В Python основных циклов всего два – «while» и «for».

Цикл с предусловием while. Цикл можно назвать самым универсальным циклом в Python, но он выполняется медленнее, чем следующий циклы. Операторы, входящие в тело цикла будут выполняться до тех пор, пока условие цикла будет истинно. В этом и плюс, но и минус. Если мы создадим неправильное условие, то циклы будет выполняться бесконечно долго.

i=1 # задаём начальное значение i

while i<=100: # проверка. пока значение i меньше или равно 100

print('i =', i)

i=i+1 # стандартный счётчик увеличивающий значение i

Цикл с параметром for. Он гораздо быстрее выполняется. Это связано с тем, что в нём нет логических проверок. Он подходит для любого итерируемого объекта (те, которые можно вычислить конечное число раз). Подходят не только числа, но и строки и списки, что делает его более универсальным.

a=0

for i in '3232453232455456': # делаем цикл по строке (а не числу!)

if i == '3':

print(i)

a=a+1 # увеличиваем счетчик троек

print('Число троек в строке =' ,a)

На практике бывает очень много ситуаций в которых нужно игнорировать часть блока операторов или наоборот, принудительно выйти из цикла, если все необходимые данные уже в нём подсчитаны. Их одинаково хорошо можно использовать в циклах for и while.

Это оператор следующего прохода continue, оператор прерывания цикла break. инструкция проверки прерывания else.