert Info Button4 Event info MouseLeftButtonDown

➤ **Example 1: Get_variableGroup0 button script**

- Query the value of each variable in VariableGroup0 and display it in the running window with a label

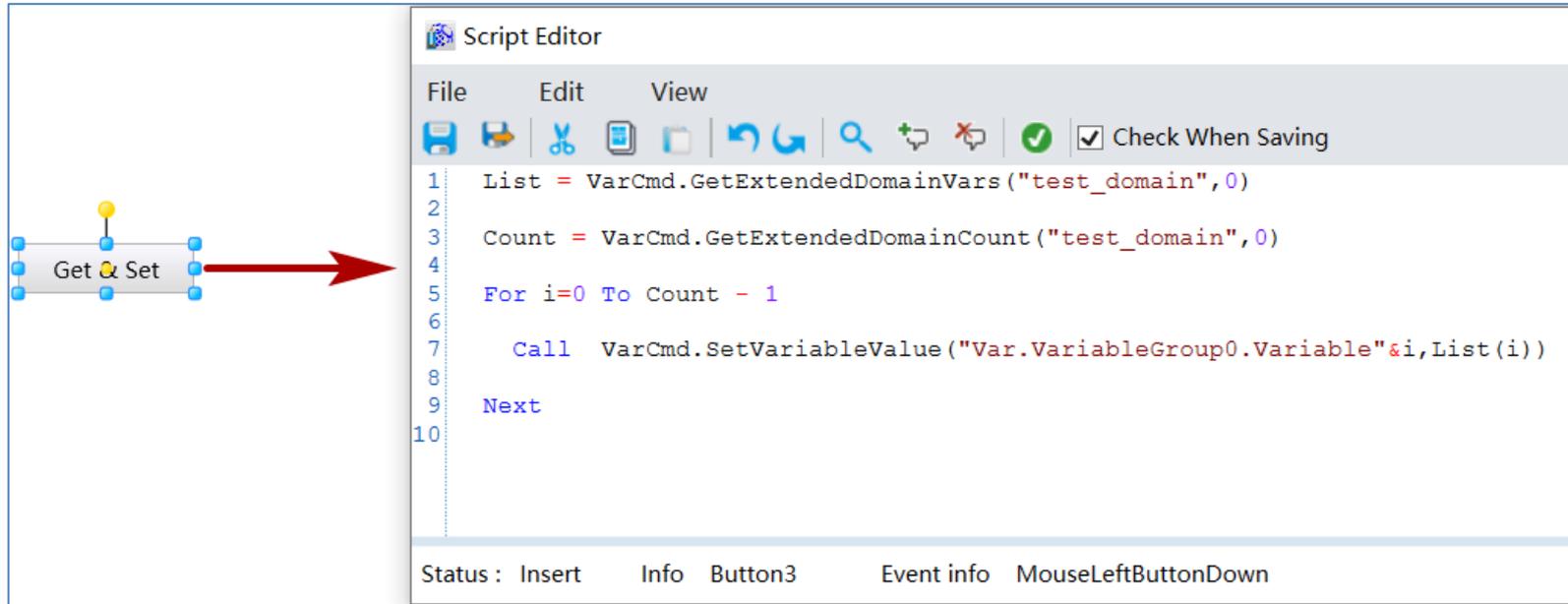


The screenshot shows a visual programming interface on the left with a button labeled "Get_variableGroup0". A red arrow points from this button to a "Script Editor" window on the right. The script editor contains the following code:

```
Script Editor
File Edit View
[Icons] [Check When Saving]
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
Status : Insert Info Button5 Event info MouseLeftButtonDown
```

➤ Example 1: Get & Set button script

- Assign the value of each variable in the extended domain test_domain to the variable group VariableGroup0



The screenshot shows a 'Script Editor' window with a menu bar (File, Edit, View) and a toolbar. The script content is as follows:

```
1 List = VarCmd.GetExtendedDomainVars("test_domain",0)
2
3 Count = VarCmd.GetExtendedDomainCount("test_domain",0)
4
5 For i=0 To Count - 1
6     Call VarCmd.SetVariableValue("Var.VariableGroup0.Variable"&i,List(i))
7
8 Next
9
10
```

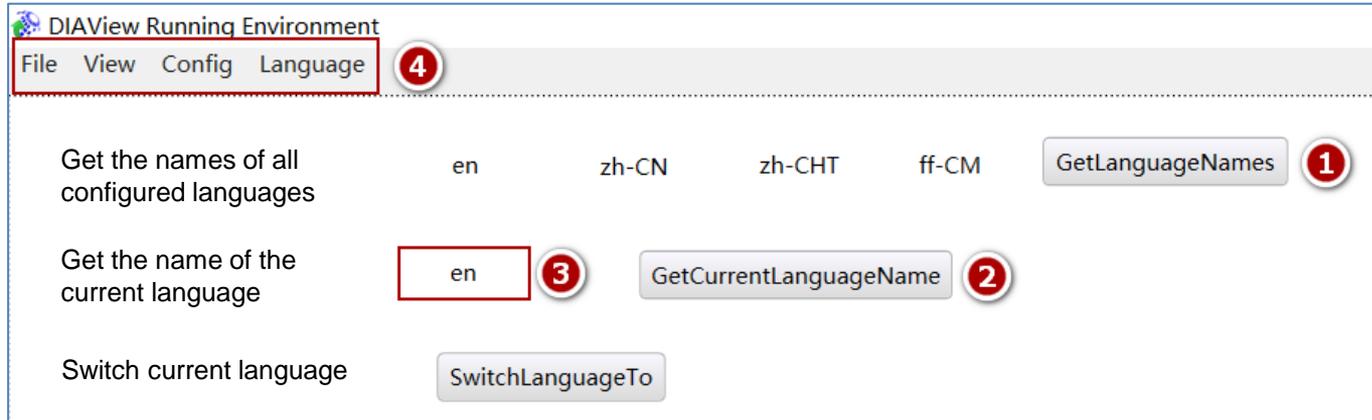
At the bottom of the editor, the status bar shows: Status: Insert Info Button3 Event info MouseLeftButtonDown

※For detailed descriptions of variable methods and properties, please refer to the user manual "CH20.3 Script Syntax and Functions-Programming Model-Variables"

- Variable dictionary batch assignment example
- **Global switching system language example**
- Variable record query example
- Query history group record example
- Database Access example

➤ Example 2:

- **Query the names of all configured languages and switch the current language**
 - ① Get the names of all configured languages as **en, zh-CN, zh-CHT, ff-CM**
 - ② Get the name of the current language
 - ③ The label content shows that the name of the current language is **en**
 - ④ Make sure the system language is **en**

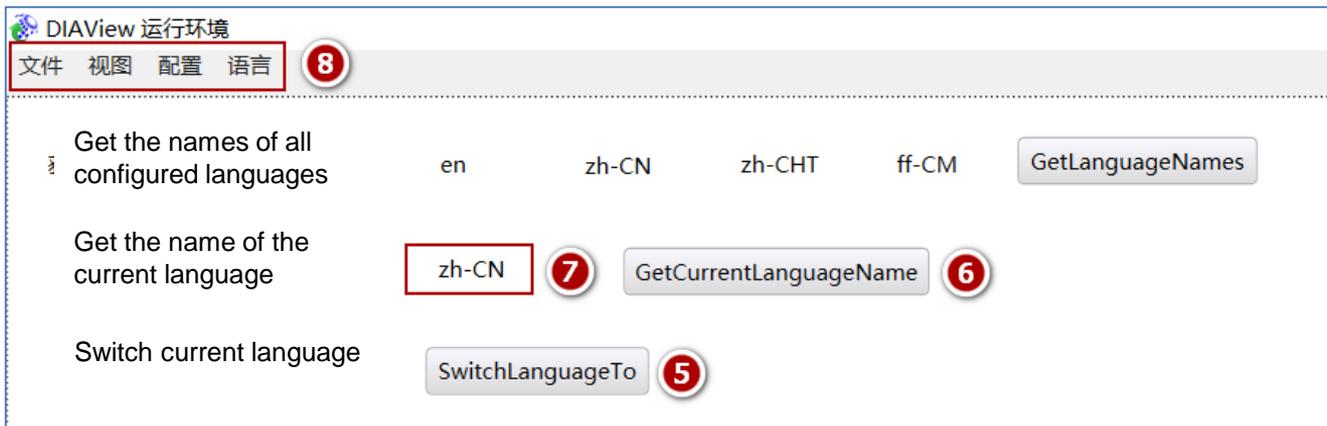


The screenshot shows the DIAView Running Environment interface with the following elements:

- Menu Bar:** File View Config Language (Step 4: Language menu is highlighted with a red box and a circled 4).
- Language Selection:** A row of language labels: en, zh-CN, zh-CHT, ff-CM. A button labeled "GetLanguageNames" (Step 1: circled 1) is positioned to the right.
- Current Language:** A text input field containing "en" (Step 3: circled 3) is highlighted with a red box. A button labeled "GetCurrentLanguageName" (Step 2: circled 2) is to its right.
- Switch Language:** A button labeled "SwitchLanguageTo" is located below the input field.

➤ Example 2:

- Query the names of all configured languages and switch the current language
 - ⑤ Switch the current language
 - ⑥ Get the name of the current language again
 - ⑦ The label content shows that the name of the current language is zh-CN
 - ⑧ Make sure the system language is zh-CN

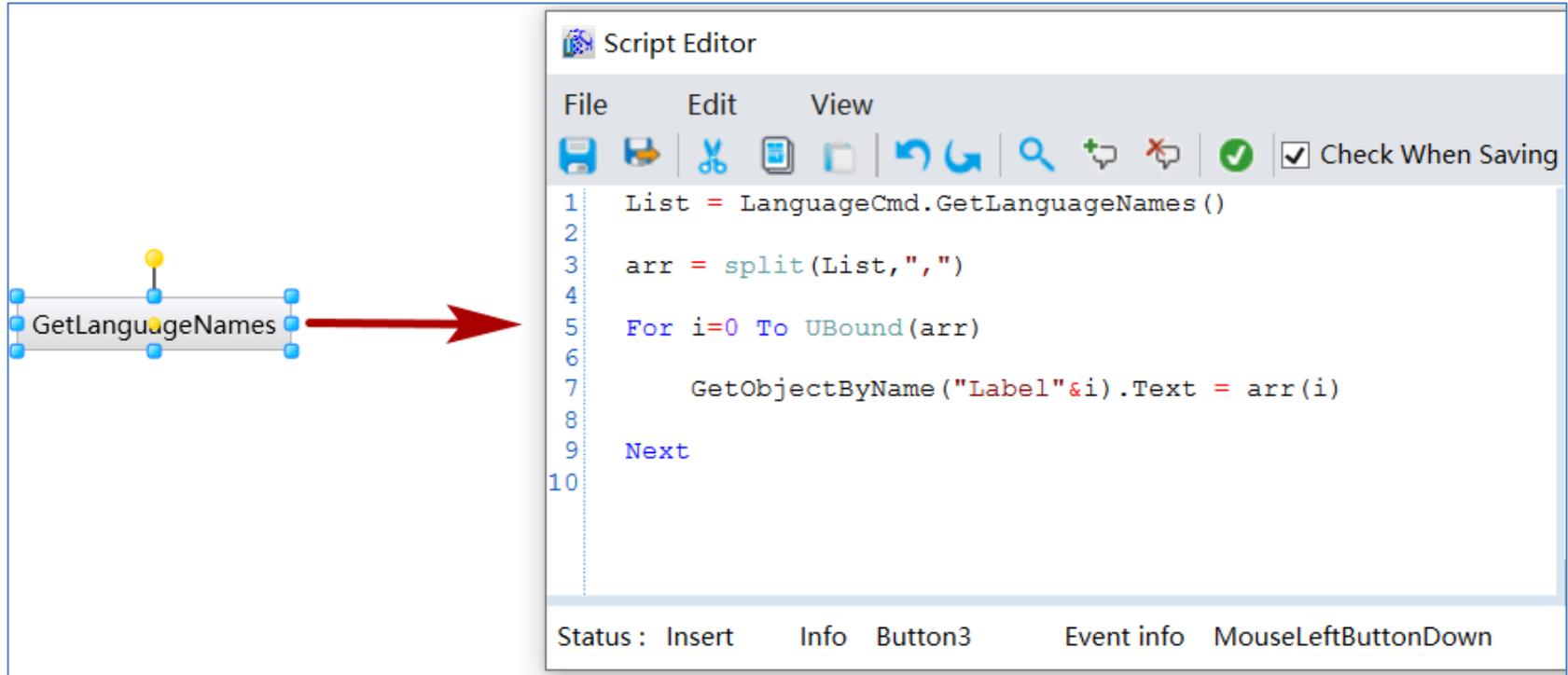


The screenshot shows the DIAView 运行环境 interface. The menu bar includes 文件, 视图, 配置, and 语言. The 语言 menu is highlighted with a red box and a circled 8. Below the menu, there are three rows of controls:

- Row 1: "Get the names of all configured languages" with radio buttons for en, zh-CN, zh-CHT, and ff-CM, and a "GetLanguageNames" button.
- Row 2: "Get the name of the current language" with a text box containing "zh-CN" (highlighted with a red box and a circled 7) and a "GetCurrentLanguageName" button (circled 6).
- Row 3: "Switch current language" with a "SwitchLanguageTo" button (circled 5).

➤ Example 2: GetLanguageNames button script

- Query the names of all configured languages and display them in the running window with tags.



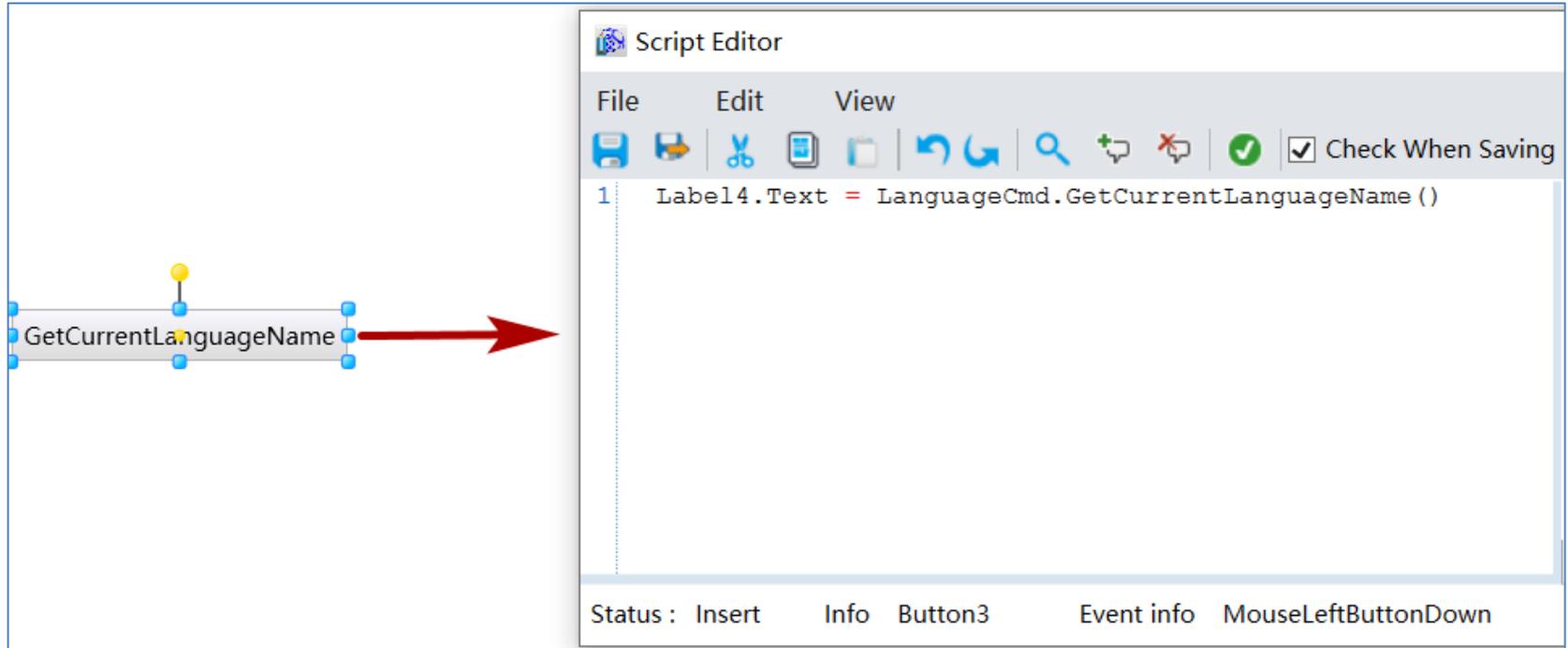
The image shows a visual programming interface on the left with a button labeled "GetLanguageNames". A red arrow points from this button to a "Script Editor" window on the right. The script editor contains the following code:

```
Script Editor
File Edit View
[Icons] [Check When Saving]
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
```

Status: Insert Info Button3 Event info MouseLeftButtonDown

➤ Example 2:GetCurrentLanguageName button script

- Query the name of the current language and display it in the running window with a label.



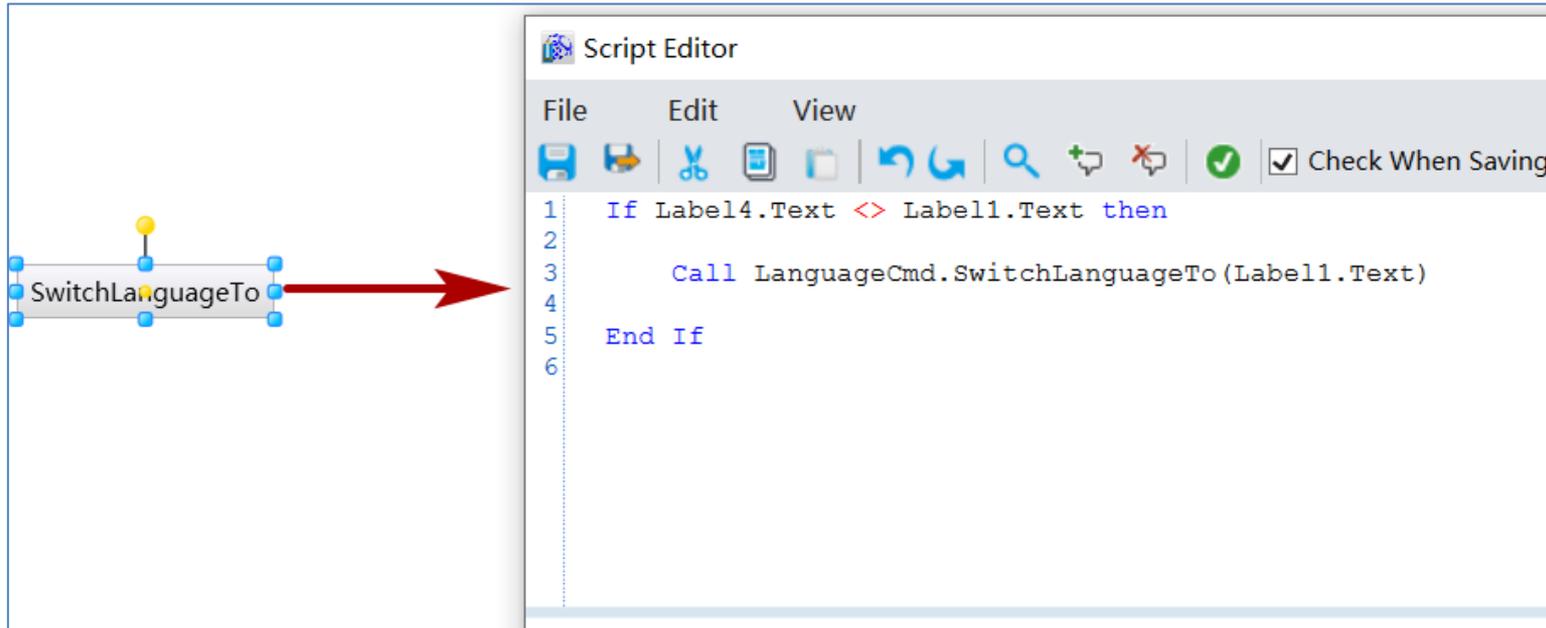
The image shows a visual programming interface on the left and a script editor on the right. A red arrow points from a button click event labeled 'GetCurrentLanguageName' to the script editor. The script editor contains the following code:

```
1 Label4.Text = LanguageCmd.GetCurrentLanguageName ()
```

The script editor window has a menu bar with 'File', 'Edit', and 'View'. Below the menu bar is a toolbar with icons for Save, Undo, Redo, Find, and other standard editing functions. A 'Check When Saving' checkbox is also present. The status bar at the bottom of the script editor shows 'Status : Insert Info Button3 Event info MouseLeftButtonDown'.

➤ Example 2: SwitchLanguageTo button script

- Switch current language.



The image shows a visual programming environment with a button labeled 'SwitchLanguageTo' on the left. A red arrow points from the button to a 'Script Editor' window on the right. The script editor contains the following code:

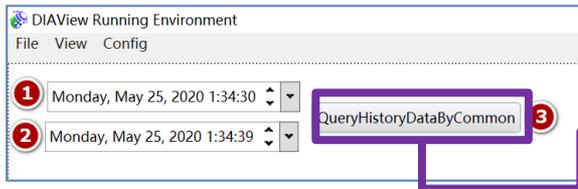
```
1  If Label14.Text <> Label11.Text then
2
3      Call LanguageCmd.SwitchLanguageTo (Label11.Text)
4
5  End If
6
```

※Note: For a detailed description of globalization methods and attributes, please refer to the user manual "CH 20.3 Script Syntax and Functions-Programming Model-Globalization"

- Variable dictionary batch assignment example
- Global switching system language example
- **Variable record query example**
- Query history group record example
- Database Access example

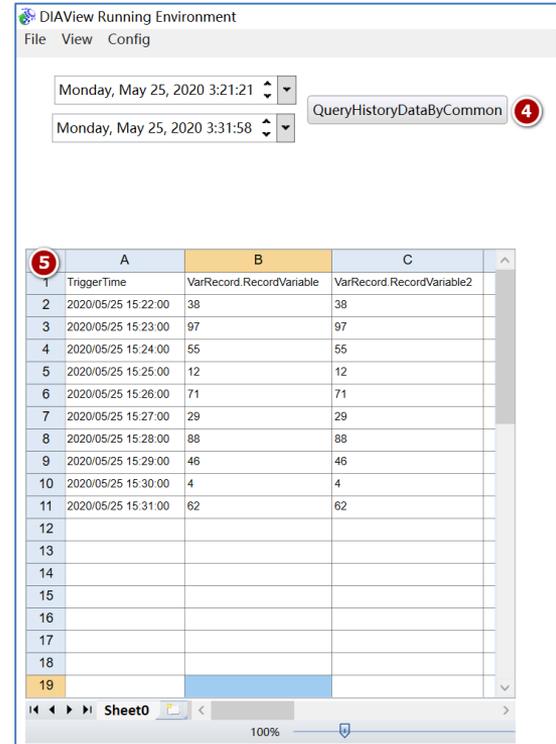
➤ **Example 3: Variable record query script**
 The variable record is read through the script and displayed in the report.

- QueryHistoryDataByCommon Button(sheetIndex, conditions, types): variable record query
 1. sheetIndex: worksheet index
 2. conditions: Variable record collection, separated by commas
 3. types: TriggeringTime, Value, can also be replaced with 0 and 1 respectively



```

DateTimePicker0.Value = Sys.StartTime
DateTimePicker1.Value = Sys.Now
Call Report0.SetWorkSheetStartTime(0,DateTimePicker0.ValueTime)
Call Report0.SetWorkSheetEndTime(0,DateTimePicker1.ValueTime)
Call Report0.SetVarRecordRuleName(0,"m1")
Call Report0.QueryHistoryDataByCommon(0,"VarRecord.RecordVariable,VarRecord.RecordVariable,VarRecord.RecordVariable2", "0,1,1")
  
```

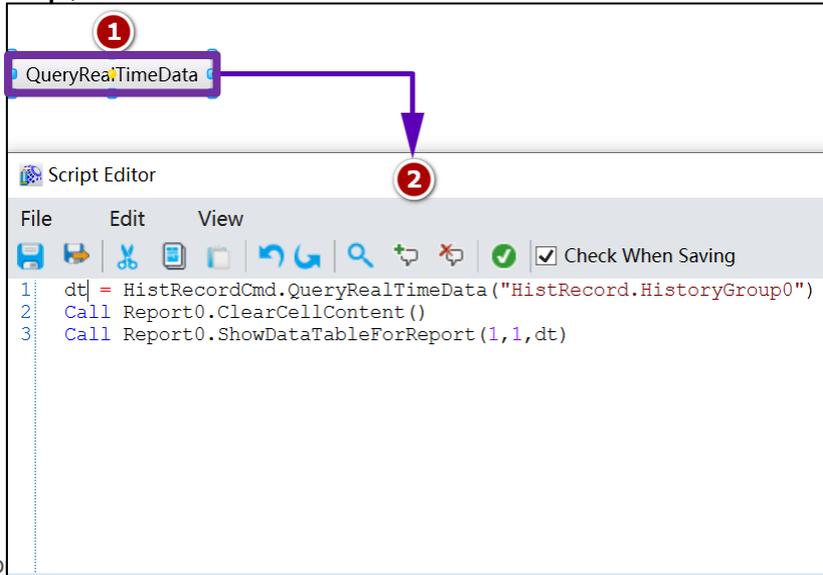


5	A	B	C
1	TriggerTime	VarRecord.RecordVariable	VarRecord.RecordVariable2
2	2020/05/25 15:22:00	38	38
3	2020/05/25 15:23:00	97	97
4	2020/05/25 15:24:00	55	55
5	2020/05/25 15:25:00	12	12
6	2020/05/25 15:26:00	71	71
7	2020/05/25 15:27:00	29	29
8	2020/05/25 15:28:00	88	88
9	2020/05/25 15:29:00	46	46
10	2020/05/25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

- Variable dictionary batch assignment example
- Global switching system language example
- Variable record query example
- **Query history group record example**
- Database Access example

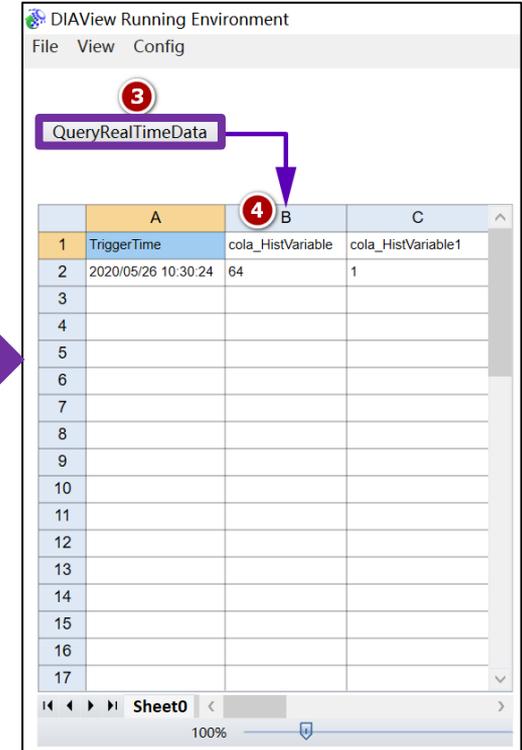
➤ Example 4: Query history group record script

- Read the real-time data of the historical record group through the script and display it with the report.
- QueryRealTimeData Button: query real-time data of historical record group;



The screenshot shows the Script Editor interface. At the top, a button labeled "QueryRealTimeData" is highlighted with a red circle labeled "1". A purple arrow points from this button to a red circle labeled "2" in the script editor area. The script editor contains the following code:

```
1| dt| = HistRecordCmd.QueryRealTimeData ("HistRecord.HistoryGroup0")
2| Call Report0.ClearCellContent ()
3| Call Report0.ShowDataTableForReport (1,1,dt)
```

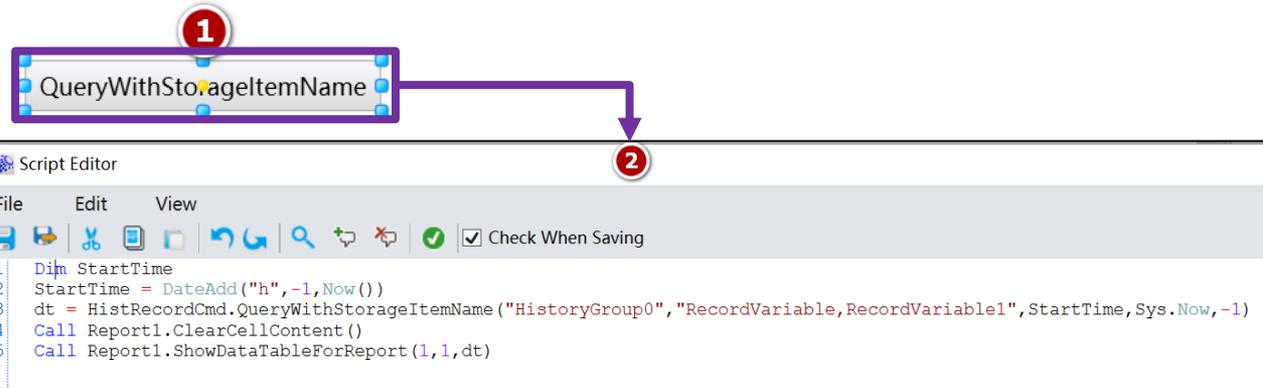


The screenshot shows the DIAView Running Environment. At the top, a button labeled "QueryRealTimeData" is highlighted with a red circle labeled "3". A purple arrow points from this button to a red circle labeled "4" in the report output table. The report output table is as follows:

	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			

- Variable dictionary batch assignment example
- Global switching system language example
- Variable record query example
- Query history group record example
- **Database Access example**

- **Example 5: Query history group record script**
 - Read the historical record group data through the script and display it with the report.
 - QueryWithStorageItemName Button: Query historical record group data, the example is to query the data of the last hour.
 - Use the **DateAdd** function to set the query time range to the past hour;

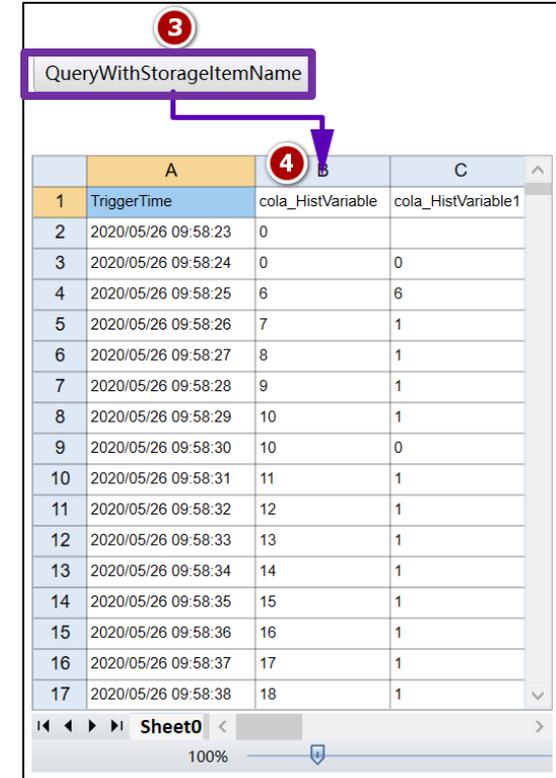


1 QueryWithStorageItemName

2 Script Editor

File Edit View

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

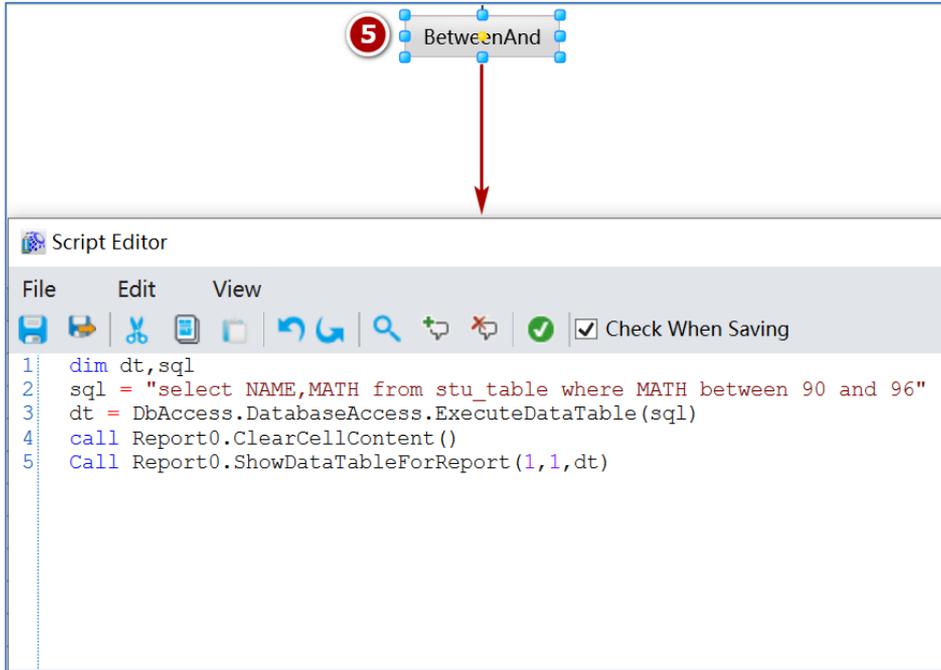
4 B

	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%

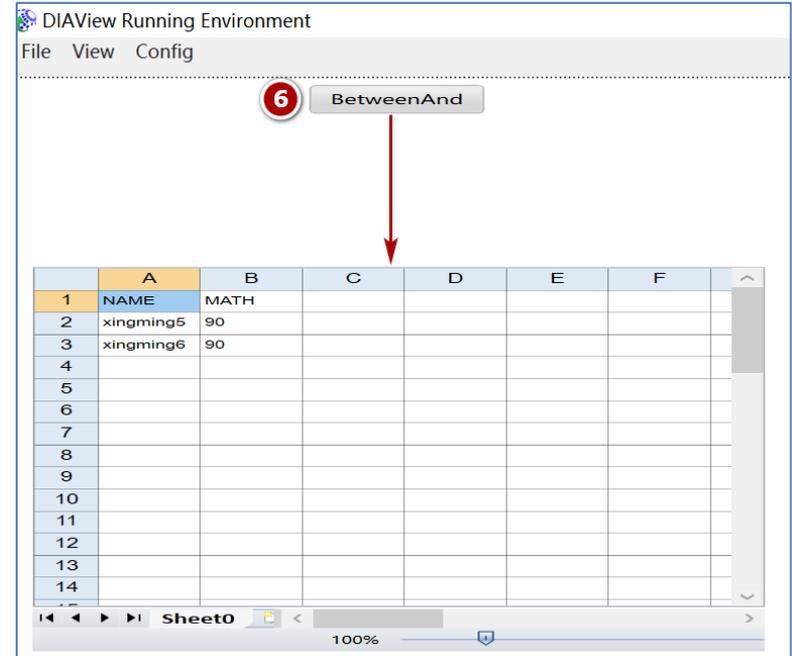
- Variable dictionary batch assignment example
- Global switching system language example
- Variable record query example
- Query history group record example
- **Database Access example**

- ⑤ Programming button event script, query table **stu table** qualified data;
- ⑥ Execute the button event script to display the queried data in the running window;



5 BetweenAnd

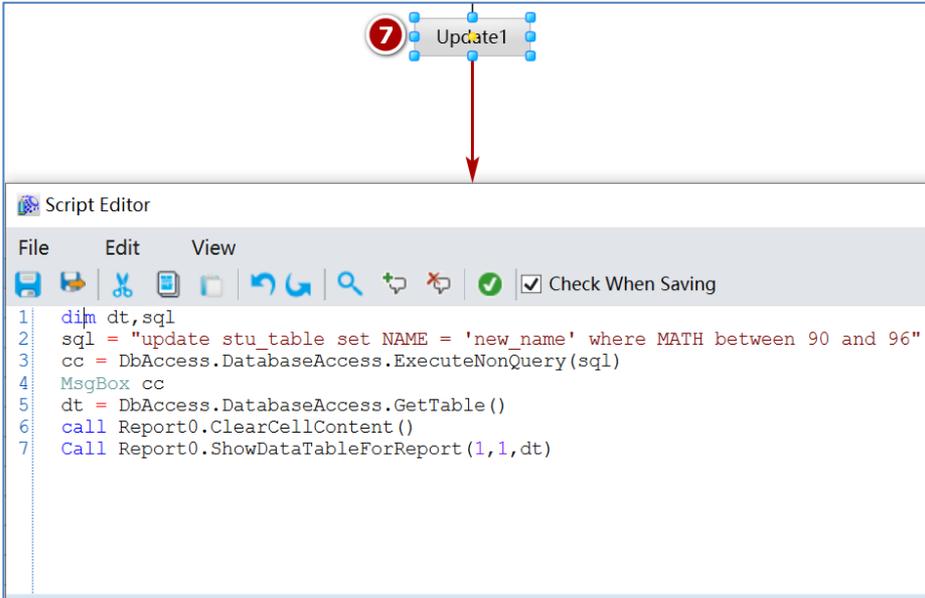
```
1 dim dt,sql
2 sql = "select NAME,MATH from stu_table where MATH between 90 and 96"
3 dt = DbAccess.DatabaseAccess.ExecuteDataTable (sql)
4 call Report0.ClearCellContent ()
5 Call Report0.ShowDataTableForReport (1,1,dt)
```



6 BetweenAnd

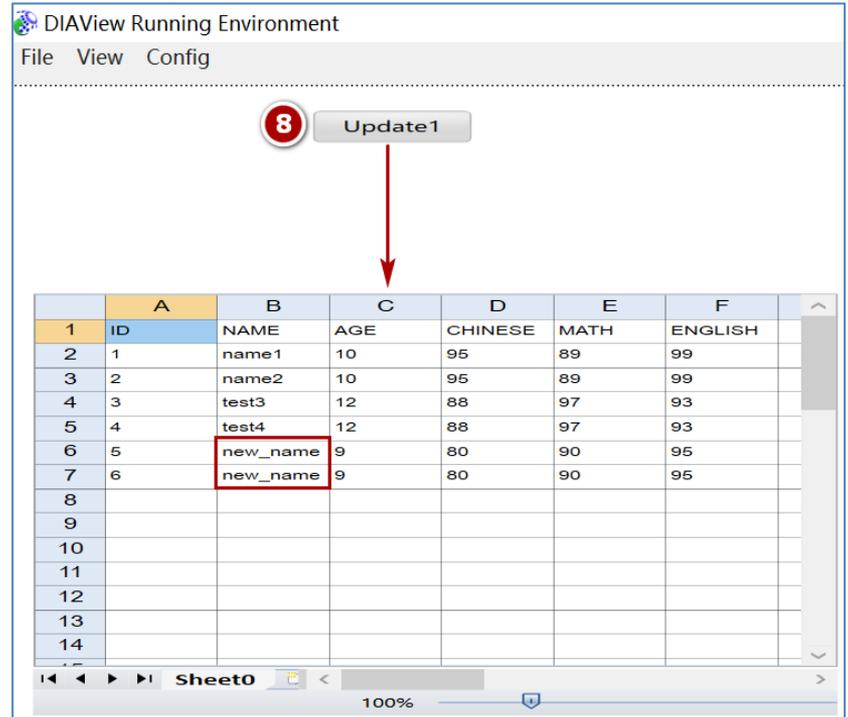
	A	B	C	D	E	F
1	NAME	MATH				
2	xingming5	90				
3	xingming6	90				
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						

- ⑦ Programming a button event script to update the qualified data in the table **stu_table**;
- ⑧ Execute the button event script and display the executed table **stu_table** in the running window;



The screenshot shows a button labeled "Update1" with a red circle containing the number 7. A red arrow points from the button to the Script Editor window below it. The Script Editor contains the following code:

```
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)
```

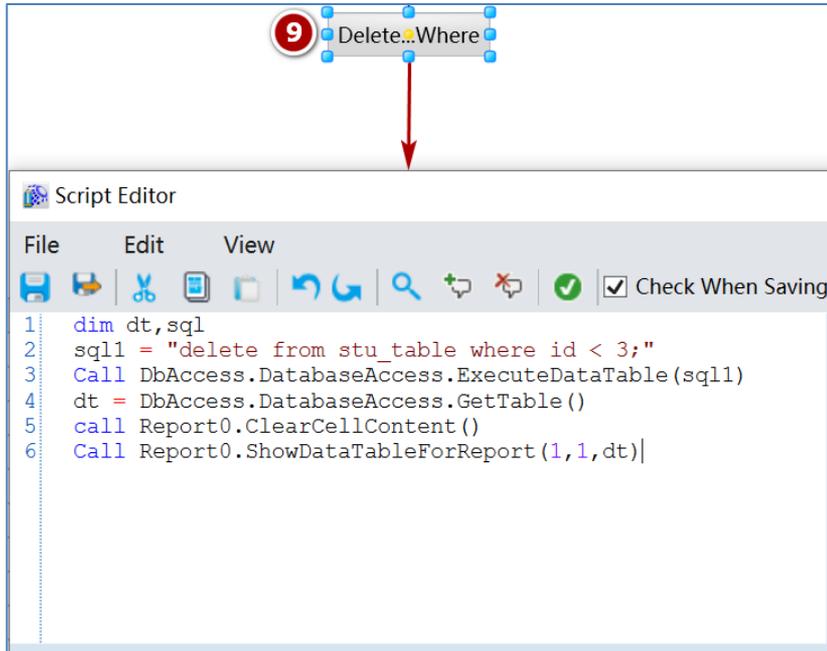


The screenshot shows the DIAView Running Environment window with a menu bar (File, View, Config) and a button labeled "Update1" with a red circle containing the number 8. A red arrow points from the button to a data table below it. The table has columns A through F and rows 1 through 14. The data is as follows:

	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							

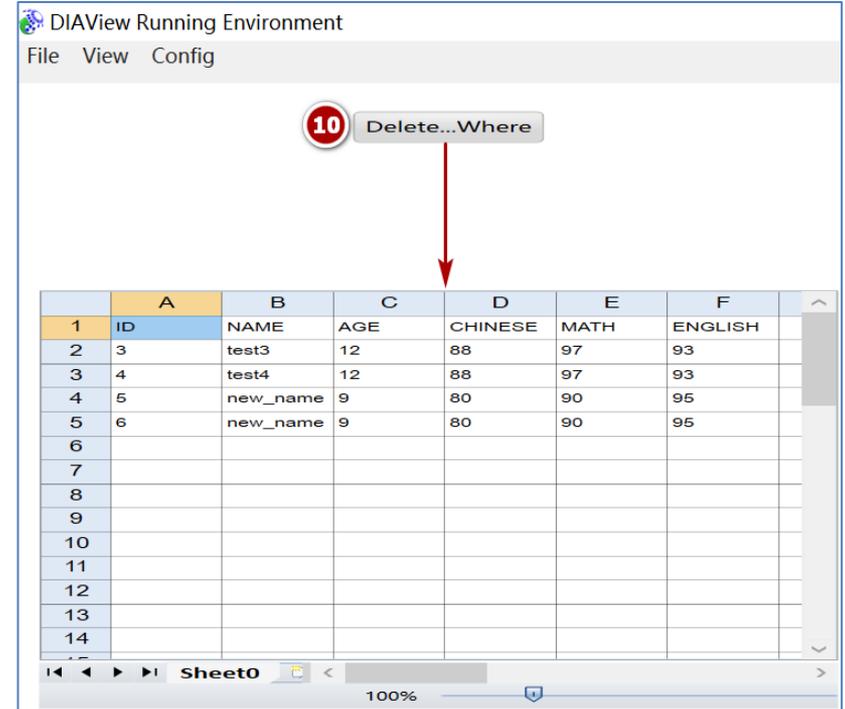
The table is displayed in a window titled "Sheet0" with a 100% zoom level.

- ⑨ Write a button event script to delete the qualified data in the table **stu table**;
- ⑩ Execute the button event script and display the executed table **stu table** in the running window;



The screenshot shows a 'Delete...Where' button with a red circle containing the number 9. A red arrow points from the button to the Script Editor below. The Script Editor has a menu bar (File, Edit, View) and a toolbar with icons for save, undo, redo, search, and other functions. A checkbox labeled 'Check When Saving' is checked. The script content is as follows:

```
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)
```



The screenshot shows the 'DIAView Running Environment' window with a menu bar (File, View, Config) and a 'Delete...Where' button with a red circle containing the number 10. A red arrow points from the button to a data table below. The table has columns A through F and rows 1 through 14. The data is as follows:

	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							

The table is displayed in a grid format with a status bar at the bottom showing 'Sheet0' and '100%' zoom.

Smarter. Greener. Together.

To learn more about Delta, please visit www.deltaww.com.

