**print**() может распечатать:

строки, числа, математические операции с числами (print(5+5)), переменные, математические операции с переменными (print(a+b)), сложение строк.

**str**() В круглые скобки принимает число, делает его строкой.

**int**() В круглые скобки принимает строку, делает ее числом.

**input**() В круглые скобки принимает строку, которую нужно распечатать перед тем как считать значение с клавиатуры и затем считывает (запоминает) строчку, которую ввел пользователь. Всегда перед командой input стоит переменная, в которую и кладется считанная строчка.

**!Так как по умолчанию считывается именно строка, чтобы считать число, нужно воспользоваться командой int()**

**ПРИМЕР: number=int(input())**

Не обязательно в круглые скобки передавать строку. Можно просто считать значение ничего не выводя перед этим.

**ПЕРЕМЕННЫЕ:**

a = 2

b = ”geography”

нельзя складывать, вычитать и сравнивать строки и числа (также строковые и численные переменные)

**УСЛОВИЯ:**

If b == 100:

print(“you win”)

если выполняется условие:

выполнить этот код

обрати внимание, что этот код находится на расстоянии отступа (здесь находится все, что нужно сделать при выполнении условия

в паре с if может идти слово else (иначе)

внутри else записывается код, который нужно выполнить, если не выполняется основное условие в if

year = int(input("В каком году вы родились?"))

if year < 2009:

print("you are older than me")

else:

print("you are younger")

**ЦИКЛЫ:**

***Цикл это многократно повторяемый участок кода***

**Цикл while**

a=10

**while** a>0:  
 a=a-1  
 print(a)

эти две строчки будут выполняться пока выполняется условие a > 0 верно.

Как только a станет равно 0, цикл закончится

**Цикл for**

word = “apple”

for char in word:

print(char)

Этот цикл проходится по каждому символу слова по очереди. Будет выведено:  
a  
p  
p  
l  
e

ВАЖНО! Вместо char мы можем написать любое слово или символ, ведь это всего лишь временная переменная, которая создаётся на время выполнения цикла, и она хранит в себе по очереди все символы слова.

Мы могли написать:

for letter in word:  
 for a in word:  
 for qwerty in word:

Главное ставить двоеточие в конце этой строки.

Также цикл for может проходиться по элементам списка:

fruits = [“grape”, “orange”, “lemon”]

for element in fruits:

print(element)

Будет напечатано:  
grape  
orange  
lemon

Цикл for может многое. Он не обязательно должен проходить по элементам (символам) какого-то списка, или слова. Он может просто прокрутить код, который находится внутри него столько раз, сколько мы попросим. Например:

for i in range(10):

if i < 5:

print(“Привет, я меньше 5”)

else:

print(“Привет, я больше 5”)

То есть блок с условием будет “прокручен” 10 раз.

Как это работает? У нас создаётся временная переменная i,   
которая пробегает значения от 0 (включительно) до 10 (не включительно). С каждым проходом цикла эта переменная сама увеличивается на 1. То есть как только i дойдёт до 10, цикл закончится сам.

Запомним команду range(). Она принимает в круглые скобки число, сколько раз нужно повторять цикл.

Также команде range() можно передавать значение, с которого нужно начать, например:

for i in range(5, 10):

Переменная i сразу будет равна 5.

Также можно задать шаг, с которым будет идти цикл.  
Это третий параметр в команде range()

for i in range(5, 20, 3):  
 print(i)

Будет распечатано:  
5  
8  
11  
14  
17

Помним, что последнее значение не включительно, поэтому 20 распечатано не будет.

**СПИСКИ**:

Создать список просто:

spisok1 = [1, 2, 3, 4, 5, 6, 7]

spisok2 = [“a”, “b”, “c”, “d”]

spisok3 = [“молоко”, “сыр”, “творог”]

Нужно лишь название списка, знак =, и значения в квадратных скобках через запятую.

В списке могут лежать числа, строки, отдельные символы и даже другие списки!

Можно создать и пустой список:  
empty\_list = []

**append()**Очень важная команда, она добавляет значение, которое мы передадим в круглые скобки, в конец списка.

Сначала пишем название нашего списка, ставим точку и пишем append.  
empty\_list.append(1)

Добавили 1 в конец списка, сейчас наш список выглядит так:

[1]

Добавим ещё:

empty\_list.append(2)

Теперь в списке лежит 2 значения: [1, 2]

Мы можем сформировать наш список с помощью цикла.

Например:

spisok = []

for i in range(10):

spisok.append(i)

Тогда после цикла в списке будут лежать все числа от 0 до 10 (не включительно, не забываем)

То есть: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]