**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 математической кибернетики и информационных технологий

**Отчет по лабораторной работе 4**

на тему «Библиотека PyQt5»

по дисциплине «Введение в ИТ»

Выполнил: студент группы БВТ2105

Полунин Илья Михайлович

Проверил:

Мкртчян Грач Маратович

Москва

2021

Цель и задачи работы

**Цель**: научиться создавать простые приложения пользуясь библиотекой PyQt5

**Задача:** Создание калькулятор с помощью Python и библиотеки PyQt5

**Содержание отчета**

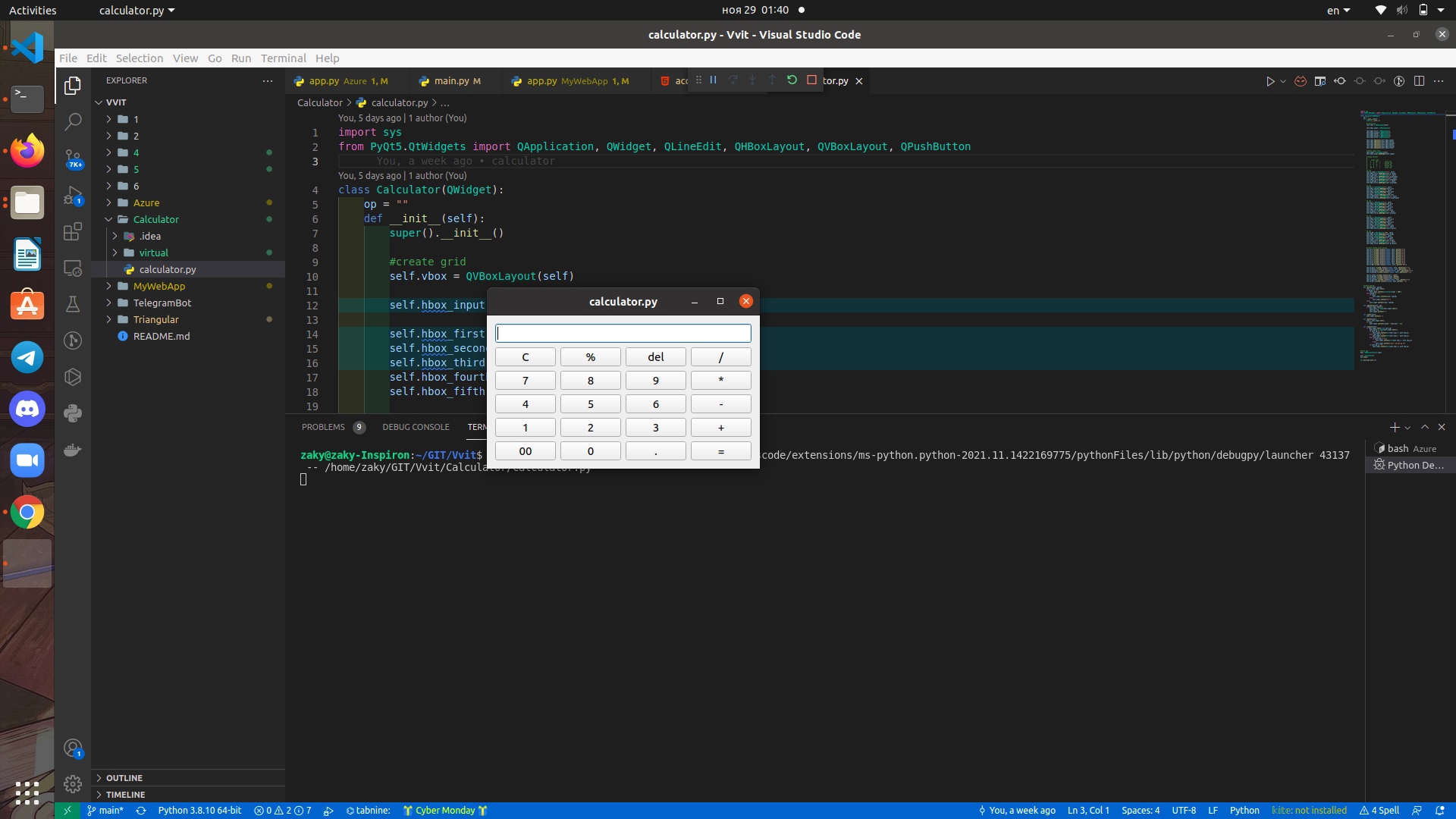
1. Титульный лист
2. Цель и техническое задание
3. Решение поставленной задачи
4. Вывод

**Выполнение:**

Листинг кода файла calculator.py:

import sys  
from PyQt5.QtWidgets import QApplication, QWidget, QLineEdit, QHBoxLayout, QVBoxLayout, QPushButton  
  
  
class Calculator(QWidget):  
 def \_\_init\_\_(self):  
 super(Calculator, self).\_\_init\_\_()  
  
 # задаем строки  
 self.vbox = QVBoxLayout(self)  
 self.hbox\_input = QHBoxLayout()  
 self.hbox\_first = QHBoxLayout()  
 self.hbox\_operations = QHBoxLayout()  
 self.hbox\_result = QHBoxLayout()  
  
 self.vbox.addLayout(self.hbox\_input)  
 self.vbox.addLayout(self.hbox\_first)  
 self.vbox.addLayout(self.hbox\_operations)  
 self.vbox.addLayout(self.hbox\_result)  
  
 # создаем поле ввода  
 self.input = QLineEdit(self)  
 self.hbox\_input.addWidget(self.input)  
  
 # создаем кнопки  
 self.b\_1 = QPushButton("1", self)  
 self.hbox\_first.addWidget(self.b\_1)  
 self.b\_2 = QPushButton("2", self)  
 self.hbox\_first.addWidget(self.b\_2)  
 self.b\_3 = QPushButton("3", self)  
 self.hbox\_first.addWidget(self.b\_3)  
 self.b\_4 = QPushButton("4", self)  
 self.hbox\_first.addWidget(self.b\_4)  
 self.b\_5 = QPushButton("5", self)  
 self.hbox\_first.addWidget(self.b\_5)  
 self.b\_6 = QPushButton("6", self)  
 self.hbox\_first.addWidget(self.b\_6)  
 self.b\_7 = QPushButton("7", self)  
 self.hbox\_first.addWidget(self.b\_7)  
 self.b\_8 = QPushButton("8", self)  
 self.hbox\_first.addWidget(self.b\_8)  
 self.b\_9 = QPushButton("9", self)  
 self.hbox\_first.addWidget(self.b\_9)  
 self.b\_0 = QPushButton("0", self)  
 self.hbox\_first.addWidget(self.b\_0)  
  
 self.b\_plus = QPushButton("+", self)  
 self.hbox\_operations.addWidget(self.b\_plus)  
 self.b\_minus = QPushButton("-", self)  
 self.hbox\_operations.addWidget(self.b\_minus)  
 self.b\_multiply = QPushButton("\*", self)  
 self.hbox\_operations.addWidget(self.b\_multiply)  
 self.b\_divide = QPushButton("/", self)  
 self.hbox\_operations.addWidget(self.b\_divide)  
  
 self.b\_result = QPushButton("=", self)  
 self.hbox\_result.addWidget(self.b\_result)  
  
 self.b\_reset = QPushButton("reset", self)  
 self.hbox\_result.addWidget(self.b\_reset)  
  
 # задаем действия для кнопок  
 self.b\_1.clicked.connect(lambda: self.\_button("1"))  
 self.b\_2.clicked.connect(lambda: self.\_button("2"))  
 self.b\_3.clicked.connect(lambda: self.\_button("3"))  
 self.b\_4.clicked.connect(lambda: self.\_button("4"))  
 self.b\_5.clicked.connect(lambda: self.\_button("5"))  
 self.b\_6.clicked.connect(lambda: self.\_button("6"))  
 self.b\_7.clicked.connect(lambda: self.\_button("7"))  
 self.b\_8.clicked.connect(lambda: self.\_button("8"))  
 self.b\_9.clicked.connect(lambda: self.\_button("9"))  
 self.b\_0.clicked.connect(lambda: self.\_button("0"))  
  
 self.b\_plus.clicked.connect(lambda: self.\_operation("+"))  
 self.b\_minus.clicked.connect(lambda: self.\_operation("-"))  
 self.b\_multiply.clicked.connect(lambda: self.\_operation("\*"))  
 self.b\_divide.clicked.connect(lambda: self.\_operation("/"))  
  
 self.b\_reset.clicked.connect(self.\_reset)  
  
 self.b\_result.clicked.connect(self.\_result)  
  
 # действие при нажатии на кнопки-цифры  
 def \_button(self, param):  
 line = self.input.text()  
 self.input.setText(line + param)  
  
 # действие при нажатии на кнопки-операции  
 def \_operation(self, op):  
 if (self.input.text()):  
 self.num\_1 = int(self.input.text())  
 self.op = op  
 self.input.setText("")  
  
 # действие при нажатии на кнопку reset (сброс)  
 def \_reset(self):  
 self.input.setText("")  
  
 # действие при нажатии на кнопку =  
 def \_result(self):  
 if (self.input.text()):  
 self.num\_2 = int(self.input.text())  
 if self.op == "+":  
 self.input.setText(str(self.num\_1 + self.num\_2))  
 elif self.op == "-":  
 self.input.setText(str(self.num\_1 - self.num\_2))  
 elif self.op == "/":  
 if self.num\_2 != 0:  
 self.input.setText(str(self.num\_1 / self.num\_2))  
 else:  
 self.input.setText("Can't divide by 0")  
 if self.op == "\*":  
 self.input.setText(str(self.num\_1 \* self.num\_2))  
  
  
# запуск  
app = QApplication(sys.argv)  
  
win = Calculator()  
win.show()  
  
sys.exit(app.exec\_())

Рисунок – запущенный калькулятор



**Вывод:**

В результате проделанной работы были основаны простейшие классы библиотеки PyQt5 и использование их в Python.