

---



# Good Day , Students

---

---



# Prayer

---

---



# Attendance

---

---


**Recap:**

**Last lesson**



---

---



# Activity

---

---



# GUI to Py Connection

---

# GUI to Py Connection

## Objectives



# GUI to Py Connection

The PyQt5 comes with a GUI builder tool called Qt Designer.

Using its simple drag and drop interface, a GUI interface can be quickly built without having to write the code.

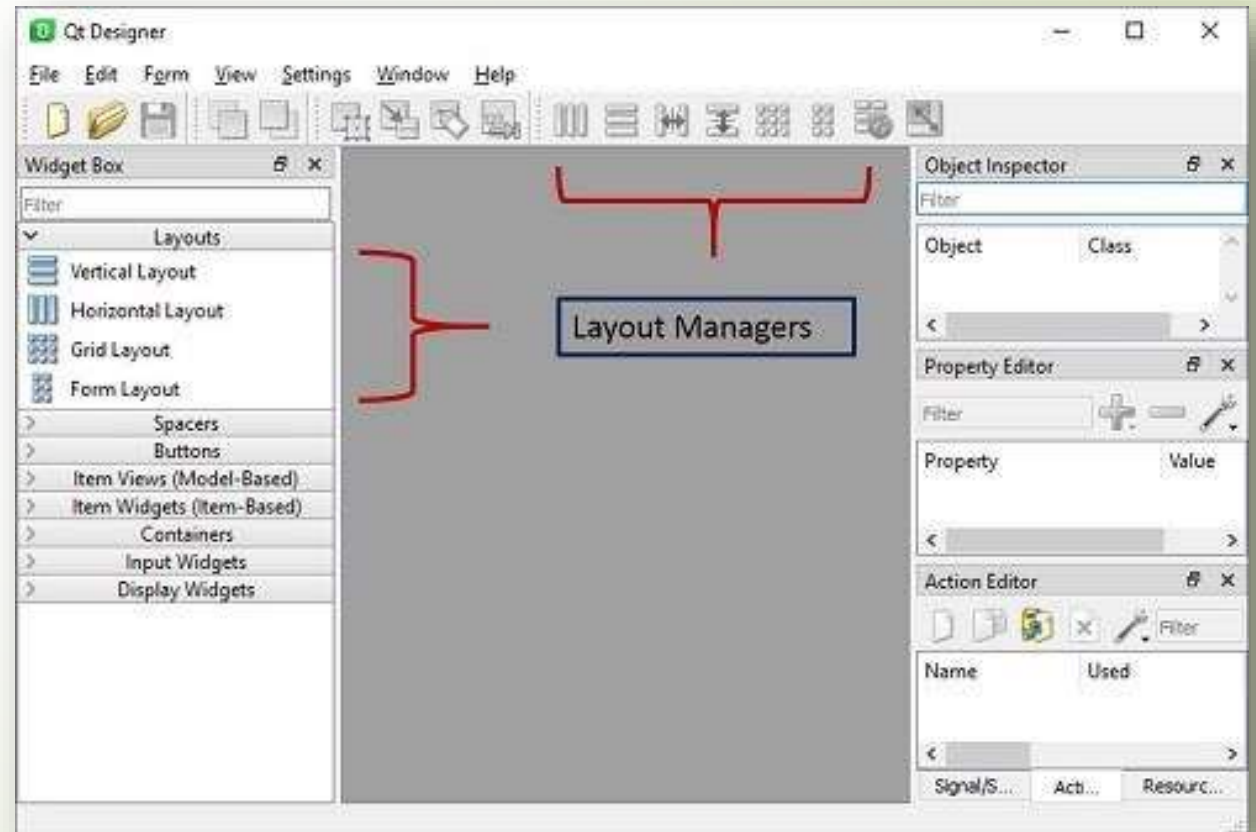
Qt Designer != IDE



# GUI to Py Connection

## 1. Layout Managers

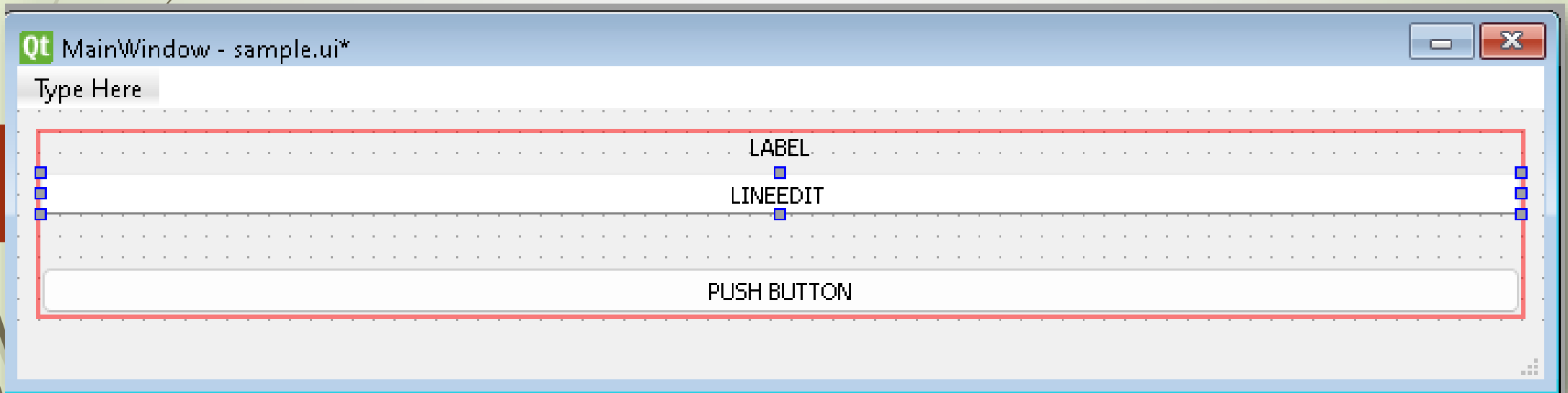
- consists of qbox layout (horizontal & vertical), qgrid layout, and qform layout



# GUI to Py Connection

## 2. Basic Widgets

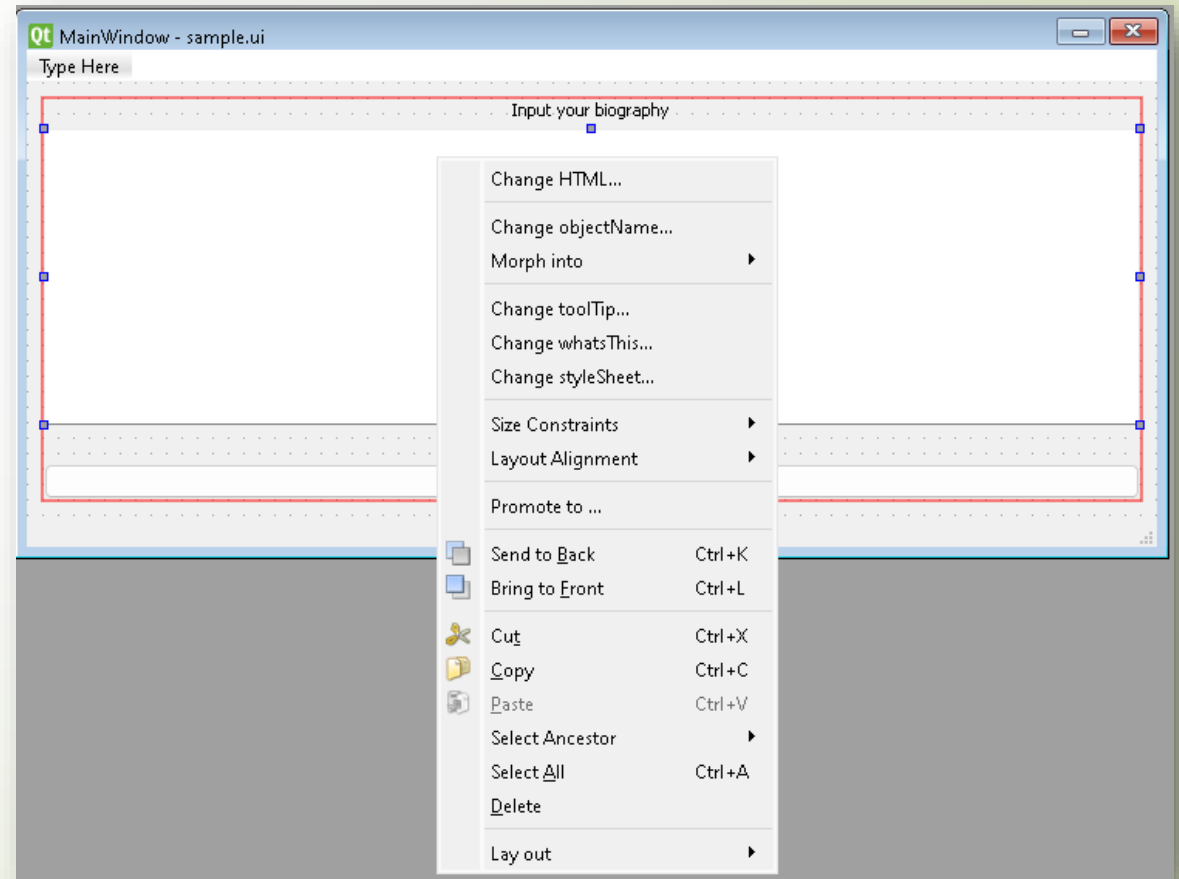
- consists of but not limited to `qlabel`, `qlineedit`, and `qpushbutton`



# GUI to Py Connection

## 3. Context Menu

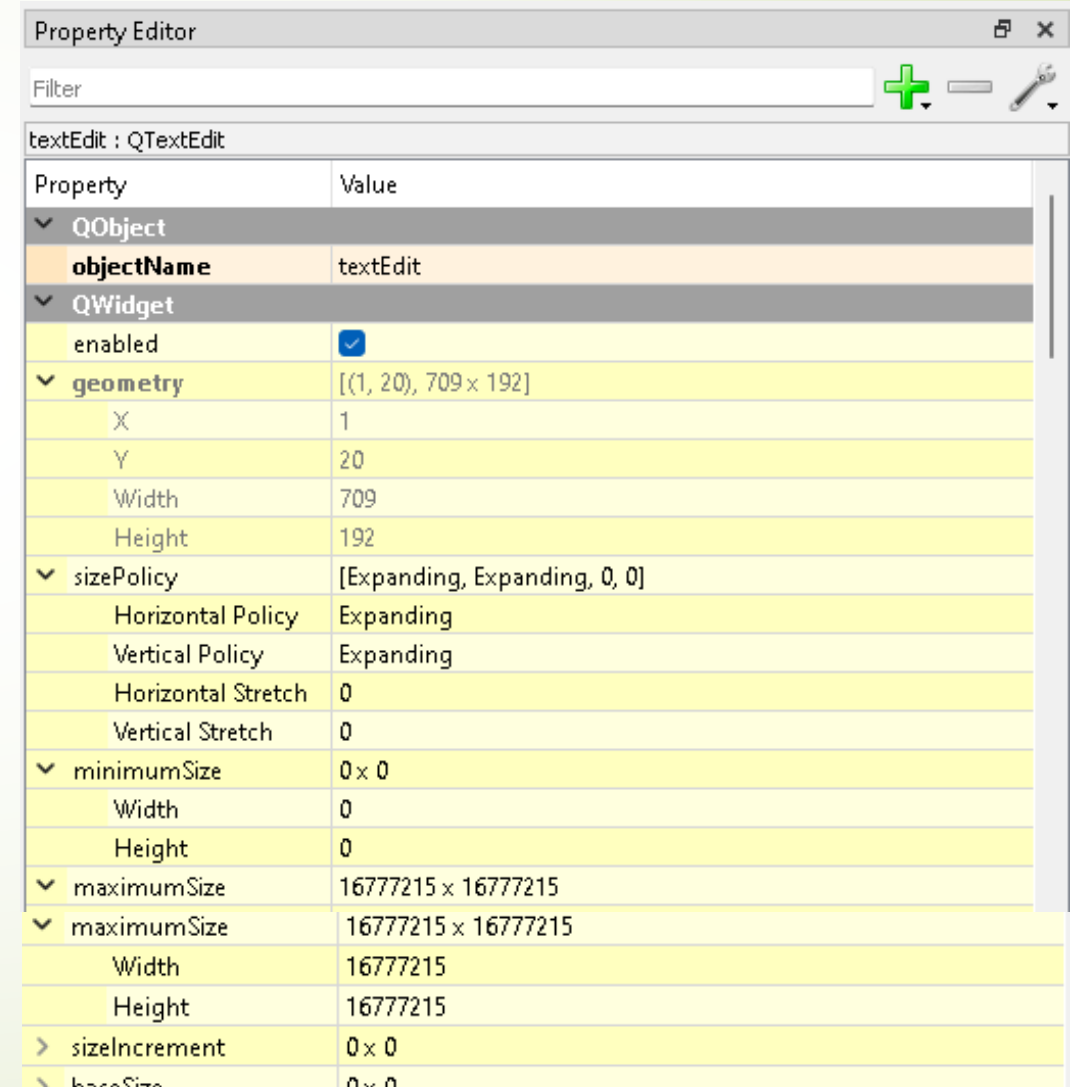
- perform multiple functions like morph object to a different widget, stylesheet (font, color, bg-color), and size



# GUI to Py Connection

## 4. Property Editor


- where most of the modification can be done



# GUI to Py Connection

## 4. Property Editor

- where most of  
the modification  
can be done

Vertical Policy	Expanding
Horizontal Stretch	0
Vertical Stretch	0
▼ minimumSize	0 x 0
Width	0
Height	0
▼ maximumSize	16777215 x 16777215
▼ maximumSize	16777215 x 16777215
Width	16777215
Height	16777215
> sizeIncrement	0 x 0
> baseSize	0 x 0
palette	Inherited
▼ font	A [MS Shell Dlg 2, 8]
Family	MS Shell Dlg 2
Point Size	8
Bold	<input type="checkbox"/>
Italic	<input type="checkbox"/>
Underline	<input type="checkbox"/>
Strikeout	<input type="checkbox"/>
Kerning	<input checked="" type="checkbox"/>
Antialiasing	PreferDefault
cursor	 Arrow
mouseTracking	<input checked="" type="checkbox"/>
tabletTracking	<input type="checkbox"/>
focusPolicy	StrongFocus
contextMenuPolicy	DefaultContextMenu
acceptDrops	<input checked="" type="checkbox"/>
> toolTip	
toolTipDuration	-1
> statusTip	
> whatsThis	
> accessibleName	
> accessibleDescription	
layoutDirection	LeftToRight
autoFillBackground	<input type="checkbox"/>

# GUI to Py Connection

## 4. Property Editor

- where most of  
the modification  
can be done

focusPolicy	StrongFocus
contextMenuPolicy	DefaultContextMenu
acceptDrops	<input checked="" type="checkbox"/>
> toolTip	
toolTipDuration	-1
> statusTip	
> whatsThis	
> accessibleName	
> accessibleDescription	
layoutDirection	LeftToRight
autoFillBackground	<input type="checkbox"/>
styleSheet	
> locale	English, United States
> inputMethodHints	ImhMultiLine
▼ QFrame	
frameShape	StyledPanel
frameShadow	Sunken
lineWidth	1
midLineWidth	0
▼ QAbstractScrollArea	
verticalScrollBarPolicy	ScrollBarAsNeeded
horizontalScrollBarPo...	ScrollBarAsNeeded
sizeAdjustPolicy	AdjustIgnored
▼ QTextEdit	
> autoFormatting	AutoNone
tabChangesFocus	<input type="checkbox"/>
> documentTitle	
undoRedoEnabled	<input checked="" type="checkbox"/>
lineWrapMode	WidgetWidth
lineWrapColumnOrW...	0
readOnly	<input type="checkbox"/>
> markdown	
> html	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN" "htt...
overwriteMode	<input type="checkbox"/>
tabStopWidth	80
tabStopDistance	80 000000

# GUI to Py Connection

## 4. Property Editor

- where most of  
the modification  
can be done

> accessibleName	
> accessibleDescription	
layoutDirection	LeftToRight
autoFillBackground	<input type="checkbox"/>
styleSheet	
> locale	English, United States
> inputMethodHints	ImhMultiLine
▼ QFrame	
frameShape	StyledPanel
frameShadow	Sunken
lineWidth	1
midLineWidth	0
▼ QAbstractScrollArea	
verticalScrollBarPolicy	ScrollBarAsNeeded
horizontalScrollBarPo...	ScrollBarAsNeeded
sizeAdjustPolicy	AdjustIgnored
▼ QTextEdit	
> autoFormatting	AutoNone
tabChangesFocus	<input type="checkbox"/>
> documentTitle	
undoRedoEnabled	<input checked="" type="checkbox"/>
lineWrapMode	WidgetWidth
lineWrapColumnOrW...	0
readOnly	<input type="checkbox"/>
> markdown	
> html	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN" "htt...
overwriteMode	<input type="checkbox"/>
tabStopWidth	80
tabStopDistance	80.000000
acceptRichText	<input checked="" type="checkbox"/>
cursorWidth	1
> textInteractionFlags	TextSelectableByMouse TextSelectableByKeyboard TextEditabl...
> placeholderText	



# GUI to Py Connection

## Connecting UI to Python

```
from PyQt5 import QtWidgets, uic  
import sys
```



Import PyQt5 and its  
Widgets first



# GUI to Py Connection

## Connecting UI to Python

```
class MainWindow(QtWidgets.QMainWindow):
    def __init__(self):
        super(MainWindow, self).__init__()
        uic.loadUi('sample.ui', self)
        self.window2 = window_2()
        self.setWindowTitle('Biography Editor')
        self.pushButton.clicked.connect(lambda: self.prompt())

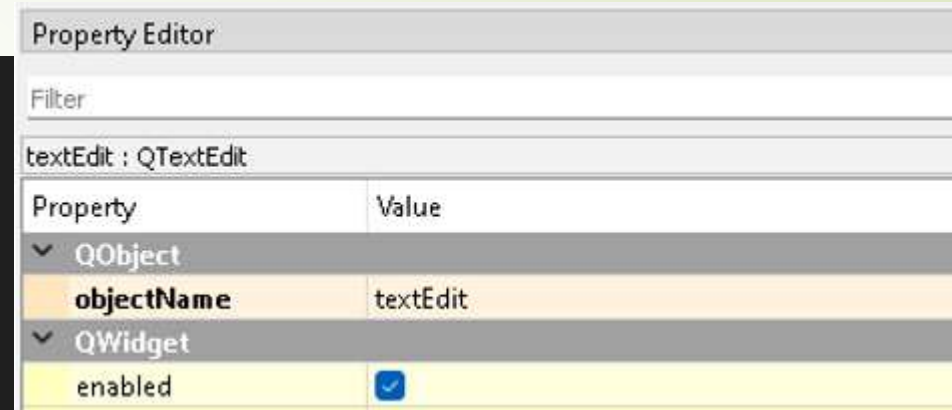
    def btnclicked(self):
        self.label2.setText('Input is Saved!')
        self.window2.label3.setStyleSheet('background-color : green')
        self.window2.plainTextEdit.setPlainText(self.textEdit.toPlainText())
        self.textEdit.clear()
        self.window2.show()
```

**Create a MainWindow Class**

# GUI to Py Connection

## Connecting UI to Python

```
class MainWindow(QtWidgets.QMainWindow):  
    def __init__(self):  
        super(MainWindow, self).__init__()  
        uic.loadUi('sample.ui', self)  
        self.window2 = window_2()  
        self.setWindowTitle('Biography Editor')  
        self.pushButton.clicked.connect(lambda: self.prompt())  
  
    def btnclicked(self):  
        self.label2.setText('Input is Saved!')  
        self.window2.label3.setStyleSheet('background-color : green')  
        self.window2.plainTextEdit.setPlainText(self.textEdit.toPlainText())  
        self.textEdit.clear()  
        self.window2.show()
```



# GUI to Py Connection

## Connecting UI to Python

```
app = QtWidgets.QApplication(sys.argv)
Window = MainWindow()
Window.show()
sys.exit(app.exec_())
```

Create an instantiation of  
MainWindow class

# GUI to Py Connection

## Test your skills



# GUI to Py Connection

Test your skills

Instructions: The class will be divided into 2 groups and each group will complete the following code snippet:

```
from [1] import QtWidgets, [2]  
import [3]  
  
app = QtWidgets.[4](sys.argv)  
window = MainWindow()  
window.[5]  
sys.exit(app.exec_())
```

# GUI to Py Connection


Test your skills

```
from 1 import QtWidgets, 2
import 3

app = QtWidgets.4(sys.argv)
Window = MainWindow()
Window.5
sys.exit(app.exec_())
```

# GUI to Py Connection


## Questions?





# GUI to Py Connection


To summarize  
it all...





# GUI to Py Connection

Quiz



# GUI to Py Connection

## Quiz

1. A GUI builder tool in PyQt5 with simple drag and drop function for making interface without writing a code.
  - a. PyQt5
  - b. Qt for Python
  - c. Qt Designer
  - d. PyQt5 Designer
2. Also known as containers where most of the widgets are placed into so that objects are displayed properly.
  - a. Property Editor
  - b. Layout Managers
  - c. Basic Widgets
  - d. Context Menu

# GUI to Py Connection

## Quiz

3. This is where you can perform multiple functions like morph object to a different widget, stylesheet (font, color, bg-color), and size.

- a. Property Editor
- b. Layout Managers
- c. Basic Widgets
- d. Context Menu

4. This is where most of the modification and manipulation can be done.

- a. Property Editor
- b. Layout Managers
- c. Basic Widgets
- d. Context Menu

# GUI to Py Connection

## Quiz

5. This consists of buttons, labels, windows, dialogs, progress bars etc.
- a. Property Editor
  - b. Layout Managers
  - c. Basic Widgets
  - d. Context Menu

# GUI to Py Connection

Key to Correction

1. C

2. B

3. D

4. A

5. C

# GUI to Py Connection

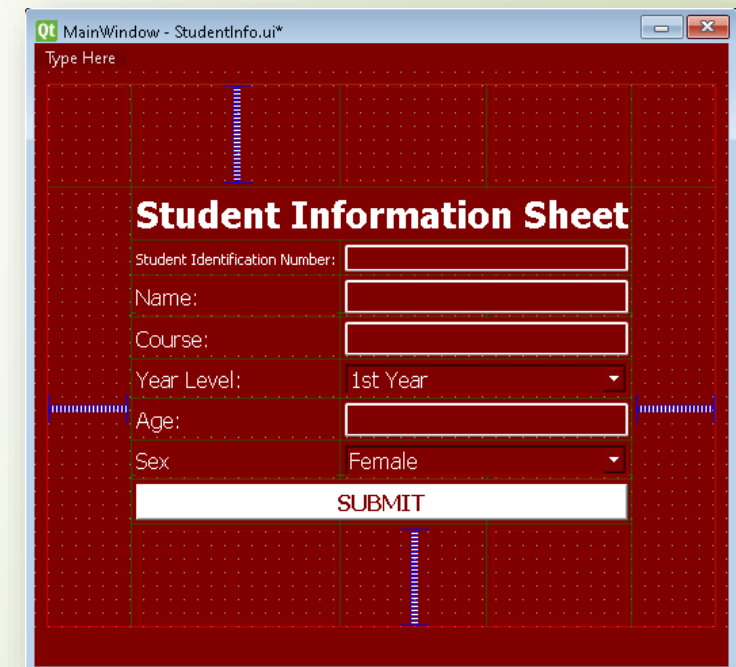
## Assignment

Direction: Create GUI interface with the following specifications:

- Label "Student Information Sheet"
- Form Layout with widgets inside:
  - Student ID Number
  - Name
  - Course
  - Age
  - Combo box for Sex and Year Level
- Push button "Submit"
- Change bg color, font and font size.
- Window must be interactive when being resized.

Deadline = Dec. 14, 2023

Submit the UI file in Google Classroom.



# GUI to Py Connection

## Advanced Assignment

Directions: Do advance research on:

GUI Program & Database Integration





# GUI to Py Connection

## References

Martin Fitzpatrick (2023, March 16). The complete PyQt5 tutorial — Create GUI applications with Python; PythonGUIs.

[https://www.pythonguis.com/pyqt5-tutorial/?fbclid=IwAR1wOr8vivrmjtj\\_2Fi7BiJA62geewVBSWxr7DKHemm1QTK6jBobkx9BlnQ](https://www.pythonguis.com/pyqt5-tutorial/?fbclid=IwAR1wOr8vivrmjtj_2Fi7BiJA62geewVBSWxr7DKHemm1QTK6jBobkx9BlnQ)

abhileknathdas111 (2022, December 19). Introduction to PyQt5; Geeksforgeeks.

<https://www.geeksforgeeks.org/python-introduction-to-pyqt5/?fbclid=IwAR1SWIYLRbuFcMpuL2abHCvgiizKtUQkJX5Kawo6eRnfjGb97w2R0I45M0E>

Michael Herrmann (2023). PyQt5 tutorial: fman build system [https://build-system.fman.io/pyqt5-tutorial?fbclid=IwAR362ANumNSnudHpUYZAiQZDESjSBmn6w43A732U1xQGtY2f\\_jGRAY-NvQU](https://build-system.fman.io/pyqt5-tutorial?fbclid=IwAR362ANumNSnudHpUYZAiQZDESjSBmn6w43A732U1xQGtY2f_jGRAY-NvQU)

Tutorialspoint (2023). PyQt5 Tutorial [https://www.tutorialspoint.com/pyqt5/pyqt5\\_using\\_qt\\_designer.htm](https://www.tutorialspoint.com/pyqt5/pyqt5_using_qt_designer.htm)



---

GUI to Py Connection

Thank You



---