

outline

GUI (start with GUIDE)

Figure File (.fig)

- do the UI layout.
- Properties of an object can be edited using the property inspector.
- Tag Property is the name of an object and unique for each object.

m-file (.m)

- Consists of four functions:
our focus is `untitledToolOpeningFcn` and `objectName_Callback`.
- In the callback function, we use:
 - `VariableName = get(handles.TagValue, capitalized 'PropertyName');`
 - `Set(handles.TagValue, 'PropertyName', PropertyValue);`

↓
You can use the functions `str2double` & `num2str`

Building a graphical user interface

note:

The GUI is contained in two files:

- figure file (.fig): Contains ^{the} graphical layout information.
- m-file (.m): Contains the main GUI function & some subfunctions.

- To open the GUIDE window:
write >> guide in the command window

- MATLAB class called uicontrol → contains most of these GUI objects

- the name of each object is the value of the Tag property & it is a unique value for each object.

note:

Callback Functions: each object has a callback function & is executed when an object is activated.

For example, a button is activated when the user presses and releases it.

the name of the callback function has the form:

`TagValue_Callback`

~~note:~~

important note:

the handle of the object: is the address of the object in the memory

note:

The functions `str2double` & `num2str` are frequently used when coding in the m-file.

- m-file has code contains:

untitledTool \rightarrow the main function that creates the tool itself.

untitledTool_OpeningFcn \rightarrow ~~this~~ this function is executed once the program is run. it is like constructors in C++.
just before the tool is made visible.
untitledTool_OutputFcn \rightarrow advanced function, ignore it.

+

objectName_Callback \rightarrow executed when the object is activated.

The used functions for the programmer

in the GUIDE window

To edit an object \rightarrow use Property Inspector window by double-clicking on it.

note: the Tag property is an important property.

Important note: during writing the code, we deal with Property-Value Pairs. it ~~will~~ have the notation that the name of the property is capitalized.
(first letter)

In the m-file

To communicate (edit or ~~get~~ get value) with a GUI object, we need to know the handle of the object.

① To know the handle of an object:

to know the handle of an object, we search for the handle of the object whose $\langle \text{Property} \rangle$ matches the specified $\langle \text{Value} \rangle$

→ To do that we use the handles structure:

`handles.xSlider` → returns the handle of the slider with Tag "xSlider"

② Communicating with the GUI object:

To do that we use the get & set command

→ The general form of the get command:

$\langle \text{Var} \rangle = \text{get}(\langle \text{handle} \rangle, \langle \text{Property name} \rangle);$

ex.

~~get(hText, 'String');~~ `get(hText, 'String');`

returns the value of the String property of the text object

→ The general form of the set command:

$\text{set}(\langle \text{handle} \rangle, \langle \text{Property name} \rangle, \langle \text{Property value} \rangle);$

ex.

`set(hText, 'String', num2str(x));`

`set(hSlider, 'Value', 9);`