**Date :** 28th - 08 - 2020
**Morning Session** : 9am – 11.00 PM
**By ~** Rohan Kumar

## Topics: Python Loops

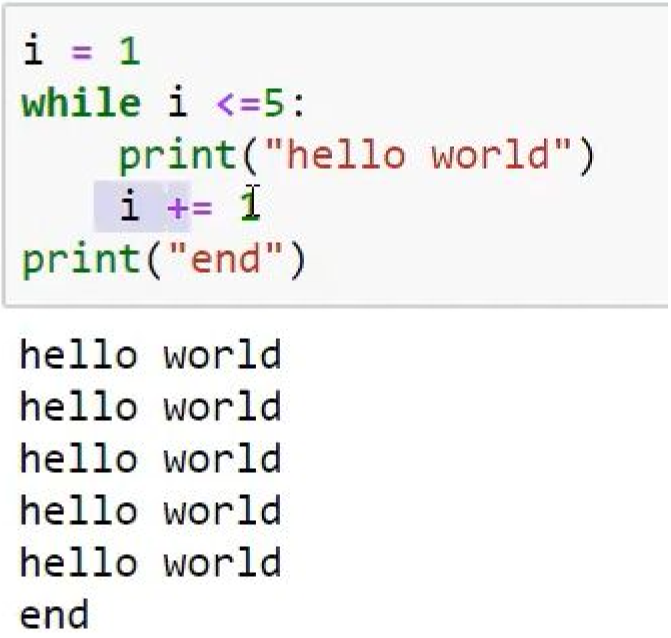
**Loop:** A loop statement allows us to execute a statement or group of statements multiple times.

1) **While**
2) **For**

**While:** 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.

**Syntax:**

```
while expression:
    statement(s)
```

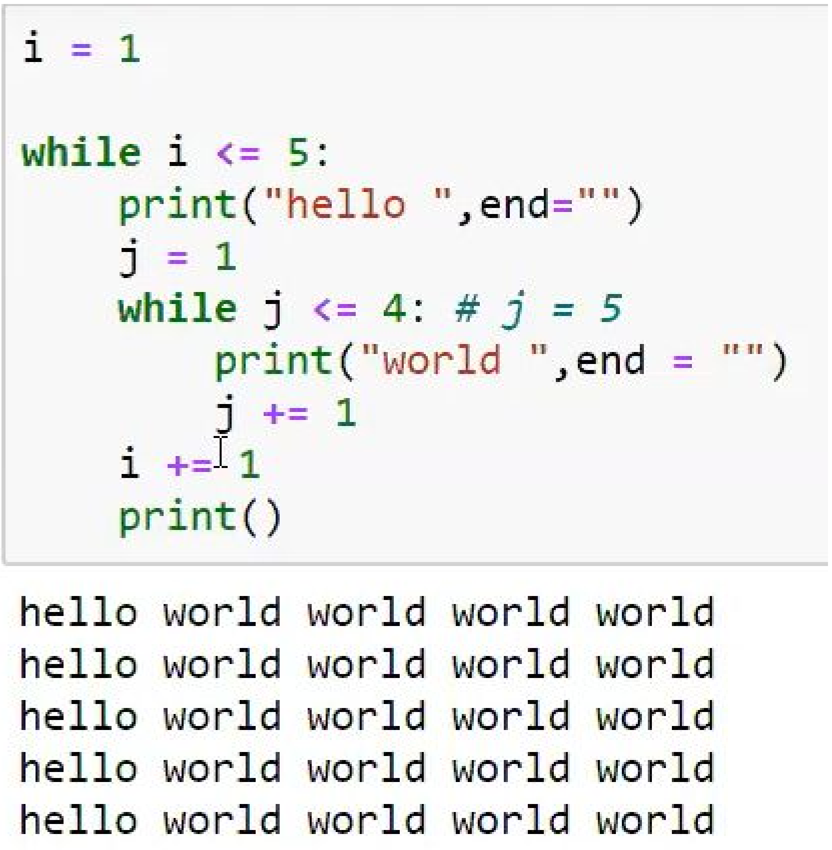


```
i = 1
while i <=5:
    print("hello world")
    i += 1
print("end")
```

```
hello world
hello world
hello world
hello world
hello world
end
```

**Always in while loop there things shouble there**

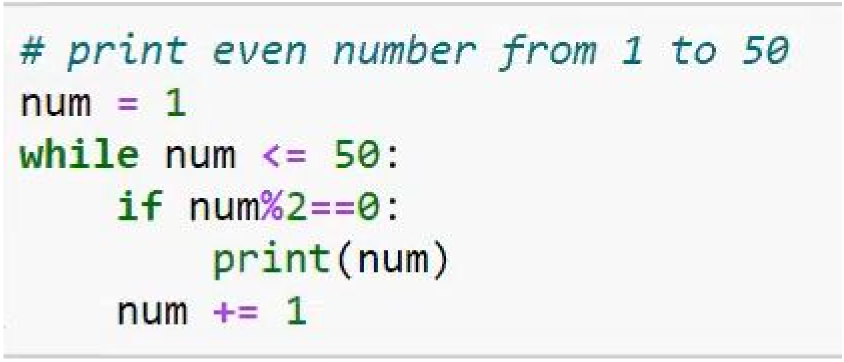
1) **Initialize 2) condition 3) increment or decrement**



```
i = 1

while i <= 5:
    print("hello ",end="")
    j = 1
    while j <= 4: # j = 5
        print("world ",end = "")
        j += 1
    i += 1
    print()
```

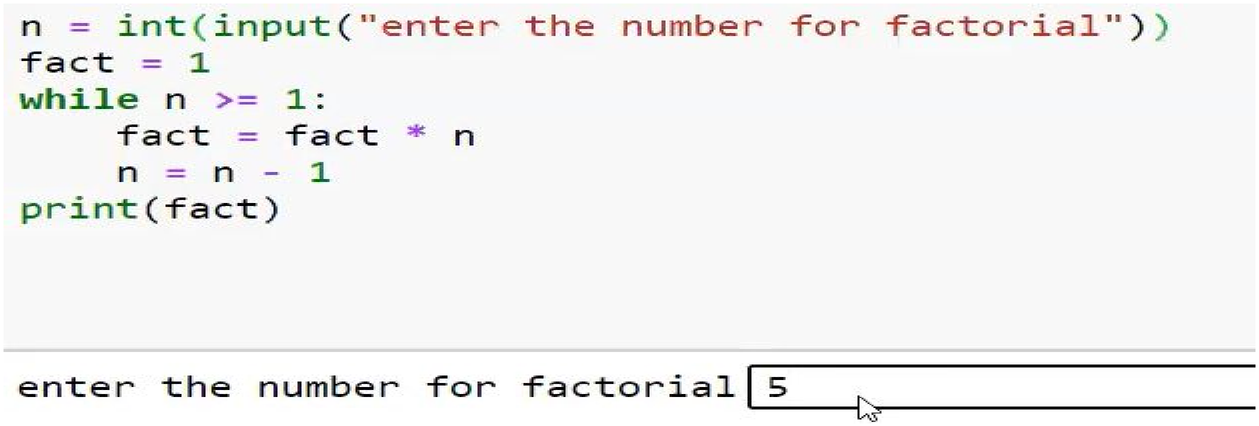
```
hello world world world world
hello world world world world
hello world world world world
hello world world world world
hello world world world world
```



```
# print even number from 1 to 50
num = 1
while num <= 50:
    if num%2==0:
        print(num)
    num += 1
```

**Output:** 1 to 50 even numbers

## Factorial:



```
n = int(input("enter the number for factorial"))
fact = 1
while n >= 1:
    fact = fact * n
    n = n - 1
print(fact)
```

```
enter the number for factorial 5
```

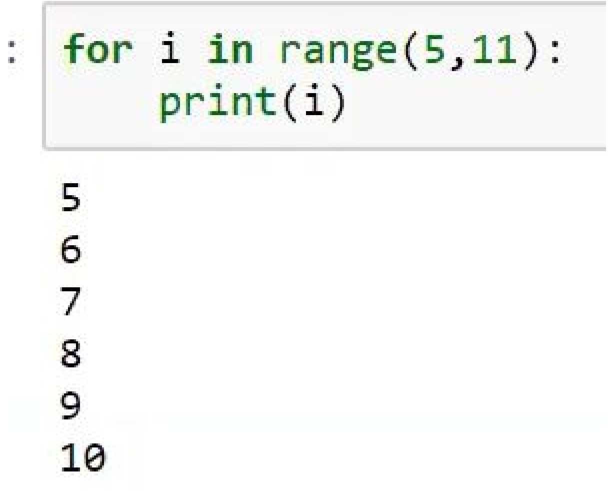
**Output:** 120

**For Loop:** Executes a sequence of statements multiple times and abbreviates the code that manages the loop variable.

**Syntax:**

```
for iterator_var in sequence:

    statements(s)
```



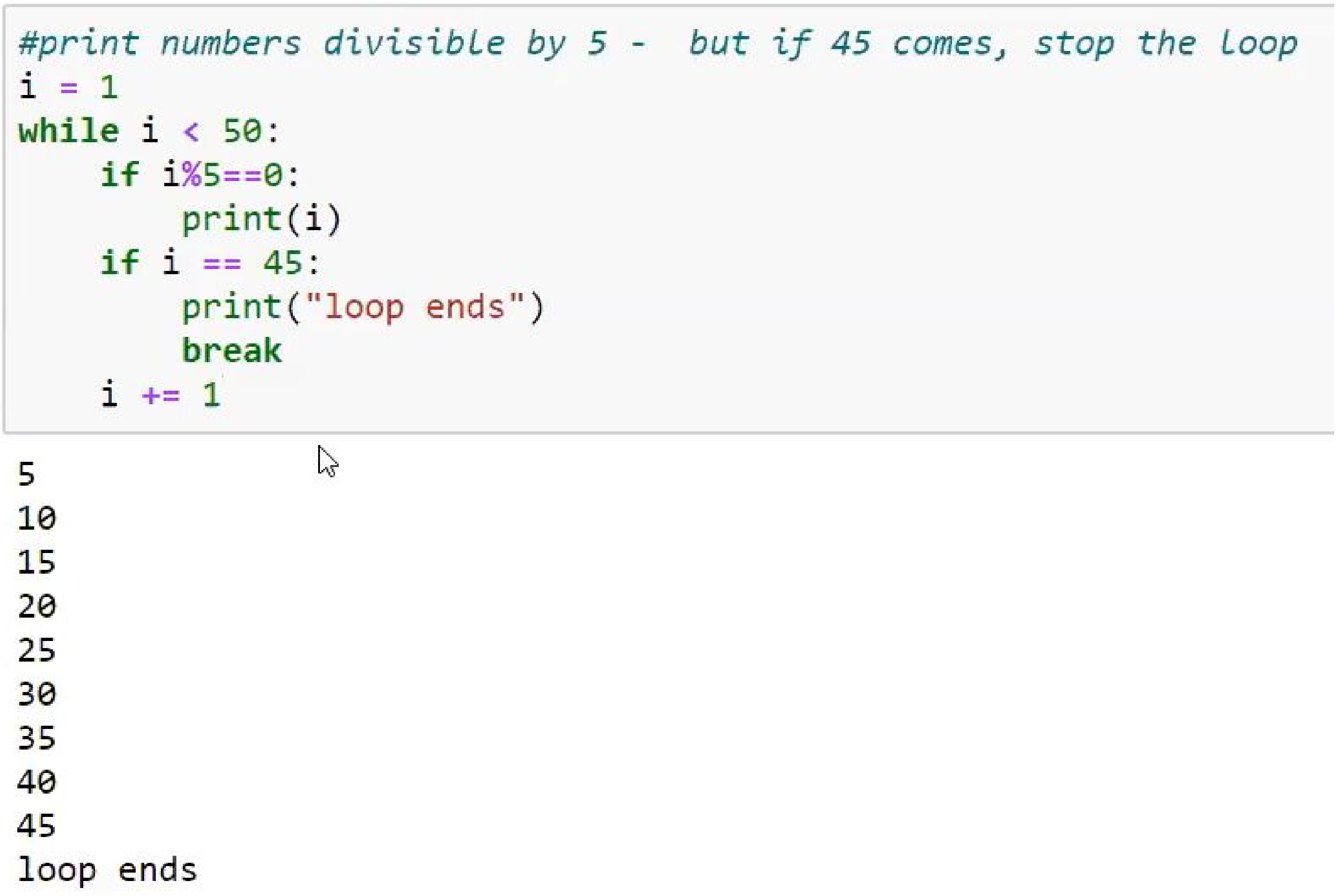
```
for i in range(5,11):
    print(i)
```

```
5
6
7
8
9
10
```

**Loop Controls :** Loop control statements change execution from its normal sequence. When execution leaves a scope, all automatic objects that were created in that scope are destroyed.

1) **Break**
2) **Continue**
3) **pass**

1) **Break :** Terminates the loop statement and transfers execution to the statement immediately following the loop.



```
#print numbers divisible by 5 -  but if 45 comes, stop the loop
i = 1
while i < 50:
    if i%5==0:
        print(i)
    if i == 45:
        print("loop ends")
        break
    i += 1
```

```
5
10
15
20
25
30
35
40
45
loop ends
```

2) **Continue :** It returns the control to the beginning of the while loop.. The continue statement rejects all the remaining statements in the current iteration of the loop and moves the control back to the top of the loop.

3) **Pass:** It is used when a statement is required syntactically but you do not want any command or code to execute. The pass statement is a *null* operation; nothing happens when it executes. The pass is also useful in places where your code will eventually go, but has not been written yet.