

# DIView

Basic programming

朱明杰

2020/05/15



# Outline

- Script editor overview
- Basic syntax usage

# Purpose

After this chapter, you will learn...

- ... The use of VBScript basic syntax

# Outline

- **Script editor overview**
- Basic syntax usage



# Script editor overview

- The DIAView configuration software is comprehensive and powerful, but the functions required by customers are different. Some functions are customized according to customer needs. The DIAView configuration software can write related programs through a script editor to complete some special tasks and functions.
- The event configuration and user program in the DIAView configuration software all need to use the script editor to write scripts. The DIAView configuration software uses VB Script language. Users can write logic control programs according to the VB Script language grammar specifications, so as to complete special functions and enhance the usability of the system.

- Script editor overview
- Basic syntax usage

## ➤ Use of conditional statements

- A variant of the **If...Then...Else** statement allows you to choose from multiple conditions, namely adding an **Else If** clause to expand the functionality of the **If...Then...Else** statement,
- Allows you to control the program flow based on many possible. for example:

```
Dim value
If value = 0 Then
    MsgBox value
Elseif value = 1 Then
    MsgBox value
Elseif value = 2 then
    MsgBox value
Else
    MsgBox "The value is out of range!"
End If
```

## ➤ Use of loop statements

- The **For...Next** statement is used to run the statement block a specified number of times. The counter variable is used in the loop, and the value of the variable increases or decreases with each loop.
- For example: the following example repeats the procedure **MyProc** 50 times. The For statement specifies the counter variable x and its start and end values. The **Next** statement increments the counter variable by 1.

```
Sub DoMyProc50Times()  
    Dim x  
    For x = 1 To 50  
        MyProc  
    Next  
End Sub
```

## ➤ Use of Function process

- **Function** procedures are a set of VBScript statements contained between Function and End Function statements. Function procedures are similar to Sub procedures, but Function procedures can return values. Function procedures can use parameters (constants, variables, or expressions passed by the calling procedure).
- If the Function procedure has no parameters, the Function statement must contain empty brackets (). The Function procedure returns a value by the function name. This value is assigned to the function name in the procedure statement. The data type of the function return value is always **Variant**.

## ➤ Use of Fuction process

- In the following example:

The Celsius function converts Fahrenheit to **Celsius**. Sub procedure **ConvertTemp** When this function is called, the variable containing the parameter value is passed to the function. The conversion result is returned to the calling process and displayed in the message box.

```
Sub ConvertTemp()  
    temp = InputBox("Please enter the temperature in Fahrenheit.", 1)  
    MsgBox "The temperature is "& Celsius(temp) & "Celsius."  
End Sub  
Function Celsius(fDegrees)  
    Celsius = (fDegrees-32) * 5/9  
End Function
```

## ➤ Use of Sub Process

- **Sub** procedures are a set of VBScript statements contained between **Sub and End Sub** statements that perform operations but do not return values. Sub procedures can use parameters (constants, variables, or expressions passed by the calling procedure). If the Sub procedure has no parameters, the Sub statement must contain empty brackets ().
- The following Sub procedure uses two inherent (or built-in) VBScript functions, **MsgBox** and **InputBox**, to prompt the user to enter information. Then display the calculation results based on this information. The calculation is done by the Function procedure created using VBScript.

```
Sub ConvertTemp()  
    temp = InputBox("Please enter the temperature in Fahrenheit.", 1)  
    MsgBox "The temperature is "& Celsius(temp) & "Celsius."  
End Sub
```

✘Refer to "20.3 Script Syntax and Functions" in the user manual for details on the use of VBScript basic syntax.

# Smarter. Greener. Together.

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

