

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**Отчет по лабораторной работе №2.15**

**Работа с файлами в языке Python.**

**по дисциплине «Технологии программирования и алгоритмизация»**

Выполнил студент группы ИВТ-б-о-20-1

Лысенко И.А. « » \_\_\_\_\_ 20\_\_ г.

Подпись студента \_\_\_\_\_

Работа защищена « » \_\_\_\_\_ 20\_\_ г.

Проверила Воронкин Р.А. \_\_\_\_\_

(подпись)

Ставрополь 2021

**Цель работы:** приобретение навыков по работе с текстовыми файлами при написании программ с помощью языка программирования Python версии 3.x, изучение основных методов модуля os для работы с файловой системой, получение аргументов командной строки.

**Ход работы:**

1. Разобрал примеры.

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

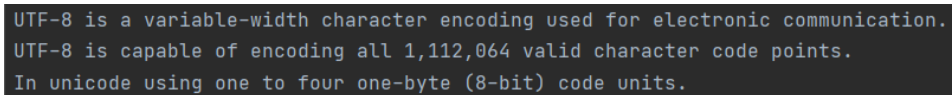
Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

Рисунок 1 – Создание виртуального окружения

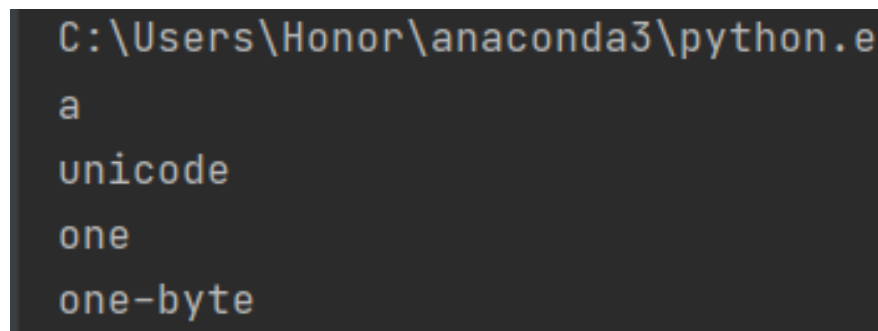
Рисунок 1 – Создание виртуального окружения

2. Выполнил первое индивидуальное задание, вариант 9. Написал программу, которая считывает английский текст из файла и выводит на экран слова текста, начинающиеся и оканчивающиеся на гласные буквы.



```
UTF-8 is a variable-width character encoding used for electronic communication.
UTF-8 is capable of encoding all 1,112,064 valid character code points.
In unicode using one to four one-byte (8-bit) code units.
```

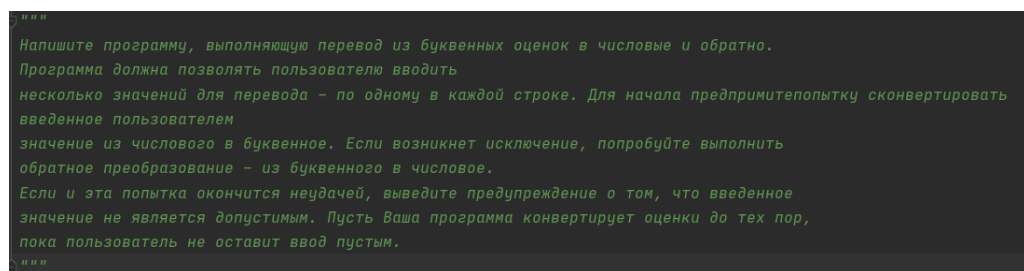
Рисунок 2.1 – Текст, над которым производилась работа.



```
C:\Users\Honor\anaconda3\python.exe
a
unicode
one
one-byte
```

Рисунок 2.2 – Работа индивидуального задания.

3. Выполнил второе индивидуальное задание, 9 вариант.



```
Напишите программу, выполняющую перевод из буквенных оценок в числовые и обратно.
Программа должна позволять пользователю вводить
несколько значений для перевода – по одному в каждой строке. Для начала предпримите попытку сконвертировать
введенное пользователем
значение из числового в буквенное. Если возникнет исключение, попробуйте выполнить
обратное преобразование – из буквенного в числовое.
Если и эта попытка окончится неудачей, выведите предупреждение о том, что введенное
значение не является допустимым. Пусть Ваша программа конвертирует оценки до тех пор,
пока пользователь не оставит ввод пустым.
```

Рисунок 3.1 – Задание

```

4
A = 5
3
3 = 0
f
f = Данное значение является недопустимым

<_io.TextIOWrapper name='oc.txt' mode='w' encoding='cp1251'>

```

Рисунок 3.2 – Работа кода

#### 4. Проверил программы через flake8.

```

(tools) PS C:\Users\Honor\desktop\r\lab_2_15> flake8
(tools) PS C:\Users\Honor\desktop\r\lab_2_15>

```

Рисунок 7 – Проверка flake8

### Ответы на вопросы.

#### 1. Как открыть файл в языке Python только для чтения?

Используя функцию `open()`, после ввода имени файла через запятую указать режим “r”.

#### 2. Как открыть файл в языке Python только для записи?

Используя функцию `open()`, после ввода имени файла через запятую указать режим “w”.

#### 3. Как прочитать данные из файла в языке Python?

Сначала необходимо открыть файл, вызвав функцию `open()`, затем использовать метод `read()`.

#### 4. Как записать данные в файл в языке Python?

Сначала необходимо открыть файл, вызвав функцию `open()`, затем использовать метод `write()`.

#### 5. Как закрыть файл в языке Python?

Использовать метод `close()` или открывать файл при помощи оператора `with`, который закрывает файл, после окончания работы с ним.

6. Изучите самостоятельно работу конструкции `with ... as`. Каково ее назначение в языке Python? Где она может быть использована еще, помимо работы с файлами?

Конструкция `with ... as` гарантирует, что критические функции выполнятся в любом случае. В основном она используется для работы с

файлами разного типа, но также может использоваться для фиксации или отката транзакции базы данных, для перенаправления стандартного вывода однопоточных программ.

7. Изучите самостоятельно документацию Python по работе с файлами. Какие помимо рассмотренных существуют методы записи/чтения информации из файла?

Метод `writelines()` – добавляет последовательность строк в файл.

8. Какие существуют, помимо рассмотренных, функции модуля `os` для работы с файловой системой?

**`os.name`** – имя операционной системы.

**`os.environ`** – словарь переменных окружения.

**`os.getpid()`** – текущий `id` процесса.

**`os.uname()`** – информация об ОС.

**`os.access(path, mode, *, dir_fd=None, effective_ids=False, follow_symlinks=True)`** – проверка доступа к объекту у текущего пользователя.

**`os.chdir(path)`** – смена текущей директории.

**`os.chmod(path, mode, *, dir_fd=None, follow_symlinks=True)`** – смена прав доступа к объекту.

**`os.link(src, dst, *, src_dir_fd=None, dst_dir_fd=None, follow_symlinks=True)`** – создаёт жёсткую ссылку.

**`os.listdir(path=".")`** – список файлов и директорий в папке.

**`os.makedirs(path, mode=0o777, exist_ok=False)`** – создаёт директорию, создавая при этом промежуточные директории.

**`os.symlink(source, link_name, target_is_directory=False, *, dir_fd=None)`** – создаёт символическую ссылку на объект.

**`os.truncate(path, length)`** – обрезает файл до длины `length`.

**`os.utime(path, times=None, *, ns=None, dir_fd=None, follow_symlinks=True)`** – модификация времени последнего доступа и изменения файла.

**os.walk**(top, topdown=True, onerror=None, followlinks=False) – генерация имён файлов в дереве каталогов.

**os.system**(command) – исполняет системную команду, возвращает код её завершения.

**os.urandom**(n) – n случайных байт.

**os.path** – модуль, реализующий некоторые полезные функции на работы с путями.

**Вывод.** Были приобретены навыки по работе с текстовыми файлами при написании программ с помощью языка программирования Python версии 3.x, изучение основных методов модуля os для работы с файловой системой, получение аргументов командной строки