

Bài giảng: Lệnh điều kiện và vòng lặp trong Python

Tác giả: Đặng Kim Thi ✨

Mục lục

1. Giới thiệu
 2. Lệnh điều kiện (Conditional Statements)
 - 2.1. Lệnh `if`
 - 2.2. Lệnh `if-else`
 - 2.3. Lệnh `if-elif-else`
 - 2.4. Toán tử ba ngôi
 - 2.5. So sánh với các ngôn ngữ khác
 3. Vòng lặp (Loops)
 - 3.1. Vòng lặp `for`
 - 3.2. Vòng lặp `while`
 - 3.3. Lệnh `break`, `continue` và `else` trong vòng lặp
 - 3.4. So sánh với các ngôn ngữ khác
 4. Bài tập thực hành
 - 4.1. Bài tập về lệnh điều kiện
 - 4.2. Bài tập về vòng lặp
 - 4.3. Bài tập nâng cao
-

1. Giới thiệu

Lệnh điều kiện và vòng lặp là hai cấu trúc quan trọng trong một chương trình. Lệnh điều kiện giúp chương trình quyết định nên thực thi một khối mã nào dựa trên điều kiện cho trước. Vòng lặp giúp lặp đi lặp lại một tổ hợp câu lệnh cho đến khi đạt được điều kiện dừng.

Python cung cấp những câu lệnh điều kiện và vòng lặp dễ hiểu và linh hoạt. So với các ngôn ngữ khác như C++, Java, Python giảm thiểu các dấu ngoặc `{}` bằng thừa độ thừa (đều dòng) và dùng các biểu thức để viết.

2. Lệnh điều kiện (Conditional Statements)

2.1. Lệnh `if`

Cú pháp:

```
if condition:
    statement
```

Ví dụ:

```
x = 10
if x > 0:
    print("x là số dương")
```

2.2. Lệnh `if-else`

```
if condition:
    statement1
else:
    statement2
```

Ví dụ:

```
x = -5
if x > 0:
    print("Số dương")
else:
    print("Số không dương")
```

2.3. Lệnh `if-elif-else`

```
if condition1:
    statement1
elif condition2:
    statement2
else:
    statement3
```

2.4. Toán tử ba ngôi

```
x = 5
y = "Dương" if x > 0 else "Âm"
```

2.5. So sánh với các ngôn ngữ khác

Ngôn ngữ	Cú pháp <code>if-else</code>
Python	<code>if x > 0:</code>
C++/Java	<code>if (x > 0) {}</code>

3. Vòng lặp (Loops)

3.1. Vòng lặp `for`

```
for i in range(5):  
    print(i)
```

3.2. Vòng lặp `while`

```
x = 5  
while x > 0:  
    print(x)  
    x -= 1
```

3.3. `break`, `continue`, `else`

```
for i in range(10):  
    if i == 5:  
        break # Dừng vòng lặp  
    if i % 2 == 0:  
        continue # Bỏ qua lần lặp này  
    print(i)
```

4. Bài tập thực hành

4.1. Bài tập về lệnh điều kiện

1. Kiểm tra số chẵn/lẻ.

Input: Một số nguyên `n`.

Output: "Chẵn" nếu `n` là số chẵn, "Lẻ" nếu `n` là số lẻ.

2. Kiểm tra năm nhuận.

Input: Một năm `y`.

Output: "Năm nhuận" nếu `y` chia hết cho 400 hoặc chia hết cho 4 nhưng không chia hết cho 100.

3. Kiểm tra một số có lớn hơn 100 hay không.

Input: Một số nguyên `n`.

Output: "Lớn hơn 100" hoặc "Nhỏ hơn hoặc bằng 100".

4. Tính thuế thu nhập dựa trên mức lương.

Input: Một số thực `lương`.

Output: Mức thuế phải đóng theo quy tắc nhất định.

5. Xác định học lực dựa trên điểm trung bình.**Input:** Điểm trung bình **dtb**.**Output:** "Giỏi", "Khá", "Trung bình", hoặc "Yếu".**6. Kiểm tra xem một số có chia hết cho 3 và 5 không.****Input:** Một số nguyên **n**.**Output:** "Chia hết cho cả 3 và 5" hoặc "Không chia hết".**7. Xác định giai đoạn tuổi.****Input:** Một số nguyên **tuổi**.**Output:** "Trẻ em", "Thanh niên", "Người lớn", "Người già".**8. Kiểm tra xem ba số có thể là ba cạnh tam giác không.****Input:** Ba số nguyên **a**, **b**, **c**.**Output:** "Là tam giác" hoặc "Không phải tam giác".**9. Xác định xem một chữ có là nguyên âm hay phụ âm.****Input:** Một ký tự **ch**.**Output:** "Nguyên âm" hoặc "Phụ âm".**10. Kiểm tra tính hợp lệ của một ngày trong tháng.****Input:** Ba số nguyên **ngày**, **tháng**, **năm**.**Output:** "Hợp lệ" hoặc "Không hợp lệ".**4.2. Bài tập về vòng lặp****1. Tính tổng các số từ 1 đến 100.****Input:** Không có.**Output:** Tổng của các số từ 1 đến 100.**2. In bảng cửu chương.****Input:** Một số nguyên **n**.**Output:** Bảng nhân của **n** từ 1 đến 10.**3. Tính giai thừa của một số.****Input:** Một số nguyên **n**.**Output:** **n!**.**4. Tính Fibonacci n-th.****Input:** Một số nguyên **n**.**Output:** Giá trị Fibonacci thứ **n**.**5. In các số nguyên tố nhỏ hơn N.****Input:** Một số nguyên **N**.**Output:** Danh sách các số nguyên tố nhỏ hơn **N**.**6. In hình tam giác sao.****Input:** Một số nguyên **h**.**Output:** Hình tam giác có **h** hàng.

7. Kiểm tra xem một số có phải là số Armstrong.**Input:** Một số nguyên **n**.**Output:** "Là số Armstrong" hoặc "Không phải số Armstrong".**8. Đảo ngược một số.****Input:** Một số nguyên **n**.**Output:** Số **n** sau khi đảo ngược.**9. Tính tổng các chữ số của một số.****Input:** Một số nguyên **n**.**Output:** Tổng các chữ số của **n**.**10. Tính số lần xuất hiện của từ trong câu.****Input:** Một chuỗi **s** và một từ **word**.**Output:** Số lần **word** xuất hiện trong **s**.

Tác giả: **Đặng Kim Thi** ✨