**УСЛОВНЫЕ**

**КОНСТРУКЦИИ И**

**ЦИКЛЫ**

**Условные конструкции**

Условная конструкция в Python предоставляет возможность

выполнения определенного блока кода, основываясь на результате

проверки логического условия. В Python используется ключевые слова

if, elif (сокращение от "else if") и else для создания условий.

**if условие1:**

# Код, выполняющийся, если условие1 истинно

**elif условие2:**

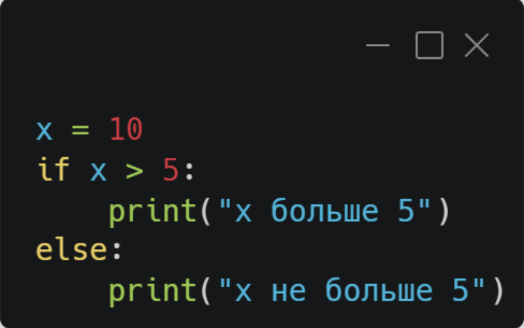
# Код, выполняющийся, если условие2 истинно

**else:**

# Код, выполняющийся, если ни одно из условий не истинно

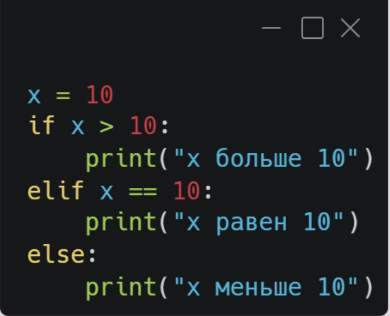
**Условные конструкции**

**if-else:**



Условные конструкции

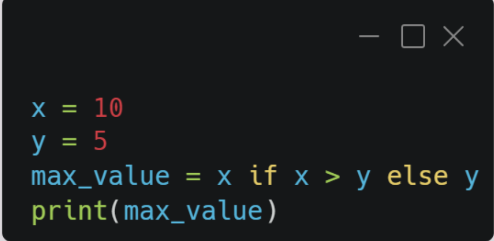
if-elif-else:



**Тернарный оператор**

x if условие else y: Возвращает x, если условие истинно, и y в

противном случае.



**Цикл while**

Цикл while в Python выполняет блок кода до тех пор, пока указанное

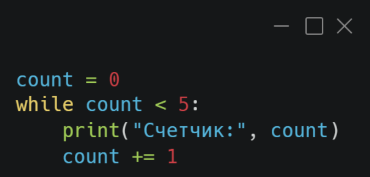
условие остается истинным. Он проверяет условие перед каждой

итерацией и завершает выполнение, как только условие становится

ложным.

**while условие:**

# код, выполняемый до тех пор, пока условие истинно



**Цикл for**

Цикл for в Python используется для итерации (перебора) элементов в

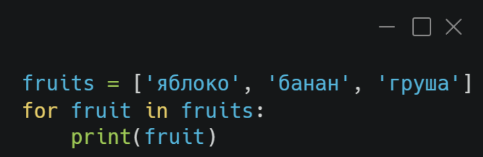
последовательности, такой как список, кортеж, строка или диапазон

чисел. Он выполняет указанный блок кода для каждого элемента в

последовательности.

**for элемент in последовательность:**

# код, выполняемый для каждого элемента



**break и continue**

break и continue - это ключевые слова в Python, которые используются в

циклах для управления выполнением цикла. Они позволяют вам

контролировать, когда следует прервать выполнение всего цикла

(break) или пропустить текущую итерацию и перейти к следующей

(continue).

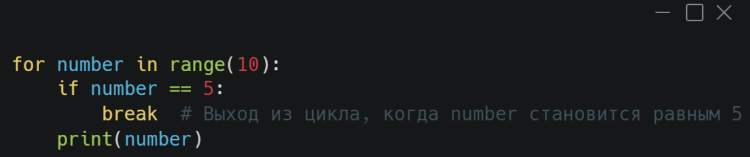
**1. break:**

Когда оператор break встречается внутри цикла, выполнение цикла

немедленно завершается, и программа продолжает выполнение после

цикла. Это полезно, например, чтобы выйти из цикла при выполнении

определенного условия.



**2. continue:**

Оператор continue применяется внутри цикла и используется для

пропуска текущей итерации цикла и перехода к следующей итерации.

Это может быть полезно, если вы хотите пропустить какие-либо

операции в текущей итерации.

