



---

# Cấu trúc điều kiện

Khóa học: Python căn bản

---

# Kiểm tra bài trước

Hỏi và trao đổi về các khó khăn gặp phải trong bài “Biến, kiểu dữ liệu”

Tóm tắt lại các phần đã học từ bài “Biến, kiểu dữ liệu và toán tử”

# Mục tiêu

---

- Trình bày được câu lệnh điều kiện
- Mô tả cú pháp của câu lệnh if, if – else, if lồng nhau, if bậc thang
- Trình bày được biểu thức điều kiện
- Sử dụng được câu lệnh điều kiện if, if – else, if lồng nhau, if bậc thang
- Sử dụng được biểu thức điều kiện

# Thảo luận

Câu lệnh điều kiện

Câu lệnh if

Câu lệnh if-else

# Các câu lệnh điều khiển

---



- Một chương trình phần mềm thực thi các câu lệnh theo trật tự từ trên xuống dưới
- Có thể thay đổi luồng thực thi của một chương trình bằng cách sử dụng các câu lệnh điều khiển luồng (*control flow statement*)
- Các câu lệnh điều khiển của JavaScript:
  - Câu lệnh điều kiện (conditional statement)
  - Câu lệnh lặp (Loop statement)
  - Câu lệnh nhảy (jump statement)



# Câu lệnh điều kiện

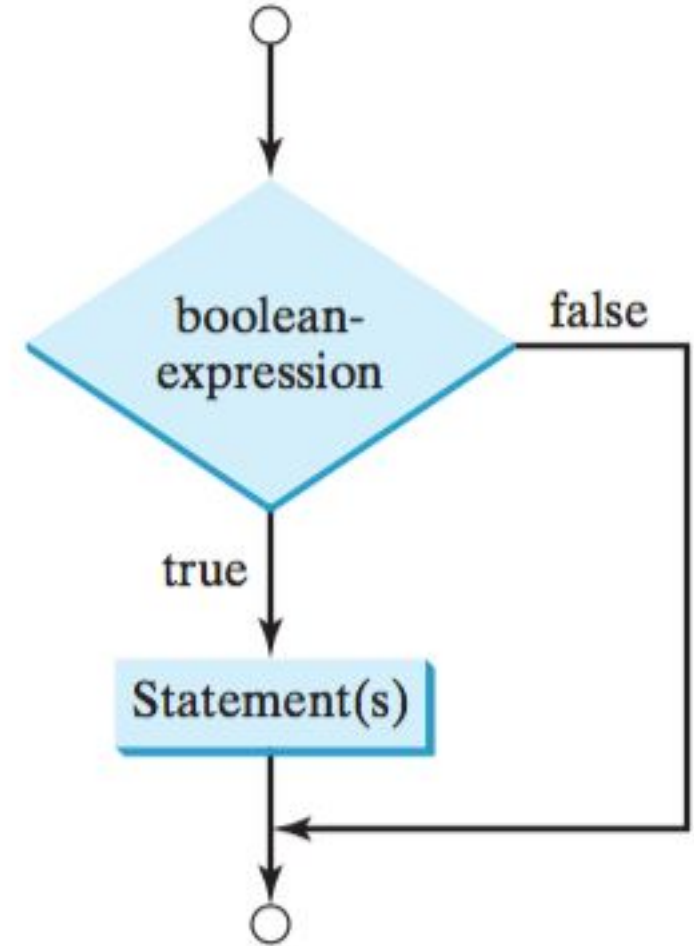
---

- Câu lệnh điều kiện còn được gọi là câu lệnh ra quyết định (decision making)
- Cho phép thay đổi luồng thực thi của chương trình
- Lựa chọn thực thi một khối lệnh dựa trên việc đánh giá một điều kiện cho trước
- Python hỗ trợ các câu lệnh điều kiện:
  - if

# Câu lệnh if



- Đánh giá một điều kiện và đưa ra lựa chọn thực thi một khối lệnh dựa trên điều kiện đó
- Nếu điều kiện có giá trị đúng (true) thì khối lệnh bên trong if sẽ được thực thi
- Nếu điều kiện có giá trị sai (false) thì khối lệnh bên trong if sẽ được bỏ qua. Luồng thực thi sẽ chuyển xuống ngay sau khối lệnh if.



# Cú pháp câu lệnh if

---



- Cú pháp:

```
if condition:  
    # one or more statements
```

Trong đó:

- `condition`: là biểu thức trả về giá trị kiểu boolean
- `statements`: Các câu lệnh sẽ được thực thi nếu điều kiện trả về **true**





# Câu lệnh if: Ví dụ

---

- **Ví dụ:**

```
i = 10
if (i > 15) :
    print ("10 is less than 15")
print ("I am Not in if")
```

- **Kết quả:** I am Not in if

Điều kiện trong câu lệnh if là False nên khối lệnh bên dưới câu lệnh if không được thực thi



# if với nhiều câu lệnh bên trong

---

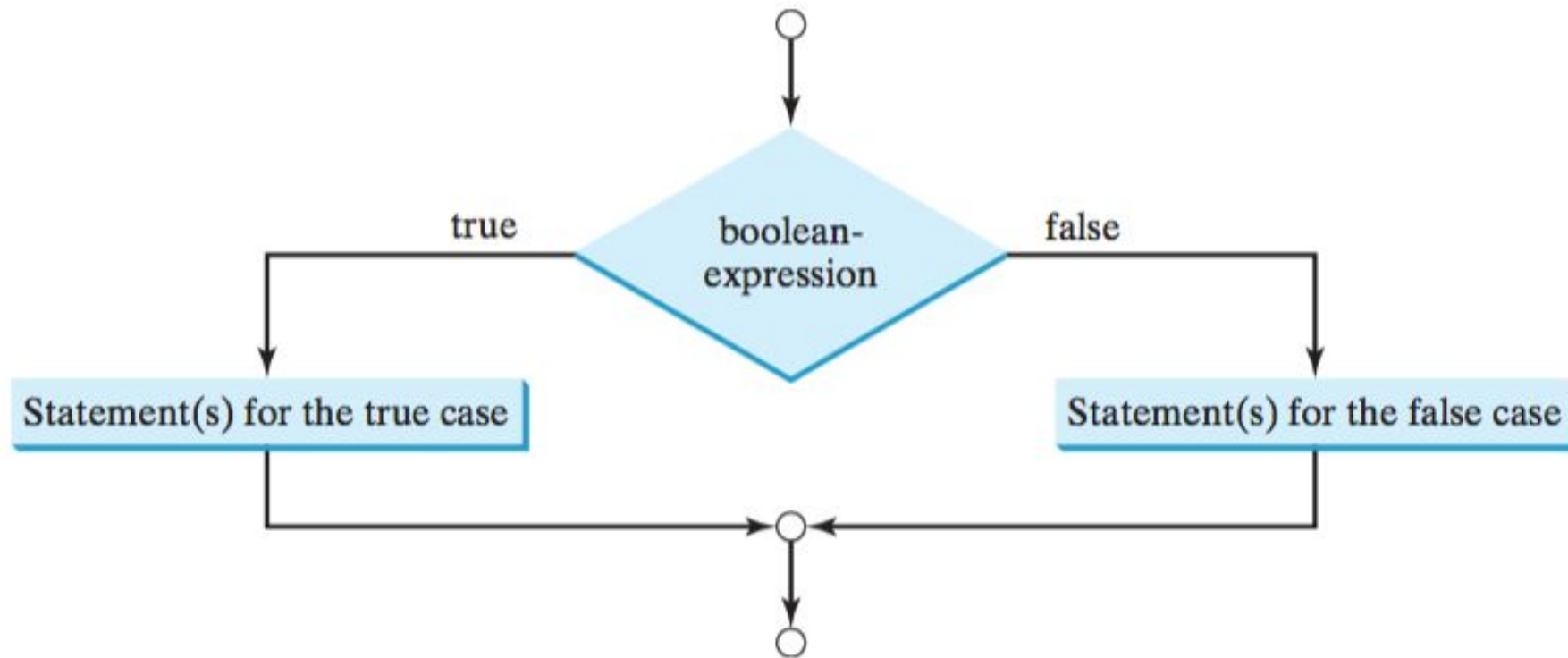
- Khi có nhiều hơn một câu lệnh bên trong **if**
- Ví dụ:

```
if (i > 0) :  
    print("i is positive")  
    print("I am in if")
```

# Câu lệnh if-else



- Câu lệnh if-else lựa chọn thực thi 1 trong 2 khối lệnh thông qua việc đánh giá một điều kiện



# Cú pháp if-else

---



- **Cú pháp**

```
if (condition) :
```

```
    # Executes this block if condition is true
```

```
else:
```

```
    # Executes this block if is false
```

Trong đó:

- condition: điều kiện để đánh giá. Nếu condition trả về **true** thì khối lệnh bên trong **if** được thực thi. Nếu condition trả về **false** thì khối lệnh trong **else** được thực thi.



# if-else: Ví dụ 1

---

- Đoạn mã sau xác định một số là số chẵn hay là số lẻ:

```
if (number % 2 == 0) :  
    print (number + " is even.")  
else:  
    print (number + " is odd.")
```

# if-else: Ví dụ 2

---



- Đoạn mã sau kiểm tra một số đưa vào có lớn hơn 15 hay không?

```
i = 20;  
if (i < 15):  
    print ("i is smaller than 15")  
    print ("i'm in if Block")  
else:  
    print ("i is greater than 15")  
    print ("i'm in else Block")
```

```
print ("i'm not in if and not in else Block")
```

# if-else: Ví dụ 2

---



- **Kết quả:**

`i is greater than 15`

`i'm in else Block`

`i'm not in if and not in else Block`

Khối mã theo sau câu lệnh `else` được thực thi vì điều kiện có trong câu lệnh `if` là `False`. Sau đó câu lệnh không nằm trong khối `if...else` sẽ được thực thi.



---

# Demo

Lệnh if

Lệnh if-else



---

# Thảo luận

Viết mã sạch

Câu lệnh if lồng nhau

If-else bậc thang



# Câu lệnh if lồng nhau (nested if)

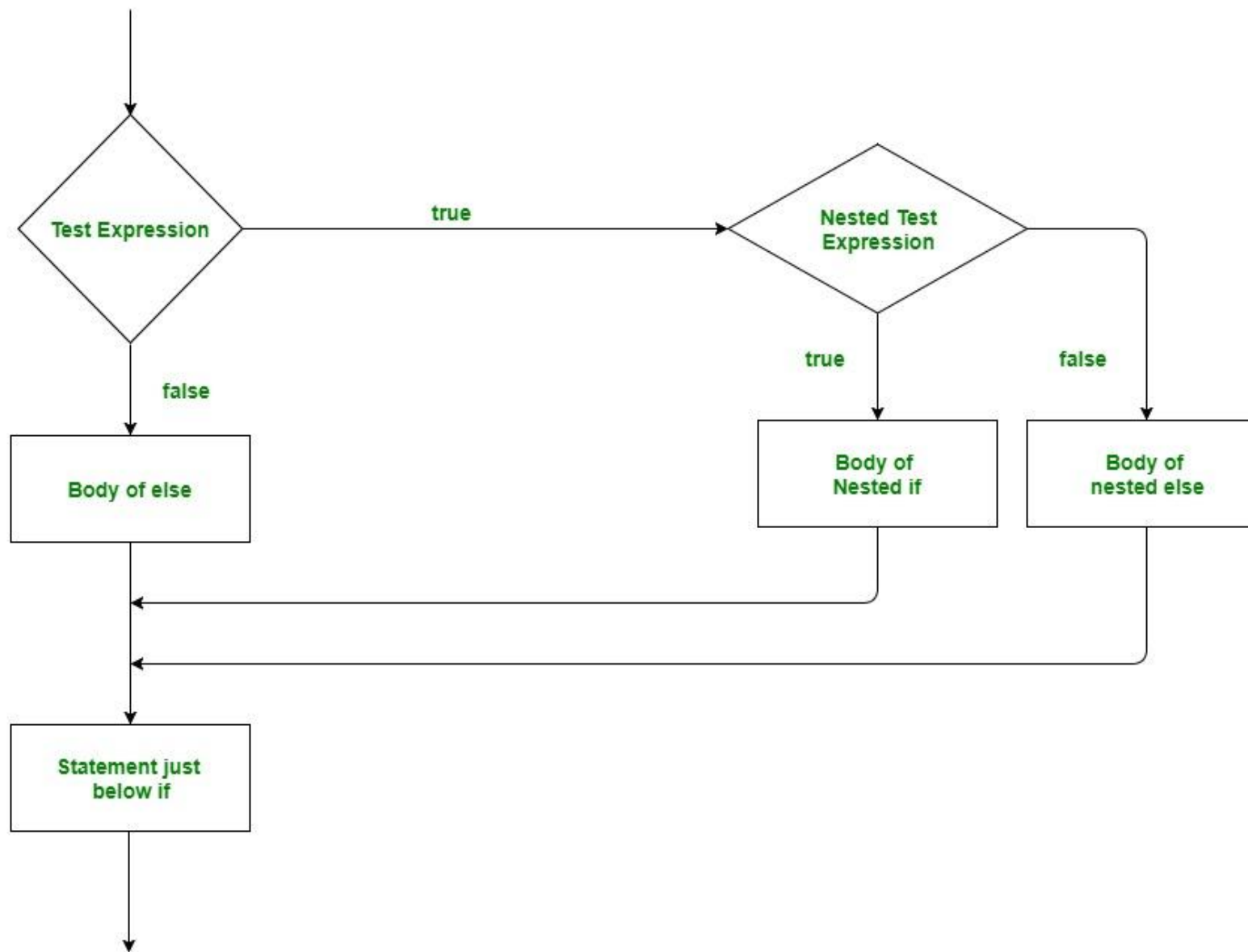
---

- Một if lồng nhau là một câu lệnh if là mục tiêu của một câu lệnh if khác. Các câu lệnh if lồng nhau có nghĩa là một câu lệnh if bên trong một câu lệnh if khác.

## Cú pháp:

```
if (condition1) :  
    # Executes when condition1 is true  
    if (condition2) :  
        # Executes when condition2 is true  
    # if Block is end here
```

# Câu lệnh if lồng nhau



# if lồng nhau: ví dụ 1

---



```
if (a > b) :  
    if (a > c) :  
        print ("Greatest number is a = " + a)  
    else:  
        print ("Greatest number is c = " + c)  
else:  
    if (b > c) :  
        print ("Greatest number is b = " + b)  
    else:  
        print ("Greatest number is c = " + c)
```

# if lồng nhau: ví dụ 2

---



## • Ví dụ:

```
i = 10
if (i == 10):
    #First if statement
    if (i < 15):
        print("i is smaller than 15")
    #Nested - if statement
    #Will only be executed if statement above
    #it is true
    if (i < 12):
        print("i is smaller than 12 too")
else:
    print("i is greater than 15")
```

## Kết quả:

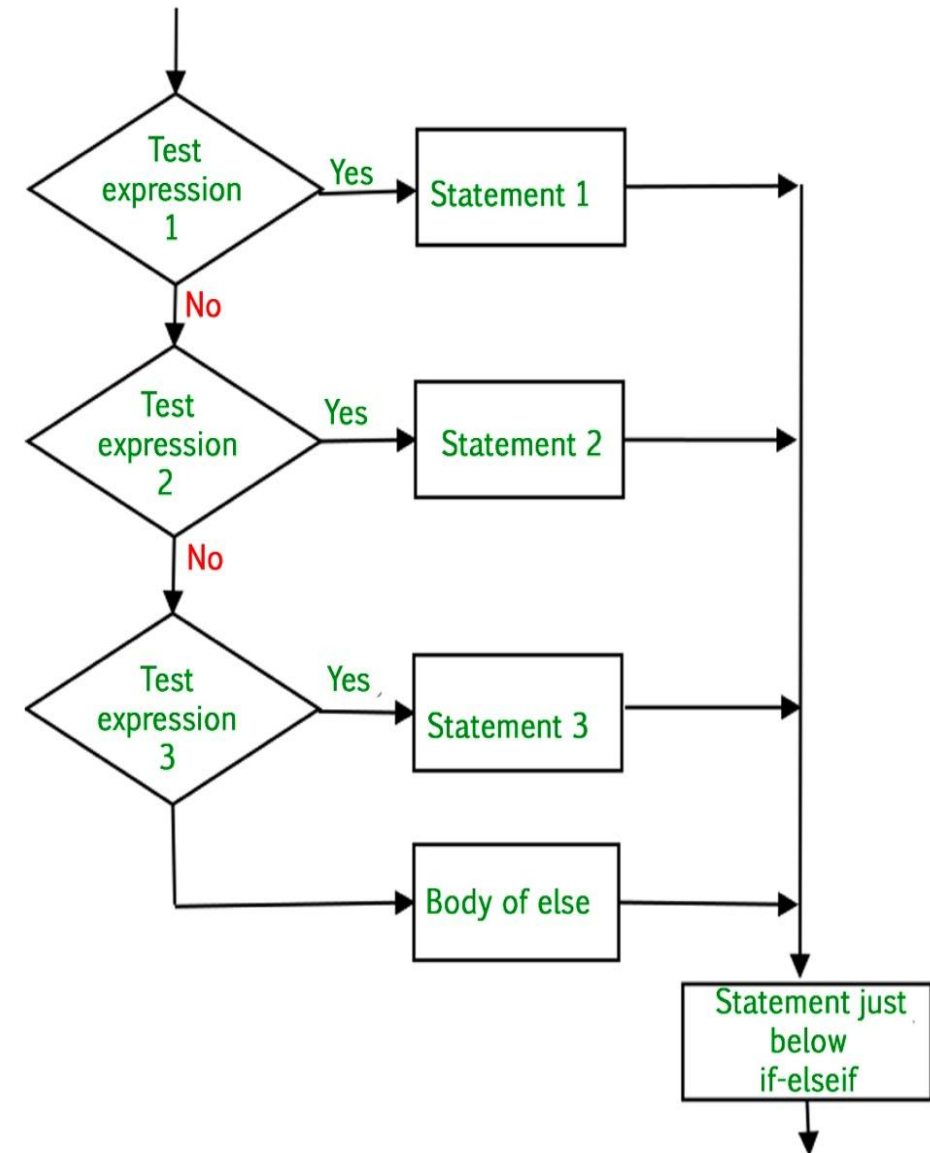
i is smaller than 15  
i is smaller than 12 too

# Câu lệnh if bậc thang



- **Cú pháp:**

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



# Câu lệnh if bậc thang

---

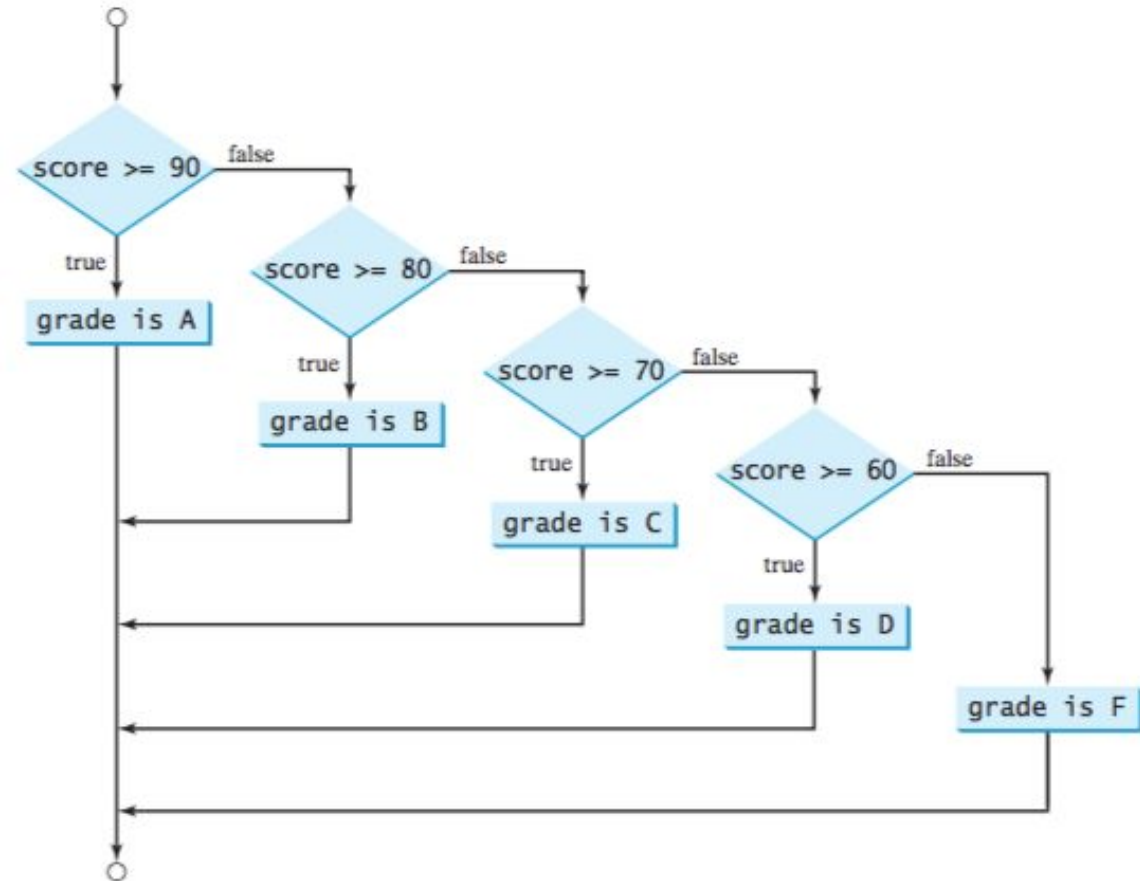


- elif là viết gọn của else if, nó cho phép chúng ta kiểm tra nhiều điều kiện.
- Nếu điều kiện là sai, nó sẽ kiểm tra điều kiện của khối elif tiếp theo và cứ như vậy cho đến hết.
- Nếu tất cả các điều kiện đều sai nó sẽ thực thi khối lệnh của else.
- Chỉ một khối lệnh trong if...elif...else được thực hiện theo điều kiện. Có thể không có hoặc có nhiều elif, phần else là tùy chọn.

# if-else bậc thang: Ví dụ



```
if (score >= 90.0):  
    print("A")  
elif (score >= 80.0):  
    print("B")  
elif (score >= 70.0):  
    print("C")  
elif (score >= 60.0):  
    print("D")  
else:  
    print("F")
```







---

# Demo

Lệnh if lồng nhau

Lệnh if bậc thang

# Các lỗi thường gặp

# Lỗi sai cú pháp

---



- **Ví dụ:**

```
num = int(input("Give me a number: "))  
if num > 0:  
    print(num, "is positive")
```

- **Kết quả:**

```
File "main.py", line 3  
    print(num, "is positive")  
    ^
```

IndentationError: expected an indented block

# Lỗi sai cú pháp

---



- **Kết luận:**

Như chúng ta đã biết, python sử dụng thụt đầu dòng để xác định một khối. Vì vậy, hãy chắc chắn rằng khối bên dưới câu lệnh if đã được viết đúng cú pháp:

if condition:

    # Statements to execute if condition is true

# Lỗi sai cú pháp toán tử quan hệ



- **Ví dụ:**

```
num = int(input("Give me a  
number: "))  
if num = 0:  
    print("You entered 0")
```

- **Kết quả:**

```
File "main.py", line 2  
    if num = 0:  
           ^  
SyntaxError: invalid  
syntax
```

- **Kết luận:**

Câu lệnh if được sử dụng để quyết định xem một câu lệnh hoặc khối câu lệnh nhất định sẽ được thực thi hay không.

Nếu một điều kiện nhất định là đúng thì một khối câu lệnh được thực thi hoặc không. Vì vậy, hãy chắc chắn rằng điều kiện đã được viết đúng cú pháp.



# Lỗi không viết đầy đủ các điều kiện cho câu lệnh if

---

- **Ví dụ:**

```
num = int(input("Give me a number: "))  
if num > 0:  
    print(num, "is positive")
```

- **Kết quả:**

Give me a number: 0

Không trả về kết quả đối với những số  $\leq 0$



# Lỗi không viết đầy đủ các điều kiện cho câu lệnh if

---

- **Kết luận:**

Hãy lường trước được những trường hợp có thể sẽ xảy ra và viết nó một cách chặt chẽ nhất để kết quả sẽ luôn được trả về

```
num = int(input("Give me a number: "))  
if num > 0:  
    print(num, "is positive")  
if num < 0:  
    print(num, "is negative")  
if num == 0:  
    print(num, "is zero")
```

# Lỗi viết quá nhiều câu lệnh if

---



- Ví dụ:

```
gas = input("What type of gas do you want? ")
if gas == "regular":
    print("Regular Gas:", 3.89*17)
if gas == "plus":
    print("Plus Gas:", 4.09*17)
if gas == "premium":
    print("Premium Gas:", 4.19*17)
if gas == "diesel":
    print("Diesel Gas:", 4.59*17)
else:
    print("Invalid Gas Type")
```



# Lỗi viết quá nhiều câu lệnh if



- **Kết luận:**

Python sẽ đánh giá tất cả các câu lệnh if để xác định xem chúng có đúng không. Trong khi một điều kiện trong câu lệnh if elif else đúng, Python sẽ ngừng đánh giá các điều kiện khác.

```
gas = input("What type of gas do you want? ")
if gas == "regular":
    print("Regular Gas:", 3.89*17)
elif gas == "plus":
    print("Plus Gas:", 4.09*17)
elif gas == "premium":
    print("Premium Gas:", 4.19*17)
elif gas == "diesel":
    print("Diesel Gas:", 4.59*17)
else:
    print("Invalid Gas Type")
```

# Lỗi viết quá nhiều điều kiện lồng nhau

---



Nếu có thể, hãy tận dụng các toán tử logic để có được những dòng code gọn gàng và hiệu quả nhất

Thay vì:

```
dayssent = int(input("How many days in a row have you sent your friend a message?"))
daysreceived = int(input("How many days in a row has your friend sent you a message? "))
if dayssent >= 100:
    if daysreceived >= 100:
        print("you receive a hundred emoji")
    else:
        print("not there yet")
```

# Lỗi viết quá nhiều điều kiện lồng nhau

---



Nên viết thành:

```
dayssent = int(input("How many days in a row have you sent your friend a message? "))
daysreceived = int(input("How many days in a row has your friend sent you a message? "))
if dayssent >= 100 and daysreceived >= 100:
    print("you receive a hundred emoji")
else:
    print("not there yet")
```

# Biểu thức điều kiện

# Biểu thức điều kiện



- Biểu thức điều kiện đánh giá một biểu thức dựa vào một điều kiện cho trước
- Biểu thức điều kiện là một toán tử 3 ngôi
- Cú pháp:

*condition* **if** *expression\_true\_case* **else** *expression\_false\_case*

Trong đó:

- *condition*: biểu thức điều kiện dùng để đánh giá
- *expression\_true\_case* : biểu thức sẽ được sử dụng trong trường hợp **true**
- *expression\_false\_case* : biểu thức sẽ được sử dụng trong trường hợp **false**

# Biểu thức điều kiện: Ví dụ

---



- Tìm giá trị lớn nhất trong 2 số:

`maxValue = number1 > number2 if number1 else number2;`

- Tính số lượng dựa trên điều kiện khoảng cách:

`count = isHere if getHereCount(index) else getAwayCount(index);`



# Tóm tắt bài học

---

- Các câu lệnh điều khiển giúp điều hướng luồng thực thi của ứng dụng
- Python hỗ trợ các câu lệnh điều khiển: điều kiện, lặp, nhảy
- Python hỗ trợ câu lệnh điều kiện: if
- Một số dạng khác của if:
  - if-else
  - if-else lồng nhau
  - if-else bậc thang
- Biểu thức điều kiện đánh giá một biểu thức dựa vào một điều kiện cho trước

# Hướng dẫn

Hướng dẫn làm bài thực hành và bài tập

Chuẩn bị bài tiếp theo