

## Conditional Statements:

### ***if statement:***

Conditional Statement in Python perform different computations or actions depending on whether a specific Boolean constraint evaluates to true or false. Conditional statements are handled by IF statements in Python. The if statement is always followed by colon which marks the indentation, that is the block of code within the if condition. If indentation is not given then the interpreter throws an error.

The boolean conditions can be of the form (a>b or a==b or a in b etc)

```
In [51]: 1 #for eg:
          2 a = 2
          3 b = 4
          4 if (a > b):
          5     print (a, ' is greater than ',b)
```

### ***else statement:***

We have seen 'if' statement executes when a condition becomes True, now if the condition is False then in that case the else statement is executed. we don't have to give any condition with the else block

```
In [4]: 1 #for eg:
         2 a = 4
         3 b = 6
         4 if (a > b):
         5     print (a, ' is greater than ',b)
         6 else:
         7     print (a, ' is less than ',b)
```

4 is less than 6

### ***elif statement:***

If we want to evaluate multiple conditions then in that case we can use elif statement or else-if statement.

It's similar to an if-else statement and the only difference is that in else we will not check the condition but in elif we will do check the condition.

```
In [5]: 1 #for eg: Let
        2 x = 3
        3 if x == 1:
        4     print ('first code')
        5 elif x == 2:
        6     print ('second code')
        7 elif x == 3:
        8     print ('third code')
        9 elif x == 4:
       10     print ('fourth code')
       11 else:
       12     print ('No condition matched')
       13
```

third code

### ***Nested if-else:***

In case of nested if-else statement, there can be another if-else statement present within the if statement. That is if one set of condition is True only then it will evaluate the second set of conditions.

```
In [10]: 1 name = 'Paul'
        2 age = 18
        3 if age < 20:
        4     if name == 'Paul':
        5         print ('Hi Paul, you are invited for an event')
        6     else:
        7         print ('Sorry, we could\'nt find your name in the list')
        8 else:
        9     print ('Sorry, this event is meant for students below 20 years')
```

Hi Paul, you are invited for an event

## **Python Loops:**

Sometimes it is required to execute the part of the code repeatedly untill the condition is met. Then the loop terminate. Python provide two ways for executing the loops.

### ***While loop:***

In python, while loop is used to execute a block of statements repeatedly until a given a condition is satisfied. And when the condition becomes false, the line immediately after the loop in program is executed.

In case of while loop, we initialise one variable, then we evaluate the condition and finally we increment/decrement the value of the variable.

```
In [12]: 1 #for eg:
        2 x = 0      #initialising
        3 while x < 3: #evaluating
        4     print (x)
        5     x += 1   #incrementing
```

```
0
1
2
```

### **For loop:**

A for loop is used to iterate over a dataset, or it iterates over a certain range:

```
In [13]: 1 #for eg:
        2 #1) Iterating over a dataset:
        3 x = ['john','paul','mike','tom'] #lets consider this dataset
        4 for i in x:
        5     print (i)
```

```
john
paul
mike
tom
```

```
In [43]: 1 #2) Iterating over a dataset(1):
        2 x = ['john','paul','mike','tom']
        3 for i in range(len(x)):
        4     print (i)
```

```
0
1
2
3
```

```
In [14]: 1 #3) Iterating over a range of values:
        2 for i in range (0,3):
        3     print (i)
```

```
0
1
2
```

### **Break, pass and continue:**

#### **Break:**

The break statement is used to break the loop. The loop breaks when it encounters the break keyword and it throws to the immediate next line after the loop. for eg:

```
In [55]: 1 for i in range(0,5):  
2         if i == 1:  
3             break  
4         print (i)  
5     print ('continue from this line after the for loop')
```

```
0  
continue from this line after the for loop
```

**continue:**

The continue statement will skip the current value and will continue from the immediate next value.  
for eg:

```
In [57]: 1 for i in range (0,5):  
2         if i == 3 or i == 2:  
3             continue  
4         print (i)
```

```
0  
1  
4
```

**Pass:**

pass is just used to pass the flow. The utility of pass statement is when certain part of the code is not ready so in order to check rest of the code we can use pass statement as it will pass the flow normally.

```
In [59]: 1 for i in range (0,5):  
2         pass  
3     print ('continue from this line')
```

```
continue from this line
```

```
In [72]: 1 x = ['+918109340944', '+918839114952', '+11234567890', '123456789012', '986524']
2 for i in x:
3     if (len(i)==13 or len(i)==12):
4         if (i.startswith('+91')):
5             print (i, 'is an Indian number')
6         elif(i.startswith('+1')):
7             print (i, 'is a U.S number')
8         else:
9             print (i, 'is not a valid number')
10    else:
11        print (i, 'length did\'nt match')
12
13
```

```
+918109340944 is an Indian number
+918839114952 is an Indian number
+11234567890 is a U.S number
123456789012 is not a valid number
986524 length did'nt match
```

### Practise Questions:

- 1 1. Program to find the greater number between the two.
- 2 2. Program that takes user input and identifies even and odd number
- 3 3. Program to identify a valid mobile number
- 4 4. define a function that takes list of numbers as argument and output the number that are less than 20
- 5 5. Count the number of letters in refrigerator without using len() function.
- 6 6. find the length of each element of array
- 7 7. print the number of vowels occurring in 'refrigerator'
- 8 8. Write a program to find the number of vowels, consonents, digits and white space characters in a string. 'I am learning python'
- 9 9. Print all letters except 'E' in cricket.
- 10 10. Write a program to make a new string with all the vowels deleted from the string "Hello, have a good day" (output : Hll, hv gd dy)
- 11 11. Write a program to find out the largest and smallest word in the string "This is python programming".
- 12 12. Write a program to check if a given string is a Palindrome. A palindrome reads same from front and back e.g.- aba, ccaacc, mom, etc.
- 13 13. a = ['read', 'feed', 'string', 'at'] , Write a Python program to add 'ing' at the end of a given string (length should be at least 3). If the given string already ends with 'ing' then add 'ly' instead. If the string length of the given string is less than 3, leave it unchanged.
- 14 14. Write a Python program to count the number of even and odd numbers from a series of numbers. [15,13,17,19,14,6,61,62]
- 15 15. Form a pattern:
- 16 \*
- 17 \*\*
- 18 \*\*\*
- 19 \*\*\*\*
- 20 \*\*\*\*\*
- 21 \*\*\*\*
- 22 \*\*\*
- 23 \*\*

```

24      *
25 16. Form a pattern:
26      *
27      ***
28      *****
29      ***
30      *
31 17. Identify a fibnocci series (1,1,2,3,5,8,13...)
32 18. Identify a prime number in range (2,30)
33 19. write a program to calculate the average of a list of numbers
34      [1,2,3,4,5,6,7,8,9,10]
35 20. Print the multiplication table of 7 as (7*1=7 and so on)
36 21. Factorial of any number n is represented by n! and is equal to 1*2*3....
    (n-1)*n. Write a program to calculate factorial of a number.
37 22. Three digit number is called Armstrong number if sum of cube of its
    digit is equal to number itself. •E.g.- 153 is an Armstrong number because
    (13)+(53)+(33) = 153. •Write all Armstrong numbers between 100 to 500.
38 23. Write a Python program to find the median of given values
39 [8,10,15,20,11,17,30,22,13,16,23]
40 24. write a program to print patter 0 as :
41      ***
42      *      *
43      *      *
44      *      *
45      *      *
46      *      *
47      ***
48 25. Write a program that gives a list of elements which is reversed for eg:
49 ['string','integer','float','boolean']  output as : ['gnirts', 'regetni',
    'taolf', 'naeloob']
50
51 26. Print the following pattern:
52      ***
53      *      *
54      *      *
55      *****
56      *      *
57      *      *
58      *      *
59 27. Print the following pattern:
60      *****
61      *      *
62      *      *
63      *      *
64      *      *
65      *      *
66      *****
67 28. Print the following pattern:
68      *****
69      *
70      *
71      *
72      *****
73
74 29. Create a while loop which repeats itself and asks for the user input.
    If the user enters the letter 'q',it quits the while loop else it
    continues the loop.

```

```

75
76 30. Print all No.s between range (2000,3000) which are divisible by 7 but
    not by 5.
77
78 31. For x = 'hello world 1234', count the number of alphabets, numbers and
    spaces
79
80 32. For x = ['john','smith','michael','david','abrahm','nicola'], format
    the list such that the first letter of every name is in capitals.
81 Expected result : ['John', 'Smith', 'Michael', 'David', 'Abrahm', 'Nicola']
82
83 33. x = ['Banett','Rose','Clare'] y = [24,29,20]
84 For this two lists, x represents the names and y represents the ages
    corresponding to names. Create a dictionary having keys as names and ages
    representing the values. Expected output: {'Banett': 24, 'Rose': 29,
    'Clare': 20}
85
86 34. x = ['right','a','heaven','is','when','the']
87 For list above, create another list in which elements are sorted out on
    basis of length of element.
88 Expected output: ['a', 'is', 'the', 'when', 'right', 'heaven']
89
90 35. t= 'program'
91 Apply string operations only and sort out the letters in the above word in
    alphabetical order.
92 Expected output: agmoprrr

```

### Answers:

In [15]:

```

1 x = int(input('enter first number '))
2 y = int(input('enter second number '))
3 if x>y:
4     print(x, ' is greater then ', y)
5 else:
6     print(y , ' is greater then ', x)

```

```

enter first number 24
enter second number 20
24 is greater then 20

```

In [16]:

```

1 x = int(input('enter a number'))
2 if x% 2 == 0:
3     print(x , 'is even')
4 else:
5
6     print(x , 'is odd')

```

```

enter a number26
26 is even

```

```
In [17]: 1 t= ['+918109340944','+918839114952','+19876543210','123456','123456789123']
2 for x in t:
3     # x = str(input('Please enter a phone number '))
4     if len(x)==13 or len(x)==12:
5         if x.startswith('+91'):
6             print (x, ' is Indian')
7         elif x.startswith('+1'):
8             print (x, ' is from U.S')
9         else:
10            print (x, ' Unable to Match the number')
11    else:
12        print (x, 'Not a valid number')
13
```

```
+918109340944 is Indian
+918839114952 is Indian
+19876543210 is from U.S
123456 Not a valid number
123456789123 Unable to Match the number
```

```
In [18]: 1 x= [25,12,8,7,21,35,50,96,102]
2 for i in range (len(x)):
3     if (x[i]<20):
4         print (x[i], ' is less than 20')
5     else:
6         print (x[i], ' is more than 20')
```

```
25 is more than 20
12 is less than 20
8 is less than 20
7 is less than 20
21 is more than 20
35 is more than 20
50 is more than 20
96 is more than 20
102 is more than 20
```

```
In [19]: 1 x = 'refrigerator'
2 counter = 0
3 for i in x:
4     counter += 1
5 print (counter)
```

```
12
```

```
In [20]: 1 r = ['abc','in','abcd','pqrst','uvwxyz']
2 for i in r:
3     print (i, 'the length is', len(i) )
4
```

```
abc the length is 3
in the length is 2
abcd the length is 4
pqrst the length is 5
uvwxyz the length is 6
```



```
In [21]: 1 x = 'refrigerator'
2 counter = 0
3 for i in x:
4     if i=='a' or i=='e' or i=='i' or i=='o' or i=='u':
5         counter += 1
6     print (counter)
```

5

```
In [22]: 1 x = 'I am learning python'
2 y = x.lower()
3 counter_1 = 0
4 counter_2 = 0
5 counter_3 = 0
6 for i in y:
7     if i=='a' or i=='e' or i=='i' or i=='o' or i=='u' :
8         counter_1+=1
9     elif i == ' ':
10        counter_2+=1
11    else:
12        counter_3+=1
13    print (counter_1)
14    print (counter_2)
15    print (counter_3)
16    len(y)
```

6

3

11

Out[22]: 20

```
In [23]: 1 n = 'cricket'
2 for i in n:
3     if i=='e':
4         continue
5     print(i)
```

c

r

i

c

k

t

```
In [24]: 1 x="Hello, have a good day"
2 y=''
3 for i in x:
4     if i!='a' and i!='e' and i!='i' and i!='o' and i!='u':
5         y +=i
6     print (y)
```

Hll, hv gd dy

```
In [25]: 1 a = "This is an umbrella"
2 a = a.split()
3 maxx = a[0]
4 max_len = len(a[0])
5 for i in a:
6     if len(i)>max_len:
7         max_len = len(i)
8         maxx = i
9 print (maxx)
```

umbrella

```
In [26]: 1 x = ['mom', 'ccaacc', 'at', 'the', 'cbc']
2 for i in x:
3     if i[0:]==i[::-1]:
4         print (i, 'is palindrome')
5     else:
6         print (i, 'is not palindrome')
```

mom is palindrome  
ccaacc is palindrome  
at is not palindrome  
the is not palindrome  
cbc is palindrome

```
In [27]: 1 a = ['read', 'feed', 'string', 'at']
2 for i in a:
3     if (i.endswith('ing')):
4         y = i+'ly'
5         print (y)
6     elif (len(i)<3):
7         print (i)
8     else:
9         y = i+'ing'
10        print (y)
```

reading  
feeding  
stringly  
at

```
In [28]: 1 y = [1,5,6,4,7,9,12,15]
2 Even_counter = 0
3 odd_counter = 0
4 for i in y:
5     if (i%2==0):
6         Even_counter +=1
7     else:
8         odd_counter +=1
9
10 print (Even_counter)
11 print (odd_counter)
```

3  
5

In [29]:

```

1 x = '*'
2 counter = 1
3 while (counter<5):
4     print (x*counter)
5     counter+=1
6 while (counter>0):
7     print (x*counter)
8     counter-=1
9

```

```

*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*

```

In [67]:

```

1 i = 1
2 j = 2
3 while i>=1:
4     a = " "*j+"*"*i+" "*j
5     print (a)
6     i = i+2
7     j = j-1
8     if i>5:
9         break
10 i = 3
11 j = 1
12 while i>=1:
13     a = " "*j+"*"*i+" "*j
14     print (a)
15     i = i-2
16     j = j+1

```

```

*
***
*****
***
*

```

In [69]:

```

1 x = [1,2,3,4,5,6,7,8,9,10]
2 y=0
3 for i in x:
4     y+=i
5 print('Avg is ', y/len(x))

```

Avg is 5.5

```
In [72]: 1 x = int(input('give the number to know its multiplication table upto 10'))
2 for i in range(1,11):
3     k=x*i
4     print ('{}*{}={}'.format(x,i,k))
```

give the number to know its multiplication table upto 107

```
7*1=7
7*2=14
7*3=21
7*4=28
7*5=35
7*6=42
7*7=49
7*8=56
7*9=63
7*10=70
```

```
In [77]: 1 #6*5*4*3*2*1
2 x = int(input('give the number to calculate its factorial '))
3 y=x
4 while (x>1):
5     x-=1
6     y*=x
7 print(y)
```

give the number to calculate its factorial 8

```
40320
```

```
In [84]: 1 # x = '153'
2 for j in range (100,500):
3     y=0
4     j=str(j)
5     for i in j:
6         y+=int(i)**3
7     if (str(y)==j):
8         print (j, 'its an armstrong no.')
9     # else:
10    #     print(j, 'its not an armstrong no.')
11
```

```
153 its an armstrong no.
370 its an armstrong no.
371 its an armstrong no.
407 its an armstrong no.
```

In [93]:

```

1 x = [8,10,15,20,11,17,30,22,13,16,23]
2 x.sort()
3 print (x)
4 if (len (x)%2==0):
5     i = int(len(x)/2)
6     j = i+1
7     print ((x[i]+x[j])/2)
8 else :
9     i = len(x)//2
10    print (x[i])
11
12

```

[8, 10, 11, 13, 15, 16, 17, 20, 22, 23, 30]  
16

In [96]:

```

1 x = '*'
2 for i in range (0,7):
3     if (i==0 or i==6):
4         print (' '+x*3+' ')
5     else:
6         print (x+' '+x)

```

```

***
*      *
*      *
*      *
*      *
*      *
***

```

In [97]:

```

1 x = ['string','integer','float','boolean']
2 y = []
3 for i in x:
4     y.append(i[::-1])
5 print (y)

```

['gnirts', 'regetni', 'taolf', 'naeloob']

In [42]:

```

1 a = 0
2 b = 1
3
4 for i in range (0,7):
5     temp = a
6     a = b
7     b = temp+b
8     print (a)

```

1  
1  
2  
3  
5  
8  
13

```
In [31]: 1 for i in range(2,15):
2         counter = 0
3         for j in range(2,i):
4             if (i%j == 0):
5                 counter = 1
6         if (counter==0):
7             print (i, ' is a prime number')
8
```

```
2 is a prime number
3 is a prime number
5 is a prime number
7 is a prime number
11 is a prime number
13 is a prime number
```

```
In [32]: 1 for i in range (0,9):
2         if i==0:
3             print (' '+'***'+ ' ')
4         elif i == 4:
5             print ('*****')
6         else:
7             print('*'+ ' '+'*')
```

```
***
*      *
*      *
*      *
*****
*      *
*      *
*      *
*      *
```

```
In [34]: 1 while True:
2         x = input('enter Q to quit ')
3         if x=='q' or x == 'Q':
4             break
5
```

```
enter Q to quit Q
```

```
In [ ]: 1 for i in range(2000,3000):
2         if i%7==0:
3             if i%5!=0:
4                 print (i)
```

```
In [35]: 1 x = 'hello world 1234'
2 char_ =0
3 dig_ =0
4 spc_ =0
5 for i in x:
6     if i.isdigit():
7         dig_ += 1
8     elif i.isalpha():
9         char_ += 1
10    else:
11        spc_ += 1
12 print (dig_)
13 print (char_)
14 print (spc_)
15
```

```
4
10
2
```

```
In [36]: 1 x = ['john','smith','michael','david','abrahm','nicola']
2 y = []
3 for i in x:
4     y.append(i[0].upper()+i[1:])
5 print (y)
```

```
['John', 'Smith', 'Michael', 'David', 'Abrahm', 'Nicola']
```

```
In [37]: 1 for i in range (0,8):
2     if i == 0 or i ==7:
3         print ('****')
4     else:
5         print ('*   *')
6
```

```
****
*   *
*   *
*   *
*   *
*   *
*   *
****
```

```
In [38]: 1 x = ['Banett','Rose','Clare']
2 y = [24,29,20]
3 z= {}
4 for i in range (len(x)):
5     z[x[i]]=y[i]
6
7 print (z)
8
```

```
{'Banett': 24, 'Rose': 29, 'Clare': 20}
```

```
In [39]: 1 i=7
2 y= ' '
3 while i>=0:
4     if (i==0 or i==7):
5         print ('*****')
6         i-=1
7     else:
8         print(y*i+'*')
9         i-=1
```

```
*****
      *
     *
    *
   *
  *
 *
*****
```

```
In [40]: 1 x = ['right', 'a', 'heaven', 'is', 'when', 'the']
2 y = []
3 for i in range(len(x)):
4     for j in x:
5         if (len(j)==i+1):
6             y.append(j)
7 print (y)
```

```
['a', 'is', 'the', 'when', 'right', 'heaven']
```

```
In [41]: 1 t= 'program'
2 n = []
3 for i in t:
4     n.append(i)
5 n.sort()
6 m = ''.join(n)
7 print (m)
```

```
agmoprr
```