Ввод данных

Для считывания строки со стандартного ввода используется функция input(), которая считывает строку с клавиатуры и возвращает значение считанной строки, которое сразу же можно присвоить переменным:

```
a = input()

b = input()
```

Правда, функция input возвращает текстовую строку. Если нужно сделать так, чтобы переменные имели целочисленные значения, то сразу же после считывания выполним преобразование типов при помощи фунцкии int, и запишем новые значения в переменные a и b:

```
a = int(a)

b = int(b)
```

Можно объединить считывание строк и преобразование типов, если вызывать функцию int для того значения, которое вернет функция input():

```
a = int(input())

b = int(input())
```

Сложнее считать значения переменных, если они записаны в отдельной строке. Здесь нужно применить к считанной строке метод split(), который разделяет строку на части по одному или двум пробелам. Затем результат выполнения этой функции присвоим кортежу из двух или нескольких чисел. Например, если в строке вводятся два числа через пробел, то считать их можно так:

```
a, b = input().split()

a = int(a)

b = int(b)
```

Аналогично, три переменные можно считать, записав слева от оператора присваивания кортеж из трех переменных:

```
a, b, c = input().split()
```

Можно также сразу же преобразовать считанные значения в числовой тип (например, int), если воспользоваться функцией map, которая применяет к каждому элементу списка заданную функцию (для преобразования к типу int нужно, соответственно, задать функцию int для применения к каждому элементу). Для начала можно просто запомнить эту конструкцию:

```
a, b, c = map(int, input().split())
```

Вывод данных

Для вывода данных используется функция print может выводить не только значения переменных, но и значения любых выражений. Например, допустима запись print(2 + 2 ** 2). Также при помощи функции print можно выводить значение не одного, а нескольких выражений, для этого нужно перечислить их через запятую:

```
a = 1

b = 2

print(a, '+', b, '=', a + b)
```

В данном случае будет напечатан текст 1 + 2 = 3: сначала выводится значение переменной a, затем строка из знака “+”, затем значение переменной b, затем строка из знака “=”, наконец, значение суммы a + b.

Обратите внимание, выводимые значение разделяются одним пробелом. Но такое поведение можно изменить: можно разделять выводимые значения двумя пробелами, любым другим символом, любой другой строкой, выводить их в отдельных строках или не разделять никак. Для этого нужно функции print передать специальный именованный параметр, называемый sep, равный строке, используемый в качестве разделителя (sep — аббревиатура от слова separator, т.е. разделитель). По умолчанию параметр sep равен строке из одного пробела и между значениями выводится пробел. Чтобы использовать в качестве разделителя, например, символ двоеточия нужно передать параметр sep, равный строке ‘:’:

```
print(a, b, c, sep = ':')
```

Аналогично, для того, чтобы совсем убрать разделитель при выводе нужно передать параметр sep, равный пустой строке:

```
print(a, '+', b, '=', a + b, sep = '')
```

Для того, чтобы значения выводились с новой строке, нужно в качестве параметра sep передать строку, состоящую из специального символа новой строки, которая задается так:

```
print(a, b, sep = '\n')
```

Символ обратного слэша в текстовых строках является указанием на обозначение специального символа, в зависимости от того, какой символ записан после него. Наиболее часто употребляется символ новой строки ‘\n’. А для того, чтобы вставить в строку сам символ обратного слэша, нужно повторить его два раза: ‘\\’.

Вторым полезным именованным параметром функции print является параметр end, который указывает на то, что выводится после вывода всех значений, перечисленных в функции print. По умолчанию параметр end равен ‘\n’, то есть следующий вывод будет происходить с новой строки. Этот параметр также можно исправить, например, для того, чтобы убрать все дополнительные выводимые символы можно вызывать функцию print так:

```
print(a, b, c, sep = '', end = '')
```

Источник: https://foxford.ru/wiki/informatika/vvod-vyvod-v-python