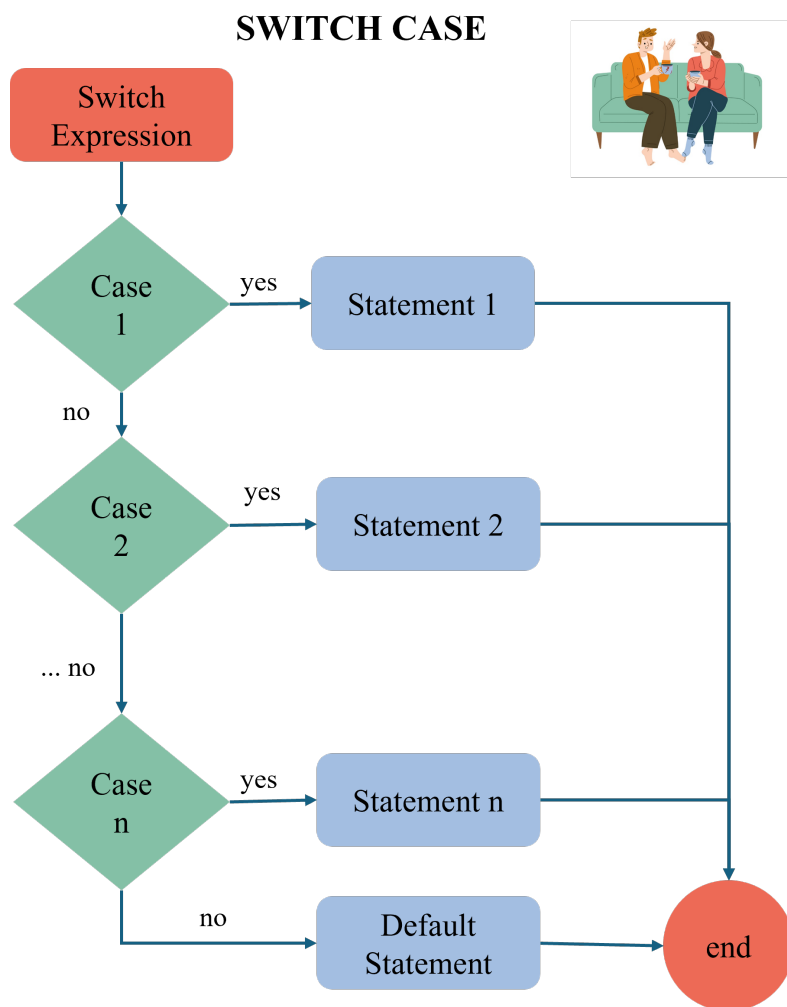


Switch-case Alternatives in Python

Dinh-Tiem Nguyen và Quang-Vinh Dinh

1 Switch Case

Switch case là một cấu trúc điều khiển trong lập trình giúp kiểm tra giá trị của một biến, biểu thức và thực hiện các hành động khác nhau tùy thuộc vào giá trị đó. Để đơn giản thì hãy tưởng tượng bạn có một chiếc hộp điều khiển với nhiều nút bấm. Mỗi nút bấm trên chiếc hộp này sẽ làm một việc khác nhau. Khi bạn nhấn vào một nút, chiếc hộp sẽ biết cần phải làm gì tiếp theo. Đó chính là cách mà switch case hoạt động.



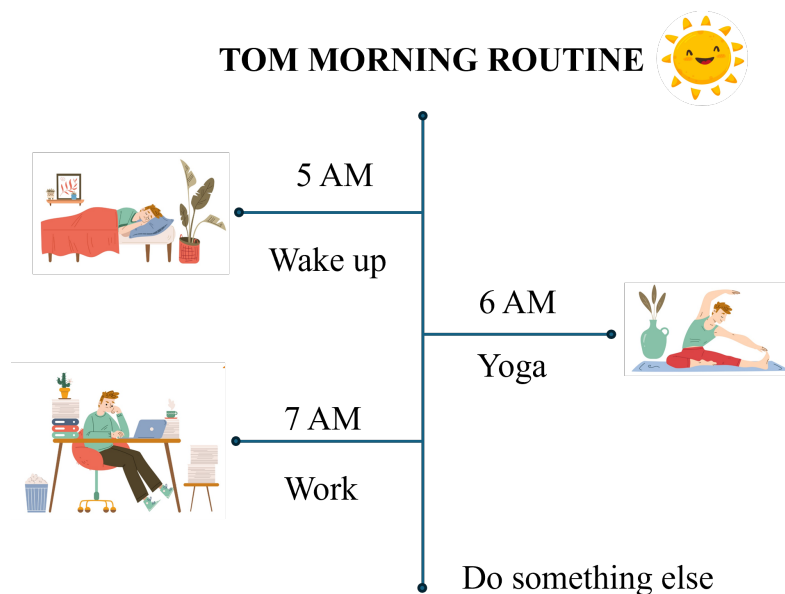
Trong biểu đồ trên thể hiện các thành phần và cách switch case hoạt động, trong đó:

- Switch Expression (Biểu thức kiểm tra): Chúng ta bắt đầu với việc kiểm tra biểu thức. Ví dụ: Hôm nay bạn muốn ăn món gì?
- Case 1: Kiểm tra trường hợp đầu tiên. Nếu biểu thức khớp với trường hợp này, chúng ta thực hiện "Statement 1" và sau đó kết thúc. Ví dụ: Case bạn muốn ăn Táo thì thực hiện Statement 1: Lấy trong tủ lạnh.

- Case 2: Nếu biểu thức không khớp với trường hợp đầu tiên, chúng ta kiểm tra trường hợp thứ hai. Nếu biểu thức khớp với trường hợp này, chúng ta thực hiện "Statement 2" và sau đó kết thúc. Ví dụ: Case bạn muốn ăn chuối thì thực hiện Statement 1: Lấy trên bàn ăn.
- Case n: Tiếp tục kiểm tra các trường hợp khác cho đến khi tìm thấy một trường hợp khớp hoặc đến trường hợp cuối cùng.
- Default Statement: Nếu không có trường hợp nào khớp, chúng ta thực hiện hành động mặc định. Đây là điều sẽ xảy ra nếu tất cả các trường hợp trên đều không đúng. ví dụ: statement mặc định khi bạn chọn một loại trái cây không khớp với bất kỳ trường hợp nào, giả sử bạn chọn bưởi mà nhà bạn không có bưởi thì giá trị mặc định trả về sẽ là: "Hãy ra chợ mua".
- End: Sau khi thực hiện một hành động (dù là từ một trường hợp cụ thể hay hành động mặc định), quá trình kết thúc.

Và đó là cấu trúc và cách switch case hoạt động, tuy nhiên trong python không có cấu trúc này. Thay vào đó, chúng ta có thể sử dụng các phương pháp khác như **if-elif-else**, **dictionary** hoặc từ Python 3.10 trở đi, chúng ta có thể sử dụng match case để làm điều tương tự.

2 Các phương pháp thay thế switch case trong python



Chúng ta sẽ tìm hiểu các phương pháp thay thế switch case qua ví dụ về thói quen vào buổi sáng của Tom. Buổi sáng, Tom thường thức dậy vào lúc 5 giờ, đến 6 giờ thì cậu ấy bắt đầu tập yoga, sau đó 7 giờ cậu ấy bắt đầu làm việc, và các thời gian còn lại thì cậu ấy làm các công việc khác. Ta sẽ viết chương trình nhập đầu vào là thời gian để xem hoạt động của Tom.

Input	Output
5 AM	Wake up
6 AM	Yoga
7 AM	Work
Giá trị khác	Do something else

2.1 If-elif-else

```
if (condition):
    statement
elif (condition):
    statement
.
.
else:
    statement
```

Cách phổ biến nhất để thay thế switch case trong python là sử dụng cấu trúc if-elif-else. Cấu trúc này bắt đầu bằng câu lệnh if và điều kiện condition, nếu điều kiện này đúng khối lệnh trong if(statement) sẽ được thực hiện. Nếu điều kiện sai sẽ chuyển sang câu lệnh elif và tiếp tục kiểm tra điều kiện. Nếu không có điều kiện nào đúng thì thực hiện khối lệnh trong else.

Sau đây là cách sử dụng if-elif-else để xây dựng chương trình :

```
1 time = input("Input time: ")
2 if time == "5 AM":
3     print("Wake up")
4 elif time == "6 AM":
5     print("Yoga")
6 elif time == "7 AM":
7     print("Work")
8 else:
9     print("Do something else")
10
```

Chương trình trên bắt đầu bằng cách yêu cầu người dùng nhập thời gian. Người dùng sẽ nhập một giá trị thời gian cụ thể (ví dụ: "5 AM", "6 AM", hoặc "7 AM"). Sau đó chương trình kiểm tra giá trị của biến thời gian với các lệnh if, elif, và else, nếu điều kiện đúng thì sẽ thực hiện câu lệnh print. Nếu người dùng nhập các giá trị khác thì thực hiện khối lệnh trong else in ra màn hình "Do something else".

2.2 Dictionary

```
dict.get(key, default=None)
```

Phương pháp thứ hai có thể thay thế switch case trong python là sử dụng Dictionary, đây là kiểu dữ liệu được lưu trữ dưới dạng các cặp khóa và giá trị. Mỗi khóa là duy nhất và liên kết với một giá trị. Sau đây là cách sử dụng dictionary để xây dựng chương trình theo dõi hoạt động của Tom.

```
1 todo = {"5 AM": "Wake up",
2         "6 AM": "Yoga",
3         "7 AM": "Work"}
4 time = input("Input time: ")
5 todo.get(time, "Do something else")
```

Đầu tiên chúng ta tạo một từ điển todo với các cặp khóa-giá trị. Mỗi khóa là một thời gian cụ thể (ví dụ: "5 AM", "6 AM", "7 AM") và mỗi giá trị là hoạt động của Tom vào thời gian đó (ví dụ: "Wake

up", "Yoga", "Work"). Tiếp theo chương trình yêu cầu người dùng nhập thời gian. Cuối cùng chúng ta sử dụng phương thức get của từ điển để tìm công việc tương ứng với thời gian nhập vào. Nếu thời gian nhập vào có trong từ điển, phương thức get sẽ trả về công việc tương ứng. Nếu thời gian nhập vào không có trong từ điển, phương thức get sẽ trả về giá trị mặc định là "Do something else".

2.3 Match case

```
match expression:
    case value1:
        #do something
    case value2:
        #do something
    ...
    case valueN:
        #do something
    case _:
        #do something
```

Cấu trúc match case trong Python cung cấp một cách tiếp cận mạnh mẽ và trực quan để xử lý các điều kiện phức tạp. So với if-else, dictionary, và switch case từ các ngôn ngữ khác, match case giúp mã dễ đọc hơn, dễ bảo trì hơn, và linh hoạt hơn trong việc kiểm tra các mẫu dữ liệu phức tạp. Cú pháp sử dụng tương tự switch case, bắt đầu bằng match và biểu thức cần kiểm tra. Bên trong khối match gồm các case, với trường hợp các case đều sai thì sử dụng cú pháp "case _".

Dưới đây là cách sử dụng match case để xây dựng chương trình theo dõi hoạt động của Tom:

```
1 time = input("Input time: ")
2 match time:
3     case "5 AM":
4         print("Wake up")
5     case "6 AM":
6         print("Yoga")
7     case "7 AM":
8         print("Work")
9     case _:
10        print("Do something else")
11
```

Trong chương trình trên, đầu tiên chúng ta sử dụng lệnh input để yêu cầu người dùng nhập thời gian và lưu giá trị đó vào biến time. Tiếp theo chúng ta kiểm tra giá trị biến time này với cấu trúc match case. Một chú ý nhỏ ở đây là dấu gạch dưới _ đại diện cho trường hợp mặc định, khi giá trị của time không khớp với bất kỳ trường hợp nào đã liệt kê.