

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ  
УНИВЕРСИТЕТ»

Институт цифрового развития

Кафедра инфо коммуникаций

**ОТЧЕТ**  
**ПО ЛАБОРАТОРНОЙ РАБОТЕ №2.19**  
**Дисциплины «Программирование на Python»**

Выполнил:

Волошин Алексей Вадимович  
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, канд. техн. наук, 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.** 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.

```
Cloning into 'Laba_2.19'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), done.
```

Рисунок 1. Клонирование репозитория

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

```
Anaconda Powershell Prompt
(base) PS C:\Users\Gaming-PC> cd C:\Users\Gaming-PC\Laba_2.19
(base) PS C:\Users\Gaming-PC\Laba_2.19> conda create -n 2.19 python=3.10
Retrieving notices: ...working... done
Collecting package metadata (current_repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 23.1.0
  latest version: 23.10.0

Please update conda by running

    $ conda update -n base -c defaults conda

Or to minimize the number of packages updated during conda update use

    conda install conda=23.10.0

## Package Plan ##

environment location: C:\Users\Gaming-PC\.conda\envs\2.19

added / updated specs:
- python=3.10
```

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

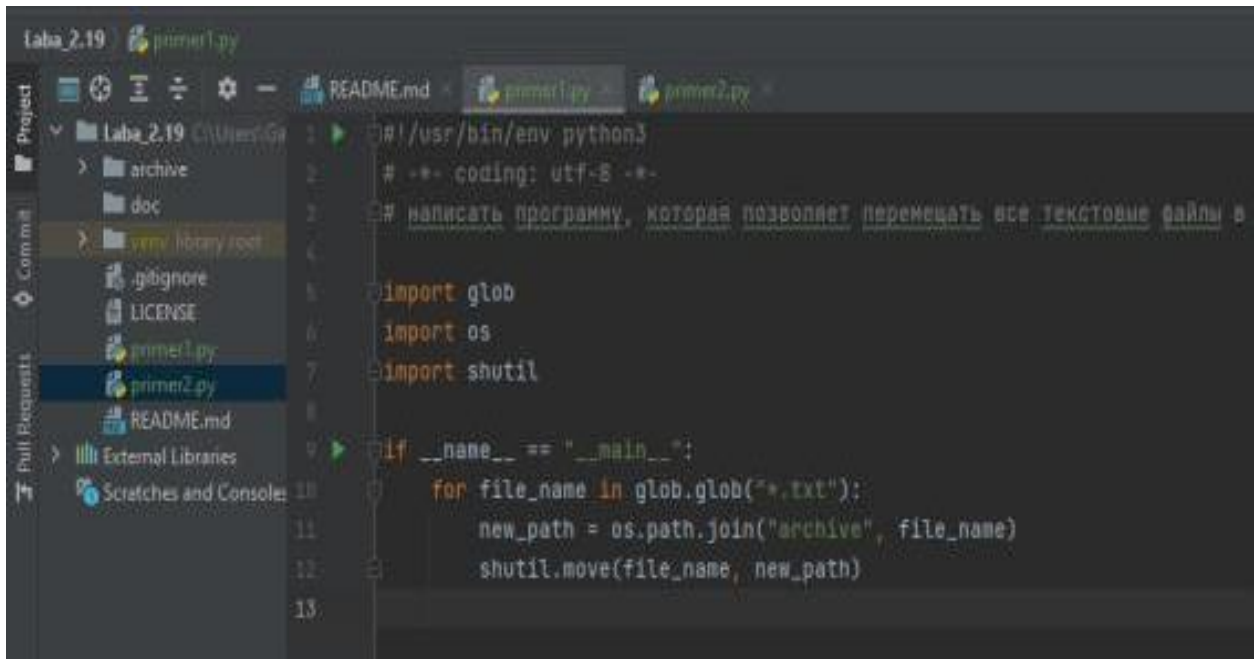


Рисунок 3. Реализация первого примера

#### Задание 4.

#### Индивидуальное задание

#### Вариант 8

Создал 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.

```

1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  import os
5
6  def realname(path, root=None):
7      if root is not None:
8          path=os.path.join(root, path)
9          result=os.path.basename(path)
10         if os.path.islink(path):
11             realpath=os.readlink(path)
12             result= '%s -> %s' % (os.path.basename(path), realpath)
13         return result
14
15  def ptree(startpath, depth=-1):
16      prefix=0
17      if startpath != '/':
18          if startpath.endswith('/'): startpath=startpath[:-1]
19          prefix=len(startpath)
20      for root, dirs, files in os.walk(startpath):
21          level = root[prefix:].count(os.sep)
22          if depth > -1 and level > depth: continue
23          indent=subindent = ''
24          if level > 0:
25              indent = '| ' * (level-1) + '|-- '
26              subindent = '| ' * (level) + '|-- '
27          print('{}{}'.format(indent, realname(root)))
28          # print dir only if symbolic link; otherwise, will be printed as root
29          for d in dirs:
30              if os.path.islink(os.path.join(root, d)):
31                  print('{}{}'.format(subindent, realname(d, root=root)))
32          for f in files:
33              print('{}{}'.format(subindent, realname(f, root=root)))

```

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.

### **Задание 5.**

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

```
(2.19) PS C:\Users\Gaming-PC\Laba_2.19> conda env export > environment.yml  
(2.19) PS C:\Users\Gaming-PC\Laba_2.19> conda deactivate
```

Рисунок 7. Деактивация ВО

Ссылка: <https://github.com/Alexei-42STR/LR2.19>

### **Ответы на контрольные вопросы:**

#### **1. Какие существовали средства для работы файловой системой до Python 3.4?**

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

#### **2. 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.

#### **3. 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')
```

#### **4. 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'`)

#### **5. Как получить путь к родительским элементам файловой системы с помощью модуля `pathlib`?**

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

#### **6. 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`.

#### **7. 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.

#### **8. Как выполнить перемещение и удаление файлов с помощью модуля `pathlib`?**

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

#### **9. Как выполнить подсчет файлов в файловой системе?**

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

#### **10. Как отобразить дерево каталогов файловой системы?**

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

#### **11. Как создать уникальное имя файла?**

Создание уникального имени файла можно осуществить с использованием модуля `tempfile`.

#### **12. 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:** приобрел навыки по работе с файловой системой с помощью библиотеки `pathlib` языка программирования Python версии 3.x.