

Открытие и чтение файла, запись в файл



Адилет
Асанкожоев



Адилет Асанкожоев

Python-разработчик в Makers.kg



План занятия

1. [Что такое файл?](#)
2. [Работа с файлами в Python](#)
3. [Основные ошибки при работе с файлами](#)
4. [Домашнее задание](#)



Что такое файл?

Files are everywhere

- хранение данных программы
- файлы конфигурации
- текстовые файлы
- картинки
- итд

Файл — именованная область данных на диске.

track.mp3

diplom.doc

100010**1001010101010101**010101100101010101**0101010101010101101**010101

File object в Python — объект, позволяющий работать с данными, как с файлом.

<https://docs.python.org/3/glossary.html#term-file-object>



Работа с файлами в Python

Модель работы с любыми ресурсами

Не только с файлами

- открыть
- основная логика:
прочитать данные, записать данные
- закрыть

- `f = open(Имя файла, режим)`
- `f.write(...), f.read()`
- `f.close()`

[Ссылка на Repl.it](#)



Упрощаем работу с ресурсами

С помощью конструкции `with` (менеджер контекста) Python сам проследит за закрытием файла

```
with open('test.txt') as f:
```

[Ссылка на Repl.it](#)

Методы объекта `File` для чтения

`read` – чтение всего файла целиком

`readline` – строчное чтение

`readlines` – чтение всех строк в список

<https://docs.python.org/3.8/tutorial/inputoutput.html#methods-of-file-objects>

<https://www.programiz.com/python-programming/file-operation>

```
1  with open('data.txt', 'r') as f:
2      |  for line in f:
3          |  |  print(f)
```

[Ссылка на Repl.it](#)

Запись в файл

```
1 with open('file.txt', 'w') as document:  
2     document.write('hello world!')  
3
```

Режимы работы с файлом

Режимы доступа

'r' — чтение (по-умолчанию)

'w' — запись

'a' — запись в конец файла

```
open(Имя файла, режим)
```

Режимы чтения/записи

'b' — двоичный режим (работаем с байтами)

't' — текстовый режим (работаем с строками, по-умолчанию)

[Ссылка на Repl.it](https://repl.it)



Основные ошибки при работе с файлами

Путь к файлу в иерархии

1. Абсолютный (от корневого каталога)
2. Относительный (от текущего каталога)

1. Нужен временный файл или конкретный путь не задан?

Используйте относительные пути

2. Работаете с конкретной директорией?

Используйте абсолютные пути

3. Помните про ОС

Windows - C:\Users\ddemidov\...

Linux - /home/users/ddemidov...

[Ссылка на Repl.it](#)

Методы для работы с путями

```
import os  
  
os.getcwd() #получить абсолютный путь текущего каталога  
os.path.join(path, *paths) #построение платформоспецифичного пути  
os.path.dirname(path) # абсолютный путь до каталога  
os.path.basename(path) # имя файла
```

Интерфейсы операционной системы:

<https://docs.python.org/3/library/os.html#module-os>

Манипуляции с путями:

<https://docs.python.org/3/library/os.path.html>



Вложенные with

Более подробно про менеджеры контекста будет в одной из ближайших лекций.

А пока свежий пример: [ссылка на Repl.it](#)

Кодировки

- Для компьютера текст (как и любые другие данные) — это набор байтов.
- Мы (люди) хотим работать со строками, байты нам читать сложно.
- Кодировка — это способ сопоставления байтов символам.
- По умолчанию в Python при работе с файлами используется utf-8.
- Для изменения кодировки в функции `open` есть параметр `encoding`.

[Ссылка на Repl.it](#)

Обзор Unicode в официальной документации:

<https://docs.python.org/3.8/howto/unicode.html>



Домашнее задание

Давайте посмотрим ваше [домашнее задание](#).

- Вопросы по домашней работе задаём в чат
- Задачи можно сдавать по частям.
- Зачёт по домашней работе предоставляется после того, как приняты **все задачи**.

**Задавайте вопросы и
пишите отзыв о лекции!**

Адилет Асанкожоев