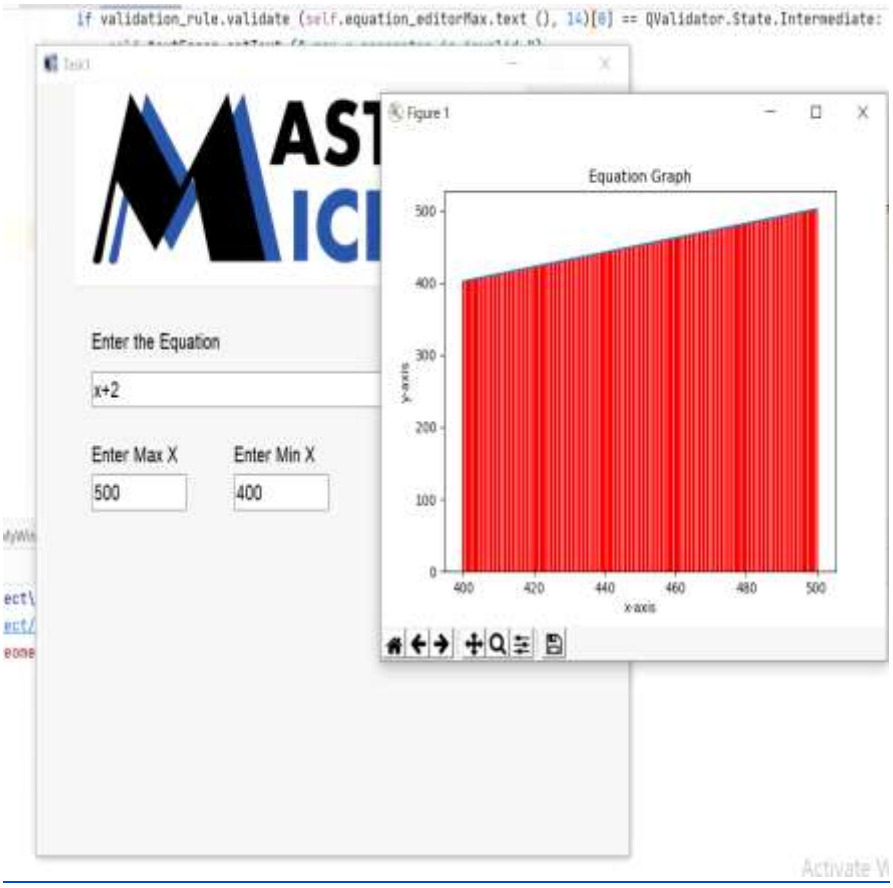
بسم الله الرحمن الرحيم

Task 1 :


Task1
MASTER MICRO
Enter the Equation
min x parameter is invalid
Enter Max X
Enter Min X
run
500
xs


Task1
MASTER MICRO
Enter the Equation
min x parameter is invalid
Enter Max X
Enter Min X
run
500
xs


Task1
Enter the Equation
x+sd43
equation must be in x format
Enter Max X
Enter Min X
run
500
400


if validation_rule.validate (self.equation_editorMax.text (), 14)[0] == QValidator.State.Intermediate:
Figure 1
Equation Graph
y-axis
x-axis
Enter the Equation
x+2
Enter Max X
Enter Min X
500
400

first i was using python 3.9 but after 1 day working i found there is no support for 3.9 for matplotlib library so i moved to 3.7 i have some issue for having 2 version but after 1 day i solved the dependency issue .

first thing in code i bulid the gui by using PySide2 lib and and set the Geometry dimension and i inserted :

-icon

-label("Macro img")

-bush putton(" run") to run the graph

-line editor ("Max X") ("Min X") ("Equation")

-Error text for showing the error reason

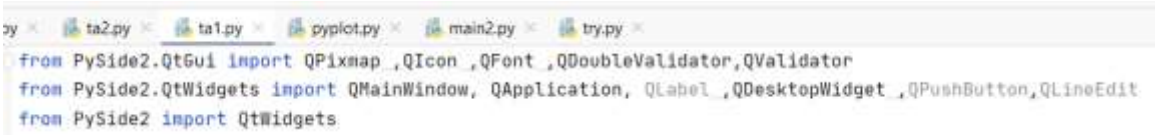


fig : the pyside2 lib

```
class MyWindow(QMainWindow):

    def __init__(self):
        super(MyWindow,self).__init__()

        self.init_ui()


    def init_ui(self):


        self.title = "Task1"
        self.setWindowTitle(self.title)
        self.setGeometry(300, 200, 250, 250)
        self.setFixedSize(750, 750)

        self.label = QtWidgets.QLabel(self)
        pixmap = QPixmap('macro.jpg')
        self.label.setPixmap(pixmap)
        self.label.setGeometry(50, -300, 1000, 800)

        self.text = QtWidgets.QLabel(self )
        self.text.setText("Enter the Equation")
        self.text.setFont(QFont('Arial', 12))
        self.text.setGeometry(70,100,200,300)

        self.equation_editor=QtWidgets.QLineEdit( self)
        self.equation_editor.setGeometry(70,280,400,35)
```

as you can see i bulid the window and setting the Title and adding Icon , Label , Equation Line editor and set the font type and the Gemometry

```
self.textMax = QtWidgets.QLabel(self )
self.textMax.setText("Enter Max X")
self.textMax.setFont(QFont('Arial', 12))
self.textMax.setGeometry(70,265,140,190)

self.equation_editorMax = QtWidgets.QLineEdit(self)
self.equation_editorMax.setGeometry(70, 380, 120, 35)
self.equation_editorMax.setFont(QFont('Arial', 12))


self.textMin = QtWidgets.QLabel(self )
self.textMin.setText("Enter Min X")
self.textMin.setFont(QFont('Arial', 12))
self.textMin.setGeometry(250,265,140,190)

self.equation_editorMin = QtWidgets.QLineEdit(self)
self.equation_editorMin.setGeometry(250, 380, 120, 35)
self.equation_editorMin.setFont(QFont('Arial', 12))

self.textError = QtWidgets.QLabel(self)
self.textError.setFont(QFont('Times', 12))
self.textError.setGeometry(480, 180, 300, 300)
self.textError.setStyleSheet("color: red")
```

```
self.run_button =QtWidgets.QPushButton(self)
self.run_button.setText("run")
self.run_button.setFont(QFont('Arial', 10))
self.run_button.setGeometry(480,350,60,30)
self.run_button.setStyleSheet("background-color: white")
self.run_button.clicked.connect(self.btn_click)
```

as you can see i made fuction pressed on " Run" button as when the user press on run the btn_click function will be called which will rexecute the graph will be showed

```
        self.setIcon()
        self.center()

    def setIcon(self):
        appIcon = QIcon("macro.JPG")
        self.setWindowIcon(appIcon)

    def center(self):
        qReact = self.frameGeometry ()
        centerpoint = QDesktopWidget ().availableGeometry ().center ()
        qReact.moveCenter (centerpoint)
        self.move (qReact.topLeft ())
```

and i also wrote 2 function first to add Macro icon when user run the gui and center to make the gui appear at the center .

so what happens when the presed click on run ?

first i will read the user input x max and x min and equation in string and then check

```
    def btn_click(self):

        equation=self.equation_editor.text()
        xmaxmiun=self.equation_editorMax.text()
        xminimum=self.equation_editorMin.text()
```

there are 2 ways to check , first

```
validation_rule = QDoubleValidator (-1000, 1000, 0)
if validation_rule.validate (self.equation_editorMax.text (), 14)[0] == QValidator.State.Invalid:
    self.textError.setText (" max x parameter is invalid ")
else:
    if validation_rule.validate (self.equation_editorMax.text (), 14)[0] == QValidator.State.Intermediate:
        self.textError.setText (" max x parameter is invalid ")
    else:

        if validation_rule.validate (self.equation_editorMin.text (), 14)[0] == QValidator.State.Invalid:
            self.textError.setText (" min x parameter is invalid ")
        else:
            if validation_rule.validate (self.equation_editorMin.text (), 14)[0] == QValidator.State.Intermediate:
                self.textError.setText (" min x parameter is invalid ")
            else:
                error_check = self.equation_check (equation, xminimum, xmaxmiun)
                print(error_check)
                if error_check == 1:
                      self.textError.setText (" equation must be in x format ")
                else:
```

i used (sequential if) why ?

to run the graph i have to make sure that the X max and X min and equation inputs are valid from the customer in case i made only 1 flag if there is error there will be no execution but in this case the user will not know the error reason , so i start first to check if Max x is valid i used QDoubleValidator and set radom limit (-1000 , 1000) so if user enter characters instead of number it will be (QValidator.State.Invalid) and get msg xparamter is invalid if user enter number not in limit it will be (QValidator.State.Invalid) get msg xparamter is invalid , so if user enter correct parameter the code will be allowed to check the x min paramter and so on .

so we checked x min & x max are valid and not empty we will call equation check function

to check that the equation is in X format like (x^2+2).

```
from sympy import *
from Equation import Expression
```

fig : used lib for equation check

```
def equation_check (self,string,stringmin,stringmax):
    mini =int(stringmin)
    maxi= int(stringmax)
    error =0
    if not string  :
        error =1
        return  error
    else:
        for i in string:
            if i.isalpha ():
                if i != "x":
                    error = 1
                    return error
        x = symbols ('x ')
        fn = (Expression (string))
        for i in range (mini, (maxi + 1)):
            x_values.append (i)
            y_values.append ((fn(i)))

        return error
```

first code take the equation in string and also X max and X min to set range to substitue the x values in the equation

if equation is empty which means customer doesnot enter any equation i return from function with error 1 else i will loop in every letter in the string if i found charcater and it isnot "x" i will return with error 1 , then after i checked the equation and equation is valid i used(symbols x) to make x as value holder(مش حرف مجهول فى دالة) not character and used fn to convert sting to equation and then i

made for lop from min value of x to max value of substitue the x in the equation

and i made two global array to hold the value of X and the result (Y) and in every step in the loop and used append method in the array to add them .

so when i return from the function (equation check) if error equal (0) i will use the two arrays (x,y) to plot and then draw the graph or if error equal 1 i will sent message to the user the equation is wrong and must be in X format .

```
import matplotlib.pyplot as plt
```

fig : library used to plot

```
else:
    self.textError.clear ()
    plt.plot (x_values, y_values)
    plt.xlabel ("x-axis")
    plt.ylabel ("y-axis")
    plt.title ("Equation Graph")
    plt.bar(x_values,y_values ,color = 'r')
    plt.show ()
```

but may some donot like this method for validation so i wrote another with idea of sending the Max and Min strings to function before checking the equation

```
equation=self.equation_editor.text()
xmaxmiun=self.equation_editorMax.text()
xminimum=self.equation_editorMin.text()
x_values_check =self.x_max_min_check(xminimum,xmaxmiun)
```

sending strings to check the values

```
def x_max_min_check(self,stringmin,stringmax):
    validation_rule = QDoubleValidator (-1000, 1000, 0)
    error = 0
    if validation_rule.validate (self.equation_editorMax.text (), 14)[0] == QValidator.State.Invalid:
        self.textError.setText (" max x parameter is invalid ")
        error =1
        return error
    if validation_rule.validate (self.equation_editorMax.text (), 14)[0] == QValidator.State.Intermediate:
        self.textError.setText (" max x parameter is invalid ")
        error =1
        return error
    if validation_rule.validate (self.equation_editorMin.text (), 14)[0] == QValidator.State.Invalid:
        self.textError.setText (" min x parameter is invalid ")
        error =1
        return error
    if validation_rule.validate (self.equation_editorMin.text (), 14)[0] == QValidator.State.Intermediate:
        self.textError.setText (" min x parameter is invalid ")
        error =1
        return error
    return error
```

and then after checking i will call equation check and if return not one i will plot if 1 i will i will retrun with text for the user "enter equation in x parameter

```
if x_values_check == 0:
    error_check = self.equation_check (equation, xminimum, xmaxmiun)
    if error_check != 1:

        self.textError.clear ()
        plt.plot (x_values, y_values)
        plt.xlabel ("x-axis")
        plt.ylabel ("y-axis")
        plt.title ("Equation Graph")
        plt.bar (x_values, y_values, color='r')
        plt.show ()
    else:
        self.textError.setText (" equation must be in x format ")
```

last thing i execute the gui to run and add sleep function so when the user try to run the Gui it will not open immediately.

```python
import sys
import time
```

fig : libraries used to show the gui

```python
app = QApplication(sys.argv)
w = MyWindow()
w.show()
time.sleep(5)
w.resize(700,700)
sys.exit(app.exec_())
```

and running the code i extra tried to convert the code to exe file to run without environment i installed pyinstaller and then i wrote the comand on cmd but i face issue

pyinstaller not support matplot and math library so exe file will be runned without showing the graph