

Experiment No 7

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 is still at the heart of AutoIt. With the introduction of many new features, however, AutoIt has become a more powerful tool than ever before.

Just few of the new and updated features include :

- GUI Automation - Create a custom graphical interface for your application.
- COM functionality Fill the gap with WSH language such as VBScript/JScript.
- Loops, Functions and expression parsing.
- An enormous number of functions for handling and manipulating Strings.
- A perl-compatible regular expression engine using the PCRE library, with native 16bit mode and UCP/UTF support.
- A powerful Recursive File List to Array function
- Easily call win32 and third-party DLL APIs from within your script.


```
#include <EditConstants.au3>
#include <GUIConstant.au3>
#include <StaticConstant.au3>
#include <WindowsConstant.au3>
GUICreate ("Calculator", 260, 230)
$idBtn1 = GUICtrlCreateButton ("1", 54, 138, 36, 29)
; Digit's buttons
```

```
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)
```

; Memory buttons

```
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 ("MA", 8, 171, 36, 29)
```

; Option Operators.

```
Local $idBtnChangeSign = GUICtrlCreateButton (" +/-", 93, 171, 36, 29)
Local $idBtnDivision = GUICtrlCreateButton ("/", 171, 73, 36, 29)
Local $idBtnMultiplication = GUICtrlCreateButton ("*", 171, 106, 36, 29)
Local $idBtnSubtract = GUICtrlCreateButton ("-", 171, 138, 36, 29)
```



```

local $id BtnAdd = GUICtrlCreateButton("+", 171, 171, 36, 29)
local $id BtnAnswer = GUICtrlCreateButton("=", 210, 141, 36, 29)
local $id BtnInverse = GUICtrlCreateButton("1/x", 210, 138, 36, 29)
local $id BtnSqrt = GUICtrlCreateButton("Sqrt", 210, 73, 36, 29)
local $id BtnPercentage = GUICtrlCreateButton("%", 210, 106, 36, 29)
local $id BtnBackspace = GUICtrlCreateButton("Backspace", 54, 37, 36, 29)
local $id BtnClearE = GUICtrlCreateButton("CE", 120, 37, 36, 29)
local $id BtnClear = GUICtrlCreateButton("C", 125, 37, 36, 29)
local $id BtnEditScreen = GUICtrlCreateButton("0.", 8, 2, 239, 23)
local $id BtnLblMemory = GUICtrlCreateButton(" ", 12, 39, 27, 26)
GUISetState()

```

Local \$msg

Do

\$msg = GUIGetMsg()

Until \$msg = \$GUI_EVENT_CLOSE

```

local $id EditScreen = GUICtrlCreateEdit("0.", 8, 2, 239, 23,
    BitOR($ES_READONLY, $ES_RIGHT),
    $WS_EX_STATICEDGE)

```

```

local $id LblMemory = GUICtrlCreateLabel("", 12, 39, 27, 26,
    $SS_SUNKEN)

```

Operations :

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

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

\$2 = InputBox("Maths", "plus Number:", "")

\$plus = \$1 + \$2

\$minus = \$1 - \$2

\$times = \$1 * \$2


```
IF $operation = "+" Then
    MsgBox (" ", "Maths", "= " & $plus)
EndIf
IF $operation = "-" Then
    MsgBox (" ", "Maths", "= " & $minus)
EndIf
IF $operation = "*" Then
    MsgBox (" ", "Maths", "= " & $times)
EndIf
```

Automate :

```
; Launch calc
Run("Calc.exe", "")
; wait for 2 seconds
sleep(2000)
IF WinExists("Calculator", "") Then
    ; Press Button 5
    ControlClick("Calculator", "", 135)
    sleep(2000)
    ; Press Button +
    ControlClick("Calculator", "", 93)
    sleep(2000)
    ; Press button 6
    ControlClick("Calculator", "", 136)
    sleep(2000)
    ; Press Button =
    ControlClick("Calculator", "", 121)
    sleep(2000)
    ; Close Calc
```

```
WinClose ("Calculator", " ")
```

```
EndIf
```

Conclusion : Thus we have studied and executed a program in Autoit v3 to Create Calculator.



1 practical7.au3 2 automate.au3 3 operation.au3

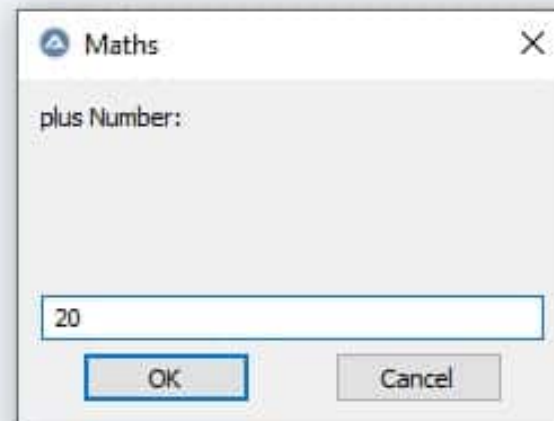
```
1 ;Automate:
2 ;Launch calc
3 Run("Calc.exe","")
4 ;wait for 2 seconds
5 sleep(2000)
6 If WinExists("Calculator","") Then
7     ;Press Button 5
8     ControlClick("Calculator","",135)
9     sleep(2000)
10    ;Press Button +
11    ControlClick("Calculator","",93)
12    sleep(2000)
13    ;Press Button 6
14    ControlClick("Calculator","",136)
15    sleep(2000)
16    ;Press Button =
17    ControlClick("Calculator","",121)
18    sleep(2000)
19    ;Close Calc
20    WinClose("Calculator","")
21 EndIf
```





1 practical7.au3 2 automate.au3 3 operation.au3

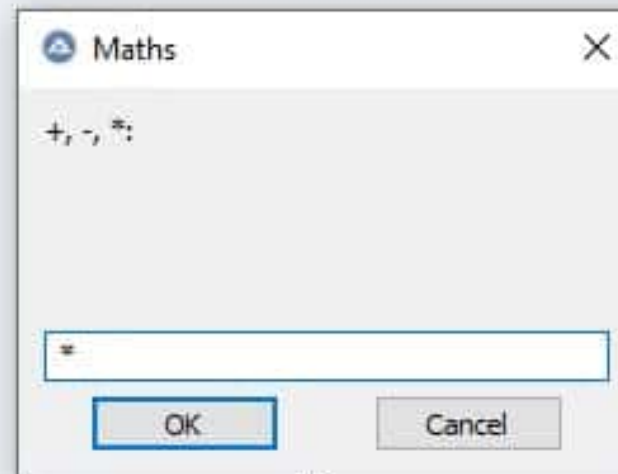
```
1 ;Operations:
2 $1 = InputBox("Maths", "Number:", "")
3 $operation = InputBox("Maths", "+, -, *:", "")
4 $2 = InputBox("Maths", "plus Number:", "")
5 $plus = $1 + $2
6 $minus = $1 - $2
7 $times = $1 * $2
8 If $operation = "+" Then
9     MsgBox("", "Maths", "= " & $plus)
10 EndIf
11 If $operation = "-" Then
12     MsgBox("", "Maths", "= " & $minus)
13 EndIf
14 If $operation = "*" Then
15     MsgBox("", "Maths", "= " & $times)
16 EndIf
```





1 practical7.au3 2 automate.au3 3 operation.au3

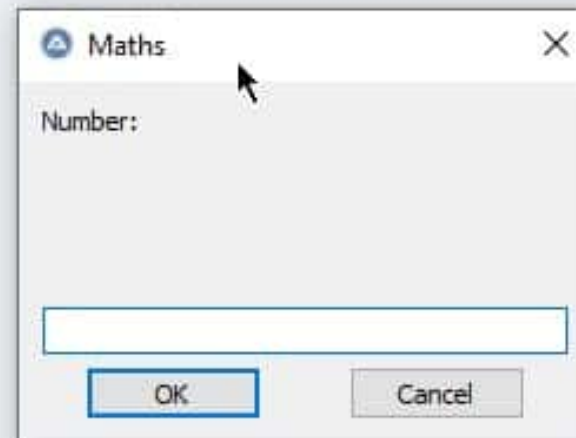
```
1 ;Operations:
2 $1 = InputBox("Maths", "Number:", "")
3 $operation = InputBox("Maths", "+, -, *:", "")
4 $2 = InputBox("Maths", "plus Number:", "")
5 $plus = $1 + $2
6 $minus = $1 - $2
7 $times = $1 * $2
8 If $operation = "+" Then
9     MsgBox("", "Maths", "= " & $plus)
10 EndIf
11 If $operation = "-" Then
12     MsgBox("", "Maths", "= " & $minus)
13 EndIf
14 If $operation = "*" Then
15     MsgBox("", "Maths", "= " & $times)
16 EndIf
```





1 practical7.au3 2 automate.au3 3 operation.au3

```
1 ;Operations:
2 $1 = InputBox("Maths", "Number:", "")
3 $operation = InputBox("Maths", "+, -, *:", "")
4 $2 = InputBox("Maths", "plus Number:", "")
5 $plus = $1 + $2
6 $minus = $1 - $2
7 $times = $1 * $2
8 If $operation = "+" Then
9     MsgBox("", "Maths", "= " & $plus)
10 EndIf
11 If $operation = "-" Then
12     MsgBox("", "Maths", "= " & $minus)
13 EndIf
14 If $operation = "*" Then
15     MsgBox("", "Maths", "= " & $times)
16 EndIf
```





1 practical7.au3 2 automate.au3 3 operation.au3

```
1 ;Operations:
2 $1 = InputBox("Maths", "Number:", "")
3 $operation = InputBox("Maths", "+, -, *:", "")
4 $2 = InputBox("Maths", "plus Number:", "")
5 $plus = $1 + $2
6 $minus = $1 - $2
7 $times = $1 * $2
8 If $operation = "+" Then
9     MsgBox("", "Maths", "= " & $plus)
10 EndIf
11 If $operation = "-" Then
12     MsgBox("", "Maths", "= " & $minus)
13 EndIf
14 If $operation = "*" Then
15     MsgBox("", "Maths", "= " & $times)
16 EndIf
```

