

Разработка программного обеспечения на языке Python

[Обзорная панель](#)[Мои курсы](#)[Разработка ПО на языке Python](#)[Программирование на языке Python](#)[Лекция 2. Управляющие конструкции и циклы](#)

Лекция 2. Управляющие конструкции и циклы

Посмотрите видеоуроки и ответьте на контрольные вопросы после лекции

Операции сравнения. Логические операции



УСЛОВНЫЕ ВЫРАЖЕНИЯ

Операции сравнения могут
сравнивать различные объекты –
строки, числа, логические

00:00 / 02:45



В прошлом видео мы рассмотрели арифметические операции в Python. В этом видео мы рассмотрим ряд операций, представляющих условные выражения, а именно операции сравнения и логические операции.

Ряд операций представляют условные выражения. Все эти операции принимают два операнда и возвращают логическое значение, которое в Python представляет тип bool. Существует только два логических значения - True (выражение истинно) и False (выражение ложно).

К такого рода операциям относят операции сравнения. Простейшие условные выражения представляют операции сравнения, которые сравнивают два значения. Python поддерживает следующие операции сравнения, которые вы можете увидеть на экране.

```
1 a = 5
2 b = 6
3 print(a == b) #False
4 print(a != b) # True
5 print(a > b)  # False
6 print(a < b)  # True
```

Рассмотри пример. Объявим 2 переменные a и b. Переменная a равна 5, а b – 6. В первом случае сравним 2 переменные и выведем значение на экран. Результатом будет false. Во втором случае проверим, а и b НЕ равны друг другу. Результатом будет true. В третьем и четвертом случае сравним на больше меньше.

Операции сравнения могут сравнивать различные объекты - строки, числа, логические значения, однако оба операнда операции должны представлять один и тот же тип.

Для создания составных условных выражений применяются логические операции. В Python имеются следующие логические операторы: `and` (логическое умножение) Возвращает `True`, если оба выражения равны `True`, `or` (логическое сложение) Возвращает `True`, если хотя бы одно из выражений равно `True`, `not` (логическое отрицание) Возвращает `True`, если выражение равно `False`

Оператор `in` возвращает `True` если в некотором наборе значений есть определенное значение. Например, строка представляет набор символов. И с помощью оператора `in` мы можем проверить, есть ли в ней какая-нибудь подстрока. Например проверим наличие подстроки `hello` в строке `hello world`. Результат будет `true`.

```
1 message = "hello world!"
2 hello = "hello"
3 print(hello in message) # True - hello есть в строке "hello world!"
```

Резюмируем, в этом видео мы рассмотрели условные выражения, к которым относятся операции сравнения и логические операции.

Условные конструкции

ПРЕДЫДУЩИЙ ЭЛЕМЕНТ КУРСА

◀ [Задание 1. Знакомство. Создание первого приложения в PyCharm](#)

Перейти на...

СЛЕДУЮЩИЙ ЭЛЕМЕНТ КУРСА

[Задание 2. Автоматическая фиксация скорости](#) ▶

© 2010-2023 Центр обучающих систем
Сибирского федерального университета, sfu-kras.ru

Разработано на платформе moodle
Beta-version (3.9.1.5.w3)

[Политика конфиденциальности](#)

[Соглашение о Персональных данных](#)

[Политика допустимого использования](#)

Контакты +7(391) 206-27-05
info-ms@sfu-kras.ru

[Скачать мобильное приложение](#)

[Инструкции по работе в системе](#)