

## Тема 2: Условные операторы ветвления if-elif-else

В этом уроке рассмотрим оператор ветвления if-elif-else. Основная цель – это дать общее представление об этих операторах и на простых примерах показать базовые принципы работы с ними.

### Условный оператор ветвления if

Оператор ветвления if позволяет выполнить определенный набор команд в зависимости от некоторого условия. Возможны следующие варианты использования.

Синтаксис оператора if выглядит так.

```
if выражение:  
    команда_1  
    команда_2  
    ...  
    команда_n
```

После оператора *if* записывается выражение. Если это выражение истинно, то выполняются команды, определяемые данным оператором. Выражение является истинным, если его результатом является число не равное нулю, непустой объект, либо логическое *True*. После выражения нужно поставить двоеточие :

**ВАЖНО:** блок кода, который необходимо выполнить, в случае истинности выражения, отделяется четырьмя пробелами слева!

Примеры:

```
if True:  
    print("True")
```

Напечатает: *True*

```
a = 3  
if a > 1:  
    print("a > 1 ")
```

Напечатает: *a > 1*

## Конструкция *if – else*

Бывают случаи, когда необходимо предусмотреть альтернативный вариант выполнения программы. Т.е. при истинном условии нужно выполнить один набор команд, при ложном – другой. Для этого используется конструкция *if – else*.

```
if выражение:
    команда_1
    команда_2
    ...
    команда_n
else:
    команда_a
    команда_b
    ...
    команда_x
```

Примеры:

```
a = 2
if a < 1:
    print("a < 1 ")
else:
    print("a > 1")
```

Напечатает: *a > 1*

## Конструкция *if – elif – else*

Для реализации выбора из нескольких альтернатив можно использовать конструкцию *if – elif – else*.

```
if выражение_1:
    команды_(блок_1)
elif выражение_2:
    команды_(блок_2)
elif выражение_3:
    команды_(блок_3)
else:
    команды_(блок_4)
```

Пример:

```
a = int(input("введите число:"))
if a < 0:
    print("число меньше 0")
elif a == 0:
    print("число равно 0")
else:
    print("число больше 0")
```

Если пользователь введет число меньше нуля, то будет напечатано “число меньше 0”, равное нулю – “число равно 0”, большее нуля – “число больше 0”.

Практическое задание:

Написать примитивный калькулятор.

Пользователь должен ввести число, потом операцию (+-/\*) и потом ещё одно число, после этого пользователь получает ответ.

Числа могут быть дробными

Пример работающего калькулятора:

```
Введите первое число 3.14
Введите операцию +
Введите второе число 12
15.14
```