from PyQt5.QtWidgets import \*

from PyQt5 import QtCore, QtGui

from PyQt5.QtGui import \*

from PyQt5.QtCore import \*

import sys

class Window(QWidget):

def \_\_init\_\_(self): # Для консольного приложения

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

self.setWindowTitle("Python ") # настройка заголовка

self.setGeometry(100, 100, 360, 350) # настройка геометрии

self.UiComponents() # вызов метода

self.show() # отображение всех виджетов

def UiComponents(self): # метод для виджетов (для кнопок и экрана)

self.label = QLabel(self) # создание метки (экрана)

self.label.setGeometry(5, 5, 350, 70) # настройка геометрии для метки

self.label.setWordWrap(True) # создание многострочной метки

self.label.setStyleSheet("QLabel" # установка таблицы стилей для метки

"{"

"border : 4px solid black;"

"background : white;"

"}")

self.label.setAlignment(Qt.AlignRight) # настройка выравнивания по метке

self.label.setFont(QFont('Arial', 15)) # шрифта

push1 = QPushButton("1", self) # создание кнопки 1

push1.setGeometry(5, 150, 80, 40) # геометрия кнопки

#дальше абсолютно так же создаюстся кнопки 2 3 4 5 6 7 8 9 0 = + - / \*

push2 = QPushButton("2", self)

push2.setGeometry(95, 150, 80, 40)

push3 = QPushButton("3", self)

push3.setGeometry(185, 150, 80, 40)

push4 = QPushButton("4", self)

push4.setGeometry(5, 200, 80, 40)

push5 = QPushButton("5", self)

push5.setGeometry(95, 200, 80, 40)

push6 = QPushButton("5", self)

push6.setGeometry(185, 200, 80, 40)

push7 = QPushButton("7", self)

push7.setGeometry(5, 250, 80, 40)

push8 = QPushButton("8", self)

push8.setGeometry(95, 250, 80, 40)

push9 = QPushButton("9", self)

push9.setGeometry(185, 250, 80, 40)

push0 = QPushButton("0", self)

push0.setGeometry(5, 300, 80, 40)

push\_equal = QPushButton("=", self)

push\_equal.setGeometry(275, 300, 80, 40)

c\_effect = QGraphicsColorizeEffect() # adding equal кнопка цветовой эффект

c\_effect.setColor(Qt.blue)

push\_equal.setGraphicsEffect(c\_effect)

push\_plus = QPushButton("+", self)

push\_plus.setGeometry(275, 250, 80, 40)

push\_minus = QPushButton("-", self)

push\_minus.setGeometry(275, 200, 80, 40)

push\_mul = QPushButton("\*", self)

push\_mul.setGeometry(275, 150, 80, 40)

push\_div = QPushButton("/", self)

push\_div.setGeometry(185, 300, 80, 40)

push\_point = QPushButton(".", self)

push\_point.setGeometry(95, 300, 80, 40)

push\_clear = QPushButton("Clear", self) # создание кнопки Очистить

push\_clear.setGeometry(5, 100, 200, 40)

push\_del = QPushButton("Del", self) # создание кнопки удалить

push\_del.setGeometry(210, 100, 145, 40)

# добавление действия к каждой кнопке

push\_minus.clicked.connect(self.action\_minus)

push\_equal.clicked.connect(self.action\_equal)

push0.clicked.connect(self.action0)

push1.clicked.connect(self.action1)

push2.clicked.connect(self.action2)

push3.clicked.connect(self.action3)

push4.clicked.connect(self.action4)

push5.clicked.connect(self.action5)

push6.clicked.connect(self.action6)

push7.clicked.connect(self.action7)

push8.clicked.connect(self.action8)

push9.clicked.connect(self.action9)

push\_div.clicked.connect(self.action\_div)

push\_mul.clicked.connect(self.action\_mul)

push\_plus.clicked.connect(self.action\_plus)

push\_point.clicked.connect(self.action\_point)

push\_clear.clicked.connect(self.action\_clear)

push\_del.clicked.connect(self.action\_del)

def action\_equal(self): #кнопка равно

equation = self.label.text() # получить текст метки

try:

ans = eval(equation) #получение ответа(находим ответ)

self.label.setText(str(ans)) # установка текста на метку (выводим ответ)

except:

self.label.setText("Wrong Input") # установка текста на метку (не получилось решить выводим ошибку)

# добавление текста метки для каждого элемента

def action\_plus(self):

text = self.label.text() # принимаем значение с поля

self.label.setText(text + " + ") # прибавляем нужный знак или число, при нажатии на него

def action\_minus(self):

text = self.label.text()

self.label.setText(text + " - ")

def action\_div(self):

text = self.label.text()

self.label.setText(text + " / ")

def action\_mul(self):

text = self.label.text()

self.label.setText(text + " \* ")

def action\_point(self):

text = self.label.text()

self.label.setText(text + ".")

def action0(self):

text = self.label.text()

self.label.setText(text + "0")

def action1(self):

text = self.label.text()

self.label.setText(text + "1")

def action2(self):

text = self.label.text()

self.label.setText(text + "2")

def action3(self):

text = self.label.text()

self.label.setText(text + "3")

def action4(self):

text = self.label.text()

self.label.setText(text + "4")

def action5(self):

text = self.label.text()

self.label.setText(text + "5")

def action6(self):

text = self.label.text()

self.label.setText(text + "6")

def action7(self):

text = self.label.text()

self.label.setText(text + "7")

def action8(self):

text = self.label.text()

self.label.setText(text + "8")

def action9(self):

text = self.label.text()

self.label.setText(text + "9")

def action\_clear(self):

self.label.setText("")

# очистка одной цифры

def action\_del(self):

text = self.label.text()

print(text[:len(text)-1])

self.label.setText(text[:len(text)-1])

App = QApplication(sys.argv) # создание приложение pyqt5

window = Window() # создание экземпляр нашего окна

sys.exit(App.exec()) #запуск приложение