

# DAY-10

## September 15

1. Write a program to check the given number is prime or not sample:

7 2,7 i =

2,3,4,5,6

CODE:

```
num = int(input())
```

```
count = 0 for i in
```

```
range(2,num,1):
```

```
if(num%i==0):
```

```
count = count+1
```

```
else:
```

```
count = count if(count==0):
```

```
print("prime number") else:
```

```
print("not a prime number") OUTPUT:
```

9 not a prime

number **Break:**

stops the execution

by the given

condition.

The break statement in Python is used to exit or “break” out of a loop (either a for or while loop) prematurely, before the loop has iterated through all its items or reached its condition. When the break statement is executed, the program immediately exits the loop, and the control moves to the next line of code after the loop.

## **CONTINUE:**

Python Continue statement is a loop control statement that forces to execute the next iteration of the loop while skipping the rest of the code inside the loop for the current iteration only, i.e. when the continue statement is executed in the loop, the code inside the loop following the continue statement will be skipped for the current iteration and the next iteration of the loop will begin.

## **PASS:**

Pass statement in Python is a null operation or a placeholder. It is used when a statement is syntactically required but we don't want to execute any code. It does nothing but allows us to maintain the structure of our program.

**EXAMPLE: for i in**

```
range(1,100,1):
```

```
    if(i==20):
```

```
        break
```

```
    else:
```

```
        print(i)
```

**OUTPUT:**

1 2

3

4 5

6 7

8 9

10 11

12 13

14 15

16 17

18 19

EXAMPLE: **for i in**

**range(1,10,1):**

**if(i==5):**

**continue    else:**

**print(i)**

OUTPUT:

1 2

3 4

6

7 8

9

EXAMPLE: **for i in**

**range(1,10,1):**

**if(i==6):        pass**

**else:**

**print(i)**

OUTPUT:

1 2

3

4

5

7

8

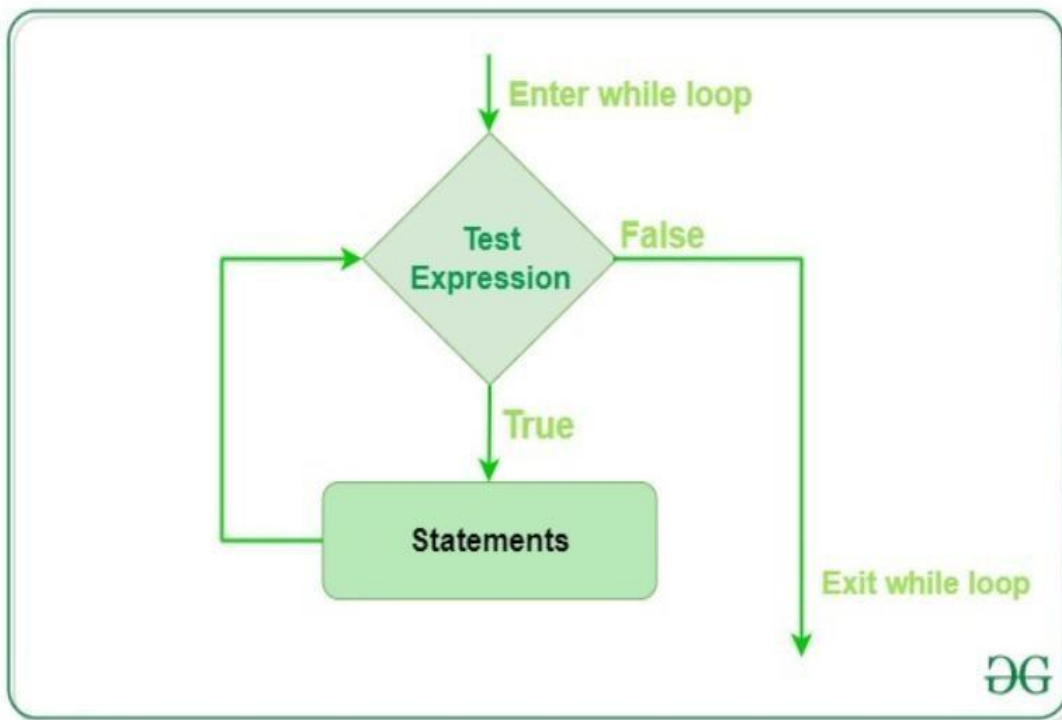
9

**While loop:** While Loop is used to execute a block of statements repeatedly until a given condition is satisfied. When the condition becomes false, the line immediately after the loop in the program is executed.

initialization while(condition):

statements

incrementation/decrementation



EXAMPLE: s

= "python" i

= 0

**while**(i<len(s)): #i = 0,1,2,3,4,5

print(s[i]) i +=1 #i = i+1

OUTPUT:

p

y t

h o n

2. Write a program to print divisibles of 3 from 1 to 20

CODE:

i = 1

**while**(i<=

20):

**if**(i%3==0

):

print(i)

**else:**

**pass**

i=i+1

OUTPUT:

3 6

9

12

15

18