

The background of the slide is a light gray gradient, decorated with numerous realistic water droplets of various sizes. Some droplets are large and prominent, while others are small and subtle, scattered across the top and bottom edges of the frame.

CS001 编程零基础 Python语言入门

第三讲

本讲内容

关系运算

逻辑运算

条件表达式

If语句

关系运算符

6种运算符。关系运算的结果为bool类型，True或者False。

> <

• $5 > 3$

• $'5' < '3'$

>= <=

• $5 \geq 3$

• $'A' \leq 'a'$

== !=

• $3 == 3.0$

• $'A' != 'a'$

关系运算符

- 连续使用。只考虑相邻关系。
 - $1 < 2 < 3$
 - $'Z' > 'X' > 'D'$
 - $2.5 < 3.6 > 2.4$
 - $3 == 3.0 != 1.5 + 1.5$

逻辑运算符

- 3种运算符：and、or、not。都是关键字。逻辑运算的结果也是bool类型。

x and y的结果			
x \ y	True	False	
True	True	False	
False	False	False	

x or y的结果			
x \ y	True	False	
True	True	True	
False	True	False	

x	not x 的结果
True	False
False	True

下面逻辑运算结果是？



`1 < 2 and 2 < 3`

`3 > 5 and 5 > 4`

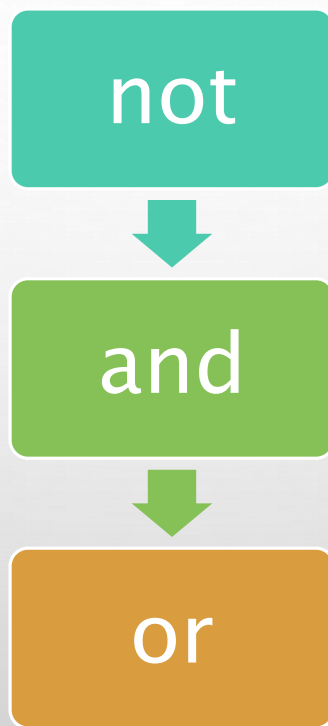
`1.2 > 1.0 or 0.9 > 1.0`

`3 != 3.0 or 'a' == 'A'`

`not 3 == 3.0`

`not 'a' != 'A'`

逻辑运算符的优先级（从高到低）



下面逻辑运算结果是？



$1 < 2$ or $2 < 3$ and $3 < 4$

not $3 > 5$ and $5 > 4$

not $3 > 5$ and $3 > 4$

not ($3 > 5$ and $5 > 4$)

条件表达式

- 凡是计算结果为**bool**类型的，都可以算是条件表达式。
- 条件表达式可以用于控制语句（包括分支和循环）。

混合运算的优先级（从高到低）

算术运算



关系运算



逻辑运算

下面运算结果是？

condition.py

$2 < 3 \text{ and } 3 < 2 * 2$

$3 ** 2 == 9 \text{ and } 3 * 3 + 4 * 4 == 5 * 5$

$1900 \% 400 == 0 \text{ or } 1900 \% 4 == 0 \text{ and not } 1900 \% 100 == 0$

$2000 \% 400 == 0 \text{ or } 2000 \% 4 == 0 \text{ and not } 2000 \% 100 == 0$

$2016 \% 400 == 0 \text{ or } 2016 \% 4 == 0 \text{ and not } 2016 \% 100 == 0$

$2017 \% 400 == 0 \text{ or } 2017 \% 4 == 0 \text{ and not } 2017 \% 100 == 0$

if语句

- 用于分支控制结构
- 形式：
 - 单分支
 - 双分支
 - 多分支
 - 嵌套分支

单分支



if 表达式:
语句块

```
#if.py  
score = int(input('Please input the score:'))  
if score >= 60:  
    print('Pass!')
```

```
Please input the score: 60  
Pass!
```

```
Please input the score: 59
```

注意：语句块部分要缩进4个字符

双分支

if2.py

if 表达式:
 语句块1
else:
 语句块2

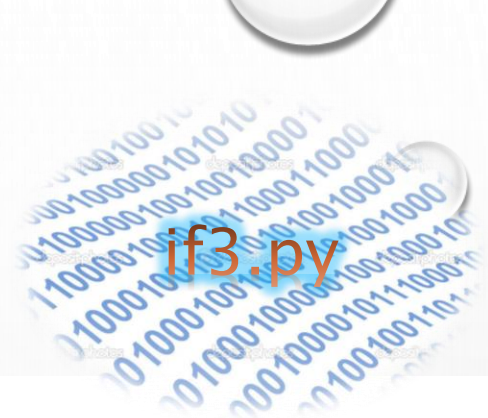
```
#if2.py
score = int(input('Please input the score:'))
if score >= 60:
    print('Pass!')
else:
    print('Fail!')
```

注意：两个分支的缩进要一致

```
Please input the score: 60
Pass!
```

```
Please input the score: 59
Fail!
```


多分支



if 表达式1:

语句块1

elif 表达式2:

语句块2

.....

elif 表达式 n :

语句块 n

else:

语句块 $n+1$

```
#if3.py
score = int(input('Please input the score:'))
if score >= 90:
    print('Excellent!')
elif score >= 80:
    print('Good!')
elif score >= 60:
    print('Pass!')
else:
    print('Fail!')
```

下面的程序为什么错误？

if3error.py

```
#if3error.py
score = int(input('Please input the score:'))
if score >= 80:
    print('Good!')
elif score >= 90:
    print('Excellent!')
elif score >= 60:
    print('Pass!')
else:
    print('Fail!')
```

嵌套分支

if4.py

```
#if4.py
score = int(input('Please input the score:'))
if score >= 80:
    if score >= 90:
        print('Excellent!')
    else:
        print('Good!')
else:
    if score >= 60:
        print('Pass!')
    else:
        print('Fail!')
```

复习阅读

- 课本第4章的4.1（除4.1.8），4.2，4.3

