

Федеральное государственное автономное  
образовательное учреждение высшего образования  
«Национальный исследовательский университет  
ИТМО»

Факультет Информационных технологий и  
программирования

*Лабораторная работа №5*

*IDE и Unit-тесты*

Выполнил: Нечаев Александр Сергеевич,

группа М3111

Проверил: Повышев Владислав Вячеславович

Санкт-Петербург

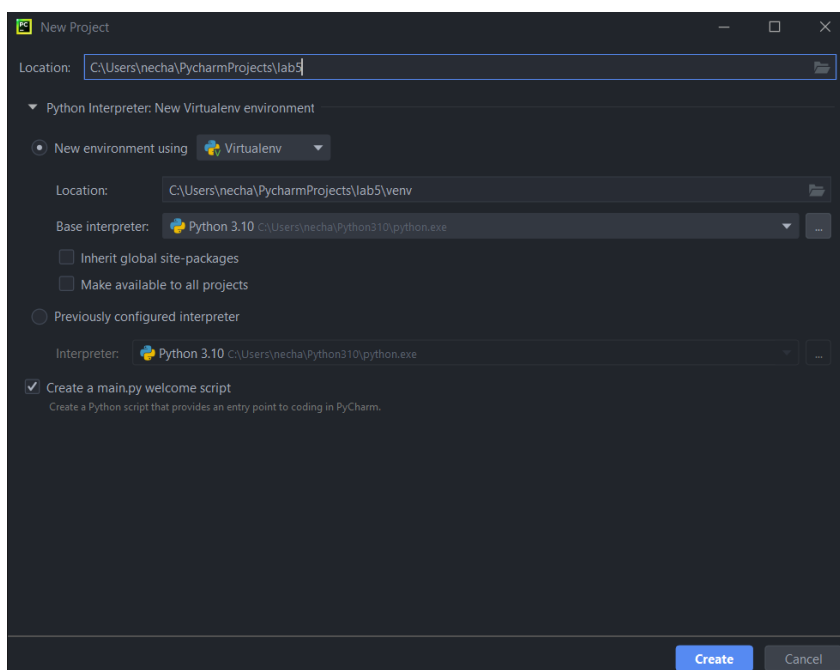
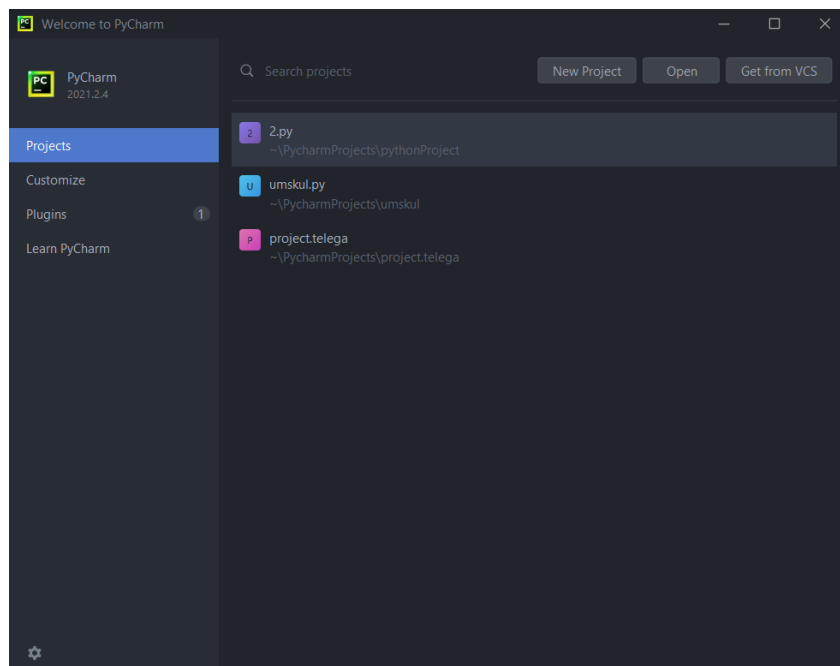
2022

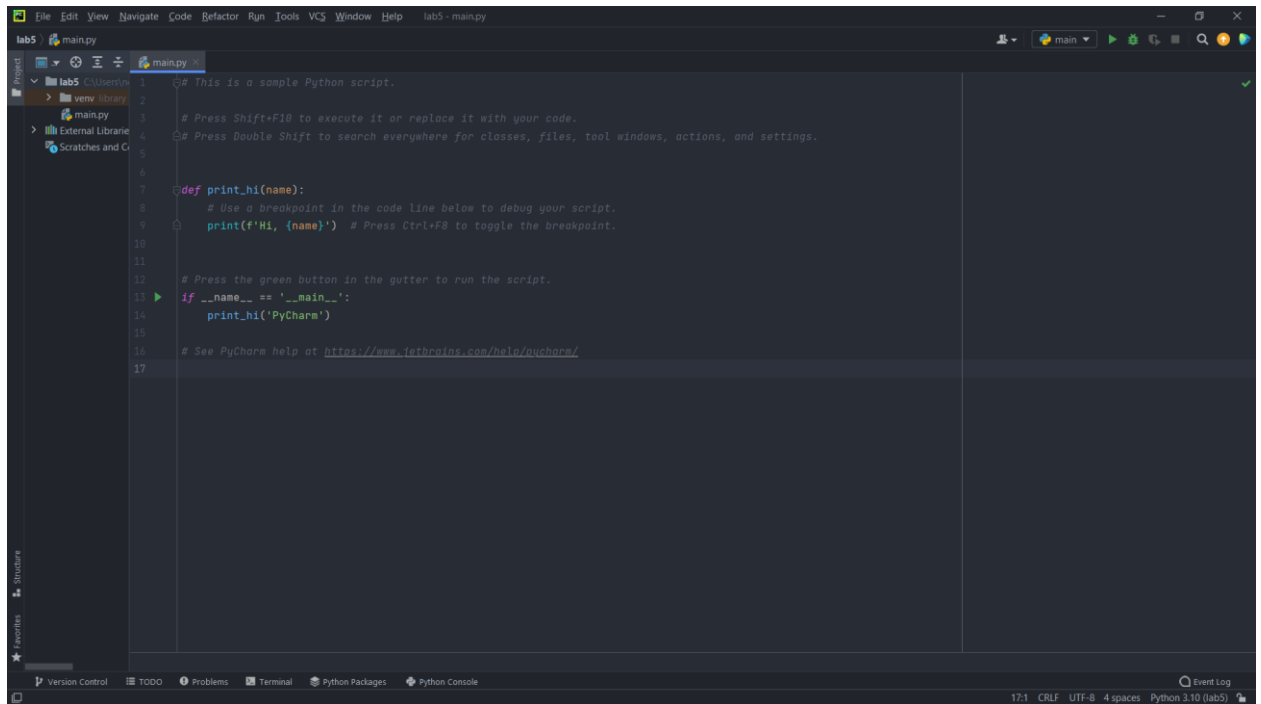
IDE (Integrated Development Environment) и редактор кода – это виды программного обеспечения, предназначенные для работы над приложениями, их разработки и тестирования. Возможности этих инструментов отличаются.

Редактор кода является программой, работающей как текстовый редактор, только более подходящий для написания кода.

IDE – это ПО, которое объединяет инструменты для разработки приложений и их тестирования в едином интерфейсе. Может делать все то, что и текстовый редактор, и даже больше. В принципе этим IDE и отличается от него, она предоставляет более расширенные функции. Среда разработки дает возможность создавать крупные проекты, а также подключать Git.

1. Для нашей лабораторной работы используем IDE для Python - PyCharm. Создадим новый проект:





2.Сконфигурировав IDE для запуска и отладки кода на языке Python можем приступать к работе.

3.Напишем простой программный код, например, самый простой консольный калькулятор:

```
1 class Simp_Calculator:
2     def __init__(self):
3         pass
4
5     def fold(self, a, b):
6         return a+b
7
8     def subtract(self, a, b):
9         return a-b
10
11    def multiply(self, a, b):
12        return a*b
13
14    def divide(self, a, b):
15        if b!=0:
16            return a/b
```

4.Необходимо написать unit-тесты. Воспользуемся встроенной библиотекой unittest, которая проверяет правильность выполнения функций написанного мною проекта.

```

import unittest
from Simp_Calculator import Simp_Calculator

class testSimpCalculator(unittest.TestCase):
    def setUp(self):
        self.simple_calc = Simp_Calculator()

    def test_fold(self):
        self.assertEqual(self.simple_calc.fold(143, 100), 243)

    def test_subtract(self):
        self.assertEqual(self.simple_calc.subtract(174, 100), 74)

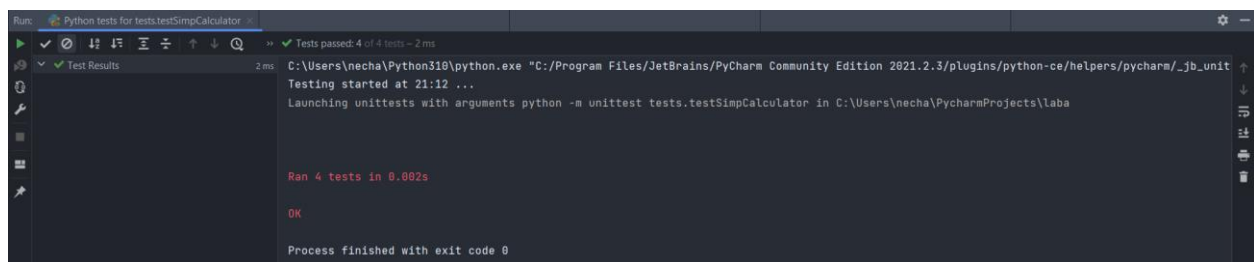
    def test_multiply(self):
        self.assertEqual(self.simple_calc.multiply(15, 15), 225)

    def test_divide(self):
        self.assertEqual(self.simple_calc.divide(198, 6), 33)

if __name__ == "__main__":
    unittest.main()

```

Запустим и проверим работу нашей программы:



### 5. Распространенные HotKeys в IDE PyCharm:

- **Ctrl + W** — выделение текущего блока. Если нажимать W несколько раз, то будут выделяться родительские блоки по иерархии.
- **Ctrl + Y** — удаляет строку
- **Ctrl + D** — дублирует строку
- **Ctrl + Shift + U** — делает выделенный текст или заглавными, или строчными буквами
- **Ctrl + Delete** — удаляет часть от курсора до конца слова
- **Ctrl + Alt + L** — форматирование кода соответствии с настройками code style.
- **Ctrl + Shift + V** — вставка из истории буфера.
- **\*Ctrl + /** — закомментировать/раскомментировать текущую строку.
- **Ctrl + Shift + /** — закомментировать/раскомментировать выделенные строки кода.
- **Ctrl + Shift + J** — к текущей строке присоединяет нижнюю.
- **Ctrl + [ или ]** — перемещает курсор в начало или конец блока с кодом, в теле функции, в теле цикла и т.д.
- **Ctrl + Shift + ] или [** — выделить код код до конца или до начала блока
- **Tab** — сделать отступ (табуляцию)
- **SHIFT + Tab** — убрать отступ
- **Ctrl + Alt + Стрелка вверх** — перемещает текущую строку вверх.

- **Ctrl + Alt + Стрелка вниз** — перемещает текущую строку вниз.
- **Ctrl + Shift + Стрелка вверх** — перемещает текущий логический блок вверх (блок if , for и т.д.).
- **Ctrl + Shift + Стрелка вниз** — перемещает текущий логический блок вниз
- **Ctrl + Space** — выводит возможные варианты автодополнения
- **Ctrl + курсор мыши над кодом** — краткая информация
- **Ctrl + G** — перейти к строке по номеру
- **Alt + Клик левой кнопкой мышки** — мультикурсор, если нужно одновременно несколько курсоров, кликаем мышкой в нужные места, куда нужно их добавить
- **Ctrl + Alt + J** — обрамление выделенного текста тегом (нужно выбрать первый пункт из всплывающего окошка)
- **Ctrl + F11** — добавить закладку с мнемоникой в текущую строку. В рамках проекта допускается до 10 нумерованных и до 26 буквенных закладок.
- **Shift + F11** — просмотр всех закладок
- **Двойной Shift** — поиск по проекту
- **Ctrl + F** — поиск в текущем файле
- **Ctrl + R** — поиск и замена
- **Ctrl+Shift+E** — вызовет всплывающее окно, в котором показаны последние куски кода в разных файлах, с которыми вы работали
- **Ctrl + Shift + F** — поиск в группе файлов (например в проекте или папке)
- **Ctrl + Shift + R** — поиск и замена в группе файлов
- **Ctrl + N** — найти в каком файле находится класс
- **Ctrl + Shift + N** — поиск файла по названию
- **Ctrl + Alt + S** — быстрый доступ к настройкам
- **Shift + F6** — переименование файла (выделите в блоке с файлами проекта нужный файл).
- **Ctrl + Insert** — создать новый файл

Вывод:

В ходе работы я узнал, что такое unit-тесты и как ими пользоваться в IDE PyCharm, а также изучил наиболее распространенные HotKeys в данной IDE.