

**Ордена трудового Красного Знамени  
Федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования  
Московский технический университет связи и информатики  
Кафедра «Сети Связи и Системы Коммутации»**

**Лабораторная работа № 6  
«Создание приложения калькулятор»**

Группа: БИН-2004  
Выполнил: Сорокин П. В.  
Проверил:  
Дата: 22.11.2022

Москва 2022

**Цель:** Создание оконного приложения калькулятор. Обработать все возможные исключения. Добавить кнопку для добавления плавающей точки. Добавить кнопки для математических операций вычитания, умножения, деления, нахождения факториала и квадратного корня. Создать для этих кнопок методы-обработчики.

### Исходный код:

```
from PyQt5 import QtCore, QtGui, QtWidgets
from PyQt5.QtWidgets import QMessageBox
from math import factorial
import math

class Ui_MainWindow(object):
    def setupUi(self, MainWindow):
        MainWindow.setObjectName("MainWindow")
        MainWindow.setFixedSize(412, 485)
        MainWindow.setSizeIncrement(QtCore.QSize(0, 400))
        self.centralwidget = QtWidgets.QWidget(MainWindow)
        self.centralwidget.setObjectName("centralwidget")
        self.label_result = QtWidgets.QLabel(self.centralwidget)
        self.label_result.setGeometry(QtCore.QRect(0, 0, 259, 87))
        font = QtGui.QFont()
        font.setFamily("Segoe UI")
        font.setPointSize(15)
        font.setBold(False)
        font.setWeight(50)
        self.label_result.setFont(font)
        self.label_result.setStyleSheet("color: rgb(255, 255, 255);\n"
                                       "border-color: rgb(255, 255, 127);\n"
                                       "background-color: rgb(158, 158, 158);\n"
                                       "border-radius:20px;")
        self.label_result.setText("")
        self.label_result.setObjectName("label_result")
        self.btn_1 = QtWidgets.QPushButton(self.centralwidget)
        self.btn_1.setGeometry(QtCore.QRect(0, 236, 100, 70))
        font = QtGui.QFont()
        font.setPointSize(12)
        font.setBold(True)
        font.setWeight(75)
        self.btn_1.setFont(font)
        self.btn_1.setStyleSheet("color: rgb(255, 255, 255);\n"
                                "background-color: rgb(0, 0, 0);\n"
                                "border-radius:20px;")
        self.btn_1.setObjectName("btn_1")
        self.btn_3 = QtWidgets.QPushButton(self.centralwidget)
        self.btn_3.setGeometry(QtCore.QRect(206, 236, 100, 70))
        font = QtGui.QFont()
        font.setPointSize(12)
        font.setBold(True)
        font.setWeight(75)
        self.btn_3.setFont(font)
        self.btn_3.setStyleSheet("color: rgb(255, 255, 255);\n"
                                "background-color: rgb(0, 0, 0);\n"
                                "border-radius:20px;")
        self.btn_3.setObjectName("btn_3")
        self.btn_2 = QtWidgets.QPushButton(self.centralwidget)
```

```

self.btn_2.setGeometry(QtCore.QRect(103, 236, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_2.setFont(font)
self.btn_2.setStyleSheet("color: rgb(255, 255, 255);\n"
                        "background-color: rgb(0, 0, 0);\n"
                        "border-radius:20px;")

self.btn_2.setObjectName("btn_2")
self.btn_4 = QtWidgets.QPushButton(self.centralwidget)
self.btn_4.setGeometry(QtCore.QRect(0, 163, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_4.setFont(font)
self.btn_4.setStyleSheet("color: rgb(255, 255, 255);\n"
                        "background-color: rgb(0, 0, 0);\n"
                        "border-radius:20px;")

self.btn_4.setObjectName("btn_4")
self.btn_5 = QtWidgets.QPushButton(self.centralwidget)
self.btn_5.setGeometry(QtCore.QRect(103, 163, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_5.setFont(font)
self.btn_5.setStyleSheet("color: rgb(255, 255, 255);\n"
                        "background-color: rgb(0, 0, 0);\n"
                        "border-radius:20px;\n"
                        "")

self.btn_5.setObjectName("btn_5")
self.btn_6 = QtWidgets.QPushButton(self.centralwidget)
self.btn_6.setGeometry(QtCore.QRect(206, 163, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_6.setFont(font)
self.btn_6.setStyleSheet("color: rgb(255, 255, 255);\n"
                        "background-color: rgb(0, 0, 0);\n"
                        "border-radius:20px;")

self.btn_6.setObjectName("btn_6")
self.btn_7 = QtWidgets.QPushButton(self.centralwidget)
self.btn_7.setGeometry(QtCore.QRect(0, 90, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_7.setFont(font)
self.btn_7.setStyleSheet("color: rgb(255, 255, 255);\n"
                        "background-color: rgb(0, 0, 0);\n"
                        "border-color: rgb(252, 252, 252);\n"
                        "border-radius:20px;")

self.btn_7.setObjectName("btn_7")
self.btn_8 = QtWidgets.QPushButton(self.centralwidget)
self.btn_8.setGeometry(QtCore.QRect(103, 90, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_8.setFont(font)
self.btn_8.setStyleSheet("color: rgb(255, 255, 255);\n"

```

```

        "background-color: rgb(0, 0, 0);\n"
        "border-radius:20px;\n"
        """)
self.btn_8.setObjectName("btn_8")
self.btn_9 = QtWidgets.QPushButton(self.centralwidget)
self.btn_9.setGeometry(QtCore.QRect(206, 90, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setItalic(False)
font.setUnderline(False)
font.setWeight(75)
font.setStrikeOut(False)
font.setKerning(True)
self.btn_9.setFont(font)
self.btn_9.setStyleSheet("color: rgb(255, 255, 255);\n"
        "border-color: rgb(255, 169, 169);\n"
        "background-color: rgb(0, 0, 0);\n"
        "border-radius:20px;")

self.btn_9.setCheckable(False)
self.btn_9.setObjectName("btn_9")
self.btn_zero = QtWidgets.QPushButton(self.centralwidget)
self.btn_zero.setGeometry(QtCore.QRect(0, 309, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_zero.setFont(font)
self.btn_zero.setStyleSheet("color: rgb(255, 255, 255);\n"
        "background-color: rgb(0, 0, 0);\n"
        "border-radius:20px;")

self.btn_zero.setObjectName("btn_zero")
self.btn_equal = QtWidgets.QPushButton(self.centralwidget)
self.btn_equal.setGeometry(QtCore.QRect(207, 382, 205, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_equal.setFont(font)
self.btn_equal.setStyleSheet("color: rgb(255, 255, 255);\n"
        "background-color: rgb(255, 170,
127);\n"
        "border-radius:20px;")

self.btn_equal.setObjectName("btn_equal")
self.btn_mult = QtWidgets.QPushButton(self.centralwidget)
self.btn_mult.setGeometry(QtCore.QRect(309, 236, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_mult.setFont(font)
self.btn_mult.setStyleSheet("color: rgb(255, 255, 255);\n"
        "background-color: rgb(0, 0, 0);\n"
        "border-radius:20px;")

self.btn_mult.setObjectName("btn_mult")
self.btn_divide = QtWidgets.QPushButton(self.centralwidget)
self.btn_divide.setGeometry(QtCore.QRect(309, 309, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_divide.setFont(font)
self.btn_divide.setStyleSheet("color: rgb(255, 255, 255);\n"
        "background-color: rgb(0, 0, 0);\n"

```

```

        "border-radius:20px;")
self.btn_divide.setObjectName("btn_divide")
self.btn_minus = QtWidgets.QPushButton(self.centralwidget)
self.btn_minus.setGeometry(QtCore.QRect(309, 163, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_minus.setFont(font)
self.btn_minus.setStyleSheet("color: rgb(255, 255, 255);\n"
                             "background-color: rgb(0, 0, 0);\n"
                             "border-radius:20px;")

self.btn_minus.setObjectName("btn_minus")
self.btn_plus = QtWidgets.QPushButton(self.centralwidget)
self.btn_plus.setGeometry(QtCore.QRect(309, 90, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_plus.setFont(font)
self.btn_plus.setStyleSheet("color: rgb(255, 255, 255);\n"
                             "background-color: rgb(0, 0, 0);\n"
                             "border-radius:20px;")

self.btn_plus.setObjectName("btn_plus")
self.btn_dot = QtWidgets.QPushButton(self.centralwidget)
self.btn_dot.setGeometry(QtCore.QRect(0, 382, 204, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_dot.setFont(font)
self.btn_dot.setStyleSheet("color: rgb(255, 255, 255);\n"
                             "background-color: rgb(0, 0, 0);\n"
                             "border-radius:20px;")

self.btn_dot.setObjectName("btn_dot")
self.btn_delete = QtWidgets.QPushButton(self.centralwidget)
self.btn_delete.setGeometry(QtCore.QRect(262, 0, 150, 87))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_delete.setFont(font)
self.btn_delete.setStyleSheet("color: rgb(255, 255, 255);\n"
                             "background-color: rgb(255, 170,
127);\n"
                             "border-radius:20px;")

self.btn_delete.setObjectName("btn_delete")
self.btn_sqrt = QtWidgets.QPushButton(self.centralwidget)
self.btn_sqrt.setGeometry(QtCore.QRect(206, 309, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn_sqrt.setFont(font)
self.btn_sqrt.setStyleSheet("color: rgb(255, 255, 255);\n"
                             "background-color: rgb(0, 0, 0);\n"
                             "border-radius:20px;")

self.btn_sqrt.setObjectName("btn_sqrt")
self.btn_faktorial = QtWidgets.QPushButton(self.centralwidget)
self.btn_faktorial.setGeometry(QtCore.QRect(103, 309, 100, 70))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)

```

```

self.btn_faktorial.setFont(font)
self.btn_faktorial.setStyleSheet("color: rgb(255, 255, 255);\n"
                                   "background-color: rgb(0, 0, 0);\n"
                                   "border-radius:20px;")

self.btn_faktorial.setObjectName("btn_faktorial")
MainWindow.setCentralWidget(self.centralwidget)
self.menubar = QtWidgets.QMenuBar(MainWindow)
self.menubar.setGeometry(QtCore.QRect(0, 0, 428, 26))
self.menubar.setObjectName("menubar")
MainWindow.setMenuBar(self.menubar)
self.statusbar = QtWidgets.QStatusBar(MainWindow)
self.statusbar.setObjectName("statusbar")
MainWindow.setStatusBar(self.statusbar)

self.retranslateUi(MainWindow)
QtCore.QMetaObject.connectSlotsByName(MainWindow)

self.add_functions()

self.is_equal = False

def retranslateUi(self, MainWindow):
    _translate = QtCore.QCoreApplication.translate
    MainWindow.setWindowTitle(_translate("MainWindow", "Калькулятор"))
    self.label_result.setText(_translate("MainWindow", "")) #удалили 0
    self.btn_1.setText(_translate("MainWindow", "1"))
    self.btn_3.setText(_translate("MainWindow", "3"))
    self.btn_2.setText(_translate("MainWindow", "2"))
    self.btn_4.setText(_translate("MainWindow", "4"))
    self.btn_5.setText(_translate("MainWindow", "5"))
    self.btn_6.setText(_translate("MainWindow", "6"))
    self.btn_7.setText(_translate("MainWindow", "7"))
    self.btn_8.setText(_translate("MainWindow", "8"))
    self.btn_9.setText(_translate("MainWindow", "9"))
    self.btn_zero.setText(_translate("MainWindow", "0"))
    self.btn_equal.setText(_translate("MainWindow", "="))
    self.btn_mult.setText(_translate("MainWindow", "*"))
    self.btn_divide.setText(_translate("MainWindow", "/"))
    self.btn_minus.setText(_translate("MainWindow", "-"))
    self.btn_plus.setText(_translate("MainWindow", "+"))
    self.btn_dot.setText(_translate("MainWindow", "."))
    self.btn_delete.setText(_translate("MainWindow", "delete"))
    self.btn_sqrt.setText(_translate("MainWindow", "√"))
    self.btn_faktorial.setText(_translate("MainWindow", "!n"))

    def add_functions(self):
        self.btn_zero.clicked.connect(lambda:
self.write_number(self.btn_zero.text()))
        self.btn_1.clicked.connect(lambda:
self.write_number(self.btn_1.text()))
        self.btn_2.clicked.connect(lambda:
self.write_number(self.btn_2.text()))
        self.btn_3.clicked.connect(lambda:
self.write_number(self.btn_3.text()))
        self.btn_4.clicked.connect(lambda:
self.write_number(self.btn_4.text()))
        self.btn_5.clicked.connect(lambda:
self.write_number(self.btn_5.text()))
        self.btn_6.clicked.connect(lambda:
self.write_number(self.btn_6.text()))
        self.btn_7.clicked.connect(lambda:
self.write_number(self.btn_7.text()))
        self.btn_8.clicked.connect(lambda:

```

```

self.write_number(self.btn_8.text())
    self.btn_9.clicked.connect(lambda:
self.write_number(self.btn_9.text()))
    self.btn_mult.clicked.connect(lambda:
self.write_number(self.btn_mult.text()))
    self.btn_divide.clicked.connect(lambda:
self.write_number(self.btn_divide.text()))
    self.btn_minus.clicked.connect(lambda:
self.write_number(self.btn_minus.text()))
    self.btn_plus.clicked.connect(lambda:
self.write_number(self.btn_plus.text()))
    self.btn_dot.clicked.connect(lambda:
self.write_number(self.btn_dot.text()))

    self.btn_delete.clicked.connect(self.Delete)

    self.btn_faktorial.clicked.connect(self.Faktorial)

    self.btn_sqrt.clicked.connect(self.SQRT)

    self.btn_equal.clicked.connect(self.results)

def write_number(self, number):
    if self.label_result.text() == "" or self.is_equal:
        self.label_result.setText(number)
        self.is_equal = False
    else:
        self.label_result.setText (self.label_result.text() + number)

def Faktorial(self):
    if not self.is_equal:
        n = int((self.label_result.text()))
        if n>=0:
            n = factorial(n)
        else:
            n = ""
            er3 = QMessageBox()
            er3.setWindowTitle("ОШИБКА")
            er3.setText("Факториал определен только для натуральных чисел
и нуля.")

            er3.setIcon(QMessageBox.Warning)

            er3.exec()
        self.label_result.setText(str(n))
        self.is_equal = True

def SQRT(self):
    if not self.is_equal:
        s = int((self.label_result.text()))
        if s>=0:
            s = math.sqrt(s)
        else:
            s = ""
            er3 = QMessageBox()
            er3.setWindowTitle("ОШИБКА")
            er3.setText("Нельзя вычислить корень отрицательного числа.")
            er3.setIcon(QMessageBox.Warning)

            er3.exec()
        self.label_result.setText(str(s))
        self.is_equal = True

def Delete(self):

```

```

a = self.label_result.text()
a = a[:-1]
self.label_result.setText(a)

def results(self):
    if not self.is_equal:
        try:
            res = eval(self.label_result.text()) # eval-обрабатывает
строку в математическое уравнение
        except ZeroDivisionError:
            res = ""
            er2 = QMessageBox()
            er2.setWindowTitle("ОШИБКА")
            er2.setText("На ноль делить нельзя")
            er2.setIcon(QMessageBox.Warning)

            er2.exec()
            self.label_result.setText(str(res))
            self.is_equal = True
        else:
            error = QMessageBox()
            error.setWindowTitle("ОШИБКА")
            error.setText("Сейчас это действие нельзя выполнить")
            error.setIcon(QMessageBox.Warning)

            error.setStandardButtons(QMessageBox.Ok|QMessageBox.Reset)
            error.setInformativeText("Если попробуете снова, калькулятор
закроется")
            error.setDetailedText("Если хотите ввести новое математическое
уравнение, начните его писать сразу после получения значений предыдущего
математического уравнение.")

            error.buttonClicked.connect(self.popup_action)

            error.exec()

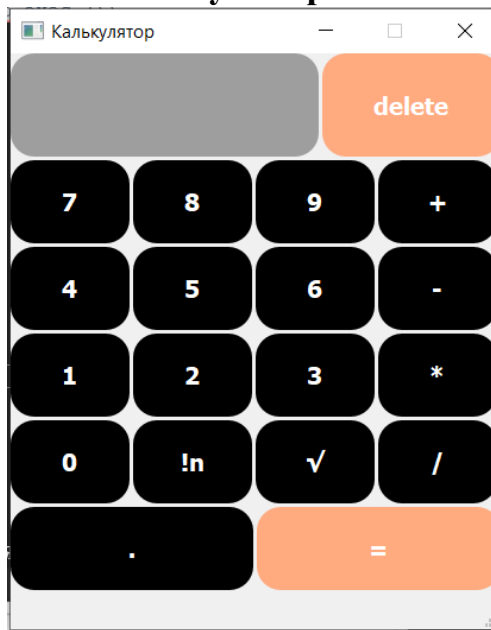
def popup_action(self, btn):
    if btn.text() == "Ok":
        print("Print Ok")
    elif btn.text() == "Reset":
        self.label_result.setText("")
        self.is_equal = False

if __name__ == "__main__":
    import sys
    app = QtWidgets.QApplication(sys.argv)
    MainWindow = QtWidgets.QMainWindow()
    ui = Ui_MainWindow()
    ui.setupUi(MainWindow)
    MainWindow.show()
    sys.exit(app.exec_())

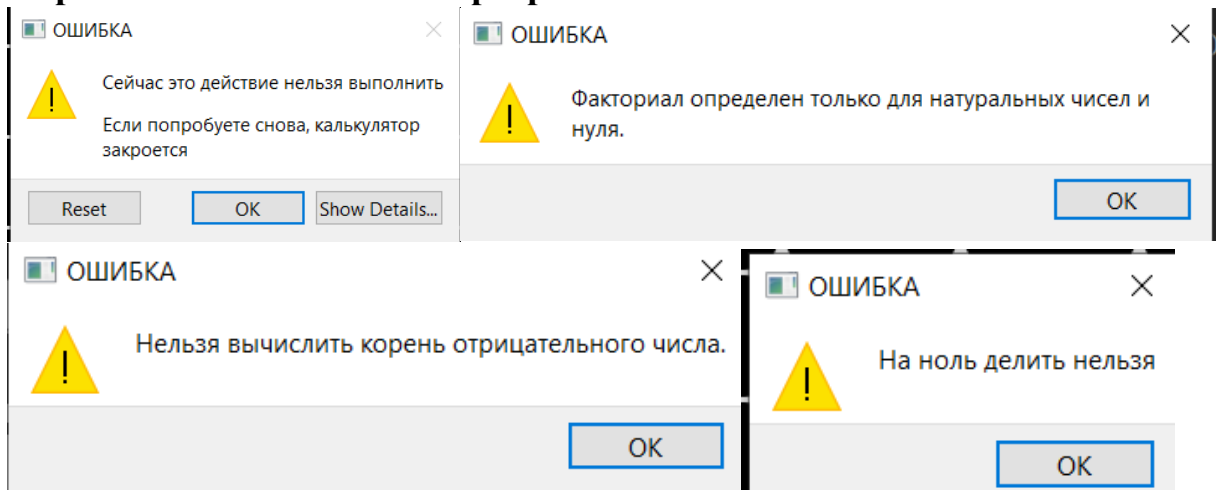
```



### Окно калькулятора:



### Обработка исключений в программе:



**Вывод:** В ходе данной лабораторной работы, я создал приложение-калькулятор, с помощью которого можно проводить различные простые математические вычисления.