

Теоретический материал к занятию Работа с файлами 2 занятие.

Еще один подход при работе с файлами — использовать инструкцию `with`, которая упрощает обработку ошибок, а также задач по закрытию и очистке.

В таком случае инструкция `close` не нужна, потому что `with` автоматически закрывает файл.

Вот как это реализовать в коде.

```
with open('example.txt') as file:  
    print(file.read())
```

Как мы видим в данном случае мы не использовали `close()`. В целом остальная работа с файлом проходит также.

Давайте теперь познакомимся ближе с функциями чтения и записи данных в файл

Функция `read()` используется для чтения содержимого файла после открытия его в режиме чтения (`r`).

```
file.read(size)
```

- `file` = объект файла
- `size` = количество символов, которые нужно прочитать. Если не указать, то файл прочитается целиком.

```
f = open('example.txt', 'r')  
f.read(7)  # чтение 7 символов из example.txt
```

```
f = open('example.txt', 'r')  
f.read(7)  # чтение 7 символов из example.txt  
f.read(7)  # чтение следующих 7 символов
```

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.

```
x = open('example.txt', 'r')  
x.readline()  # прочитать первую строку  
x.readline(2)  # прочитать вторую строку  
x.readlines()  # прочитать все строки
```

Функция `write()` используется для записи в файлы Python, открытые в режиме записи.

Если пытаться открыть файл, которого не существует, в этом режиме, тогда будет создан новый.

Предположим, файла example2.txt не существует. Он будет создан при попытке открыть его в режиме записи.

```
f = open('example2.txt','w') # открытие в режиме записи
f.write('Hello \n World') # запись Hello World в файл
f.close() # закрытие файла
```

Методы файла:

file.close()	закрывает открытый файл
file.flush()	очищает внутренний буфер
file.next()	возвращает следующую строку файла
file.read(n)	чтение первых n символов файла
file.readline()	читает одну строчку строки или файла
file.readlines()	читает и возвращает список всех строк в файле
file.tell()	возвращает текущую позицию в файле
file.truncate(n)	уменьшает размер файл. Если n указала, то файл обрезается до n байт, если нет – до текущей позиции
file.write(str)	добавляет строку str в файл
file.writelines(sequence)	добавляет последовательность строк в файл