

# Form Generator

## Contents table

1	Form generator.....	1
1.1	Calling formHandler.....	1
1.2	Data description.....	1
1.2.1	Field Name.....	1
1.2.2	Field Label.....	1
1.2.3	Length.....	1
1.2.4	Type .....	1
1.2.5	Default value.....	2
1.2.6	Item list.....	2
1.3	Returned Values.....	2
1.4	Remarks.....	2
1.4.1	Data presentation.....	2
1.4.2	Controls.....	2
1.4.3	CallBack function.....	2
1.4.4	Handling Buttons.....	3
2	Others functions and utilities.....	3
2.1	Retrieve data.....	3
2.2	The Sand Box.....	3
3	Open problems.....	3

## 1 Form generator

The script `formHandler` is a AutoIt script (version v3.3.8.0), which allows to build and handle forms; it is sufficiently generalised for a wide use.

### 1.1 Calling formHandler

The function `formHandler` has three parameters, the first is the title of the generated form, the second is a list of field's description separated by semicolon and the third, optional, is a handle to an existent window or a name of a CallBack function (see Remarks section).

Without the third parameter `formHandler` is called as `PopUP`, otherwise the form is created in the parent window.

### 1.2 Data description

Every field is characterised by a list of attributes which must be comma separated and are in this order: Field Name, Filed Label, Length, Type, Default Value and Items list.

#### 1.2.1 Field Name

Is the name of the field, which is returned with value.

#### 1.2.2 Field Label

Label of the Field or caption of button, if omitted Field Name is used.

#### 1.2.3 Length

The length of the field or the buttons for Type **BL**; it is number of characters.

#### 1.2.4 Type

- **B** button;
- **BL** button List, every button will have a Label Field as caption;
- **F** file;
- **D** folder;

- **N** numeric field (right aligned) and controlled;
- **P** password field;
- **CKB** check box, values are 1 for selected check boxes, 4 otherwise, if Default value is 1 then the check box appears checked;
- **CMB** combo box;
- **CMT** combo box modifiable;
- **T** text field is the default if the Type is omitted;
- **U** not modifiable field.

An **M** after Type means that the field is mandatory, this is ignored for type **B** and **U**.

The Types are accepted also in lower case.

### 1.2.5 Default value

This value appears in the Text, Combos and Check boxes and is restored when the `Reset` button is clicked. This is also used for build forms with data inside.

### 1.2.6 Item list

For `combos` and `Button List` is a item list separated by `|`, a couple of `| (| )` insert an empty item.

For Type **B** Buttons, this parameter contains the value which will be returned when the button is clicked.

For combo or button List if you would return a code associated to the description, the item has the form: `code,value`.

Example:

```
Unit,Measure Unit,5,CMTM,Kg,||Kg,Kilos|Lt,Liters|Mc,CubicMeters|Wh,Watt/hour
```

## 1.3 Returned Values

The function return a two dimension array where the column 0 (0 index) contains the field Names and the column 1 contains the values; if the form is cancelled an empty string is returned.

## 1.4 Remarks

### 1.4.1 Data presentation

The data are presented in the order they appear in the list of parameters, with the exception of the buttons that appear together to buttons inserted at the bottom by `formHandler`.

### 1.4.2 Controls

Numeric fields, if the type is **N**.

### 1.4.3 Callback function

`formHandler` works in polling mode that is with `GUIOnEventMode=0`; this means which the events of the parent window aren't handled; if you need to handle also these events you must use a `Callback` function.

For handle Events of the controls external to `formHandler`, they must be coded in a function besides the identifiers of controls must be declared Global and the third parameter of `formHandler` is the name of that function; the script below is a sample:

```
#Include <Array.au3>
#include <WindowsConstants.au3>
#include "formhandler.au3"
$parentWindow = GUICreate("Parent Window", 450, 350, 100, 100)
$helpMenu = GUICtrlCreateMenu("?")
Global $infoItem = GUICtrlCreateMenuItem("&Info", $helpMenu)
AutoItSetOption("GUIResizeMode", $GUI_DOCKALL)
$records = "Name,,12,CMB,Beta,||K1,Alfa|K2,Beta|K3,Delta" _
          & " ;Nome2,Ok,7,BL,,K1,Alfa|K2,Beta|
K3,Delta;Nome3,ClickMe,7,B,,Gamma"
_ArrayDisplay(formHandler("Backup",$records,"checkEvents"),"Data")
Do
```

```

    $msg = GUIGetMsg()
    checkEvents($msg)
Until $msg = $GUI_EVENT_CLOSE
Func checkEvents($msg)
    If $msg = $infoItem Then
        MsgBox(0, "Credits", "Condor Informatique - Turin" _
            & @CRLF & "Form Handler Sample" _
            & @CRLF & "Program developped with AutoIt3")
    EndIf
EndFunc

```

#### 1.4.4 Handling Buttons

`formHandler` insert the `Ok` button, the `Ignore` button and the `Reset` button, this is function of the widgets in the form:

- the `Ignore` button is always present,
- the `Reset` button is present if there are data fields (e.g. `Type F, D, N, P, CKB, CMB, CMT`),
- the `Ok` button is present if there are only data fields.

## 2 Others functions and utilities

### 2.1 Retrieve data

The function `TakeData($array, $key)`<sup>1</sup> can be used for retrieve data by name instead of by position:

Example:

```

$records = "BackupName,Backup Name,15, TM; FileName, File Name, 40, FM; ...
$aReturn = formHandler("Backup", $records)
MsgBox(0, "", TakeData($aReturn, "FileName"))

```

### 2.2 The Sand Box

The script `SandBoxFormHandler` help to understand, build parameters and test `formHandler`.

## 3 Open problems

- Orphan ; crash
- Add Radio Buttons
- Add Hidden Fields
- CallBack

---

<sup>1</sup>This function works also on the array returned by `IniReadSection` function.