

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 №1
дисциплины «Анализ данных
Вариант №5

Выполнил:
Михеева Елена Александровна
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., канд. техн. наук,
доцент, доцент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

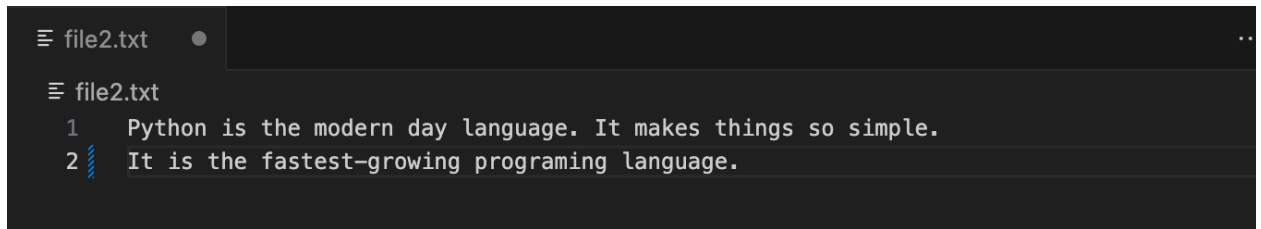
Ставрополь, 2024 г.

Тема: Работа с файлами в языке Python

Цель: 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.

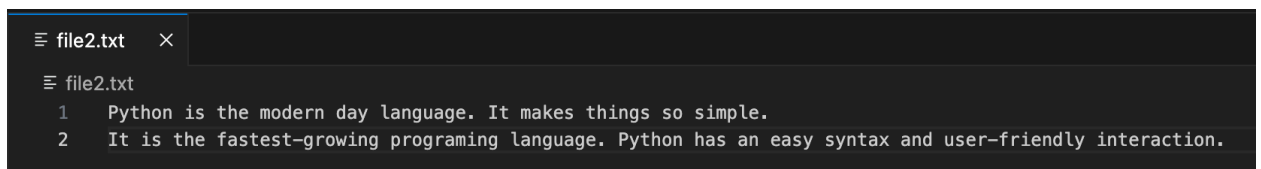
Порядок выполнения работы:

1. Были проработаны примеры лабораторной работы с 1 по 17.



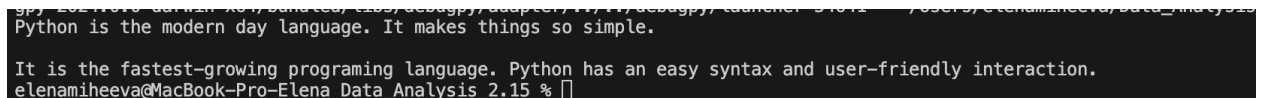
```
file2.txt
file2.txt
1 Python is the modern day language. It makes things so simple.
2 It is the fastest-growing programming language.
```

Рисунок 1. Результат выполнения примера №1



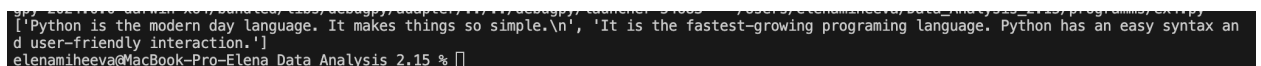
```
file2.txt
file2.txt
1 Python is the modern day language. It makes things so simple.
2 It is the fastest-growing programming language. Python has an easy syntax and user-friendly interaction.
```

Рисунок 2. Результат выполнения примера №2



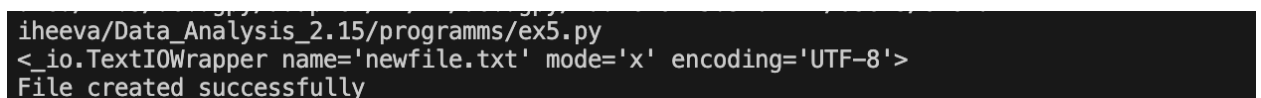
```
Python is the modern day language. It makes things so simple.
It is the fastest-growing programming language. Python has an easy syntax and user-friendly interaction.
elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.15 %
```

Рисунок 3. Результат выполнения примера №3



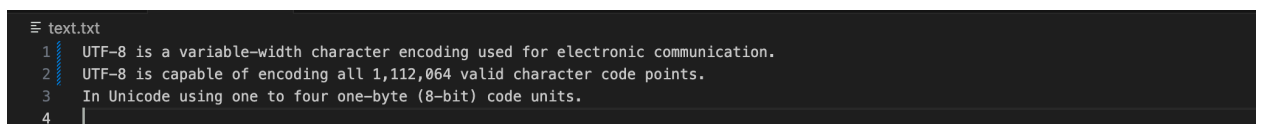
```
['Python is the modern day language. It makes things so simple.\n', 'It is the fastest-growing programming language. Python has an easy syntax and user-friendly interaction.']
elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.15 %
```

Рисунок 4. Результат выполнения примера №4



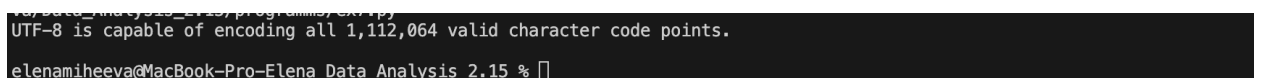
```
iheeva/Data_Analysis_2.15/programms/ex5.py
<_io.TextIOWrapper name='newfile.txt' mode='x' encoding='UTF-8'>
File created successfully
```

Рисунок 5. Результат выполнения примера №5



```
text.txt
1 UTF-8 is a variable-width character encoding used for electronic communication.
2 UTF-8 is capable of encoding all 1,112,064 valid character code points.
3 In Unicode using one to four one-byte (8-bit) code units.
4
```

Рисунок 6. Результат выполнения примера №6



```
UTF-8 is capable of encoding all 1,112,064 valid character code points.
elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.15 %
```

Рисунок 7. Результат выполнения примера №7

```
The filepointer is at byte : 0  
After reading, the filepointer is at: 10  
elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.15 %
```

Рисунок 8. Результат выполнения примера №8

```
programms > ex9-14.py > ...  
1  #!/usr/bin/env python3  
2  # -*- coding: utf-8 -*-  
3  
4  
5  import os  
6  
7  
8  if __name__ == "__main__":  
9      with open("file4.txt", "w") as fileptr:  
10         fileptr.write(  
11             "Simple file\n"  
12         )  
13         # rename file2.txt to file3.txt  
14         os.rename("file4.txt", "file5.txt")  
15  
16         # deleting the file named file3.txt  
17         os.remove("file5.txt")  
18  
19         # creating a new directory with the name new  
20         os.mkdir("new")  
21  
22         path = os.getcwd()  
23         print(path)  
24         # removing the new directory  
25         os.rmdir("new")  
26
```

Рисунок 9. Результат выполнения примера №9-14

```
● elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.15 % python3 programs/ex15.py arg1 arg2 arg3  
Number of arguments: 4 arguments  
Argument List: ['programs/ex15.py', 'arg1', 'arg2', 'arg3']
```

Рисунок 10. Результат выполнения примера №15

```
● elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.15 % python3 programs/ex16.py Knowledge Hut 21  
Argument #0 is programs/ex16.py  
Argument #1 is Knowledge  
Argument #2 is Hut  
Argument #3 is 21  
No. of arguments passed is 4
```

Рисунок 11. Результат выполнения примера №16

```
● elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.15 % python3 programs/ex17.py 12  
Secret Password: lJyg64Sk@41o  
○ elenamiheeva@MacBook-Pro-Elena Data_Analysis_2.15 %
```

Рисунок 12. Результат выполнения примера №17

2. Были выполнены индивидуальные задания №1 и №2 согласно варианту №5.

```

programms > individual_1.py > ...
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  # Вариант №5. Написать программу, которая считывает текст из файла
5  # и выводит его на экран, меняя местами каждые два соседних слова.
6
7
8  def swap(text):
9      words = text.split()
10     for i in range(0, len(words)-1, 2):
11         words[i], words[i+1] = words[i+1], words[i]
12     return ' '.join(words)
13
14
15 if __name__ == '__main__':
16     with open("file2.txt", "r") as fileptr:
17         content = fileptr.read()
18         print("Исходный файл: ")
19         print(content)
20
21         swapped_text = swap(content)
22         print("\nТекст с поменяными соседними словами:")
23         print(swapped_text)
24

```

Рисунок 13. Программа индивидуального задания №1

Исходный файл:
Python is the modern day language. It makes things so simple.
It is the fastest-growing programing language. Python has an easy syntax and user-friendly interaction.

Текст с поменяными соседними словами:
is Python modern the language. day makes It so things It simple. the is programing fastest-growing Python langua
ge. an has syntax easy user-frienly and interaction.

Рисунок 14. Пример выполнения индивидуального задания №1

```

programms > individual_2.py > ...
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  # Написать программу, которая будет находить самое длинное слово в файле.
5  # В качестве результата программа должна выводить на экран длину
6  # самого длинного слова и все слова такой длины. Принимайте за значимые
7  # буквы любые непробельные символы, включая цифры и знаки препинания.
8
9
10 def find_longest_words(file_content):
11     words = file_content.split()
12     max_length = max(len(word) for word in words)
13     longest_words = [word for word in words if len(word) == max_length]
14     return max_length, longest_words
15
16
17 if __name__ == '__main__':
18     with open("file2.txt", "r") as fileptr:
19         content = fileptr.read()
20         print("Исходный файл: ")
21         print(content)
22         length, longest_words = find_longest_words(content)
23         print(f"Длина самого длинного слова: {length}")
24         print(f"Самые длинные слова: {' '.join(longest_words)}")

```

Рисунок 15, Программа индивидуального задания №2

```
Исходный файл:  
Python is the modern day language. It makes things so simple.  
It is the fastest-growing programming language. Python has an easy syntax and user-friendly  
Длина самого длинного слова: 15  
Самые длинные слова: fastest-growing
```

Рисунок 16. Пример выполнения индивидуального задания №2

Ответы на контрольные вопросы:

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

Чтобы открыть файл для чтения, мы используем режим `r` или `rb`.
`file = open("text.txt", 'r').`

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

Чтобы открыть файл для записи, мы используем режим `w` или `wb`.
`file = open("text.txt", 'w').` Также можно использовать режим `a` или `ab`, чтобы не перезаписывать файл, а дозаписывать в него.

3. Как прочитать данные из файла в языке Python?

Чтобы прочитать файл с помощью сценария Python, Python предоставляет метод `read()`. Метод `read()` считывает строку из файла. Он может читать данные как в текстовом, так и в двоичном формате. Также можно использовать цикл `for`: `for i in fileptr: print(i)`; также можно использовать метод `readline()`, который читает строки файла с самого начала, т. е. если мы используем его два раза, мы можем получить первые две строки файла. Python также предоставляет метод `readlines()`, который используется для чтения строк. Возвращает список строк до конца файла (EOF).

4. Как записать данные в файл в языке Python?

Запись данных в файл. Записать данные в файл можно с помощью метода `write()`.

5. Как закрыть файл в языке Python?

После того, как мы открыли файл, и выполнили все нужные операции, нам необходимо его закрыть. Для закрытия файла используется функция `close()`.

6. Изучите самостоятельно работу конструкции `with ... as`. Каково ее назначение в языке?

Конструкция `with ... as` в языке Python предназначена для обеспечения управления ресурсами с автоматическим освобождением этих ресурсов после завершения блока кода. Она часто используется с объектами, которые поддерживают протокол менеджеров контекста.

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

Метод `writelines()`:

Данный метод принимает список строк в качестве аргумента и записывает каждую строку списка в файл.

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

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

`os.chown (path, uid, gid, *, dir_fd=None, follow_symlinks=True)` - меняет `id` владельца и группы (Unix).

`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.