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 «Анализ данных»

|  |
| --- |
| Выполнил:  Кенесбаев Хилол Куат улы  2 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) |
| Руководитель практики: Воронкин Р.А., доцент кафедры инфокоммуникаций  (подпись) |

Отчет защищен с оценкой \_ Дата защиты

Ставрополь, 2024 г.

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

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

# Порядок выполнения работы:

Создал новый репозиторий, клонировал его, в нем создал ветку developer и перешел на нее.

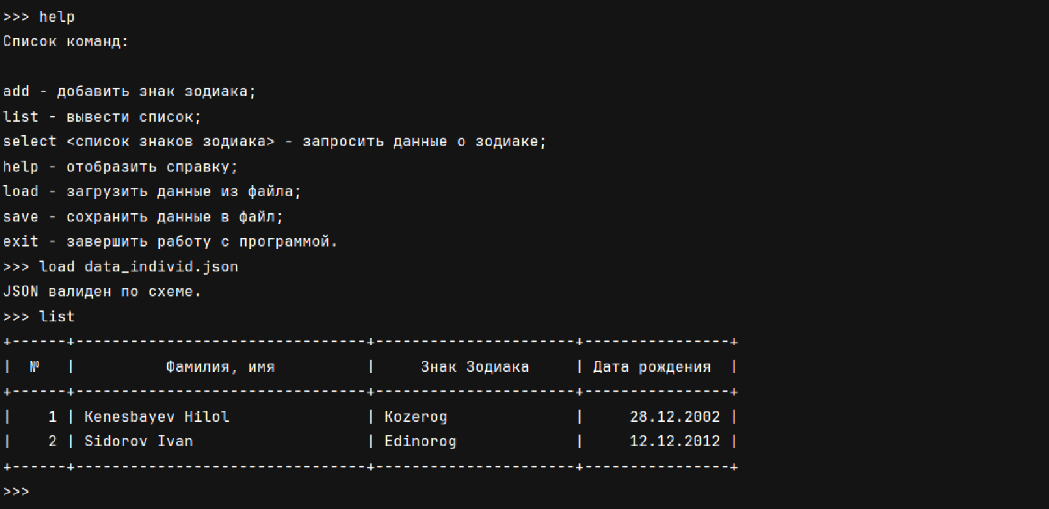
**Выполнил индивидуальное задание 1:** Для своего варианта лабораторной работы 2.17 добавьте возможность хранения файла данных в домашнем каталоге пользователя. Для выполнения операций с файлами необходимо использовать модуль pathlib. 

Рисунок 1. Вывод программы индивидуального задания 1

**Выполнил индивидуальное задание 2:** Разработайте аналог утилиты tree в Linux. Используйте возможности модуля argparse для управления отображением дерева каталогов файловой системы. Добавьте дополнительные уникальные возможности в данный программный продукт.

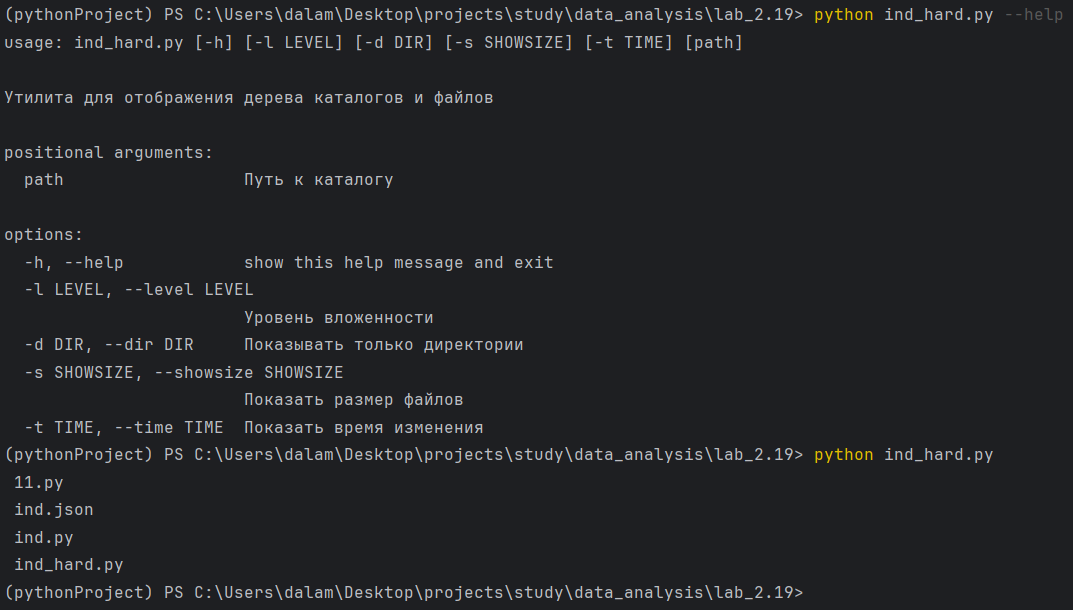


Рисунок 2. Вывод программы аналога tree

# Ответы на контрольные вопросы:

1. Какие существовали средства для работы с файловой системой до Python 3.4?
   * Методы строк, например path.rsplit('\\', maxsplit=1)[0]
   * Модуль os.path
2. Что регламентирует PEP 428?

Модуль Pathlib – Объектно-ориентированные пути файловой системы.

1. Как осуществляется создание путей средствами модуля pathlib?

Есть несколько разных способов создания пути. Прежде всего, существуют classmethods наподобие cwd() (текущий рабочий каталог) и home() (домашний каталог вашего пользователя). Также можно создать путь при помощи: pathlib.Path(«путь»).

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

При помощи метода resolve().

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

При помощи свойства parent.

1. Как выполняются операции с файлами с помощью модуля pathlib? source.replace(destination)
   * перемещение; file\_to\_delete.unlink()
   * удаление файлов; file\_to\_write.write\_text('Привет, мир!') content = file\_to\_read.read\_text()
   * чтение и запись файлов.
2. Как можно выделить компоненты пути файловой системы с помощью модуля pathlib?

.name : имя файла без какого-либо каталога

.parent : каталог, содержащий файл, или родительский каталог, если путь является каталогом

.stem : имя файла без суффикса .suffix : расширение файла

.anchor : часть пути перед каталогами

1. Как выполнить перемещение и удаление файлов с помощью модуля pathlib?

.replace() – метод перемещения файлов

.unlink() – метод удаления файлов

1. Как выполнить подсчет файлов в файловой системе? import collections

collections.Counter(p.suffix for p in pathlib.Path.cwd().iterdir()) Counter({'.md': 2, '.txt': 4, '.pdf': 2, '.py': 1})

1. Как отобразить дерево каталогов файловой системы? def

tree(directory):

print(f'+ {directory}')

for path in sorted(directory.rglob('\*')):

depth = len(path.relative\_to(directory).parts) spacer = ' ' \* depth print(f'{spacer}+ {path.name}')

1. Как создать уникальное имя файла? def unique\_path(directory, name\_pattern):

counter = 0 while True:

counter += 1

path = directory/name\_pattern.format(counter) if not path.exists(): return path

path = unique\_path(pathlib.Path.cwd(), 'test{:03d}.txt')

1. Каковы отличия в использовании модуля pathlib для различных операционных систем?

Ранее мы отмечали, что когда мы создавали экземпляр pathlib.Path, возвращался либо объект WindowsPath, либо PosixPath . Тип объекта будет зависеть от операционной системы, которую вы используете. Эта функция позволяет довольно легко писать кроссплатформенный код. Можно явно запросить WindowsPath или PosixPath, но вы будете ограничивать свой код только этой системой без каких-либо преимуществ. Такой конкретный путь не может быть использован в другой системе.

**Вывод:** в результате выполнения лабораторной работы были приобретены навыки по работе с файловой системой с помощью библиотеки pathlib языка программирования Python версии 3.x.