Python - Nested if Statement

Python supports **nested if statements** which means we can use a conditional if and if...else statement inside an existing **if statement**.

There may be a situation when you want to check for additional conditions after the initial one resolves to true. In such a situation, you can use the nested **if** construct.

Additionally, within a nested **if** construct, you can include an **if...elif...else** construct inside another **if...elif...else** construct.

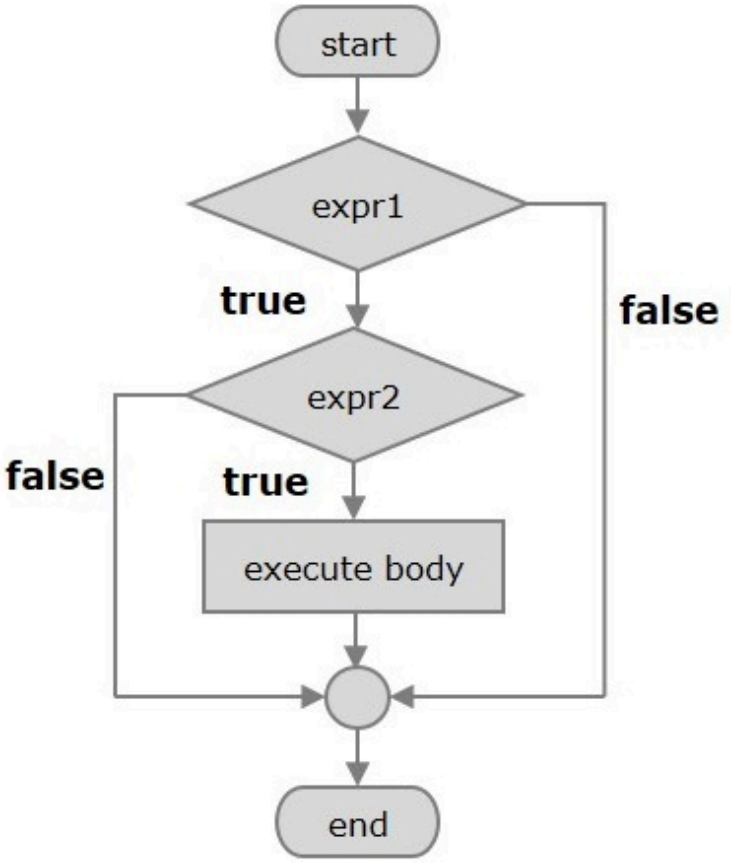
Syntax of Nested if Statement

The syntax of the nested **`if construct`** with else condition will be like this –

```
if boolean_expression1:
   statement(s)
   if boolean_expression2:
      statement(s)
```

Flowchart of Nested if Statement

Following is the flowchart of Python nested if statement –



Learn **Python** in-depth with real-world projects through our **Python certification course**. Enroll and become a certified expert to boost your career.

Example of Nested if Statement

The below example shows the working of nested if statements –

</> Open Compiler

```
num = 36
print ("num = ", num)
if num % 2 == 0:
   if num % 3 == 0:
      print ("Divisible by 3 and 2")
print("....execution ends....")
```

When you run the above code, it will display the following result –

```
num =  36
Divisible by 3 and 2
....execution ends....
```

Nested if Statement with else Condition

As mentioned earlier, we can nest **if-else** statement within an **if statement**. If the **if condition** is true, the first **if-else statement** will be executed otherwise, statements inside the **else block** will be executed.

Syntax

The syntax of the **nested if construct** with else condition will be like this –

```
if expression1:
   statement(s)
   if expression2:
      statement(s)
   else
      statement(s)
else:
   if expression3:
      statement(s)
   else:
      statement(s)
```

Example

Now let's take a Python code to understand how it works –

</> Open Compiler

```
num=8
print ("num = ",num)
if num%2==0:
   if num%3==0:
      print ("Divisible by 3 and 2")
   else:
      print ("divisible by 2 not divisible by 3")
else:
   if num%3==0:
      print ("divisible by 3 not divisible by 2")
   else:
      print ("not Divisible by 2 not divisible by 3")
```

When the above code is executed, it produces the following **output** –

```
num = 8
divisible by 2 not divisible by 3
num = 15
divisible by 3 not divisible by 2
num = 12
Divisible by 3 and 2
num = 5
not Divisible by 2 not divisible by 3
```