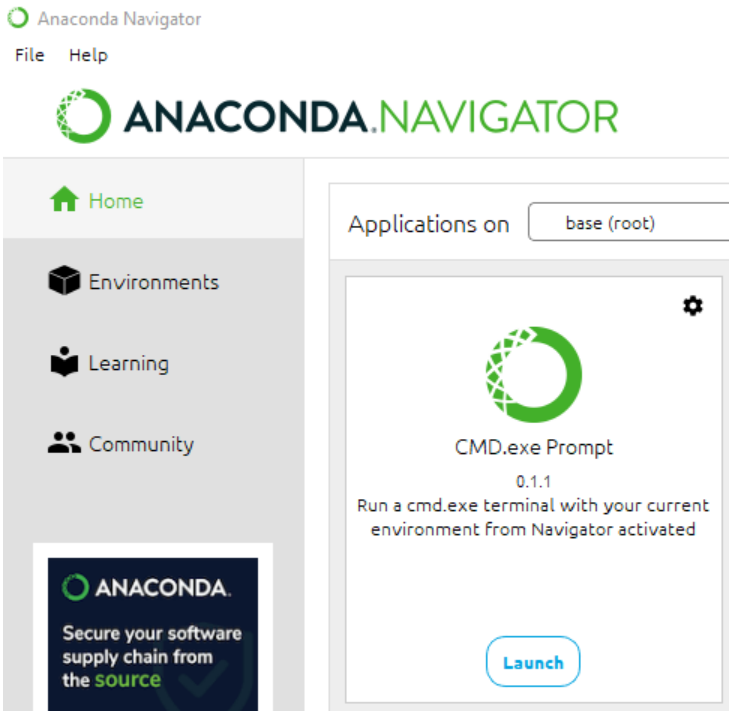
Chapter 7 – Using Custom Slots

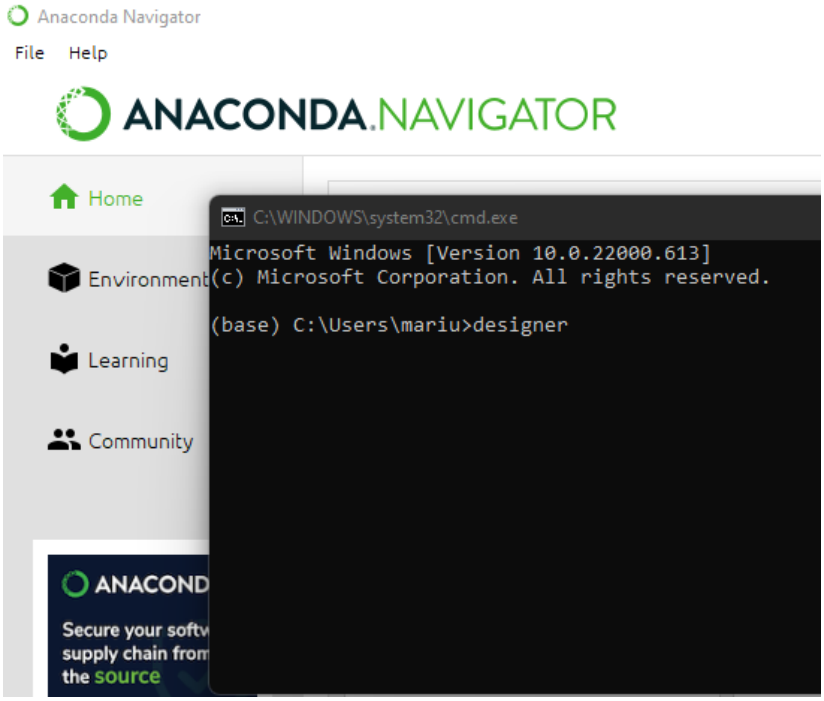
Welcome Message Application

Step 1 Open Qt Designer

- Launch the CMD.exe Prompt from within Anaconda Navigator



- In the command prompt type designer and press enter



Step 2 Selecting a template

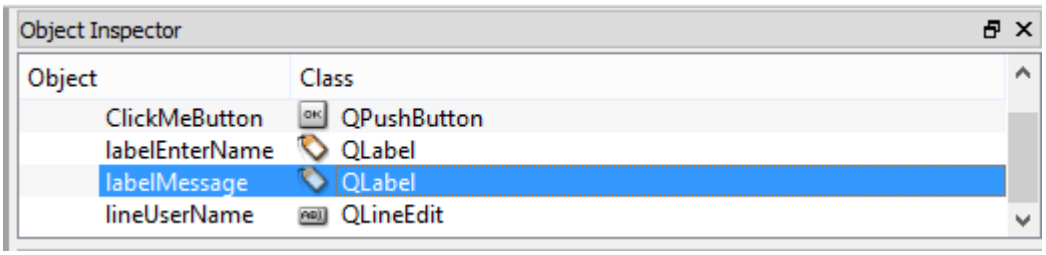
- Click on the “Dialog without Buttons”
- Click the Create button
- A new form with the caption “untitled” is created with no widgets.

Step 3 Adding widgets

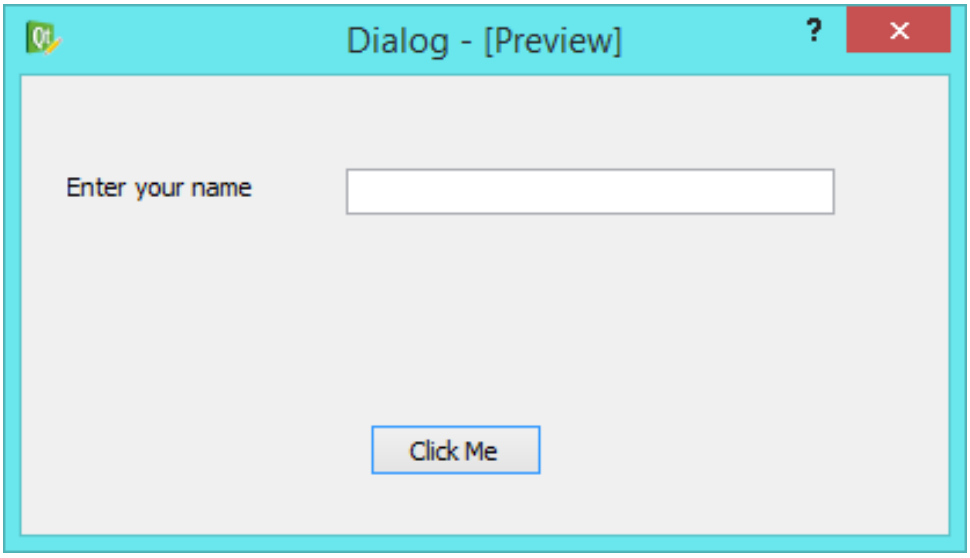
- Add the following widgets and set the properties. For now we will leave the objectNames as the default: Note that Python is case sensitive so make sure of the letter casing on objectnames

Widget	Property	Value
QLabel (Display widgets section)	objectName text	labelEnterName Enter your name
QLabel (Display widgets section)	objectName text	labelMesssage
QLineEdit (Input widgets section)	objectName	lineUserName
QPushButton (Buttons section)	objectName text	ClickMeButton Click Me

- Note that by clearing the text property of labelMessage, it will disappear from the view. You can highlight it by selecting it on the Object Inspector too change its properties



- Preview the form by pressing Ctrl+R
- Your form should look similar to the one below:



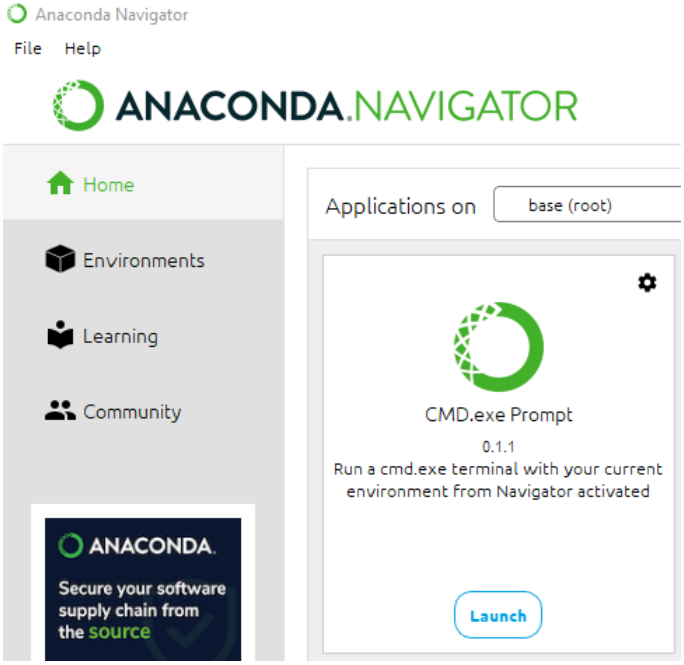
- Close the form by clicking the windows close button.

Step 4 Save the form

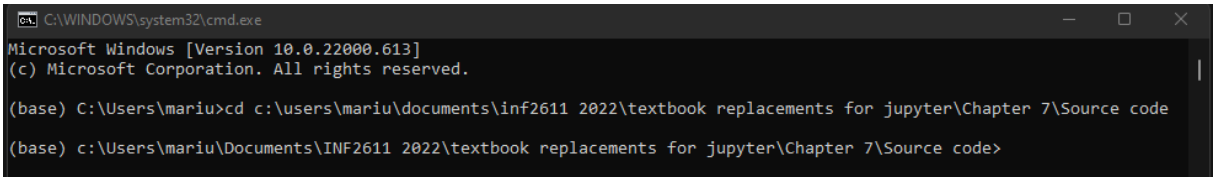
- Click File | Save
- Choose a location to save the form file to where you can easily find it so that we can copy it later to convert to a py file.
- Change the name to `welcomemsg.ui` **(note the case!! Python is case sensitive)**

Step 5 Convert the .ui file to a .py file

- The .ui file created by Qt Designer is an xml file that cannot be used in Python
- The .ui file must be converted to a Python .py script file.
- We will use a command prompt utility for converting the .ui file to a Python script
- Launch the CMD.exe Prompt from within Anaconda Navigator

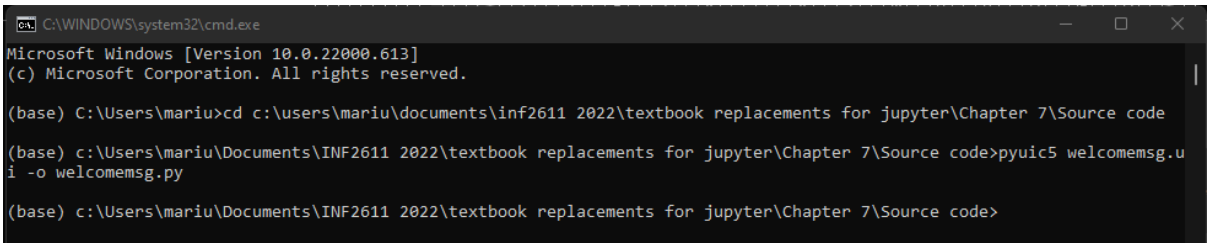


- Change to the folder where you saved your .ui file using the cd command as shown in the screenprint example



- Type `pyuic5 welcomemsg.ui -o welcomemsg.py` and press Enter **(note the case!! Python is case sensitive, so even on file names the case must be the same throughout)**

- The screen will pause for a while and return a blank line if the conversion was successful

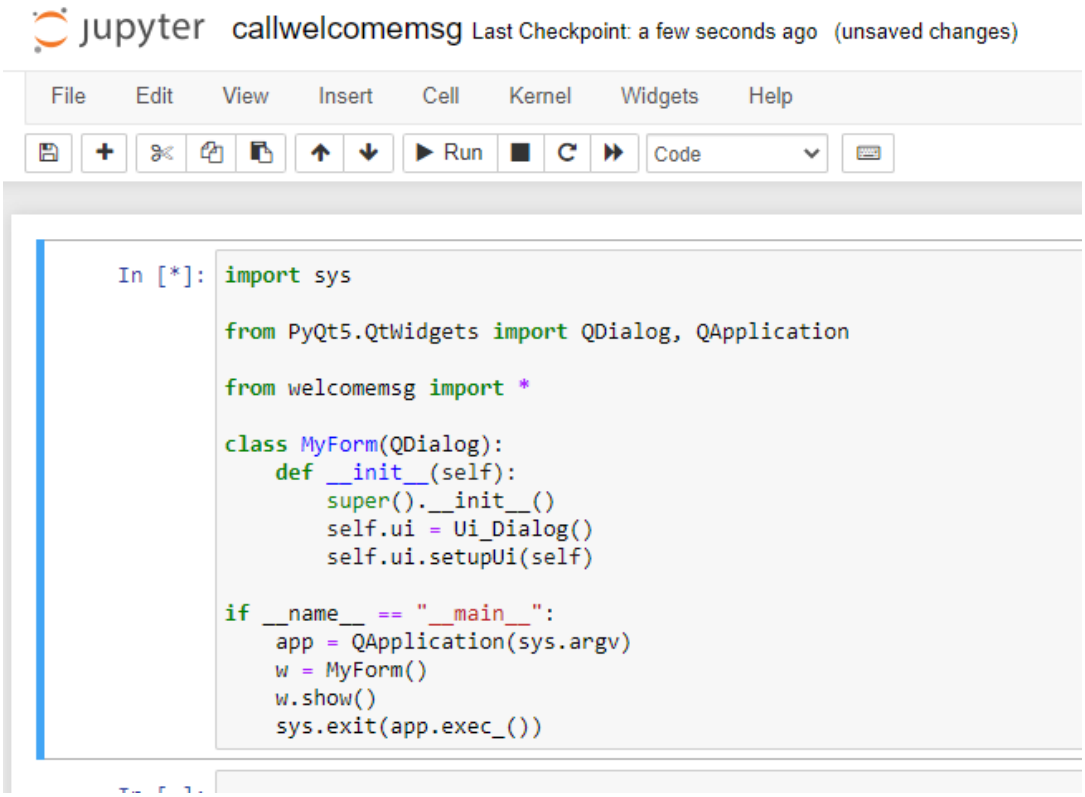


- Copy the `welcomemsg.ui` and `welcomemsg.py` files folder for this application under your documents if it is not already there

Step 6 Create a source file (.pyw) that imports the .py file

- We will now create a source file that will import the .py file created in step 5 above and from which we will invoke the user interface

- Open the Jupyter Notebook and create a new Python3 file and add the following code (note the indentation and case!!)



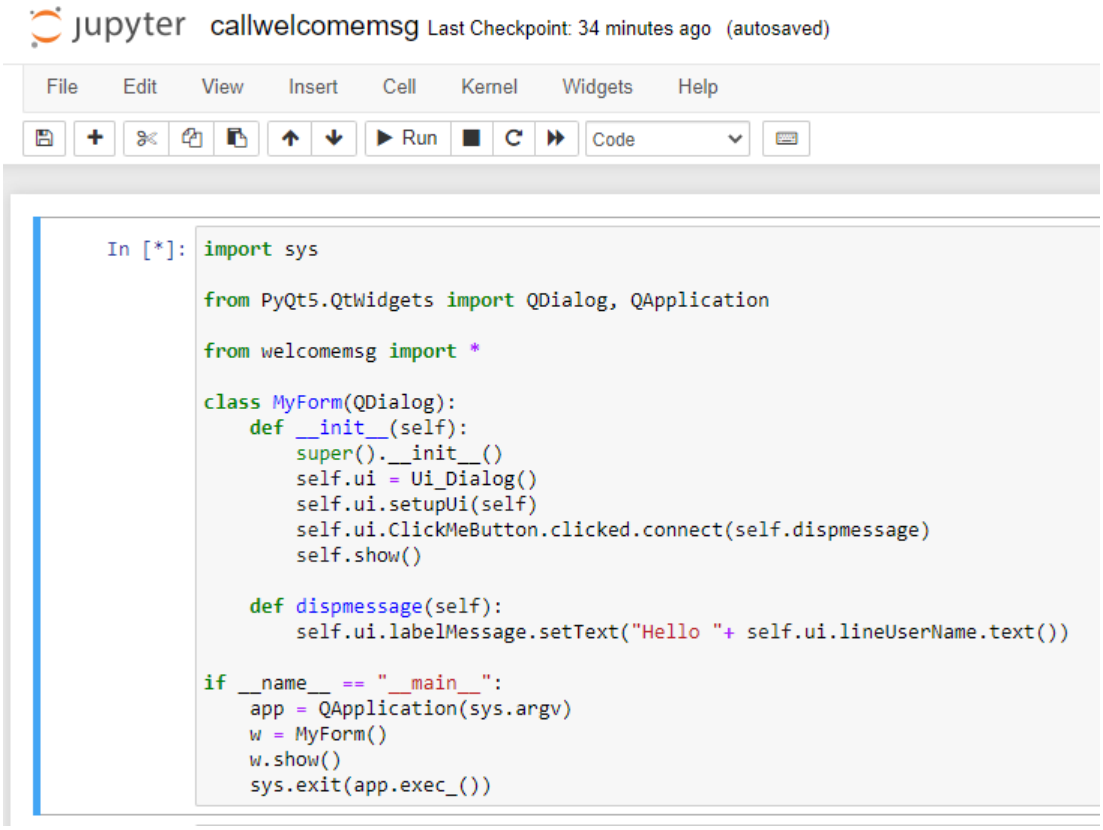
- Save the file as callwelcomemsg
- Run the application by selecting Run |Run Module
- Test your application

Step 7 Add the code for the custom signal and slot

- Add the following code to connect the clicked() signal of the ClickMeButton to a slot to change the text of labelMessage (note the indentation and case!)

```
        self.ui.ClickMeButton.clicked.connect(self.dispmessage)
        self.show()

    def dispmessage(self):
        self.ui.labelMessage.setText("Hello "+ self.ui.lineUserName.text())
```



Dispmessage is a function that we have written to set the text property of labelMessage to a string which consists of the word Hello concatenated with the text that is in the text propert of the lineUserName widget.

- Save the file
- Run the application by selecting Run |Run Module
- Test your application by entering your name in the lineEdit and clicking on the button.