Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт перспективной инженерии

Департамент цифровых, робототехнических систем и электроники

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ №6**

**дисциплины «Объектно-ориентированное программирование»**

|  |
| --- |
| Выполнил:  Евдаков Евгений Владимирович 3 курс, группа ИВТ-б-о-22-1,  09.03.01 «Информатика и вычислительная техника», направленность (профиль)  «Программное обеспечение средств вычислительной техники и автоматизированных систем», очная форма обучения  (подпись) |
| Проверил:  Воронкин Р. А., доцент департамента цифровых, робототехнических систем и электроники института перспективной инженерии  (подпись) |

Отчет защищен с оценкой Дата защиты

Ставрополь, 2024 г.

**Тема:** классы данных в Python.

**Цель:** приобретение навыков по работе с классами данных при написании программ с помощью языка программирования Python версии 3.x.

# Ход работы:

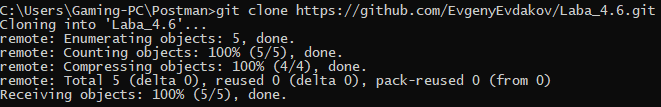
**Задание 1.** Создал общедоступный репозиторий на GitHub, в котором использована лицензий MIT и язык программирования Python, также добавил файл .gitignore с необходимыми правилами. Клонировал свой репозиторий на свой компьютер. Организовал свой репозиторий в соответствие с моделью ветвления git-flow, появилась новая ветка develop в которой буду выполнять дальнейшие задачи.

Рисунок 1. Клонирование репозитория

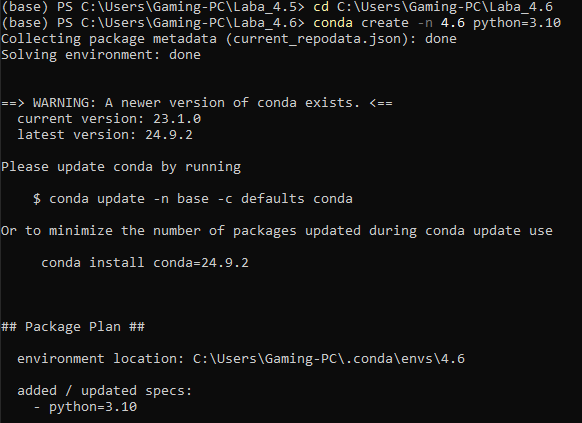
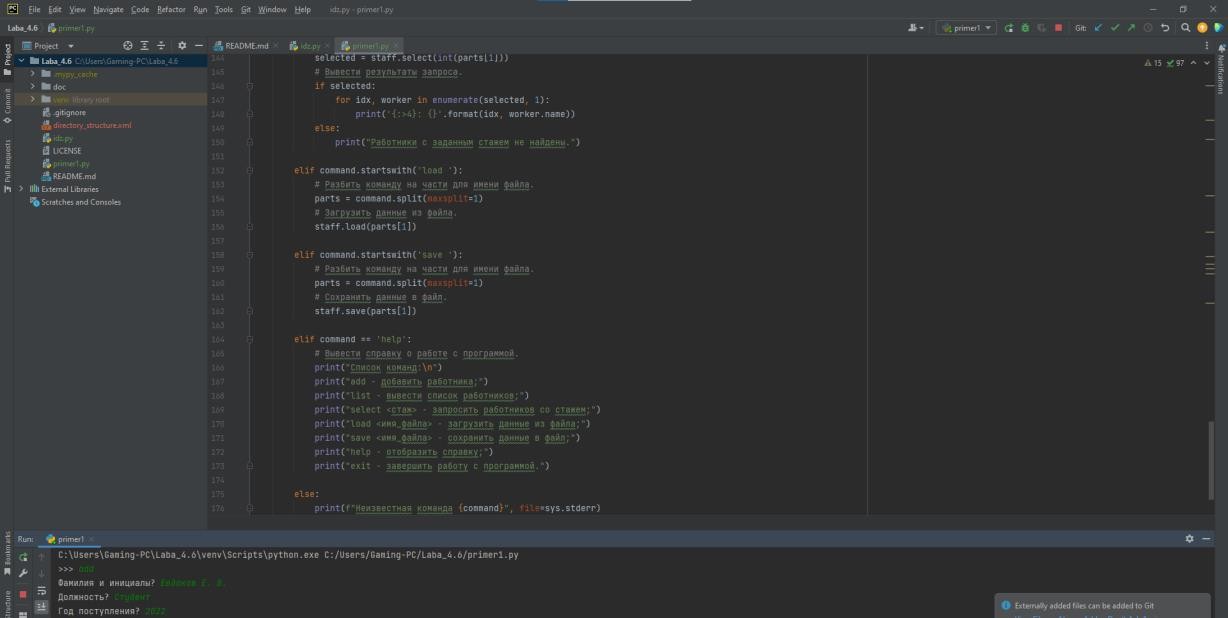
**Задание 2.** Создал виртуальное окружение conda и активировал его, также установил необходимые пакеты isort, black, flake8.

Рисунок 2. Создание виртуального окружения

**Задание 3.** Создал проект PyCharm в папке репозитория. Приступил к работе с примером. Добавил новый файл primer1.py.

**Условие примера:** Для примера 2 лабораторной работы 9 добавьте возможность работы с классами данных, а также сохранения и чтения данных в формат XML.



# Задание 4.

Рисунок 3. Выполнение первого примера

# Индивидуальное задание Вариант 6

Создал новый файл под названием idz1.py.

**Условие задания:** выполнить индивидуальное задание лабораторной работы 4.5, использовав классы данных, а также загрузку и сохранение данных в формат XML.

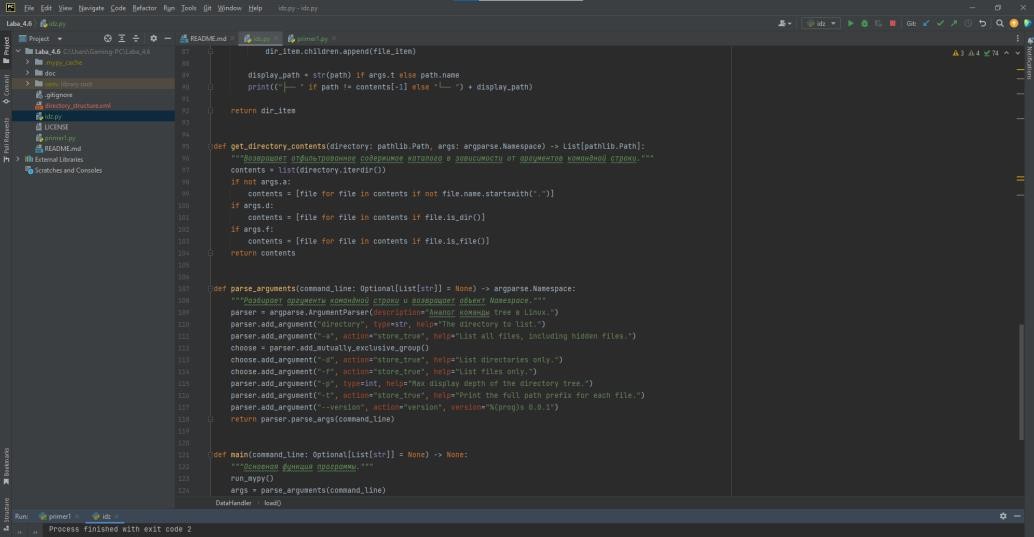


Рисунок 4. Выполнение первого индивидуального задания

Далее запустим код и проверим его на выполнение.

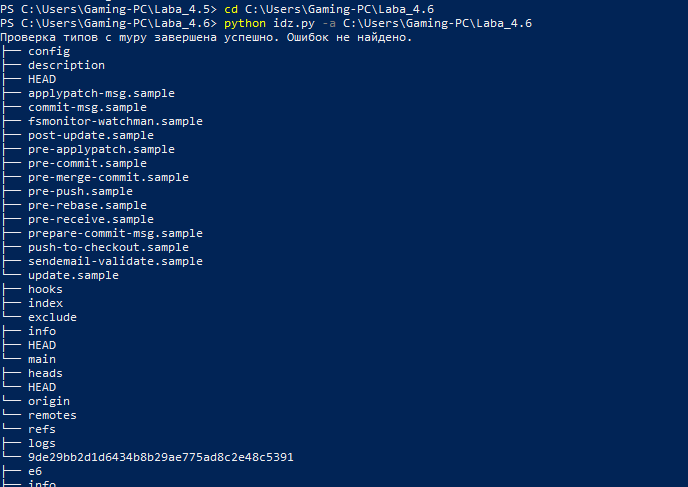


Рисунок 5. Выполнение программы

# Задание 5.

После выполнения работы на ветке develop, слил ее с веткой main и отправил изменения на удаленный сервер. Создал файл envirement.yml и деактивировал виртуальное окружение.

Рисунок 6. Деактивация ВО

Ссылка: https://github.com/EvgenyEvdakov/Laba\_4.6

# Ответы на контрольные вопросы:

1. **Как создать класс данных в языке Python?**

Классы данных (или датаклассы) в Python можно создать с помощью декоратора @dataclass из модуля dataclasses. Декоратор @dataclass автоматически добавляет полезные методы в класс, такие как init ,

repr , eq , и другие, что облегчает работу с данными.

# Какие методы по умолчанию реализует класс данных?

Декоратор @dataclass автоматически добавляет следующие методы, если они не были определены вручную:

* + init : Конструктор, который инициализирует поля, используя аргументы, переданные при создании объекта.
  + repr : Представление объекта в виде строки, удобно для его просмотра в консоли.
  + eq : Метод сравнения, позволяющий сравнивать два объекта на равенство по значению их полей.
  + lt , le , gt , ge : Методы для выполнения операций сравнения <, <=, >, >=, если в @dataclass указана опция order=True.
  + hash : Если класс помечен как неизменяемый, также будет добавлен метод hash для использования объектов в качестве ключей словаря или элементов множества.

# Как создать неизменяемый класс данных?

Чтобы создать неизменяемый класс данных, нужно использовать параметр frozen=True в декораторе @dataclass. Этот параметр запрещает изменение полей после создания объекта, делая его неизменяемым (immutable).

**Вывод:** приобрел навыки по работе с классами данных при написании программ с помощью языка программирования Python версии 3.x.