

While Loop

```
In [ ] : While --> When we want to executed a group of statement multiple times based on a given condition
         then we will While Loop.
Syntax of while loop:
    while condition:
        body
        increement/decrement
```

Examples

```
In [1]: i=1
        while i<=3:
            print(i)
            print("Hello World")
            i=i+1

1
Hello World
2
Hello World
3
Hello World
```

Infinite Loops in Python

```
In [ ] : if you want to create infinite loops in python then Within the condition you can write any constant
         value like 1,2,3,except 0,False and empty String.
```

Examples

```
In [ ] : while 1:
         print("hello")
```

```
In [ ] : while 2:
         print("Hello World")
```

```
In [ ] : while "Hello":
         print("Hello World")
```

Practice Problem Using While and For Loops

Python Program to find the Factorial of a Number Using For Loop

```
In [4]: num=int(input("Enter a number"))
        fact=1
        for i in range(1,num+1):
            fact=fact*i
        print(fact)

Enter a number6
720
```

Python Program to find the Factorial of a Number Using While Loop

```
In [6]: num=int(input("Enter a number"))
        i=1
        fact=1
        while i<num+1:
            fact=fact*i
            i=i+1
        print(fact)

Enter a number6
720
```

Python Program to Find the sum of N natural Numbers Using for loop

```
In [19]: sum=0
         n=int(input("Enter the number: "))
         for i in range(n+1):
             sum=sum+i
         print(sum)

Enter the number: 10
55
```

Python Program to Find the sum of N natural Numbers Using while loop

```
In [21]: i=0
         sum=0
         n=int(input("Enter the number"))
         while i<=n:
             sum=sum+i
             i=i+1

         print(sum)

Enter the number10
55
```

Transfer Statements

```
In [ ] : Transfer statements are basically used to Transfer the flow of the Program.
```

There are two keywords that we used for transferring the flow of the program:

- 1.break
- 2.continue

Break --> If you are using break then Break will Terminate(Stop) the loop whenever the pointer encounter to that break statement.

Continue -->If you are using continue then Continue will Skipp the current iteration of the loop whenever the pointer encounter to that Continue. statement.

Note: If you are using nested loop then break will stop the inner loop only. That means break is capabale to stop one loop at a time during iteration.

Examples of Break Statement

```
In [10]: #break statement with single loop
         for i in range(7):
             if i==4:
                 print("Break Encountered")
                 break
             print(i)

0
1
2
3
Break Encountered
```

```
In [8]: #Break Statement with nested loops:
         for i in range(4): #0,1,2,3
             for j in range(4): #0,1,2,3
                 if j==2:
                     print("Break encountered")
                     break
                 print("i value is",i ,j)

i value is 0 0
i value is 0 1
Break encountered
i value is 1 0
i value is 1 1
Break encountered
i value is 2 0
i value is 2 1
Break encountered
i value is 3 0
i value is 3 1
Break encountered
```

```
In [11]: cart = [10,20,300,500,800,400]
         for item in cart:
             if item>=500:
                 print("We cannot purchase this item")
                 break
             print(item)

10
20
300
We cannot purchase this item
```

Examples of Continue Statement

```
In [13]: #continue statement with single loop
         for i in range(7):
             if i==4:
                 print("Continue Encountered")
                 continue
             print(i)

0
1
2
3
Continue Encountered
5
6
```

```
In [14]: #Continue Statement with nested loops:
         for i in range(4): #0,1,2,3
             for j in range(4): #0,1,2,3
                 if j==2:
                     print("Continue encountered")
                     continue
                 print("i value is",i ,j)

i value is 0 0
i value is 0 1
Continue encountered
i value is 0 3
i value is 1 0
i value is 1 1
Continue encountered
i value is 1 3
i value is 2 0
i value is 2 1
Continue encountered
i value is 2 3
i value is 3 0
i value is 3 1
Continue encountered
i value is 3 3
```

```
In [15]: cart = [10,20,300,500,800,400]
         for item in cart:
             if item>=500:
                 print("We cannot purchase this item")
                 continue
             print(item)

10
20
300
We cannot purchase this item
We cannot purchase this item
400
```

Pass Statement

```
In [34]: pass keyword --> The pass statement is used as a placeholder for future code.
         When the pass statement is executed, nothing happens,
         but you avoid getting an error when empty code is not allowed.
```

Examples:

```
In [16]: x=10
         if x==10:
             pass
```

```
In [17]: def m1():
         pass
```

```
In [18]: for i in range(10):
         pass
```

For vs While

```
In [ ] : for --> loops are basically used when we know the number of iterations or sequence.
         while loop--> if we want to execute a group of statement based on certain condition.
```