



PYTHON Programming

Conditional and Iterative Statement

www.infomaxacademy.com

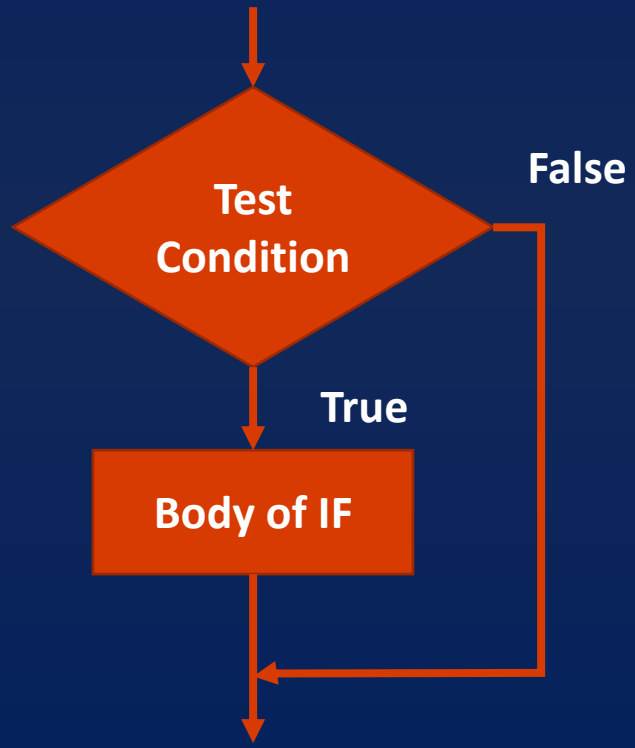
Conditional Statement

- When program uses a condition to decide which set of statement is to be executed between alternative set of statement, is called Conditional statement.
- Python provide following type of conditional construct:
  - If Statement
  - If else statement
  - If elif else statement
  - Nested if else statement

IF Statement

- The statement if tests a particular condition. Whenever that condition evaluates to true, an action or a set of action is executed. Otherwise, the action are ignored.
- Syntax:

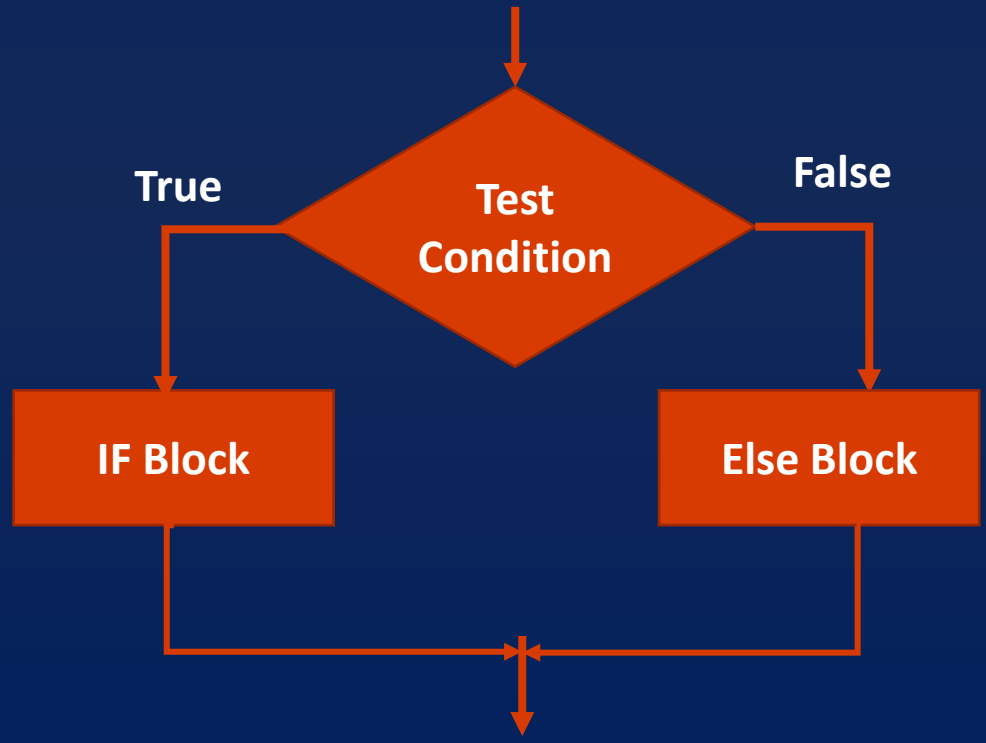
```
if condition:
    Statement 1
    Statement 2
    .....
```



IF ELSE Statement

- The if..else statement evaluates test expression and will execute the body of if only when the test condition is True.
- If the condition is False, the body of else is executed. Indentation is used to separate the blocks.
- Syntax:

```
if condition:
    Statement
    ……
else:
    Statement
    ……
```



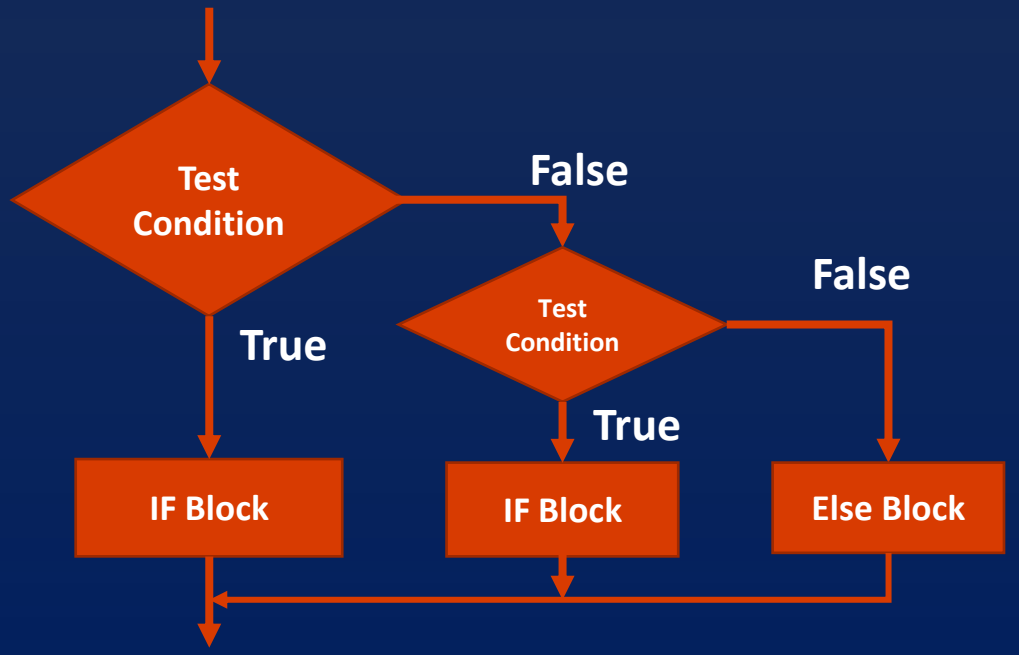
Program

- A dartboard of radius 10 units and the wall it is hanging on are represented using a two-dimensional coordinate system, with the boards center at coordinate (0,0). Variables x and y store the x-coordinate and the y-coordinate of a dart that hits the dartboard. Write a Python expression using variables x and y that evaluates to True if the dart hits (is within) the dartboard, and then evaluate the expression for these dart coordinates:
- (a) (0,0)
- (b) (10,10)
- (c) (6,6)
- (d) (7,8)

IF ELIF ELSE Statement

- The elif is short for else if. It allows us to check for multiple expressions.
- If the condition for if is False, it checks the condition of the next elif block and so on.
- If all the conditions are False, the body of else is executed.
- Only one block among the several if...elif...else blocks is executed according to the condition.
- Syntax:

```
if Condition:
    Statement
elif Condition:
    Statement
else:
    Statement
```



Nested IF ELSE Statement

- We can have a if,if else or if...elif...else statement inside another if, if else or if...elif...else statement. This is called nesting in computer programming.
- Any number of these statements can be nested inside one another. Indentation is the only way to figure out the level of nesting.
- Syntax:

```
if condition:
    if condition:
        Statement or another if Statement
    else:
        Statement or another if Statement
else:
    Statement or another if Statement
```

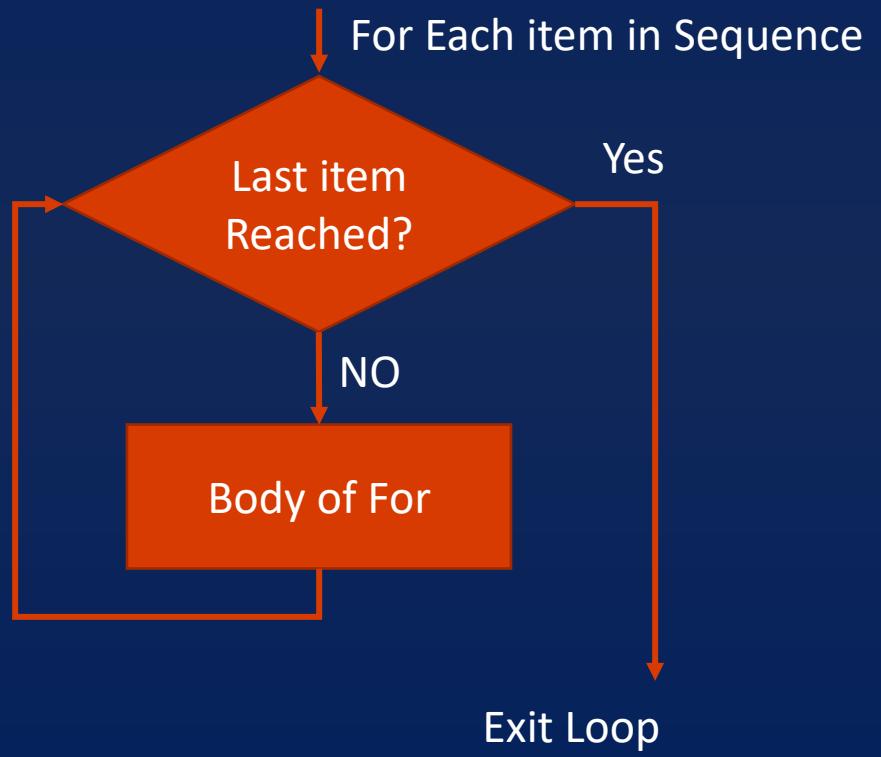
Looping Statement

- Loops are used in programming to repeat a specific block of code.
- A loop is defined as a block of instructions repeated for a certain number of times.
- Python provide two type of Loop construct:
  - for Loop
  - while loop

For Loop

- The for loop in Python is used to iterate over a sequence (list, tuple, string) or other iterable objects. Iterating over a sequence is called traversal.
- Syntax:

```
for var in sequence:
    loop body
```



range() Function

- We can generate a sequence of numbers using range() function. range(10) will generate numbers from 0 to 9 (10 numbers).
- We can also define the start, stop and step size as
  - range(start, stop, step_size)
- step_size defaults to 1 if not provided.

for loop with else block

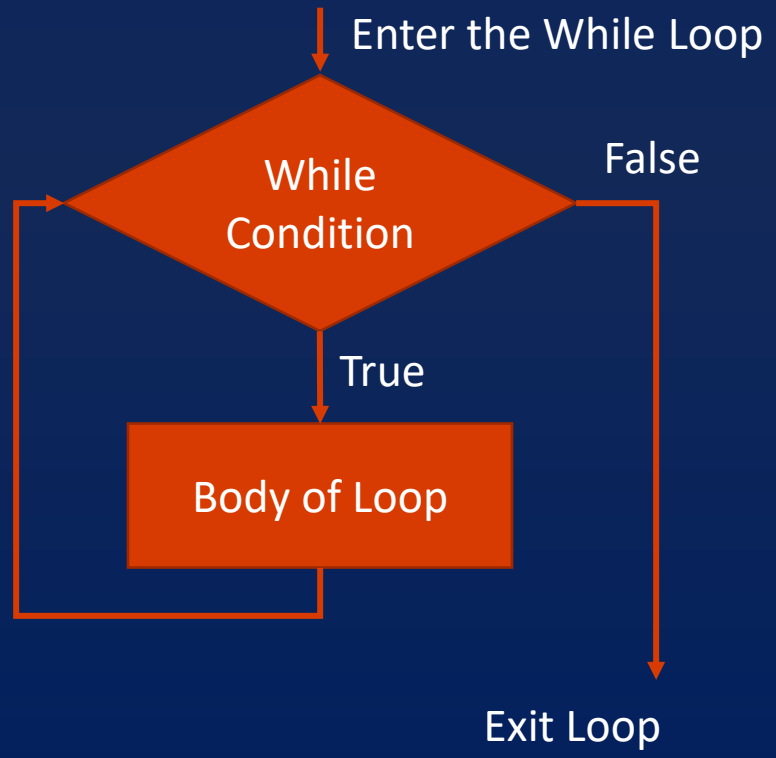
- A for loop can have an optional **else** block as well. The else part is executed if the items in the sequence used in for loop exhausts.
- The break keyword can be used to stop a for loop. In such cases, the else part is ignored.

```
n=int(input("Enter any Number : "))
for a in range(2,n):
    if n%a==0:
        print("No is Not Prime")
        break
else:
    print("No is Prime")
```

While Loop

- The while loop in Python is used to iterate over a block of code as long as the test expression (condition) is true.
- We generally use this loop when we don't know the number of times to iterate beforehand.
- Syntax:

while test_expression:
Body of while



Nested Loop

- A loop may also contain another loop within its body.
- This form of a loop inside another loop is called ***nested loop***.
- In a nested loop construction, the inner loop must terminate before the ending of outer loop.
- Example:

```
for a in range(1,6):
    for b in range(1,4):
        print("*",end="")
    print()
```

Jump Statement

- Those statement which transfer the program control, unconditionally within the program, from one place to another called jump statement.

  - ***break*** Statement
  - ***continue*** Statement
  - ***pass*** Statement

break Statement

- The break statement terminates the loop containing it. Control of the program flows to the statement immediately after the body of the loop.
- If the break statement is inside a nested loop (loop inside another loop), the break statement will terminate the innermost loop.
- Syntax:

break

continue Statement

- The continue statement is used to skip the rest of the code inside a loop for the current iteration only. Loop does not terminate but continues on with the next iteration.
- Syntax:

continue

```
for val in "string":
    if val == "i":
        continue
    print(val)
print("The end")
```

pass Statement

- In Python programming, the pass statement is a null statement. The difference between a comment and a pass statement in Python is that while the interpreter ignores a comment entirely, pass is not ignored.
- Syntax:

  pass

THANK YOU

Address: G. R. Complex Preetam Nagar, Dhoomanganj
Prayagraj Uttar Pradesh 211011

Mobile No.: 9598948810,8874588766,9532878816
Website : www.infomaxacademy.com

