**Лекция 8**

# **Файлы в python, ввод-вывод**

Файл — это всего лишь набор данных, сохраненный в виде последовательности битов на компьютере. Информация хранится в куче данных (структура данных) и имеет название «имя файла» (filename).

В Python существует два типа файлов:

1. Текстовые
2. Бинарные

**1 Текстовые файлы**

Это файлы с человекочитаемым содержимым. В них хранятся последовательности символов, которые понимает человек. Блокнот и другие стандартные редакторы умеют читать и редактировать этот тип файлов.

Текст может храниться в двух форматах: (*.txt*) — простой текст и (*.rtf*) — «формат обогащенного текста».

**2 Бинарные файлы**

В бинарных файлах данные отображаются в закодированной форме (с использованием только нулей (0) и единиц (1) вместо простых символов). В большинстве случаев это просто последовательности битов.

Они хранятся в формате (*.bin*).

Любую операцию с файлом можно разбить на три крупных этапа:

1. Открытие файла
2. Выполнение операции (запись, чтение)
3. Закрытие файла

**3 Открытие файла, *open()***

В Python есть встроенная функция *open()*. С ее помощью можно открыть любой файл на компьютере. Технически Python создает на его основе объект.

Синтаксис следующий:



Где,

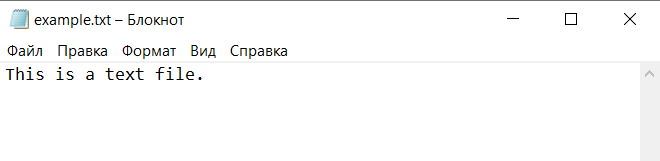
* file\_name = имя открываемого файла *(указывается в кавычках)*
* access\_mode = режим открытия файла. Он может быть: для чтения, записи и т. д. По умолчанию используется режим чтения (**r**), если другое не указано. *(указывается в кавычках)*

Всего режимов открытия файла 12, далее полный список режимов открытия файла:

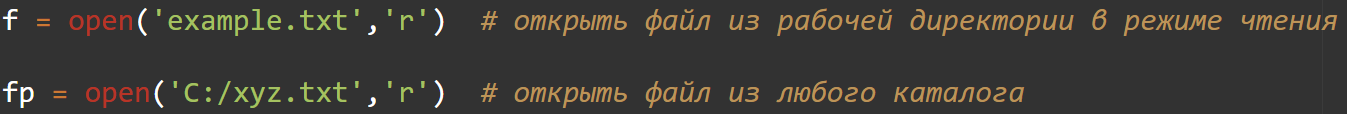
| Режим | Описание |
| --- | --- |
| r | Только для чтения. |
| w | Только для записи. Создаст новый файл, если не найдет с указанным именем. |
| rb | Только для записи. Создаст новый файл, если не найдет с указанным именем. |
| wb | Только для записи (бинарный). Создаст новый файл, если не найдет с указанным именем. |
| r+ | Для чтения и записи. |
| rb+ | Для чтения и записи (бинарный). |
| w+ | Для чтения и записи. Создаст новый файл для записи, если не найдет с указанным именем. |
| wb+ | Для чтения и записи (бинарный). Создаст новый файл для записи, если не найдет с указанным именем. |
| a | Откроет для добавления нового содержимого. Создаст новый файл для записи, если не найдет с указанным именем. |
| a+ | Откроет для добавления нового содержимого. Создаст новый файл для чтения записи, если не найдет с указанным именем. |
| ab | Откроет для добавления нового содержимого (бинарный). Создаст новый файл для записи, если не найдет с указанным именем. |
| ab+ | Откроет для добавления нового содержимого (бинарный). Создаст новый файл для чтения записи, если не найдет с указанным именем. |

### Пример

Создадим текстовый файл example.txt и сохраним его в рабочей директории.



Следующий код используется для его открытия:



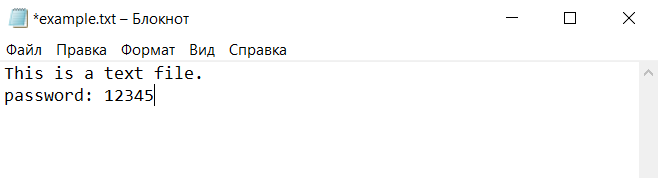
В этом примере f— переменная-указатель на файл example.txt.

**4 Считывание строк**

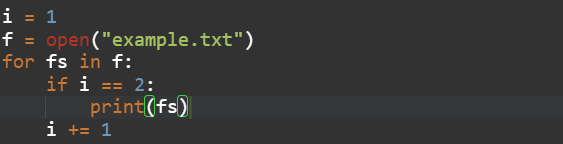
Для того чтобы работать с отдельными строками из файла можно воспользоваться функцией For

Пример

Немного изменим наш текстовый документ example.txt , добавим в него еще одну строку



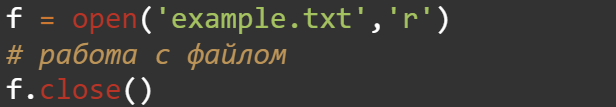
Следующий код используется для вывода 2 строки:



**5 Закрытие файла, *close()***

После открытия файла в Python его нужно закрыть. Таким образом освобождаются ресурсы и убирается мусор. Python автоматически закрывает файл, когда объект присваивается другому файлу.

Закрыть файл можно с помощью функции *close()*:



После закрытия этот файл нельзя будет использовать до тех пор, пока заново его не открыть.