Python If Else Statements – Conditional Statements

If-Else statements in Python are part of conditional statements, which decide the control of code. As you can notice from the name If-Else, you can notice the code has two ways of directions.

There are situations in real life when we need to make some decisions and based on these decisions, we decide what we should do next. Similar situations arise in programming also where we need to make some decisions and based on these decisions we will execute the next block of code.

Conditional statements in Python languages decide the direction(Control Flow) of the flow of program execution.

Types of Control Flow in Python

Python control flow statements are as follows:

1. The if statement
2. The if-else statement
3. The nested-if statement
4. The if-elif-else ladder

Python if statement

The if statement is the most simple decision-making statement. It is used to decide whether a certain statement or block of statements will be executed or not.

Syntax:

```
if condition:
    # Statements to execute if
    # condition is true
```

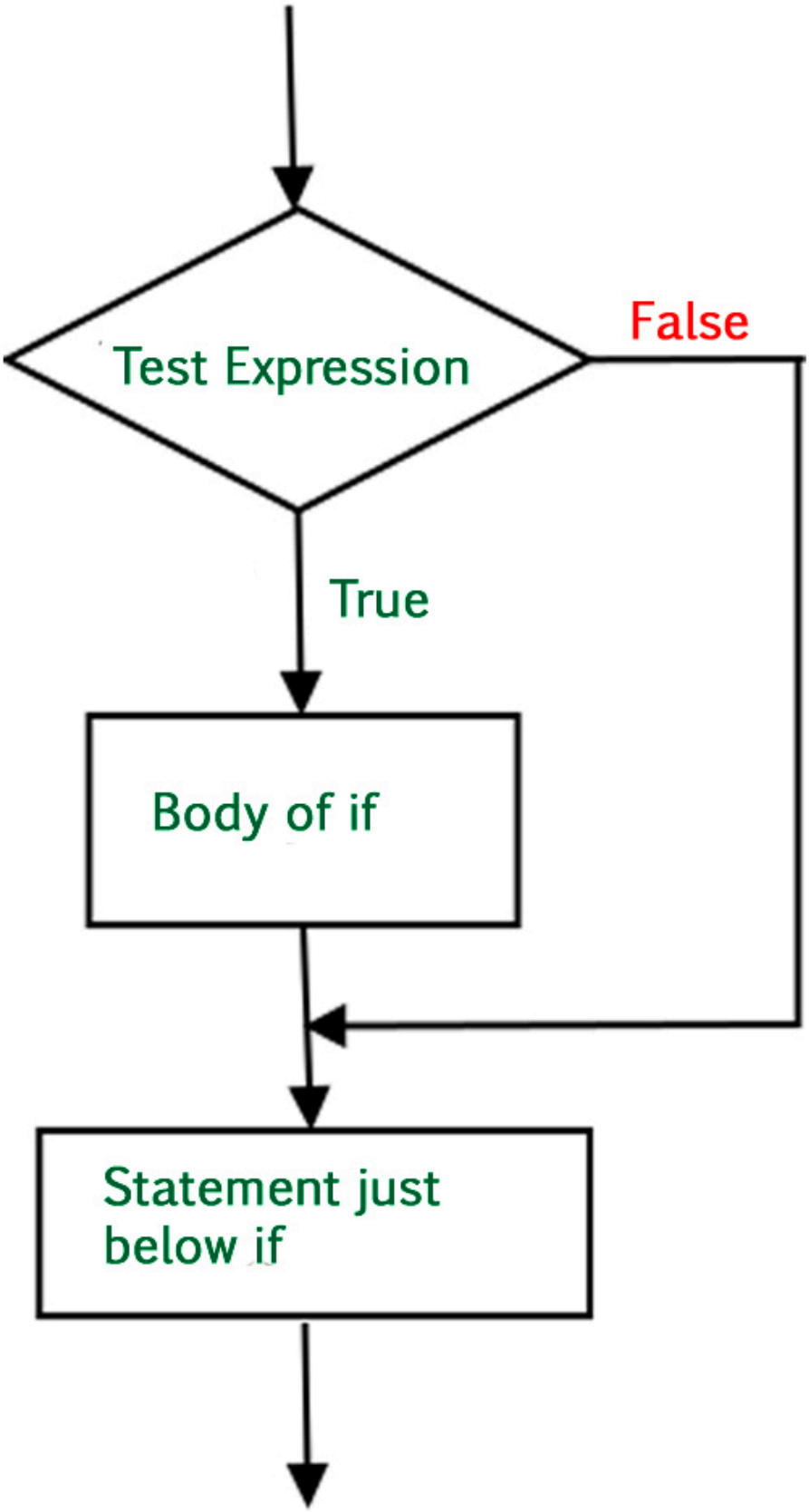
Here, the condition after evaluation will be either true or false. if the statement accepts boolean values – if the value is true then it will execute the block of statements below it otherwise not.

As we know, python uses indentation to identify a block. So the block under an if statement will be identified as shown in the below example:

```
if condition:
    statement1
statement2
# Here if the condition is true, if block
# will consider only statement1 to be inside
# its block.
```

Flowchart of Python if statement

Let’s look at the flow of code in the If statement



Flowchart of Python if statement

Example of Python if Statement

As the condition present in the if statement is false. So, the block below the if statement is executed.

Python

```
# python program to illustrate If statement

i = 10

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

Output:

```
I am Not in if
```

Python If-Else Statement

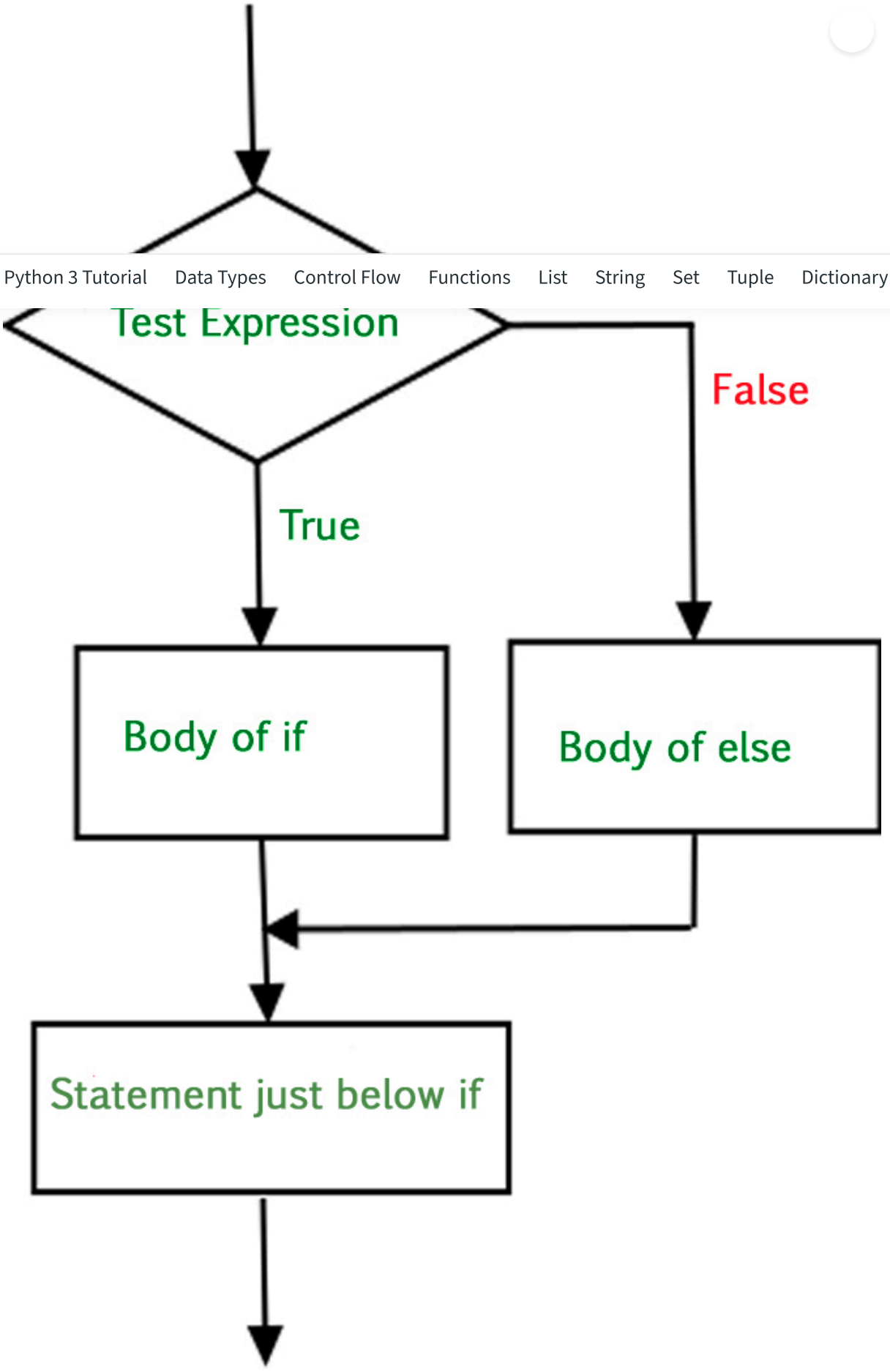
The if statement alone tells us that if a condition is true it will execute a block of statements and if the condition is false it won't. But if we want to do something else if the condition is false, we can use the else statement with the if statement to execute a block of code when the if condition is false.

Syntax of Python If-Else:

```
if (condition):
    # Executes this block if
    # condition is true
else:
    # Executes this block if
    # condition is false
```

Flowchart of Python if-else statement

Let's look at the flow of code in an if-else statement



Flowchart of Python is-else statement

Using Python if-else statement

The block of code following the else statement is executed as the condition present in the if statement is false after calling the statement which is not in the block(without spaces).

Python

```
# python program to illustrate If else statement
#!/usr/bin/python

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")
```

Output:

```
i is greater than 15
i'm in else Block
i'm not in if and not in else Block
```

Python if else statement in a List Comprehension

In this example, we are using an if statement in a list comprehension with the condition that if the element of the list is odd then its digit sum will be stored else not.

Python

```
# Explicit function
def digitSum(n):
    dsum = 0
    for ele in str(n):
        dsum += int(ele)
    return dsum


# Initializing list
List = [367, 111, 562, 945, 6726, 873]

# Using the function on odd elements of the list
newList = [digitSum(i) for i in List if i & 1]
```

```
# Displaying new list
print(newList)
```

Output :

```
[16, 3, 18, 18]
```

Nested-If Statement in Python

A nested if is an if statement that is the target of another if statement. Nested if statements mean an if statement inside another if statement.

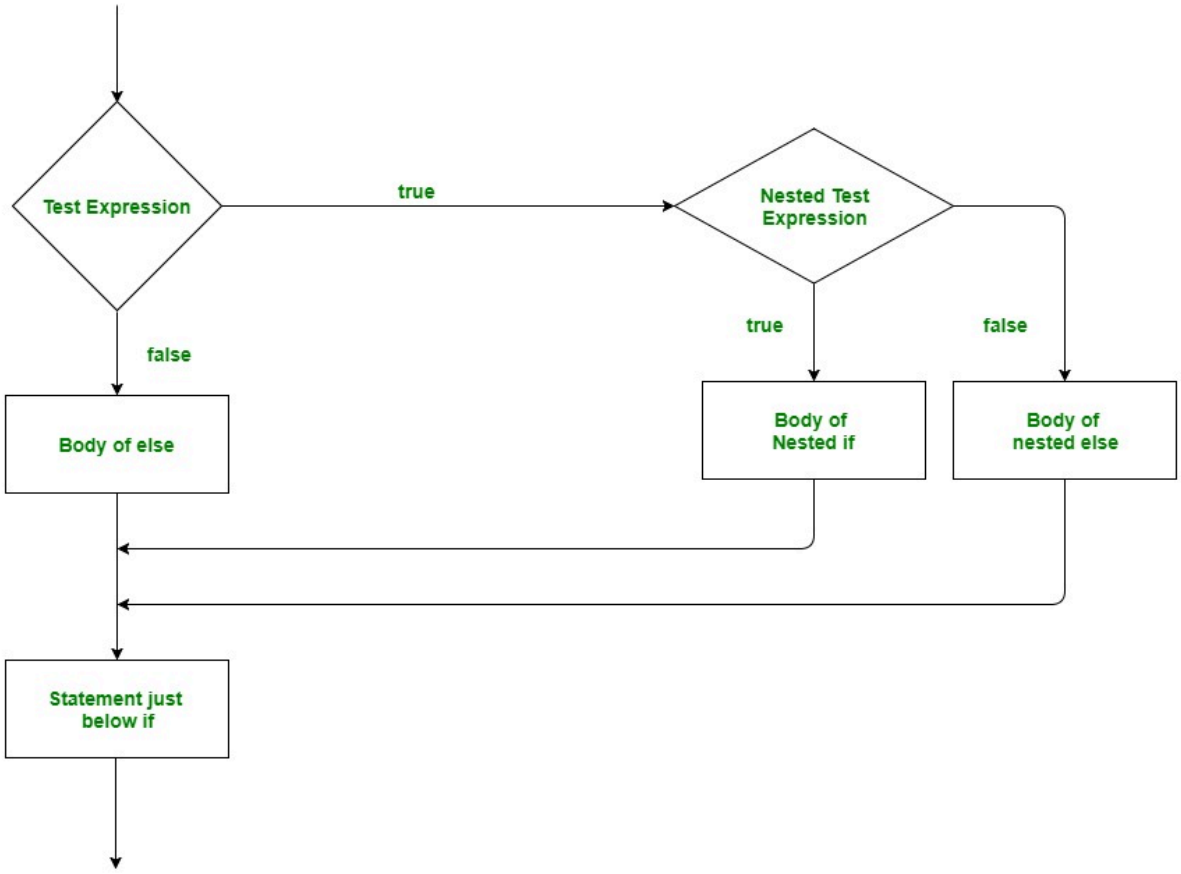
Yes, Python allows us to nest if statements within if statements. i.e., we can place an if statement inside another if statement.

Syntax:

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

Flowchart of Python Nested if Statement

Let's look at the flow of control in Nested if Statements



Flowchart of Python Nested if statement

Example of Python Nested if statement

In this example, we are showing nested if conditions in the code, All the If conditions will be executed one by one.

Python

```
# python program to illustrate nested If statement

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")
```

Output:

```
i is smaller than 15
i is smaller than 12 too
```

Python if-elif-else Ladder

Here, a user can decide among multiple options. The if statements are executed from the top down.

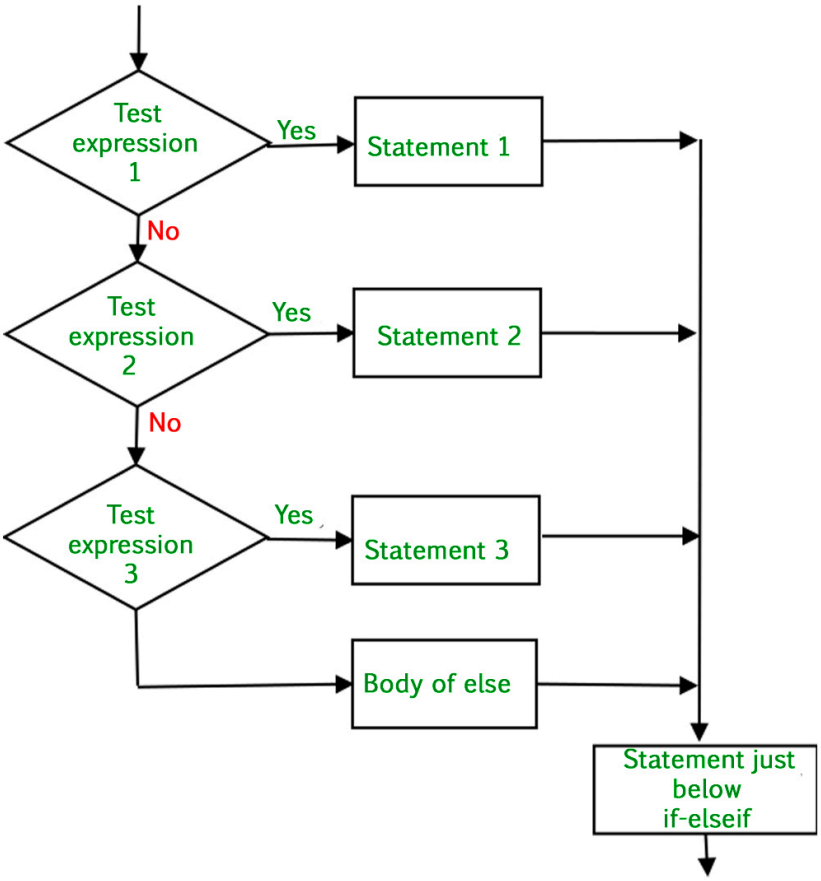
As soon as one of the conditions controlling the if is true, the statement associated with that if is executed, and the rest of the ladder is bypassed. If none of the conditions is true, then the final “else” statement will be executed.

Syntax:

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

Flowchart of Python if-elif-else ladder

Let’s look at the flow of control in if-elif-else ladder:



Flowchart of if-elif-else ladder

Example of Python if-elif-else ladder

In the example, we are showing single if condition, multiple elif conditions, and single else condition.

Python

```
# Python program to illustrate if-elif-else ladder
#!/usr/bin/python

i = 20
if (i == 10):
    print("i is 10")
elif (i == 15):
    print("i is 15")
elif (i == 20):
    print("i is 20")
else:
    print("i is not present")
```

Output:

```
i is 20
```

Short Hand if statement

Whenever there is only a single statement to be executed inside the if block then shorthand if can be used. The statement can be put on the same line as the if statement.

Syntax:

```
if condition: statement
```

Example of Python if shorthand

In the given example, we have a condition that if the number is less than 15, then further code will be executed.

Python

```
# Python program to illustrate short hand if
i = 10
if i < 15: print("i is less than 15")
```

Output:

```
i is less than 15
```

Short Hand if-else statement

This can be used to write the if-else statements in a single line where only one statement is needed in both the if and else blocks.

Syntax:

```
statement_when_True if condition else statement_when_False
```

Example of Python if else shorthand

In the given example, we are printing True if the number is 15, or else it will print False.

Python

```
# Python program to illustrate short hand if-else
```

```
i = 10
print(True) if i < 15 else print(False)
```

Output:

```
True
```

Python if-else Statements Exercise Questions

Below are two Exercise Questions on Python if-else statements. We have covered 2 important exercise questions based on the odd-even program and the eligible age-to-vote program.

Q1. Odd-even practice exercise using if-else statements

Python

```
number = 8
if number % 2 == 0:
    print("The number is even.")
else:
    print("The number is odd.")
```

Output

```
The number is even.
```

Q2. Eligible to vote exercise questions using if-else statements

Python

```
age = 15
if age >= 18:
    print("You are eligible to vote.")
else:
    print("You are not eligible to vote.")
```

Output

```
You are not eligible to vote.
```

In this article, we have covered all the variations of if-else statements. Conditional statements are a very important concept in Programming as it is

used in loops and many programs. You can use any of the if-else variations depending on your needs.

Similar Reads:

- Python3 – if , if..else, Nested if, if-elif statements
- Using Else Conditional Statement With For loop in Python
- How to use if, else & elif in Python Lambda Functions

Last Updated : 27 Dec, 2023

Previous

Python Functions

Next

For Loop in Python

Similar Reads

Using Else Conditional Statement With For loop in Python

One Liner for Python if-elif-else Statements

Conditional Statements in Python

Python3 - if , if..else, Nested if, if-elif statements

Python Else Loop

Lambda with if but without else in Python