Laboratory Activity No. 4 - Introduction to GUI Development using Pycharm	
SILANG, ELIJAH YOHANCE	10/16/2024
CPE21S1 / CPE009	Mrs. Maria Rizette Sayo

Main.py

main.py | registration.py | Class App.py | gui_buttons.py | gui_text.py

```python
import sys
from PyQt5.QtWidgets import QWidget, QLineEdit, QPushButton, QApplication, QLabel
from PyQt5.QtGui import QIcon
from registration import App

if __name__ == '__main__':
    app = QApplication(sys.argv)
    ex = App()
    ex.show()
    sys.exit(app.exec_())
```

Registration.py

main.py | registration.py | Class App.py | gui_buttons.py | gui_text.py

```python
import sys
from PyQt5.QtWidgets import QWidget, QLineEdit, QPushButton, QApplication, QLabel
from PyQt5.QtGui import QIcon

from tkinter import messagebox

3 usages
class App(QWidget):

    def __init__(self):
        super().__init__()
        self.title = "PyQt Registration"
        self.x = 200
        self.y = 200
        self.width = 300
        self.height = 300
        self.initUI()

    1 usage
    def initUI(self):
        self.setWindowTitle(self.title)
        self.setGeometry(self.x, self.y, self.width, self.height)
        self.setWindowIcon(QIcon('pythonico.ico'))
        self.textboxlbl = QLabel("Registration", self)
        self.textboxlbl.move(120, 20)
        self.textboxlbl2 = QLabel("First name:", self)
        self.textboxlbl2.move(20, 70)
        self.textboxentry = QLineEdit(self)
        self.textboxentry.move(80, 70)
        self.textboxlbl3 = QLabel("Last name:", self)
        self.textboxlbl3.move(20, 100)
        self.textboxentry2 = QLineEdit(self)
        self.textboxentry2.move(80, 100)
        self.textboxlbl4 = QLabel("Username:", self)
        self.textboxlbl4.move(20, 130)
        self.textboxentry3 = QLineEdit(self)
        self.textboxentry3.move(80, 130)
        self.textboxlbl5 = QLabel("Password:", self)
        self.textboxlbl5.move(20, 160)
        self.textboxentry4 = QLineEdit(self)
```


GUI Labels

main.py | registration.py | Class App.py | gui_buttons.py | gui_text.py | gui_labels.py

```python
import sys
from PyQt5.QtWidgets import QWidget, QLineEdit, QMainWindow, QApplication, QLabel
from PyQt5.QtGui import QIcon

1 usage
class App(QWidget):

    def __init__(self):
        super().__init__()
        self.title = "PyQt Line Edit"
        self.x = 200
        self.y = 200
        self.width = 300
        self.height = 300
        self.initUI()

    1 usage
    def initUI(self):
        self.setWindowTitle(self.title)
        self.setGeometry(self.x, self.y, self.width, self.height)
        self.setWindowIcon(QIcon('pythonico.ico'))
        self.textboxlbl = QLabel("Hello world!", self)
        self.textboxlbl.move(100, 150)
        self.textboxlbl = QLabel("This program is written in Pycharm", self)
        self.textboxlbl.move(100, 170)
        self.show()
if __name__ == '__main__':
    app = QApplication(sys.argv)
    ex = App()
    sys.exit(app.exec_())
```


GUI Text	main.py registration.py Class App.py gui_buttons.py gui_text.py 1 import sys 2 from PyQt5.QtWidgets import QWidget, QLineEdit, QMainWindow, QApplication 3 from PyQt5.QtGui import QIcon 1 usage 4 class App(QWidget): 5 def __init__(self): 6 super().__init__() 7 self.title = "PyQt Line Edit" 8 self.x = 200 9 self.y = 200 10 self.width = 300 11 self.height = 300 12 self.initUI() 1 usage 13 def initUI(self): 14 self.setWindowTitle(self.title) 15 self.setGeometry(self.x, self.y, self.width, self.height) 16 self.setWindowIcon(QIcon('pythonico.ico')) 17 self.textbox = QLineEdit(self) 18 self.textbox.move(20, 20) 19 self.textbox.resize(280, 40) 20 self.textbox.setText("Set this text value") 21 self.show() 22 if __name__ == '__main__': 23 app = QApplication(sys.argv) 24 ex = App() 25 sys.exit(app.exec_()) 26

GUI Buttons	

main.py | registration.py | Class App.py | gui_buttons.py | gui_text.py

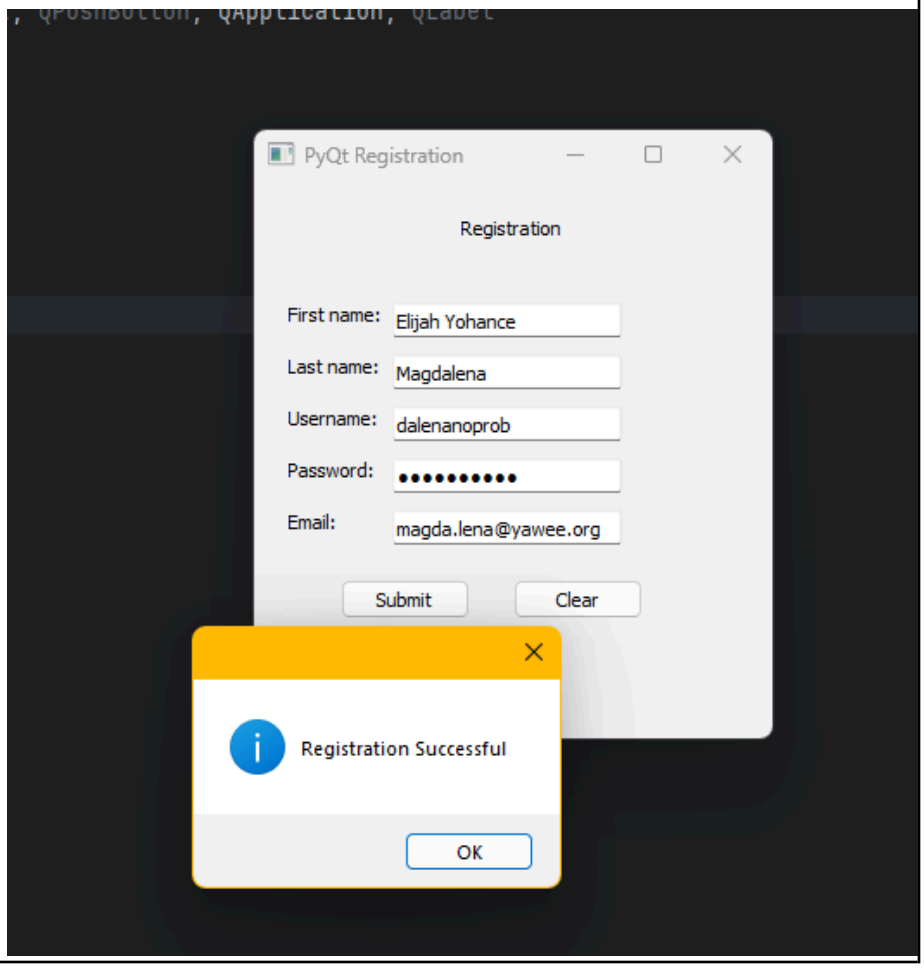
```python
import sys
from PyQt5.QtWidgets import QWidget, QPushButton, QMainWindow, QApplication
from PyQt5.QtGui import QIcon

1 usage
class App(QWidget):

    def __init__(self):
        super().__init__()
        self.title = "PyQt Button"
        self.x = 200
        self.y = 200
        self.width = 300
        self.height = 300
        self.initUI()

    1 usage
    def initUI(self):
        self.setWindowTitle(self.title)
        self.setGeometry(self.x, self.y, self.width, self.height)
        self.setWindowIcon(QIcon('pythonico.ico'))
        self.button = QPushButton('Click me', self)
        self.button.setToolTip("YES!")
        self.button.move(100, 70)
        self.button2 = QPushButton('Register', self)
        self.button2.setToolTip('Does nothing')
        self.button2.move(100, 95)
        self.show()

if __name__ == '__main__':
    app = QApplication(sys.argv)
    ex = App()
    sys.exit(app.exec_())
```

Class app	main.py registration.py Class App.py gui_buttons.py gui_text.py 1 import sys 2 from PyQt5.QtWidgets import QMainWindow, QApplication 3 from PyQt5.Qt6ui import QIcon 4 1 usage 5 class App(QMainWindow): 6 def __init__(self): 7 super().__init__() 8 self.title = "First OOP GUI" 9 self.initUI() 10 1 usage 11 def initUI(self): 12 self.setWindowTitle(self.title) 13 self.setGeometry(200, 200, 300, 300) 14 self.setWindowIcon(QIcon('pythonico.ico')) 15 self.show() 16 17 if __name__ == '__main__': 18 app = QApplication(sys.argv) 19 Main = App() 20 sys.exit(app.exec_())
INTERFACE OUTPUT	

QUESTIONS:

1. What are the common GUI Applications that general end-users such as home users, students, and office employees use? (give at least 3 and describe each)
   Web Browsers (e.g., Chrome, Firefox)
   Browsers let users access the internet, browse websites, stream media, and use web apps through a simple interface with navigation buttons and tabs.

   Word Processors (e.g., Word, Google Docs)
   Word processors allow users to create and edit documents, offering tools for formatting text, inserting images, and more, making them ideal for writing tasks.

   Spreadsheet Software (e.g., Excel, Google Sheets)
   Spreadsheets help users organize and analyze data with a grid interface, allowing for formulas, data entry, and visualizations like charts.

2. Based from your answer in question 1, why do you think home users, students, and office employees use those GUI programs?
   These groups use GUI programs like web browsers, word processors, and spreadsheets because they are user-friendly and make everyday tasks simpler. Web browsers help them access information quickly, word processors assist in creating and editing documents easily, and spreadsheets allow them to organize and analyze data without needing technical knowledge. These programs are designed to be intuitive, so users don't have to rely on complex commands.

3. How does Pycharm help developers in making GUI applications, what would be the difference if developers made GUI programs without GUI Frameworks such as Pycharm or Tkinter?
   PyCharm helps developers by providing a dedicated environment where they can write, debug, and manage code efficiently. It supports various GUI frameworks, making the development process faster and more manageable. Without frameworks like PyCharm or Tkinter, developers would need to manually handle complex tasks like drawing windows and buttons, leading to longer development times and more errors. GUI frameworks simplify creating user interfaces by providing pre-built elements.

4. What are the different platforms a GUI program may be created and deployed on? (Three is required then state why might a program be created on that specific platform)
   Windows: Often used because it's widely adopted, especially in businesses and schools. Programs created here are usually for office productivity or educational purposes.
   macOS: Popular in creative industries like design or video editing. Developers may target this platform for applications needing high-quality graphical interfaces.
   Linux: Common in development environments or for specialized software. Programs here are often created for customization or open-source projects, with more control over the system.

5. What is the purpose of app = QApplication(sys.argv), ex = App(), and sys.exit(app.exec_())?
   These lines are typical in PyQt or PySide-based GUI applications. app = QApplication(sys.argv) initializes the application and manages command-line arguments. ex = App() creates an instance of the main application window. sys.exit(app.exec_()) starts the event loop and ensures the program keeps running until the user closes the window, after which it safely exits. These commands ensure the GUI functions correctly and responds to user input.

Conclusion:

In conclusion, while developing a GUI registration system using Python, I encountered an issue related to the missing `qtwidgets` package, which had not been installed on the PC in the computer lab. This error highlighted the importance of ensuring that all required dependencies and libraries are installed before running or developing a project. Despite this challenge, I realized that creating a GUI is not as difficult as I initially thought, especially when following clear instructions from the instructor. The experience has strengthened my understanding of GUI development, showing that with the right guidance, it is a manageable and rewarding process.