

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 №3
Модули и пакеты

Выполнил:

Боженко Александр Иванович
2 курс, группа ИТС-б-о-21-1,
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, 2022 г.

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.*

Открытие файла

Python предоставляет функцию `open()` , которая принимает два аргумента: имя файла и режим доступа, в котором осуществляется доступ к файлу. Функция возвращает файловый объект, который можно использовать для выполнения различных операций, таких как чтение, запись и т. д

```
file object = open(<file-name>, <access-mode>, <buffering>)
```

Доступ к файлам можно получить с помощью различных режимов, таких как чтение, запись или добавление. посмотрим на простой пример, чтобы открыть файл с именем «file.txt»(хранящийся в том же каталоге) в режиме чтения и распечатать его содержимое на консоли

```
#opens the file file.txt in read mode
```

```
fileptr = open("file.txt", "r")
```

```
if fileptr:
```

```
print("file is opened successfully")
```

```
file is opened successfully
```

Метод close()

После того, как все операции будут выполнены с файлом, мы должны закрыть его с помощью нашего скрипта Python, используя метод `close()` . Любая незаписанная информация уничтожается после вызова метода `close()` для файлового объекта.

```
fileobject.close()
```

```
# opens the file file.txt in read mode
```

```
fileptr = open("file.txt", "r")
```

```
if fileptr:
```

```
print("file is opened successfully")
```

```
#closes the opened file
```

```
fileptr.close()
```

После закрытия файла мы не можем выполнять какие-либо операции с файлом. Файл необходимо правильно закрыть. Если при выполнении некоторых операций с файлом возникает какое-либо исключение, программа завершается, не закрывая файл.

```
try:
```

```
fileptr = open("file.txt")
```

```
# perform file operations
```

```
finally:
```

```
fileptr.close()
```

Оператор with

Он полезен в случае манипулирования файлами. Используется в сценарии, когда пара операторов должна выполняться с блоком кода между ними.

```
with open(<file name>, <access mode>) as <file-pointer>:
```

```
#statement suite
```

Преимущество использования оператора with заключается в том, что он обеспечивает гарантию закрытия файла независимо от того, как закрывается вложенный блок.

Всегда рекомендуется использовать оператор with для файлов. Если во вложенном блоке кода возникает прерывание, возврат или исключение, тогда он автоматически закрывает файл, и нам не нужно писать функцию close() . Это не позволяет файлу исказиться.

```
with open("file.txt", 'r') as f:
```

```
content = f.read();
```

```
print(content)
```

Ход работы

- 1) Создадим общедоступный репозиторий на GitHub
- 2) Решим задачи с помощью языка программирования Python3. и отправим их на GitHub.

Вывод пример

```
f:/2 курс/языки программирования/5/1.py"
107
```

Рисунок 1. Результат.

Вывод пример 2

```
"f:/2 курс/языки программирования/5/2.py"
57
```

Рисунок 2. Результат.

Вывод пример 3

```
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.
```

Рисунок 3. Результат.

Вывод пример 4

```
['Python is the modern day language. It makes things so simple.\n', 'I
t is the fastest-growing programing language Python has an easy syntax
and user-friendly interaction.\n', ' Python has an easy syntax and us
er-friendly interaction.']
```

Рисунок 4. Результат.

Вывод пример 5

```
<_io.TextIOWrapper name='newfile1.txt' mode='x' encoding='cp1251'>
File created successfully
```

Рисунок 5. Результат.

Вывод пример 6

```
Ничего в этой жизни не дается легко. Чтобы достичь своих целей необходимо
идти на определенные жертвы – тратить свои силы, время, ограничивать себ
я в чем-либо. Иногда бывают моменты, когда хочется все бросить и отказат
ся от мечты. В такие моменты вспомните, как много вы получите, если пойд
е дальше и как много потеряете, если сдадитесь. Цена успеха, как правило
, меньше, чем цена неудачи.
```

Рисунок 6. Результат.

Вывод пример 7

```
The filepointer is at byte : 0
After reading, the filepointer is at: 224
```

Рисунок 7. Результат.

Вывод пример 8

```
None
```

Рисунок 8. Результат.

Вывод пример 9-10

```
f:/2 курс/языки программирования/5/9.py"
PS F:\2 курс\языки программирования\5>
```

Рисунок 9. Результат

Вывод пример 11

```
Number of arguments: 1 arguments
Argument List: ['f:/2 курс/языки программирования/5/11.py']
```

Рисунок 10. Результат

Вывод пример 12

```
Argument #0 is f:/2 курс/языки программирования/5/12.py
No. of arguments passed is 1
```

Рисунок 11. Результат

Индивидуальная 1

```
Написать программу
которая считывает текст из файла
и выводит на экран только предложения
не содержащие запятых.
```

Рисунок 12. Результат

Индивидуальная 2

```
['непробельные']
```

Рисунок 13. Результат

Контрольные вопросы

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

```
fileptr = open("file.txt", "r")
```

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

```
fileptr = open("file.txt", "w")
```

3. Как прочитать данные из файла в языке Python?

```
fileptr = open("file.txt", "r")
```

4. Как записать данные в файл в языке Python?

```
fileptr = open("file.txt", "w")
```

5. Как закрыть файл в языке Python?

```
fileptr.close()
```

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

Конструкция with ... as используется для оборачивания выполнения блока инструкций менеджером контекста.

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

Функция readline() используется для построчного чтения содержимого файла. Она используется для крупных файлов. С ее помощью можно получать доступ к любой строке в любой момент.

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

Обработка путей

Вывод:приобретение навыков по работе с менеджером пакетов pip и виртуальными окружениями с помощью языка программирования Python версии 3.x