

# Основы программирования на языке Python

Тема 13: Введение в файлы.  
Работа с файлами.

# Введение

Считается, что первая книга появилась в древнем Египте примерно в 2400 году до нашей эры. Это событие спровоцировало огромный скачок в развитии всего человечества, так как с помощью книги человек получил возможность передавать свои знания через поколения.

# Введение

Дело в том, что наши программы сейчас словно древние люди: информация не может сохраняться между запусками программы (также как не могла сохраняться после смерти человека).

# Понятие файла

Под файлом в компьютере понимается некоторая именованная область на ПЗУ (постоянном запоминающем устройстве).

# Понятие файловой системы

Под файловой системой понимается принцип логической организации данных на ПЗУ. Например, файловая система отвечает на вопросы: где начинается файл и где он заканчивается, часть какого файла лежит в конкретной области ПЗУ, как расположены части некоторого конкретного файла в ПЗУ и многие другие. Некоторый участок памяти в ПЗУ становится файлом именно в контексте конкретной файловой системы.

# Хранение файлов

Часто файл разбивается на блоки (к примеру, по 4 кб), эти блоки могут храниться в разных частях ПЗУ. Это нужно для ускорения некоторых операций, вроде



В файловой системе:

Файл 1: 1, 2, 5, 6

Файл 2: 3, 4

# Хранения файла

Доступ к файлам замедляется из-за их фрагментации. Особенно это плохо для жестких дисков, так как в нем есть подвижные части, которым нужно время, чтобы переместиться, и еще они подвержены износу.

# Виртуальные файлы

В Unix системах существуют, так называемые, виртуальные файлы. Работа с ними происходит, как с обычными файлами так, будто они находятся на ПЗУ компьютера. Но на самом деле это не так, а считываемые данные из таких файлов генерируются операционной системой. Например, в Unix-подобных системах существует файл `/dev/random`. Считанные из него байты являются случайной (или псевдослучайной) последовательностью.

# Стандартный поток ввода

```
from sys import stdin
```

```
from_console = stdin.read()  
print(from_console)
```

```
Привет, как дела?
```

```
Все хорошо
```

```
^D
```

```
Привет, как дела?
```

```
Все хорошо
```

```
|
```

```
Process finished with exit code 0
```

Для окончания ввода:

В PyCharm – ctrl+D

В cmd – ctrl+C

# Открытие и закрытие файла

```
file = open('file.txt', encoding='UTF-8-sig')
```

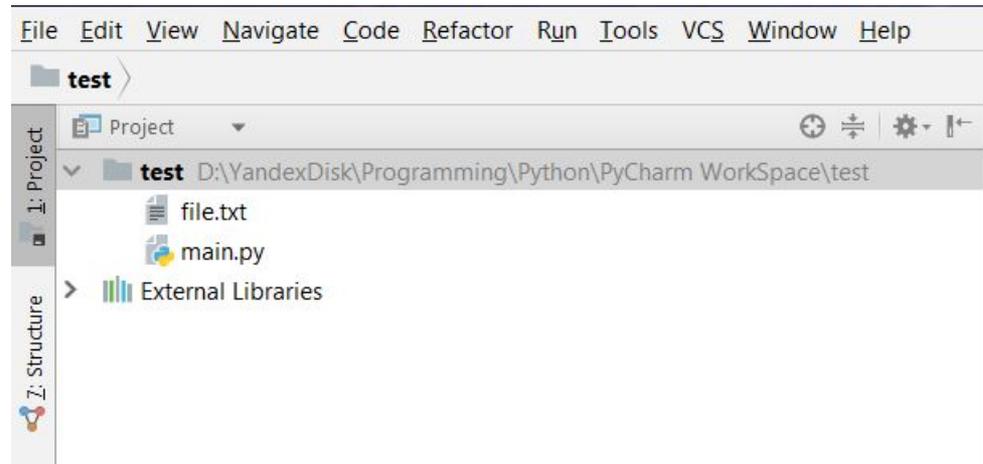
```
file_content = file.read()
```

```
file.close()
```

```
print(file_content)
```

```
В чащах юга жил был цитрус?  
Да, но фальшивый экземпляр!
```

```
Process finished with exit code 0  
|
```



# Другие способы чтения файла

```
file = open('file.txt', encoding='UTF-8-sig')
```

```
file_lines = file.readlines()
```

```
print(file_lines)
```

```
file.close()
```

```
['В чащах юга жил был цитрус? \n', 'Да, но фальшивый экземпляр!']
```

```
Process finished with exit code 0
```

# Другие способы чтения файла

```
file = open('file.txt', encoding='UTF-8-sig')
```

```
file_lines = [x.rstrip('\n') for x in file.readlines()]  
print(file_lines)
```

```
file.close()
```

```
['В чащах юга жил был цитрус? ', 'Да, но фальшивый экземпляр!']
```

```
Process finished with exit code 0
```

# Другие способы чтения файла

```
file = open('file.txt', encoding='UTF-8-sig')
```

```
for line in file: # то же, что и for line in file.readlines():  
    print(line)
```

```
file.close()
```

```
В чащах юга жил был цитрус?
```

```
Да, но фальшивый экземпляр!
```

```
Process finished with exit code 0
```

```
|
```

# Менеджер контекста

```
file = open('file.txt', encoding='UTF-8-sig')
```

```
word_counter = 0
```

```
for line in file:
```

```
    word_counter += len(line.split())
```

```
print(word_counter)
```

```
file.close()
```

```
with open('file.txt', encoding='UTF-8-sig') as file:
```

```
    word_counter = 0
```

```
    for line in file:
```

```
        word_counter += len(line.split())
```

```
    print(word_counter)
```

# Запись в текстовом режиме

```
with open('new_file.txt', mode='w', encoding='UTF-8') as file:  
    for i in range(10):  
        file.write(str(i))
```

# Режимы открытия файлов

Режим	Положение указателя	Чтение или запись
r (по умолчанию)	В начале	Чтение
r+	В начале	Чтение/Запись
w	В начале	Запись
w+	В начале	Чтение/Запись
a	В конце	Запись
a+	В конце	Чтение/Запись

При этом в при открытии файла в режиме r+, файл должен существовать, а при открытии в режиме w+ файл будет создан, если его не существует.

# Стандартны поток вывода

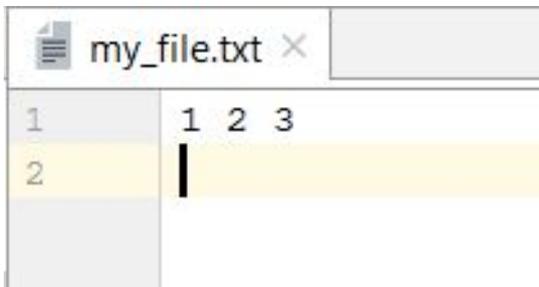
```
import sys
```

```
sys.stdout.write('Пишу в консоль без print!')  
sys.stdout.write('Еще одна строка.')
```

```
Пишу в консоль без print!Еще одна строка.  
Process finished with exit code 0  
|
```

# Запись в файл с помощью print

```
with open('my_file.txt', mode='w') as file:  
    print(1, 2, 3, file=file)
```



```
my_file.txt x  
1 1 2 3  
2 |
```

# Своя реализация функции print

```
import sys
```

```
to_print = [1, 2, 3, 'Строка']  
sep = ''  
end = '\n'  
file = sys.stdout
```

```
file.write(sep.join([str(x) for x in to_print]) + end) # это происходит внутри функции print
```

```
to_print = ['Еще строка', 0.5]  
file.write(sep.join([str(x) for x in to_print]) + end)
```

```
1 2 3 Строка
```

```
Еще строка 0.5
```

```
Process finished with exit code 0
```

```
█
```

# Основы программирования на языке Python

Тема 13: Введение в файлы.  
Работа с файлами.