

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ ФГАОУ ВО
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Кафедра инфокоммуникаций

Языки программирования

Отчет по лабораторной работе №5

Работа с файлами в языке Python

Выполнил студент группы ИТС-б-о-21-1

Романов Платон Дмитриевич

« » _____ 20__ г.

Подпись студента _____

Проверил: Доцент, к.т.н, доцент
кафедры инфокоммуникаций

Воронкин А. В.

Работа защищена с оценкой: _____

(подпись)

Ставрополь, 2022

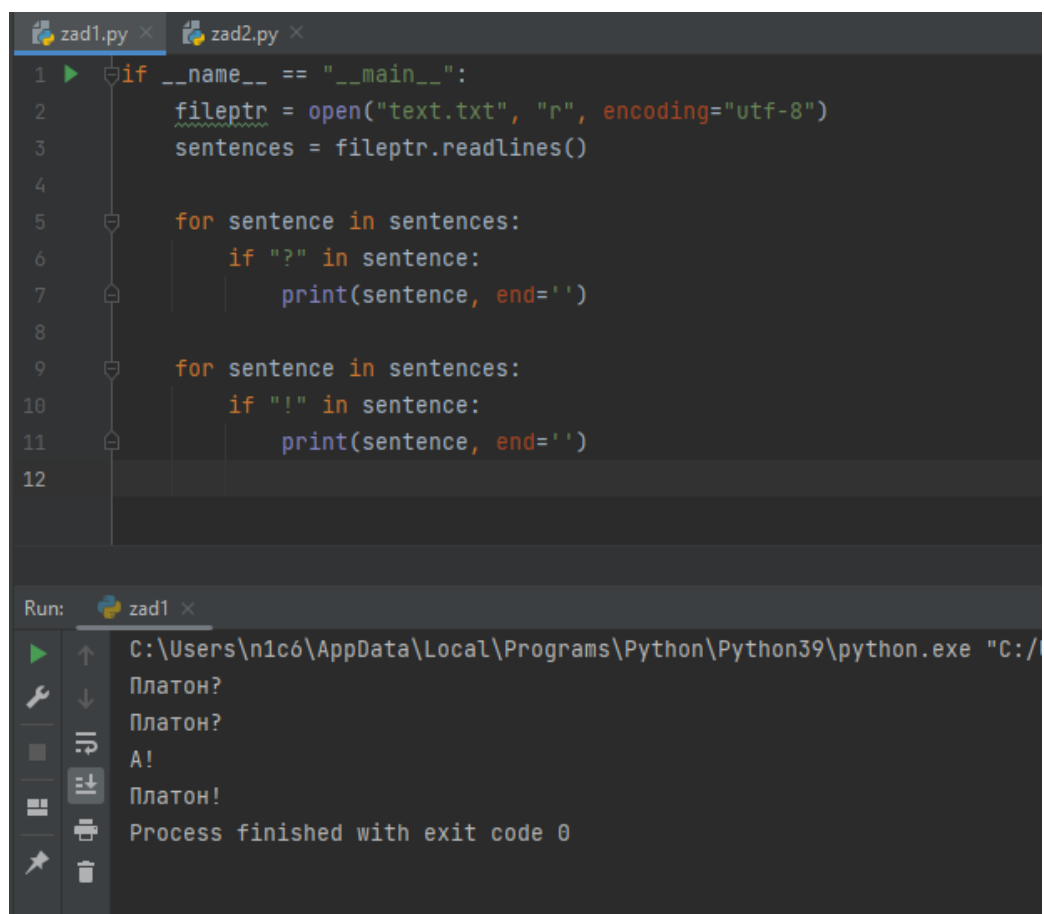
Цель работы: приобретение навыков по работе с текстовыми файлами при написании программ с помощью языка программирования Python версии 3.x, изучение основных методов модуля os для работы с файловой системой, получение аргументов командной строки.

Порядок выполнения работы:

Общедоступный репозиторий на GitHub

Задача 1.

15. Написать программу, которая считывает текст из файла и выводит на экран сначала вопросительные, а затем восклицательные предложения.

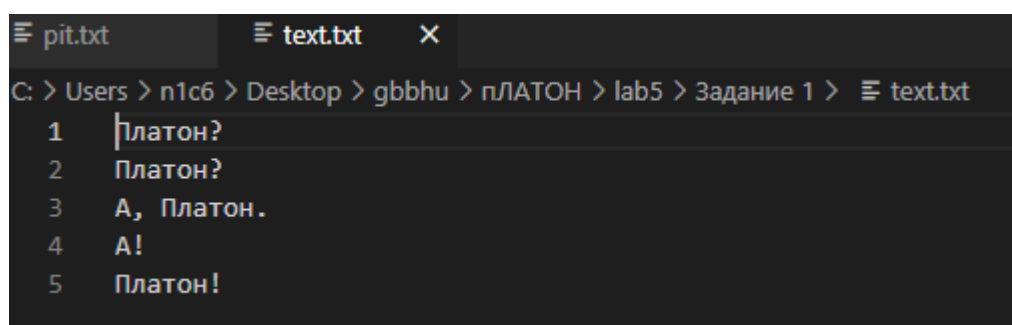


```
1  if __name__ == "__main__":
2      fileptr = open("text.txt", "r", encoding="utf-8")
3      sentences = fileptr.readlines()
4
5      for sentence in sentences:
6          if "?" in sentence:
7              print(sentence, end='')
8
9      for sentence in sentences:
10         if "!" in sentence:
11             print(sentence, end='')
12
```

Run: zad1 x

```
C:\Users\n1c6\AppData\Local\Programs\Python\Python39\python.exe "C:/
Платон?
Платон?
А!
Платон!
Process finished with exit code 0
```

Рисунок 1. Код для первой задачи и проверка его работоспособности.

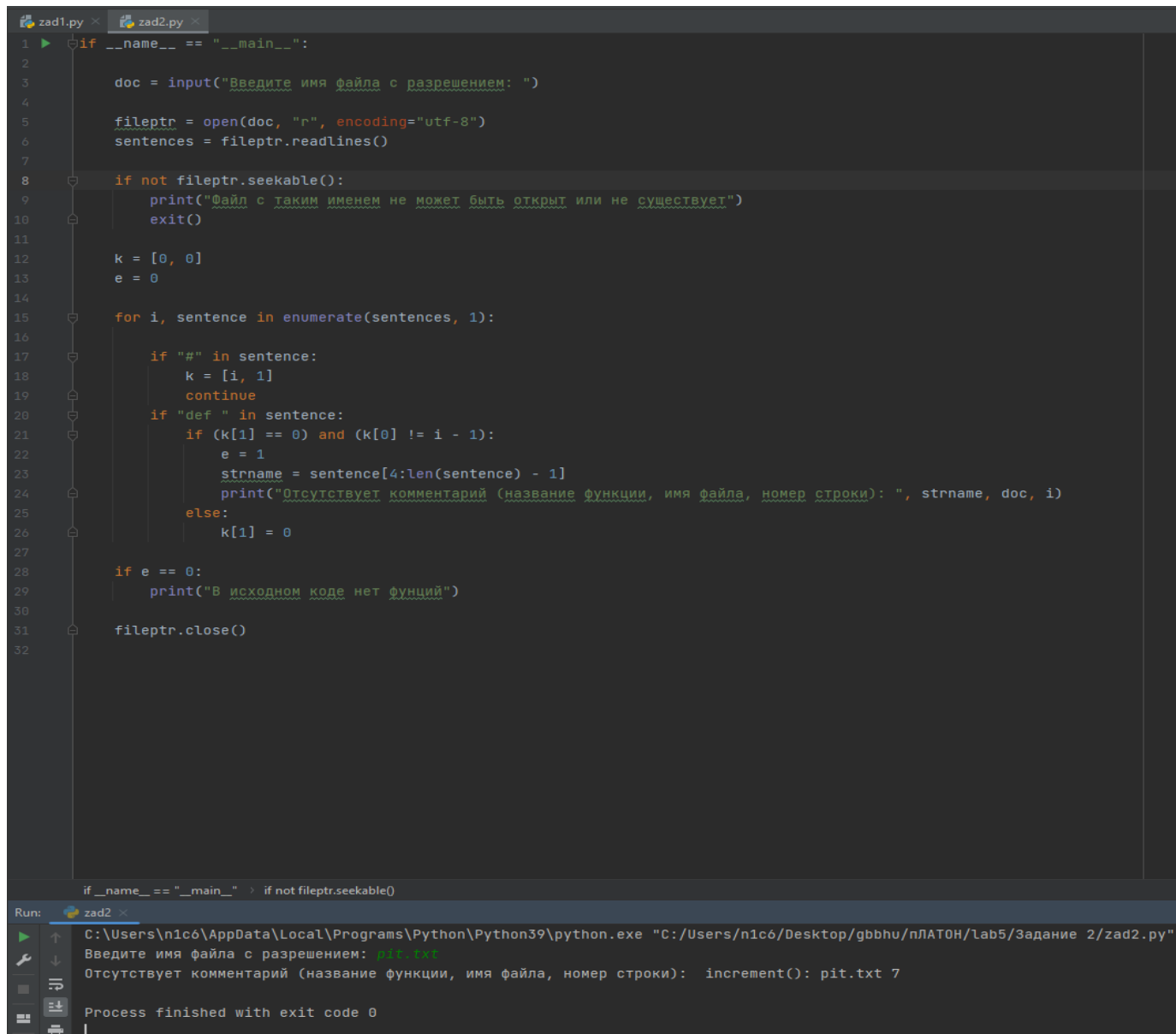


```
1  Платон?
2  Платон?
3  А, Платон.
4  А!
5  Платон!
```

Рисунок 2. Текстовый документ к коду

Задача 2.

15. При написании функций хорошей практикой считается предварение ее блоком комментариев с описанием назначения функции, ее входных параметров и возвращаемого значения. Но некоторые разработчики просто не пишут комментарии к своим функциям. Другие честно собираются написать их когда-нибудь в будущем, но руки так и не доходят. Напишите программу, которая будет проходить по файлу с исходным кодом на Python и искать функции, не снабженные блоком комментариев. Можно принять за аксиому, что строка, начинающаяся со слова `def`, следом за которым идет пробел, будет считаться началом функции. И если функция документирована, предшествующая строчка должна начинаться со знака `#`. Перечислите названия всех функций, не снабженных комментариями, вместе с именем файла и номером строки, с которой начинается объявление функции. Одно или несколько имен файлов с кодом на языке Python пользователь должен передать в функцию в качестве аргументов командной строки. Для файлов, которые не существуют или не могут быть открыты, должны выдаваться соответствующие предупреждения, после чего должна быть продолжена обработка остальных файлов.



```
1  if __name__ == "__main__":
2
3      doc = input("Введите имя файла с разрешением: ")
4
5      fileptr = open(doc, "r", encoding="utf-8")
6      sentences = fileptr.readlines()
7
8      if not fileptr.seekable():
9          print("Файл с таким именем не может быть открыт или не существует")
10         exit()
11
12         k = [0, 0]
13         e = 0
14
15         for i, sentence in enumerate(sentences, 1):
16
17             if "#" in sentence:
18                 k = [i, 1]
19                 continue
20             if "def " in sentence:
21                 if (k[1] == 0) and (k[0] != i - 1):
22                     e = 1
23                     strname = sentence[4:len(sentence) - 1]
24                     print("Отсутствует комментарий (название функции, имя файла, номер строки): ", strname, doc, i)
25                 else:
26                     k[1] = 0
27
28         if e == 0:
29             print("В исходном коде нет функций")
30
31         fileptr.close()
32
```

Run: zad2

C:\Users\n1c6\AppData\Local\Programs\Python\Python39\python.exe "C:/Users/n1c6/Desktop/gbbhu/нЛАТОН/lab5/Задание 2/zad2.py"

Введите имя файла с разрешением: pit.txt

Отсутствует комментарий (название функции, имя файла, номер строки): increment(): pit.txt 7

Process finished with exit code 0

Рисунок 3. Код для второй задачи и проверка его работоспособности

```

C: > Users > n1c6 > Desktop > gbbhu > ПЛАТОН > lab5 > Задание 2 > pit.b
1  #рандомный код
2  def compute_surface(radius):
3      from math import pi
4      return pi * radius * radius
5
6  i = 0
7  def increment():
8      global i
9      i += 1
10
11  #сам не знаю что это
12  def func1(a, b):
13      # f n
14      def inner_func(x):
15          return x*x*x
16
17      return inner_func(a) + inner_func(b)

```

Рисунок 4. Текстовый документ к коду

Ответы на контрольные вопросы:

Контрольные вопросы:

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

```
fileptr = open("Имяфайла", "r")
```

```
with open("file1.txt", "r") as fileptr:
```

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

```
fileptr = open("Имяфайла", "w")
```

```
with open("file1.txt", "w") as fileptr:
```

3. Как прочитать данные из файла в языке Python?

```

# open the file2.txt in read mode. causes error if no such file exists.
fileptr = open("file2.txt", "r")

# stores all the data of the file into the variable content
content = fileptr.read(10)

# prints the type of the data stored in the file
print(type(content))

# prints the content of the file
print(content)

# closes the opened file
fileptr.close()

```

4. Как записать данные в файл в языке Python?

```
# open the file.txt in write mode.
fileptr = open("file2.txt", "a")

# overwriting the content of the file
fileptr.write(" Python has an easy syntax and user-friendly interaction.")

# closing the opened file
fileptr.close()
```

5. Как закрыть файл в языке Python?

```
# closes the opened file
fileptr.close()
```

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 используется для оборачивания выполнения блока инструкций менеджером контекста. Иногда это более удобная конструкция, чем try...except...finally. Самый распространённый пример использования этой конструкции - открытие файлов. with ... as, как правило, является более удобной и гарантирует закрытие файла в любом случае.

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

- `r` – открывает файл в режиме только для чтения. Указатель файла существует в начале. Файл по умолчанию открывается в этом режиме, если не передан режим доступа.
- `rb` – открывает файл в двоичном формате только для чтения. Указатель файла существует в начале файла.
- `r+` – открывает для чтения и записи. Указатель файла также существует в начале.
- `rb+` – открывает в двоичном формате. Указатель файла присутствует в начале файла.
- `w` – только для записи. Он перезаписывает файл, если он существовал ранее, или создает новый, если файл с таким именем не существует. Указатель имеется в начале файла.

- `wb` – открывает файл для записи только в двоичном формате. Перезаписывает файл, если он существует ранее, или создает новый, если файл не существует. Указатель файла существует в начале файла.
- `w+` – для записи и чтения обоих. Он отличается от `r+` в том смысле, что он перезаписывает предыдущий файл, если он существует, тогда как `r+` не перезаписывает ранее записанный файл. Он создает новый файл, если файл не существует. Указатель файла существует в начале файла.
- `wb+` – открывает файл для записи и чтения в двоичном формате. Указатель файла существует в начале файла.
- `a` – открывает файл в режиме добавления. Указатель файла существует в конце ранее записанного файла, если он существует. Он создает новый файл, если не существует файла с таким же именем.
- `ab` – открывает файл в режиме добавления в двоичном формате. Указатель существует в конце ранее записанного файла. Он создает новый файл в двоичном формате, если не существует файла с таким же именем.
- `a+` – открывает файл для добавления и чтения. Указатель файла остается в конце файла, если файл существует. Он создает новый файл, если не существует файла с таким же именем.
- `ab+` – открывает файл для добавления и чтения в двоичном формате. Указатель файла остается в конце файла.

Вывод: приобрел навыки по работе с текстовыми файлами при написании программ с помощью языка программирования Python, основных методов модуля `os` для работы с файловой системой, получение аргументов командной строки.

