**Работа с файлами**

Файлы, в общем случае, делятся на два вида:

* **Текстовые** файлы – текст, разбитый на строки, состоящий из различных символов. Из специальных символов данный тип файлов содержит только символы перехода на новую строку ‘\n’
* **Двоичные** файлы – может содержать любые символы, также может являться рисунками, звуками, видео, в общем, любой файл можно открыть как двоичный.

Работа с файлами на любом языке программирования строится по принципу сэндвича: хлебом (верхней и нижней частью) являются процедуры открытия и закрытия файлов, а начинкой (то, что находится между хлебом) является работой с файлом. Именно в начинке происходит чтение информации из файла и её обработка



Файлы можно открыть для следующих манипуляций:

* **Чтение** информации из файла (для того, чтобы файл можно было прочесть, он должен существовать)
* **Запись** информации в файл (вся информация, которая находилась в файле удаляется и записывается новая информация, если файл до этого не существовал, то он создается)
* **Добавление** информации в файл (информация в файл добавляется в самый конец, после старой информации, которая была в файле)

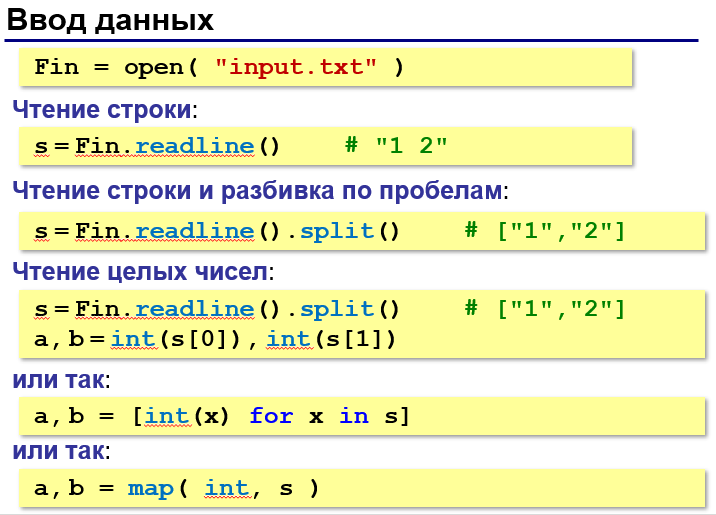
На языке Python открытие файла делается при помощи функции open(file\_name, mode, encoding), где параметрами являются:

* **file\_name** – название файла и путь к нему
* **mode** – режим открытия файла (чтение, запись, добавление и т.п)
* **encoding** – кодировка, в которой будет происходить считывание файла (utf-8, utf-16, ASCII и т.п.)

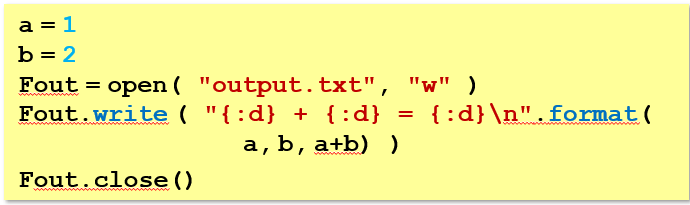
Закрытие файла выполняется при помощи метода close(), вызываемой без параметров.



**Чтение данных** осуществляется несколькими способами:

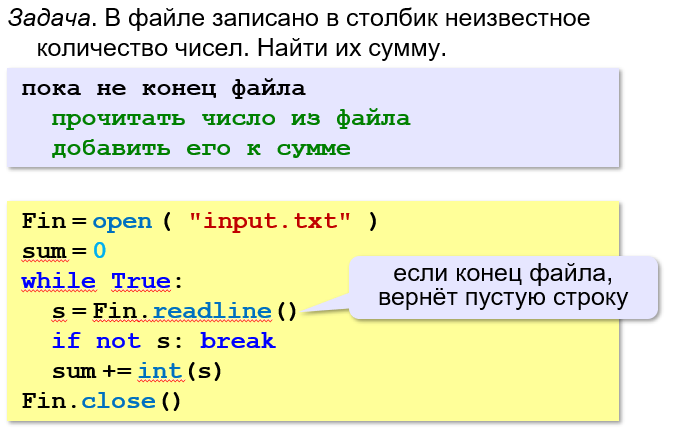


**Запись данных** в файл осуществляется следующим образом:

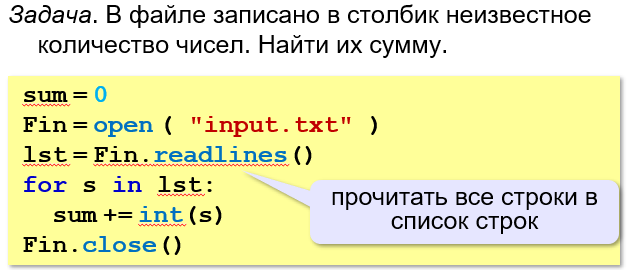


Следует отметить, что перед записью информации в файл, необходимо преобразовать всю записываемую информацию в строковый формат.

Выше был описан способ, когда нам заранее известно количество строк, которые необходимо прочитать из файла. Но чаще всего перед нами стоит задача **читать информацию из файла, в котором заранее неизвестно количество строк**. Для решения данной задачи можно использовать следующую конструкцию (в ней решается задача, в которой необходимо посчитать сумму цифр, записанных в файл по одному в строке и их количество неизвестно)



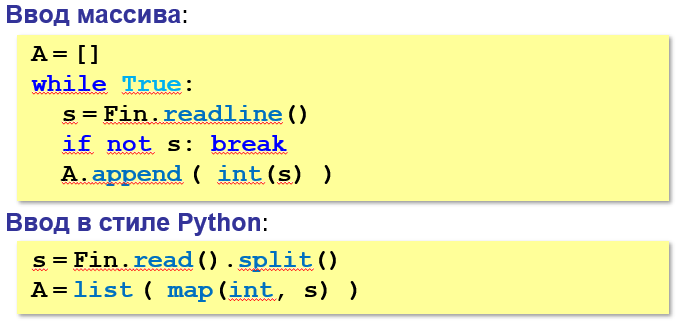
Есть **ещё один способ**, считать все числа из файла разом и добавить их в список, затем перевести все строки, записанные в массив, в числовой тип данных и найти их сумму.



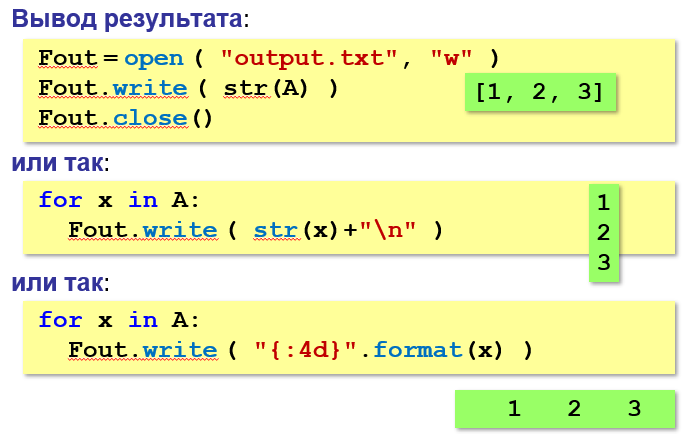
Также, есть **ещё один способ** решения данной задачи. Для этого циклу for в качестве итерируемого элемента можно передать результат работы функции open(). После считывания информации данным способом, не нужно закрывать файл, он закроется автоматически.



Если **данные из файла необходимо записать в список** и обработать их как единый набор данных (например, отсортировать), то можно использовать следующую конструкцию



Если же необходимо **записать информацию из списка в файл**, то можно использовать следующие конструкции



**Задачи:**

**№1**

Получить 10 случайных четных чисел и записать их в файл chet.txt

**№2**

Найти сумму всех чисел, записанных в файл chet.txt

**№3**

Переписать все числа из файла chet.txt, кратные 6 в файл chet6.txt

**№4**

Напишите программу, которая находит среднее арифметическое всех чисел, записанных в файле в столбик, и выводит результат в другой файл.

**№5**

Напишите программу, которая находит минимальное и максимальное среди чётных положительных чисел, записанных в файле, и выводит результат в другой файл. Учтите, что таких чисел может вообще не быть.

**№6**

В файле в столбик записаны целые числа, сколько их – неизвестно. Напишите программу, которая определяет длину самой длинной цепочки идущих подряд одинаковых чисел и выводит результат в другой файл.