**Министерство цифрового развития связи и массовых коммуникаций Российской Федерации.**

**Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования**

**"Московский технический университет связи и информатики"**

**Кафедра математическая кибернетика и информационные технологии**

**Отчет по лабораторной работе № 5**

**“Калькулятор”**

**по дисциплине**

**"Введение в ИТ"**

Выполнил: студент гр. БИН2001 Варламов М. Д.

Проверил: Аршинов Е.А.

Москва, 2021 г.

import sys  
import sympy  
from PyQt5.QtWidgets import QApplication, QWidget, QGridLayout, QPushButton, QLineEdit  
   
class Window(QWidget):  
 def \_\_init\_\_(self):  
 super(Window, self).\_\_init\_\_()  
 grid = QGridLayout()  
 self.calcText = QLineEdit()  
 self.setLayout(grid)  
 grid.addWidget(self.calcText, 0, 0, 1, 4)  
 names = ['Cls', 'Bck', '(', ')',  
 '7', '8', '9', '/',  
 '4', '5', '6', '\*',  
 '1', '2', '3', '-',  
 '0', '.', '=', '+']  
 positions = [(i+1,j) for i in range(6) for j in range(4)]  
 buttons = []  
 for position, name in zip(positions, names):  
 button = QPushButton(name)  
 buttons.append(button)  
 grid.addWidget(button, \*position)  
   
 for keyindx in range(0, len(names)):  
 buttons[keyindx].clicked.connect(lambda ch, text=names[keyindx]: self.butonact(text))  
   
 self.move(300, 150)  
 self.setWindowTitle('Calculator')  
 self.show()  
   
 def make\_calculate(self):  
 try:  
 return eval(str(sympy.sympify(self.calcText.text(), evaluate=True)))  
 except Exception as e:  
 print(e)  
 return 'ERROR'  
   
 def butonact(self, param):  
 nowLine = self.calcText.text()  
 if param in ['7', '8', '9', '/',  
 '4', '5', '6', '\*',  
 '1', '2', '3', '-',  
 '0', '.', '+', '(', ')']:  
 if len(nowLine) > 0:  
 if nowLine[-1] in ['/', '\*', '-', '.', '+'] and param in ['/', '\*', '-', '.', '+', '=']:  
 pass  
 else:   
 self.calcText.setText(nowLine + str(param))  
 else:   
 self.calcText.setText(nowLine + str(param))  
 elif param in ['Cls', 'Bck', '=']:  
 if param == 'Cls':  
 self.calcText.setText('')  
 elif param == 'Bck':  
 if len(nowLine) > 0:  
 self.calcText.setText(nowLine[:-1])  
 elif param == '=':  
 res = str(self.make\_calculate())  
 self.calcText.setText(res)  
   
   
app = QApplication(sys.argv)  
win = Window()  
win.show()   
sys.exit(app.exec())

На рисунке 1 представлены результаты вычислений, на рисунке 2 результаты при вводе ошибочных данных

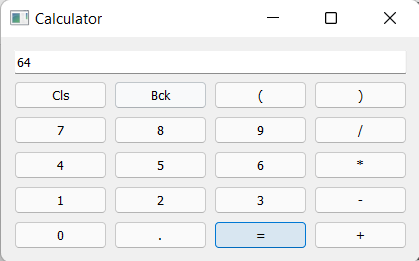
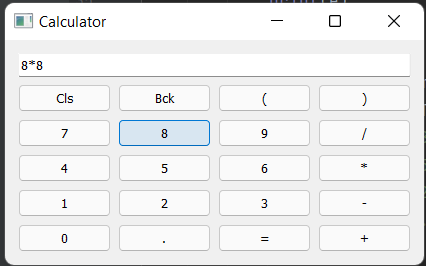


Рисунок 1

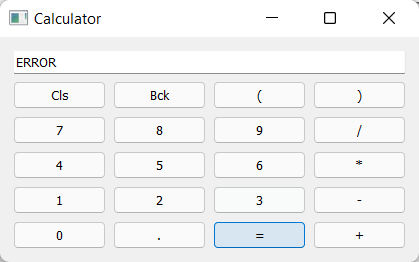
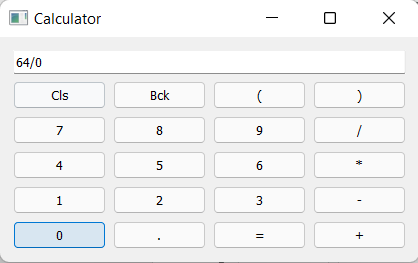


Рисунок 2