GROUP 13 MEMBERS: Section: BS
CPE12S1 - B21
Dhafny Buenafe
Lauper Xavier Francisco
Christian Ed Efa

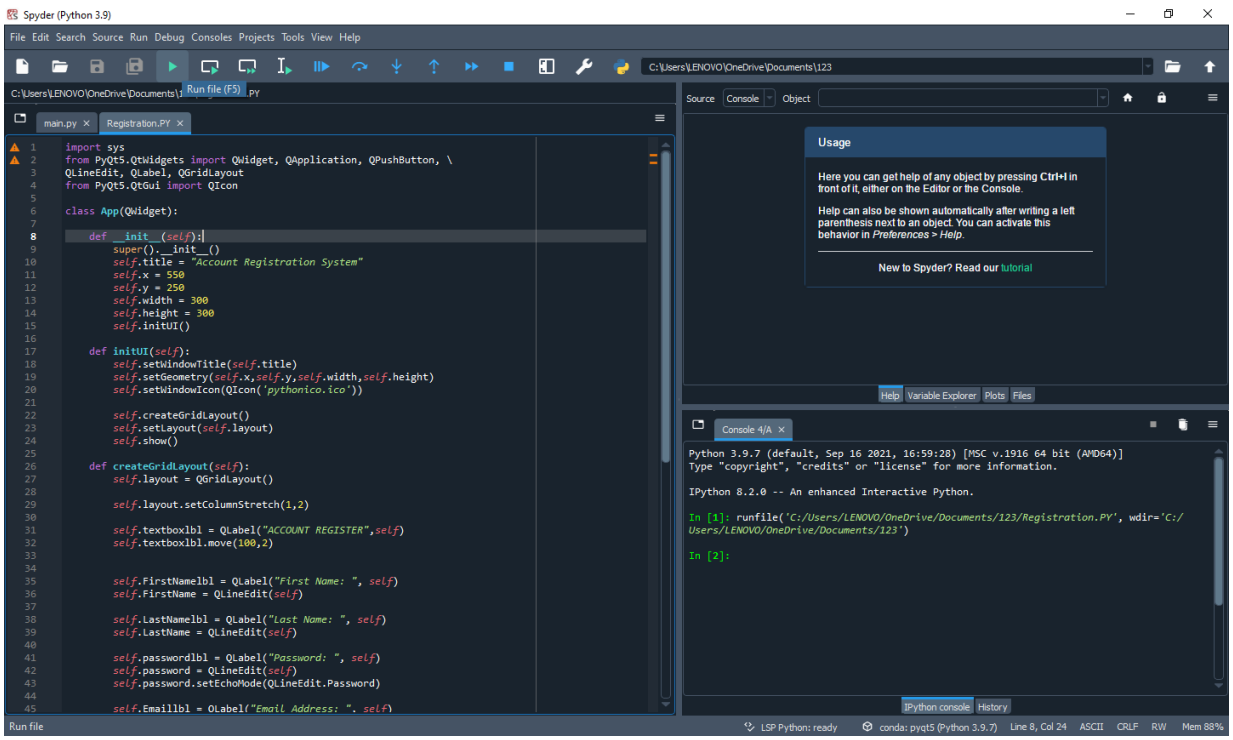
Hands-on Activity 6.1 Introduction to GUI Development using PyQt5#

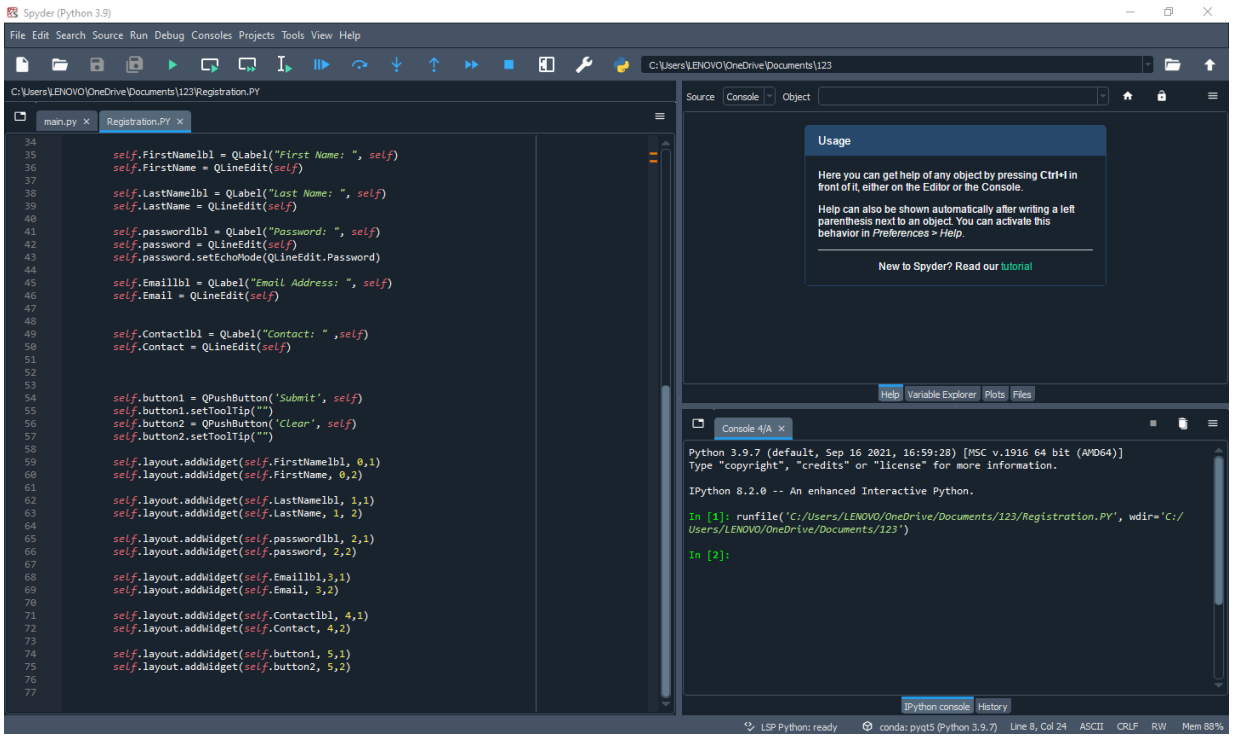
In [ ]:

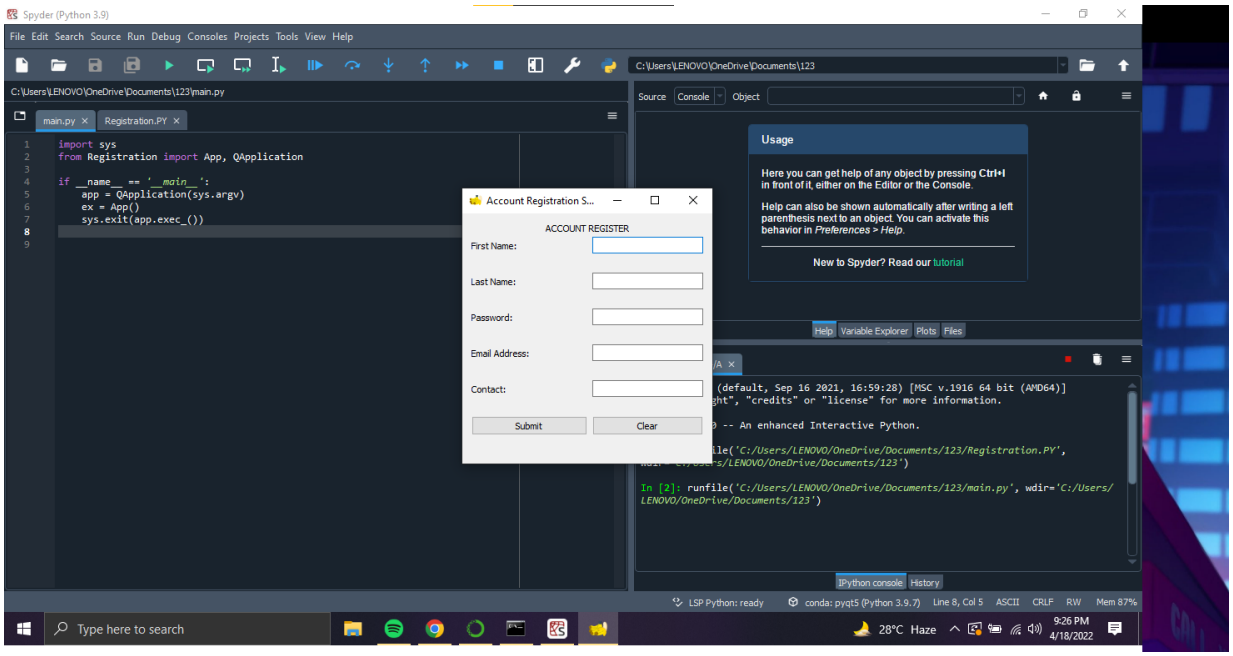
In [ ]:

Task
Create an Object-Oriented GUI Application for a simple Account Registration System with the following required information:
first name, last name, username, password, email address, contact number.

Requirements:
• The GUI must be centered on your screen.
• The GUI Components should be organized according to the order of information required using Absolute Positioning.
• The position of the components should be automatically computed based on the top component.
• All the text fields should be accompanied with their corresponding label on the left side while the text field is on
the right side.
• There should a program title other than the Window Title.
• There should be a submit button and clear button at the bottom (submit button on the left, clear button on the right).
• The program should be launched on main.py while the GUI Codes should be on a separate file called
registration.py


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\LENOVO\OneDrive\Documents\123
Run file (F5)
main.py
Registration.PY
import sys
from PyQt5.QtWidgets import QWidget, QApplication, QPushButton, \
QLineEdit, QLabel, QGridLayout
from PyQt5.QtGui import QIcon
class App(QWidget):
def __init__(self):
super().__init__()
self.title = "Account Registration System"
self.x = 550
self.y = 250
self.width = 300
self.height = 300
self.initUI()
def initUI(self):
self.setWindowTitle(self.title)
self.setGeometry(self.x,self.y,self.width,self.height)
self.setWindowIcon(QIcon('pythonico.ico'))
self.createGridLayout()
self.setLayout(self.layout)
self.show()
def createGridLayout(self):
self.layout = QGridLayout()
self.layout.setColumnStretch(1,2)
self.textboxlbl = QLabel("ACCOUNT REGISTER",self)
self.textboxlbl.move(100,2)
self.FirstNamelbl = QLabel("First Name: ", self)
self.FirstName = QLineEdit(self)
self.LastNamelbl = QLabel("Last Name: ", self)
self.LastName = QLineEdit(self)
self.passwordlbl = QLabel("Password: ", self)
self.password = QLineEdit(self)
self.password.setEchoMode(QLineEdit.Password)
self.Emaillbl = QLabel("Email Address: ", self)
Source Console Object
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
Help Variable Explorer Plots Files
Console 4/A
Python 3.9.7 (default, Sep 16 2021, 16:59:28) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 8.2.0 -- An enhanced Interactive Python.
In [1]: runfile('C:/Users/LENOVO/OneDrive/Documents/123/Registration.PY', wdir='C:/Users/LENOVO/OneDrive/Documents/123')
In [2]:
IPython console History
Run file
LSP Python: ready
conda: pyqt5 (Python 3.9.7)
Line 8, Col 24
ASCII
CRLF
RW
Mem 88%


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\LENOVO\OneDrive\Documents\123
C:\Users\LENOVO\OneDrive\Documents\123\Registration.PY
main.py
Registration.PY
self.FirstNamelbl = QLabel("First Name: ", self)
self.FirstName = QLineEdit(self)
self.LastNamelbl = QLabel("Last Name: ", self)
self.LastName = QLineEdit(self)
self.passwordlbl = QLabel("Password: ", self)
self.password = QLineEdit(self)
self.password.setEchoMode(QLineEdit.Password)
self.Emaillbl = QLabel("Email Address: ", self)
self.Email = QLineEdit(self)
self.Contactlbl = QLabel("Contact: " ,self)
self.Contact = QLineEdit(self)
self.button1 = QPushButton('Submit', self)
self.button1.setToolTip("")
self.button2 = QPushButton('Clear', self)
self.button2.setToolTip("")
self.layout.addWidget(self.FirstNamelbl, 0,1)
self.layout.addWidget(self.FirstName, 0,2)
self.layout.addWidget(self.LastNamelbl, 1,1)
self.layout.addWidget(self.LastName, 1, 2)
self.layout.addWidget(self.passwordlbl, 2,1)
self.layout.addWidget(self.password, 2,2)
self.layout.addWidget(self.Emaillbl,3,1)
self.layout.addWidget(self.Email, 3,2)
self.layout.addWidget(self.Contactlbl, 4,1)
self.layout.addWidget(self.Contact, 4,2)
self.layout.addWidget(self.button1, 5,1)
self.layout.addWidget(self.button2, 5,2)
Source Console Object
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
Help Variable Explorer Plots Files
Console 4/A
Python 3.9.7 (default, Sep 16 2021, 16:59:28) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 8.2.0 -- An enhanced Interactive Python.
In [1]: runfile('C:/Users/LENOVO/OneDrive/Documents/123/Registration.PY', wdir='C:/Users/LENOVO/OneDrive/Documents/123')
In [2]:
IPython console History
LSP Python: ready
conda: pyqt5 (Python 3.9.7)
Line 8, Col 24
ASCII
CRLF
RW
Mem 88%



In [ ]:

QUESTIONS

1. What are the common GUI Applications that general end-users such as h
ome users, students, and office employees
use? (give at least 3 and describe each)
Microsoft Windows - also called Windows and Windows OS, computer operat
ing system (OS) developed by Microsoft Corporation to run personal compu
ters (PCs).
Android -is a mobile operating system developed by Google. It is used b
y several smartphones and tablets.
Google Chrome - Google Chrome browser is an open source program for acc
essing the World Wide Web and running Web-based applications. The Google
Chrome Web browser is based on the open source Chromium project

2. Based from your answer in question 1, why do you think home users, st
udents, and office employees use those GUI
programs?
- They use GUI programs because it can help then in everyday life for ex
ample google chrome it may help students to search some of their needs i
n class or they can see all that they need in just one click only. GUI p
rgrams makes it easier to live our life because it is very useful now a
days.

3. How does PyQt5 help developers in making GUI applications, what would
be the difference if developers made GUI
programs without GUI Frameworks such as PyQt5 or Tkinter?
- Using PyQt allows you to create GUI apps using a powerful language lik

e Python. It exposes the whole Qt API. When the user is using a GUI fram eworks in programming they can cross-platform in other program so the co de will works on other application. And when using a GUI framework it wo uld be easier to program a application.

4. What are the different platforms a GUI program may be created and dep loyed on? (Three is required then state why
might a program be created on that specific platform)

- Due to its cross-platform nature, PyQt has become the most generally u sed and well-known GUI platform.Tkinter offers a common interface to the Tk GUI toolkit and its Python APIs. And then, with Kivy, developers can write code once and publish it across several platforms, while built-in OpenGL ES 2 support allows for cutting-edge graphics and design methods. Both Android and iOS applications show the Kivy GUI framework in action.

5. What is the purpose of app = QApplication(sys.argv), ex = App(), and sys.exit(app.exec_())?

- QApplication extends QGuiApplication with QWidget-specific capabilitie s. It handles widget startup and shutdown.
The sys.argv parameter contains command-line arguments. The terminal ca n execute Python programs. It allows us to regulate the scripts' launch.
- ex = App() the purpose of it is to initialize and run the program.
-The sys.exit function assures a clean exit. The environment will be to ld how the application finished.
The exec_ method contains an underscore. It is because the exec is a Py thon keyword. And therefore, exec_ was used instead.

In [ ]:

CONCLUSION

In the conclusion frameworks exist to make developers' lives simpler. They enable the establishment of certain applications or the use of certain languages. Using a Python GUI framework makes creating a user interface much simpler than coding it by hand. We should know how to properly use those frameworks and their tools so that we can surely code good applications. You'll need a variety of graphical components, including as buttons, checkboxes, toggles, dropdown lists, search fields, sliders, tooltips, and more - anything your user needs to engage with the functionality given by your app.

HONOR PLEDGE

“I accept responsibility for my role in ensuring the integrity of the wo rk submitted by the group in which I participated.”

In [ ]: