

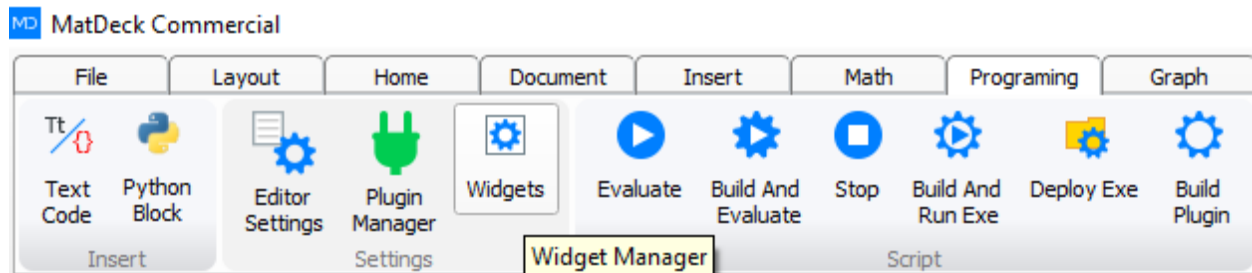
MatDeck GUI Designer

The MatDeck application contains elements of different types of graphical user interface. Lists of implemented elements, ways to create them and implement them in MatDeck documents, descriptions of GUI function arguments and much more are described in [MatDeck GUI User Manual](#).

Besides these individual GUI elements, we have implemented *GUI Designer*. GUI Designer is a software development tool which is used to simplify the creation of widgets and applications, allowing you to arrange GUI elements simply by drag and dropping them.

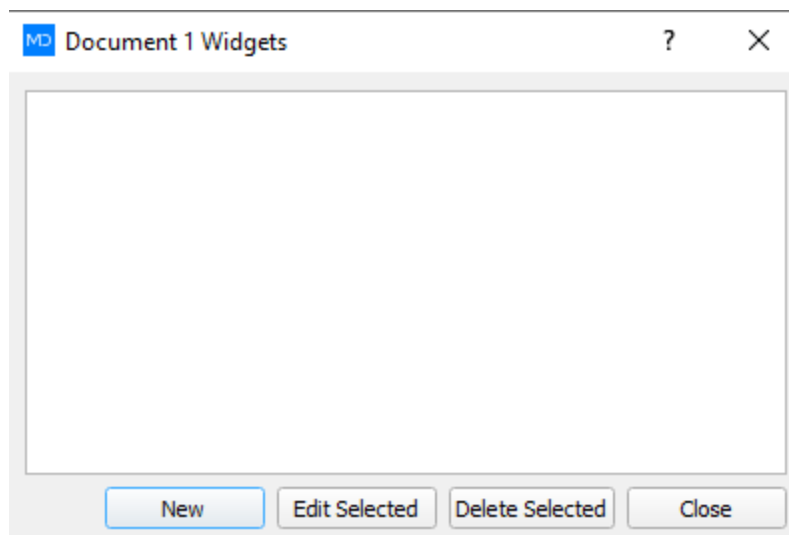
In the MatDeck GUI Designer, there is one main widget on which we place all the other elements. This widget is the parent object for all the other elements that we have placed on the form. Inserting any new container or element on top of another object makes this object the parent for the newly inserted object and there is no limitation in depth for this nesting.

To create a widget and use the GUI Designer in the MatDeck document, go to the Programming tab and press the Widgets button (Picture 1).



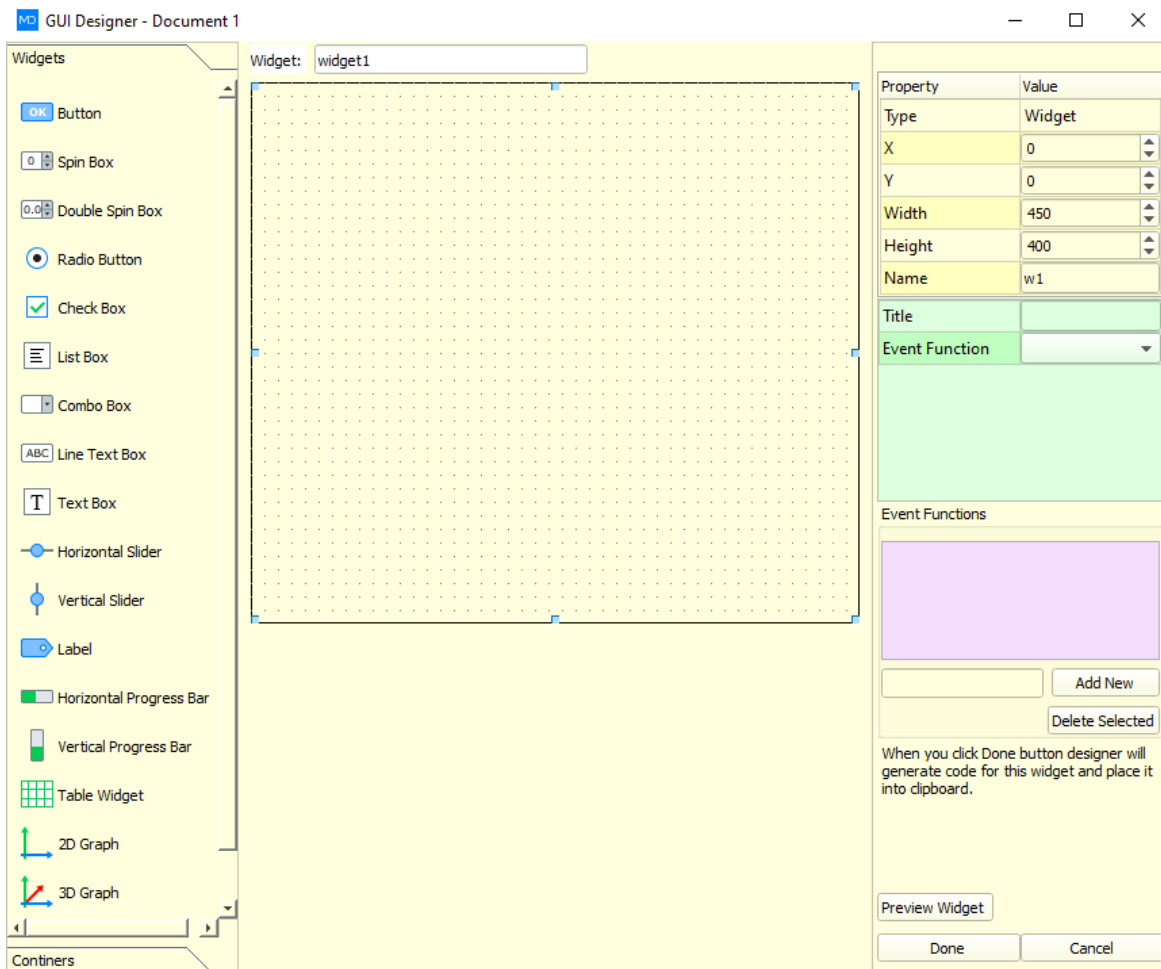
Picture 1: Widgets button

The new windows will open which will contain a list of all the widgets created in the current MatDeck document (Picture 2). From this window, you can create new widgets, edit or delete existing ones.



Picture 2: Document widgets editor

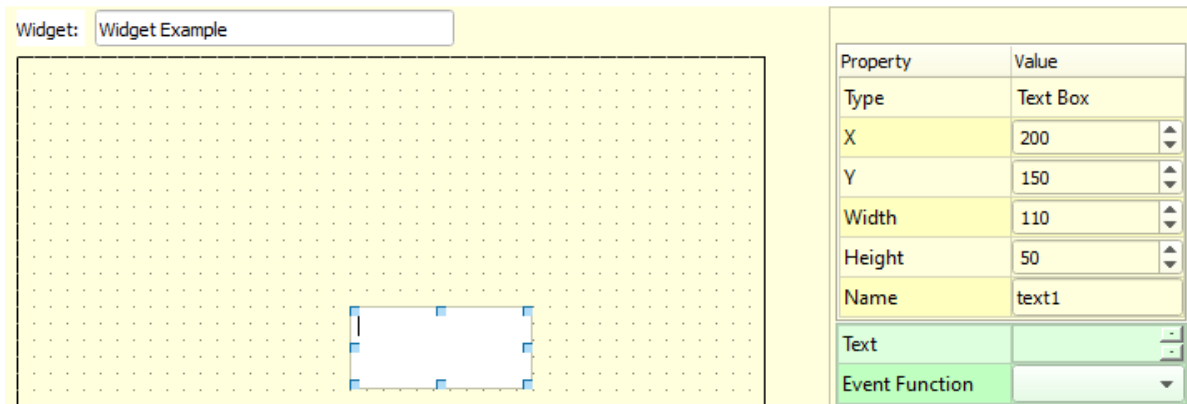
Once you choose to create a new widget or edit an existing one, the GUI Designer will open. The default look of an empty GUI Designer is shown in the picture bellow.



Picture 3: GUI Designer

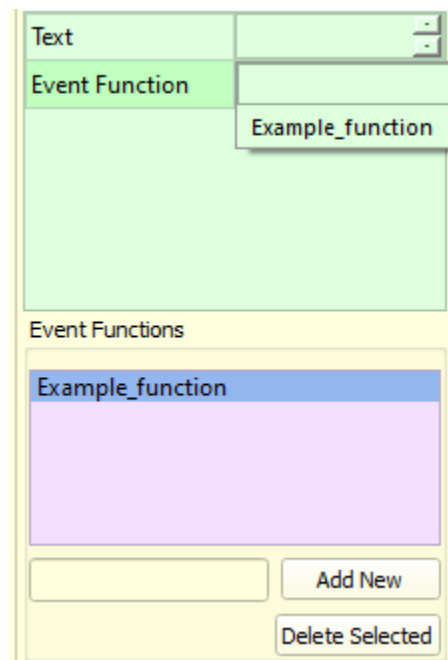
On the left side of the GUI Designer are GUI elements; in the middle of Designer is the designated working area of the widget where GUI elements are placed. To place one of the elements, just press on the preferred GUI element and select a place on the working area of the widget to place it.

For example, we will place a Text Box on the widget. On the right side of the Designer, the properties of the selected object are displayed and these properties can be modified from here (Picture 4). Change the Widget Title to your preferred choice as we have below.



Picture 4: GUI Element Property

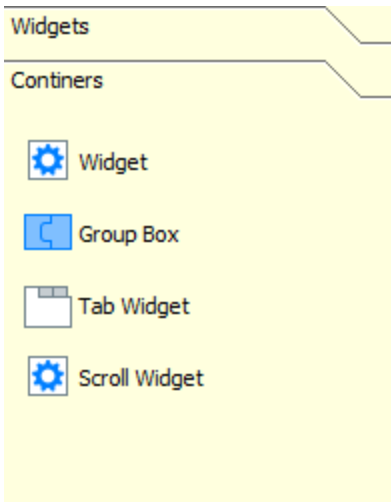
You also have the option to link event functions for the selected GUI element. The Event function list is empty on newly created Widgets; we have to define the new event function via the Event Functions part of the Designer. After the new event function is created, it will appear in the Event Function list for the selected GUI element (Picture 5).



Picture 5: Event Functions

The Event Functions container is a global container which contains all the created functions for the Widget we are currently creating/editing. On the Event Function property of the selected GUI element you can select a defined function from the Event Functions container.

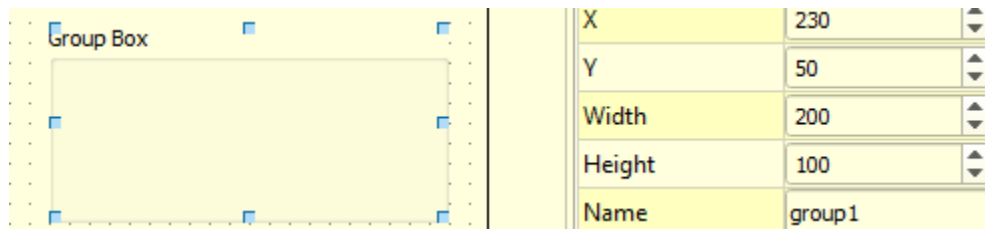
If we open the Containers menu, located in the bottom left part of the GUI Designer, a new list of objects will appear. These objects can be placed on the main widget; they are the carrier for GUI elements.



Picture 6: Containers

Inserting any of these containers on the main form is the same as inserting the other GUI elements. Select the preferred container and select the place on the working area where you want to place it.

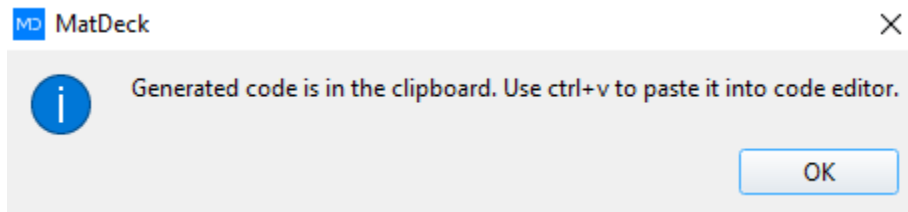
When the container or element is selected, you can resize it by moving the blue squares around the box or you can resize it by changing the Width and Height values located on the Property table of the selected object.



Picture 7: Resizing of object

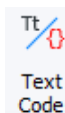
The option *Preview Widget* does exist where, at any time, you can preview the exact look of the widget you are working on.

When you finish working on the widget design, by pressing the *Done* button, the designer will generate code for the widget and place it into the clipboard. If a widget has been created and the *Done* button is pressed for the first time, the notification window (Picture 8) will appear. After the first time you the *Done* button is pressed, the code will be updated without showing this notification.

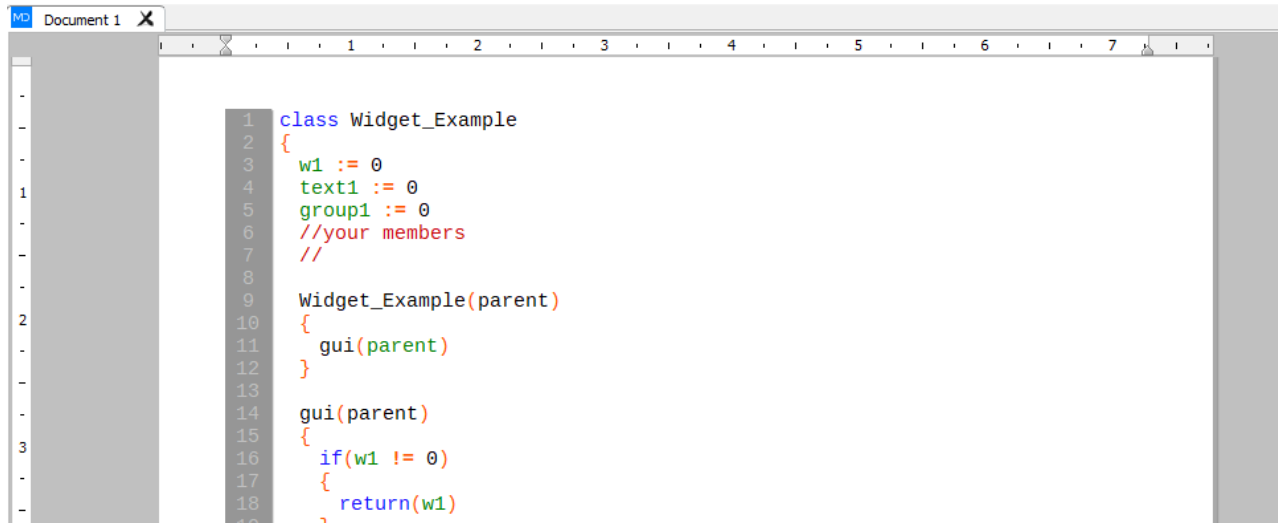


Picture 8: Notification Window

To place the generated code onto the MatDeck document, we have to place the cursor on the preferred position and turn on the Code mode. We can turn this mode on via the Programming tab by using the



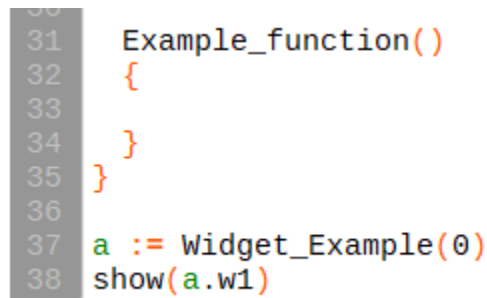
button, or by using the *Ctrl. + i* key combination. When Code mode is on, paste the code from the clipboard onto the document.



```
1 class Widget_Example
2 {
3     w1 := 0
4     text1 := 0
5     group1 := 0
6     //your members
7     //
8
9     Widget_Example(parent)
10 {
11     gui(parent)
12 }
13
14 gui(parent)
15 {
16     if(w1 != 0)
17     {
18         return(w1)
19     }
20 }
```

Picture 9: Widget Code

For every Event function we have created on the GUI Designer, the function in the code will appear with an empty body leaving the user to code whatever they have planned.



```
30
31 Example_function()
32 {
33
34 }
35 }
36
37 a := Widget_Example(0)
38 show(a.w1)
```

Picture 10: Event Function In Code

At the end of code, a constructor for the created widget is added in the combination with the `show()` function. The purpose of these two lines is to create the widget as a stand alone application when you evaluate the current document.

Notice: The Preferred order of creating/changing widget designs and generating or updating code is from the GUI Designer. If you change the code directly in the MatDeck document and open and save the changes in the GUI Designer afterwards, all the changes that were made directly in code will be lost.