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

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития Кафедра инфокоммуникаций

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ №2.19**

**дисциплины «Основы кроссплатформенного программирования»**

|  |
| --- |
| Выполнил:  Пинаева Диана Олеговна  2 курс, группа ИТС-б-о-22-1,  11.03.02 «Инфокоммуникационные технологии и системы связи», направленность (профиль)  «Инфокоммуникационные системы и сети», очная форма обучения  (подпись) |
| Руководитель практики: Воронкин Р. А., доцент кафедры инфокоммуникаций  (подпись) |

Отчет защищен с оценкой Дата защиты

Ставрополь, 2023 г.

**Тема:** Работа с файловой системой в Python3 с использованием модуля pathlib

**Цель:** приобретение навыков по работе с файловой системой с помощью библиотеки pathlib языка программирования Python версии 3.x.

# Ход работы:

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

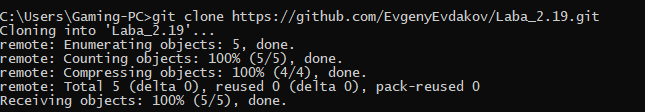


Рисунок 1. Клонирование репозитория

Задание 2.Создала виртуальное окружение conda и активировала его, также установила необходимые пакеты isort, black, flake8, pyinputplus.

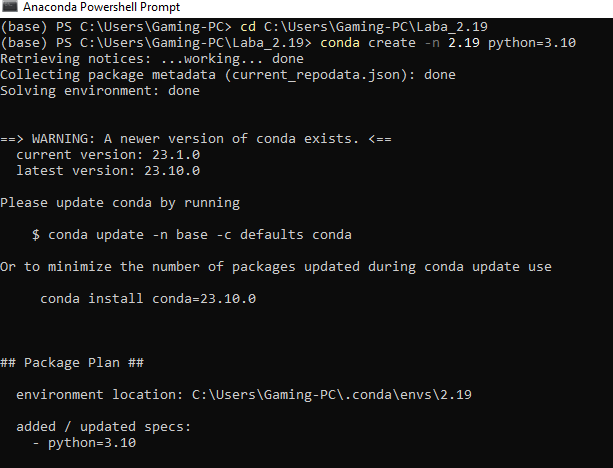


Рисунок 2. Создание виртуального окружения

Задание 3.Создала проект PyCharm в папке репозитория. Приступила к работе с примером. Добавила новый файл primer1.py.

Условие примера:написать программу, которая позволяет перемещать все текстовые файлы в один каталог, для этого необходимо вначале создать папку archive и зачем уже выполнять код.

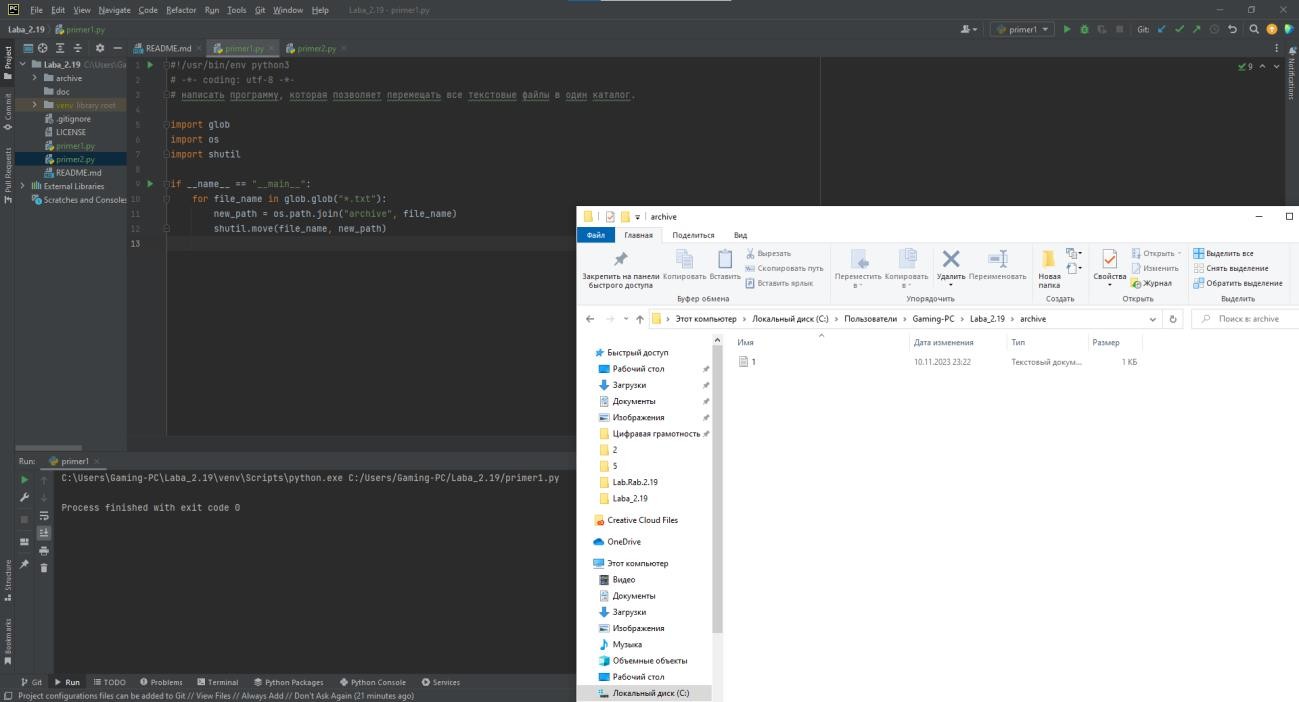


Рисунок 3. Реализация первого примера

# Индивидуальное задание

# Вариант 10

Создала новый файл под названием idz.py.

Условие задания:Для своего варианта лабораторной работы 2.17 добавьте возможность хранения файла данных в домашнем каталоге пользователя. Для выполнения операций с файлами необходимо использовать модуль pathlib.

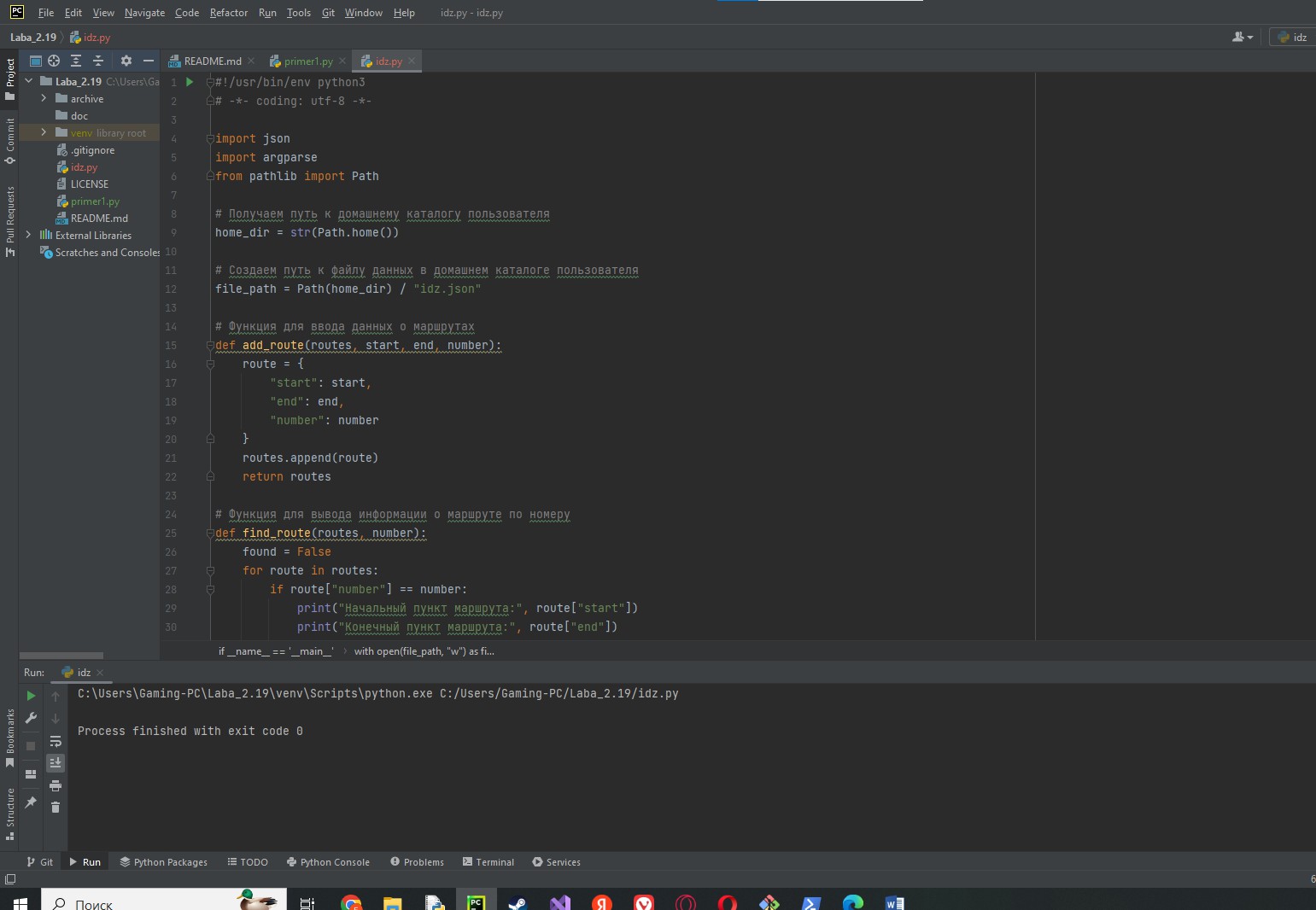


Рисунок 4. Код индивидуального задания

Path.home() возвращает экземпляр объекта Path, который представляет домашнюю директорию пользователя. Чтобы получить строковое представление этого пути, используется функция str().

В отличие от этого, os.path.join() применяется для создания полного пути к файлу, объединяя путь к домашней директории пользователя (home\_dir) и имя файла (file\_name). Этот подход гарантирует, что файл будет сохранен в домашней директории пользователя, а не в текущей рабочей директории программы.

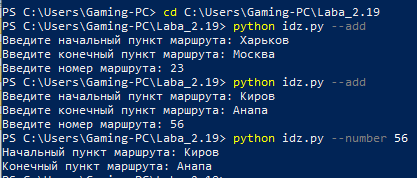


Рисунок 5. Результат индивидуального задания

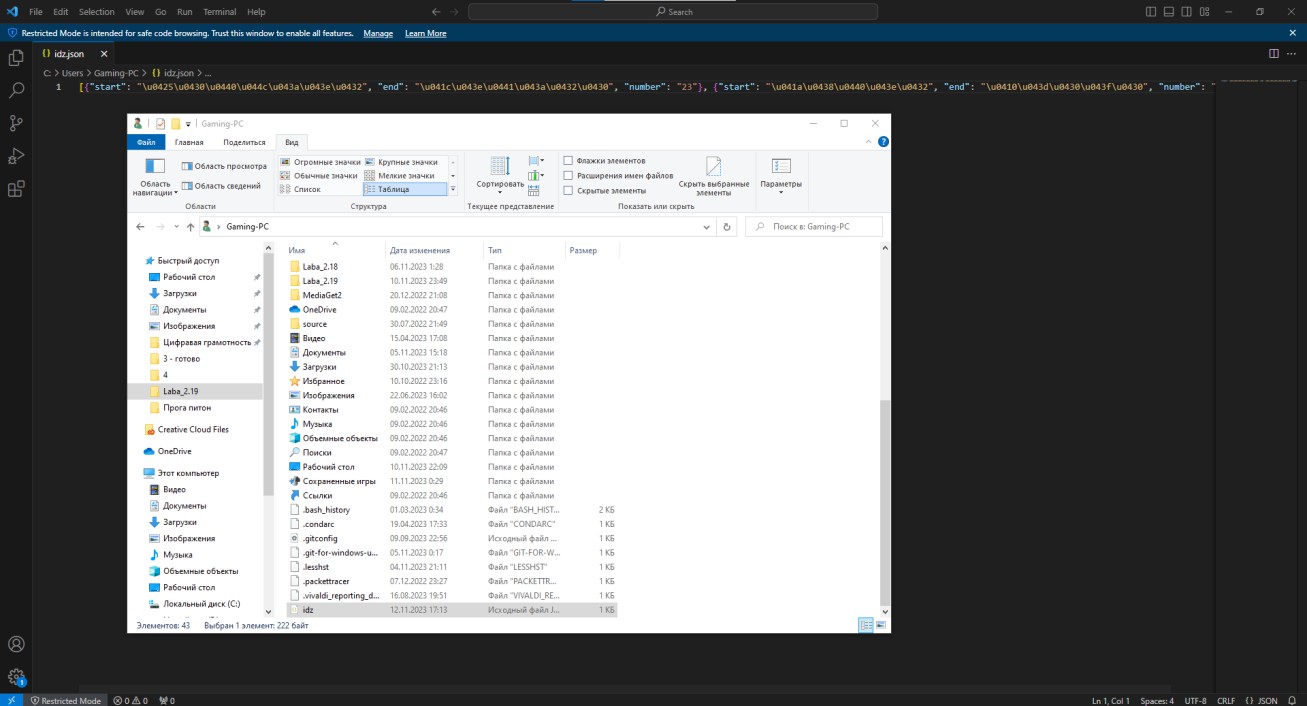


Рисунок 6. Файл json

# Задание 4.

После выполнения работы на ветке develop, слила ее с веткой main и отправила изменения на удаленный сервер. Создала файл envirement.yml и деактивировала виртуальное окружение.

Рисунок 7. Деактивация ВО

Ссылка: <https://github.com/Diana-372/Lab2.19.git>

# Ответы на контрольные вопросы:

1. Какие существовали средства для работыс файловой системой до Pytho 3.4?

До версии Python 3.4 для работы с файловой системой использовались модули os и os.path.

# Что регламентирует PEP 428?

PEP 428 регламентирует добавление модуля pathlib в стандартную библиотеку Python. Этот модуль предоставляет объектно-ориентированный интерфейс для работы с путями к файлам и каталогам.

# Как осуществляется создание путей средствами модуля pathlib?

Создание путей средствами модуля pathlib осуществляется путем создания экземпляров класса Path. Например:

from pathlib import Path

path = Path('/some/directory/file.txt')

# Как получить путь дочернего элемента файловой системы с помощью модуля pathlib?

Для получения пути дочернего элемента файловой системы используется оператор “/” (child\_path = path / 'child\_file.txt')

# Как получить путь к родительским элементам файловой системы с помощью модуля pathlib?

Для получения пути к родительским элементам файловой системы используется метод parent.

# Как выполняются операции с файлами с помощью модуля pathlib?

Операции с файлами, такие как чтение, запись и другие, выполняются с использованием методов объектов типа Path.

# Как можно выделить компоненты пути файловой системы с помощью модуля pathlib?

Компоненты пути файловой системы могут быть выделены с использованием различных атрибутов и методов объектов типа Path, таких как name, suffix, stem и другие.

# Как выполнить перемещение и удаление файлов с помощью модуля pathlib?

Перемещение и удаление файлов с помощью модуля pathlib осуществляется методами rename и unlink соответственно.

# Как выполнить подсчет файлов в файловой системе?

Подсчет файлов в файловой системе можно выполнить, например, с использованием функции rglob.

# Как отобразить дерево каталогов файловой системы?

Для отображения дерева каталогов файловой системы можно воспользоваться рекурсивным обходом каталогов, например, с использованием метода rglob и цикла.

# Как создать уникальное имя файла?

Создание уникального имени файла можно осуществить с использованием модуля tempfile.

# Каковы отличия в использовании модуля pathlib для различных операционных систем?

Модуль pathlib создавался с учетом переносимости между операционными системами, и его использование практически идентично на разных платформах. Однако, при работе с абсолютными путями, следует учитывать различия в разделителях пути между операционными системами (например, \ в Windows и / в Unix-like системах).

**Вывод:** приобрела навыки по работе с файловой системой с помощью библиотеки pathlib языка программирования Python версии 3.x.