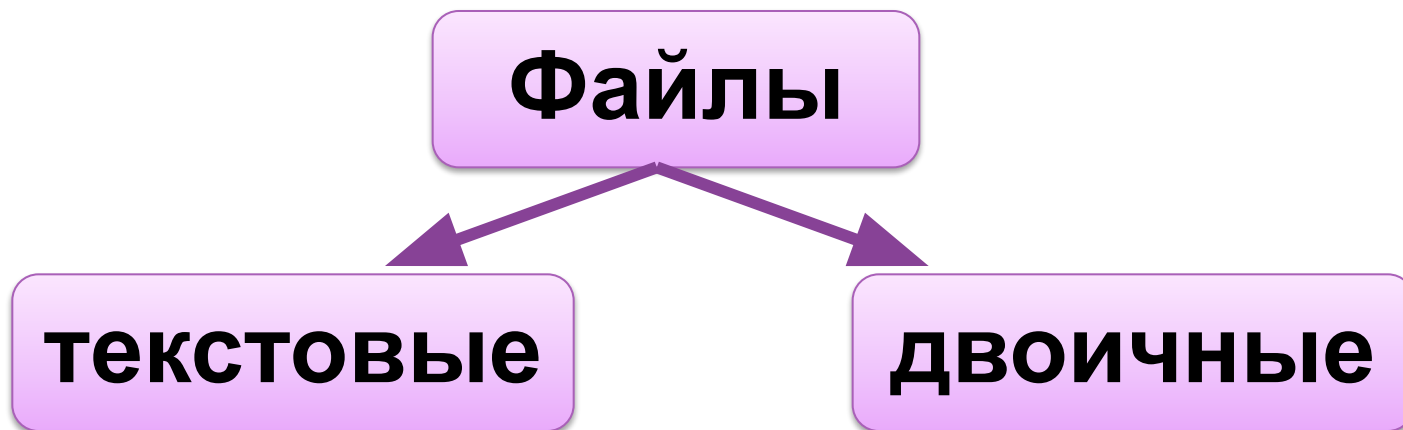


Работа с файлами



Какие бывают файлы?



«*plain text*»:

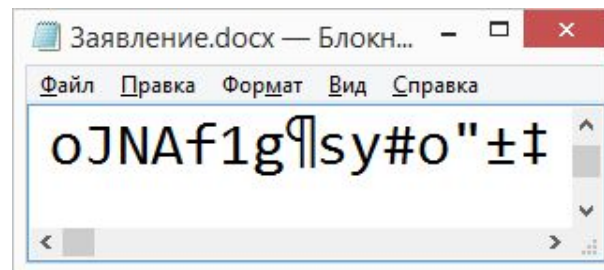
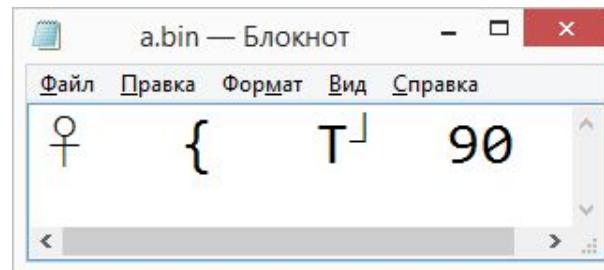
- для чтения человеком
- текст, разбитый на строки;
- из специальных символов только символы перехода на новую строку

12

123

1234

- любые символы
- рисунки, звуки, видео, ...



ПРИНЦИП СЭНДВИЧА



файловые переменные-
указатели

по умолчанию – на
чтение (режим "r")

```
Fin = open ("input.txt")  
Fout = open ("output.txt", "w")  
    # здесь работаем с файлами  
Fin.close()  
Fout.close()
```

"r" – чтение
"w" – запись
"a" – добавление

Ввод данных

```
Fin = open("input.txt")
```

Чтение строки:

```
s = Fin.readline()      # "1 2"
```

Чтение строки и разбивка по пробелам:

```
s = Fin.readline().split()      # ["1", "2"]
```

Чтение целых чисел:

```
s = Fin.readline().split()      # ["1", "2"]  
a, b = int(s[0]), int(s[1])
```

Или с помощью генератора:

```
a, b = [int(x) for x in s]
```

или так:

```
a, b = map(int, s)
```

Вывод данных в файл

```
a = 1
```

```
b = 2
```

```
Fout = open( "output.txt", "w" )
```

```
Fout.write( "{:d} + {:d} = {:d}\n".format(  
            a, b, a+b) )
```

```
Fout.close()
```



Все данные преобразовать в строку!



Чтение неизвестного количества данных

Задача. В файле записано в столбик неизвестное количество чисел. Найти их сумму.

пока не конец файла
прочитать число из файла
добавить его к сумме

```
Fin = open ("input.txt")  
sum = 0  
while True:  
    s = Fin.readline ()  
    if not s: break  
    sum += int (s)  
Fin.close ()
```

если конец файла,
вернёт пустую
строку

Чтение неизвестного количества данных

Задача. В файле записано в столбик неизвестное количество чисел. Найти их сумму.

```
sum = 0
Fin = open ("input.txt")
lst = Fin.readlines ()
for s in lst:
    sum += int (s)
Fin.close ()
```

прочитать все строки
в список строк

Чтение неизвестного количества данных

Задача. В файле записано в столбик неизвестное количество чисел. Найти их сумму.

```
sum = 0
with open ("input.txt") as Fin:
    for s in Fin:
        sum += int(s)
```

или так:

```
sum = 0
for s in open ("input.txt") :
    sum += int(s)
```



Не нужно закрывать файл!

ОБРАБОТКА МАССИВОВ

Задача. В файле записаны в столбик целые числа. Вывести в другой текстовый файл те же числа, отсортированные в порядке возрастания.



В чем отличие от предыдущей задачи?



Для сортировки нужно удерживать все элементы в памяти одновременно.

ОБРАБОТКА МАССИВОВ

Ввод массива:

```
A = []  
while True:  
    s = Fin.readline()  
    if not s: break  
    A.append(int(s))
```

Ввод в стиле Python:

```
s = Fin.read().split()  
A = list(map(int, s))
```

Сортировка:

```
A.sort()
```



ОБРАБОТКА МАССИВОВ

Вывод результата:

```
Fout = open ("output.txt", "w")  
Fout.write (str(A))  
Fout.close()
```

[1, 2, 3]

или так:

```
for x in A:  
    Fout.write (str(x) + "\n")
```

1
2
3

или так:

```
for x in A:  
    Fout.write ("{:4d}".format(x))
```

1 2 3

Обработка строк

Задача. В файле записано данные о собаках: в каждой строке кличка собаки, ее возраст и порода: **Мухтар 4 немецкая овчарка**

Вывести в другой файл сведения о собаках, которым меньше 5 лет.

пока не конец файла Fin

 прочитать строку из файла Fin

 разобрать строку – выделить возраст

 если возраст < 5 то

 записать строку в файл Fout

ЧТЕНИЕ ДАННЫХ ИЗ ФАЙЛА

Чтение одной строки:

```
s = Fin.readline ()
```

Разбивка по пробелам:

```
data = s.split ()
```

Выделение возраста:

```
sAge = data [1]
```

```
age = int (sAge)
```

Кратко всё вместе:

```
s = Fin.readline ()
```

```
age = int (s.split () [1])
```

Обработка строк

Полная программа:

```
Fin = open ("input.txt")
Fout = open ("output.txt", "w")
while True:
    s = Fin.readline()
    if not s: break
    age = int (s.split() [1])
    if age < 5:
        Fout.write (s)
Fin.close()
Fout.close()
```

Обработка строк

или так:

```
lst = Fin.readlines ()
for s in lst:
    age = int ( s.split() [1])
    if age < 5:
        Fout.write (s)
```

или так:

```
for s in open ("input.txt") :
    age = int (s.split() [1])
    if age < 5:
        Fout.write (s)
```

**Спасибо за
внимание!**