

PYTHON

CHEATSHEET

CHƯƠNG 3

CẤU TRÚC QUYẾT ĐỊNH 1 CHIỀU IF

Cấu trúc quyết định một chiều sử dụng câu lệnh if. Nếu biểu thức điều kiện sau if là **Đúng** (True), khối lệnh bên trong sẽ được thực thi. Nếu biểu thức điều kiện là **Sai** (False), chương trình sẽ bỏ qua khối lệnh đó và tiếp tục thực thi các lệnh sau khối lệnh if.

BIỂU THỨC BOOLEAN VÀ TOÁN TỬ SO SÁNH

Biểu thức Boolean đặt một câu hỏi và tạo ra kết quả Đúng hoặc Sai (True/False hoặc Yes/No), được sử dụng để điều khiển luồng chương trình. Các Toán tử so sánh (Comparison Operators) nhìn vào biến nhưng không thay đổi giá trị của biến. Bảng Toán tử:

- <: Nhỏ hơn (Less than).
- <=: Nhỏ hơn hoặc Bằng (Less than or Equal to).
- ==: Bằng (Equal to). (Lưu ý: = là dùng cho phép gán).
- >=: Lớn hơn hoặc Bằng (Greater than or Equal to).
- >: Lớn hơn (Greater than).
- !=: Khác (Not equal).

CẤU TRÚC QUYẾT ĐỊNH 2 CHIỀU IF/ELSE

Cấu trúc này cho phép chương trình thực hiện một việc nếu biểu thức logic là đúng, và một việc khác nếu biểu thức đó là sai.

Nó giống như một ngã ba đường-chúng ta phải chọn một trong hai đường chứ không thể chọn cả hai.

Cú pháp:

```
if expression:  
    # Làm điều này nếu biểu thức ĐÚNG
```

```
else:  
    # Làm điều khác nếu biểu thức SAI
```

CẤU TRÚC QUYẾT ĐỊNH ĐA CHIỀU IF/ELIF/ELSE

Đây là cách để xử lý nhiều hơn hai điều kiện. Chương trình sẽ kiểm tra các điều kiện tuần tự:

- Kiểm tra if đầu tiên. Nếu **Đúng**, thực hiện khối lệnh và kết thúc cấu trúc.
- Nếu if đầu tiên **Sai**, chuyển sang kiểm tra elif tiếp theo.
- Thực hiện khối lệnh else nếu **tất cả** các điều kiện if và elif trước đó đều **Sai**.

* **Lưu ý:** Chỉ **một** khối lệnh (tương ứng với điều kiện đúng đầu tiên) trong toàn bộ cấu trúc if/elif/else được thực thi.

LÙI ĐẦU DÒNG (INDENTATION)

Python sử dụng việc lùi đầu dòng (Indentation) để xác định phạm vi (scope) của khối lệnh, ví dụ như sau một câu lệnh if hoặc for (sau dấu hai chấm :)

Quy tắc:

- **Tăng lùi đầu dòng** sau câu lệnh if hoặc for (sau dấu :) .
 - **Duy trì lùi đầu dòng** để chỉ ra phạm vi của khối lệnh.
 - **Giảm lùi đầu dòng** về mức của câu lệnh if hoặc for để kết thúc khối lệnh.
- * **Cảnh báo:** Python rất quan trọng về mức lùi đầu dòng. Việc trộn lẫn tab và khoảng trắng có thể gây ra lỗi lùi đầu dòng (indentation errors).

XỬ LÝ LỖI VỚI TRY/EXCEPT

Cấu trúc try / except được dùng để bao quanh một đoạn mã "nguy hiểm" có khả năng gây lỗi (ví dụ: chuyển đổi chuỗi không phải số thành số nguyên).

Nguyên tắc hoạt động:

- Nếu mã trong khối try hoạt động (không lỗi), khối except sẽ bị bỏ qua.
 - Nếu mã trong khối try thất bại (xảy ra lỗi), chương trình sẽ nhảy ngay đến khối except và tiếp tục thực thi từ đó, giúp chương trình không bị dừng đột ngột.
- * Đây là một "lưới an toàn" (Safety net) để xử lý các lỗi như nhập liệu không phải số từ người dùng.