**Lab Exercise 15- Line Chart using PyQt**

**Lab Exercise: Creating a Line Chart in PyQt**

Creating a lab exercise for a line chart in PyQt involves guiding students through building a PyQt application that displays a line chart using the Matplotlib library. Below is a lab exercise that demonstrates how to create such an application:

**Objective: Create a PyQt application that displays a line chart using Matplotlib.**

Requirements:

* PyQt5: You should have PyQt5 installed.
* Matplotlib: Install Matplotlib using pip install matplotlib.

**Instructions:**

* Import the required modules:

import sys

from PyQt5.QtWidgets import QApplication, QMainWindow, QVBoxLayout, QWidget

from matplotlib.backends.backend\_qt5agg import FigureCanvasQTAgg as FigureCanvas

from matplotlib.figure import Figure

* Create a class for the PyQt application:

class LineChartApp(QMainWindow):

def \_\_init\_\_(self):

super().\_\_init\_\_()

self.initUI()

def initUI(self):

self.setGeometry(100, 100, 800, 600)

self.setWindowTitle('Line Chart Example')

central\_widget = QWidget(self)

self.setCentralWidget(central\_widget)

layout = QVBoxLayout()

central\_widget.setLayout(layout)

# Create a FigureCanvas to embed the Matplotlib plot

self.canvas = FigureCanvas(Figure())

layout.addWidget(self.canvas)

self.draw\_line\_chart() # Create and display the line chart

def draw\_line\_chart(self):

x = [1, 2, 3, 4, 5]

y = [10, 15, 13, 17, 20]

ax = self.canvas.figure.add\_subplot(111)

ax.plot(x, y, marker='o', linestyle='-')

ax.set\_xlabel('X-Axis')

ax.set\_ylabel('Y-Axis')

ax.set\_title('Line Chart Example')

self.canvas.draw()

* **Create a function to run the application:**

def run\_app():

app = QApplication(sys.argv)

ex = LineChartApp()

ex.show()

sys.exit(app.exec\_())

if \_\_name\_\_ == '\_\_main\_\_':

run\_app()

Save this code in a Python file, for example, line\_chart\_app.py.

Instruct the students to run the program, which will create a PyQt application displaying a simple line chart. Students can modify the x and y lists to experiment with different data for the line chart.

This exercise demonstrates how to integrate Matplotlib with PyQt to display a line chart, and students can use this as a starting point to explore more advanced charting and graphing capabilities in their PyQt applications.