

## Practical No. 07

Aim: Create any GUI application e.g. calculator and automate using autoIT v3 toolkit

## Practical No. 07

Aim: Create any GUI application e.g. Calculator and automate using autoit V3 tool.

## Theory :

AutoIt is constantly evolving as a programming language. It started as an add-on tool to automate basic tasks in GUI's of other programs, and task automation (such as sending a keystroke or clicking a button.) is still at the heart of AutoIt. With the introduction of many new features, such however, AutoIt has become a more powerful tool than ever before.

Just a few of the new and updated features include :

- GUI Automation - Create a custom graphical interface for your application.
- COM (object) functionality fills the gap of with WSH languages such as VBScript / JScript .
- Loops, Functions and expression parsing.
- An enormous number of functions handling and manipulating strings.
- A Perl-compatible regular expression engine using the PCRE Library , with native 16 bit mode and UCS/UTF support .
- A powerful recursive File List to Array function .
- Easily call Win32 and third-party DLL APIs from within your script .

Calc GUI :

```
#include <EditConstants.au3>
#include <GUIConstantsEx.au3>
#include <StaticConstants.au3>
#include <WindowsConstants.au3>

GUICreate("Calculator", 260, 230)
$idBtn1 = GUICtrlCreateButton("1", 54, 138, 36, 29)
Local $idBtn0 = GUICtrlCreateButton("0", 54, 171, 36, 29)
Local $idBtn1 = GUICtrlCreateButton("1", 54, 138, 36, 29)
Local $idBtn2 = GUICtrlCreateButton("2", 93, 138, 36, 29)
Local $idBtn3 = GUICtrlCreateButton("3", 132, 138, 36, 29)
Local $idBtn4 = GUICtrlCreateButton("4", 54, 106, 36, 29)
Local $idBtn5 = GUICtrlCreateButton("5", 93, 106, 36, 29)
Local $idBtn6 = GUICtrlCreateButton("6", 132, 106, 36, 29)
Local $idBtn7 = GUICtrlCreateButton("7", 54, 73, 36, 29)
Local $idBtn8 = GUICtrlCreateButton("8", 93, 73, 36, 29)
Local $idBtn9 = GUICtrlCreateButton("9", 132, 73, 36, 29)
Local $idBtnPeriod = GUICtrlCreateButton(".", 132, 171, 36, 29)

Local $idBtnMClear = GUICtrlCreateButton("MC", 8, 73, 36, 29)
Local $idBtnMRestore = GUICtrlCreateButton("MR", 8, 106, 36, 29)
Local $idBtnMStore = GUICtrlCreateButton("MS", 8, 138, 36, 29)
Local $idBtnMAdd = GUICtrlCreateButton("M+", 8, 171, 36, 29)

Local $idBtnChangeSign = GUICtrlCreateButton("+/-", 93, 171, 36, 29)
Local $idBtnDivision = GUICtrlCreateButton("/", 171, 73, 36, 29)
Local $idBtnMultiplication = GUICtrlCreateButton("*", 171, 106, 36, 29)
```

Local \$idBtnSubtract = GUICreateButton (" - ", 171, 138, 36, 29)  
Local \$idBtnAddition = GUICtrlCreateButton (" + ", 171, 171, 36, 29)  
Local \$idBtnAnswer = GUICtrlCreateButton (" = ", 210, 171, 36, 29)  
Local \$idBtnInverse = GUICtrlCreateButton (" 1/x ", 210, 138, 36, 29)  
Local \$idBtnSqrt = GUICtrlCreateButton (" sqrt ", 210, 73, 36, 29)  
Local \$idBtnPercentage = GUICtrlCreateButton (" .% ", 210, 106, 36, 29)  
Local \$idBtnBackspace = GUICtrlCreateButton (" backspace ", 54, 37, 63, 29)  
Local \$idBtnClearE = GUICtrlCreateButton (" CE ", 120, 37, 62, 29)  
Local \$idBtnClear = GUICtrlCreateButton (" C ", 185, 37, 62, 29)  
Local \$idEditScreen = GUICtrlCreateButton (" 0. ", 8, 2, 239, 23)  
Local \$idLbMemory = GUICtrlCreateButton (" ", 12, 39, 27, 26)  
GUI SetState ()  
Local \$msg  
DO  
\$msg = GUIGetMsg()  
Until \$msg = \$GUI\_EVENT\_CLOSE

```
Local $idEdtScreen = GUICtrlCreateButton("0.", 8, 2, 239, 23, BitOr($ES_READONLY,  
$ES_RIGHT), $WS_EX_STATICEDGE)  
Local $idLblMemory = GUICtrlCreateButton("", 12, 39, 27, 26, $SS_SUNKEN)
```

## Operations:

`$1 = InputBox("Maths", "Number:", "")`

\\$operation = InputBox("Maths", "+,-,\*,:","")

S2 = InputBox ("Maths", "plus number : ", "")

$$\$plus = \$1 + \$2$$

$$\$_{\text{minus}} = \$1 - \$2$$

$$\$times = \$1 * \$2$$

Conclusion:

Thus, we have studied and executed a program in AutoIT v3 to create calculator and automated it using AutoIT v3.