Python | Output using print() function

- The simplest way to produce output is using the print() function where you can pass zero or more expressions separated by commas.
- This function converts the expressions you pass into a string before writing to the screen.

In [ ]:

```
print('Hello World')
```

In [ ]:

```
a = 10
b = 20
x = a + b
print(a)
```

In [ ]:

```
Case Sensitive Language
print <> Print <> PRINT
```

In [ ]:

```
print("Hello World") #quotations doesn't matter
```

In [ ]:

```
print('Hello world')
            print('Hello World')
```

input ( )

- This function first takes the input from the user and then evaluates the expression, which means Python automatically identifies whether user entered a string or a number or list.
- If the input provided is not correct then either syntax error or exception is raised by python

eval ( )

- The eval() method parses the expression passed to it and runs python expression(code) within the program.

In [ ]:

```
# Write first Program

num1 = eval(input('Enter your weight in kilograms:'))
print('Your weight in Pounds is: ', num1*2.2)
```

In [ ]:

```
type(num1)
```

In [ ]:

```
# input function by default comes as a str
```

In [ ]:

```
a = input('Your age:')
```

In [ ]:

```
type(name)
```

In [ ]:

```
name = input('Please enter your name:')
print('Hello',name,'!!!')
```

In [ ]:

```
num1 = eval(input('Enter your weight in kilograms:'))
num2 = eval(input('Enter your Height in Cms:'))
print('Your Height is {} and your weight in Pounds is:{}'.format(num2, num1*2.2))
```

In [ ]:

```
num1
```

In [ ]:

```
print('The avg weight of a product:',num1, sep = '---')
```

In [ ]:

```
num1 = eval(input('Enter your weight in kilograms:'))
num2 = eval(input('Enter your Height in Cms:'))
print('Your Height is {} and your weight in Pounds is:{}'.format(num2, num1*2.2))
```

In [ ]:

```
print(num1, 'is the Random number generated', sep = '---')
```

In [ ]:

```
print('The first line is', end = '---')
print('And the second line is')
```

In [ ]:

```
num = range(5,10)
num
```

In [ ]:

```
range(100,50, -2)
```

In [ ]:

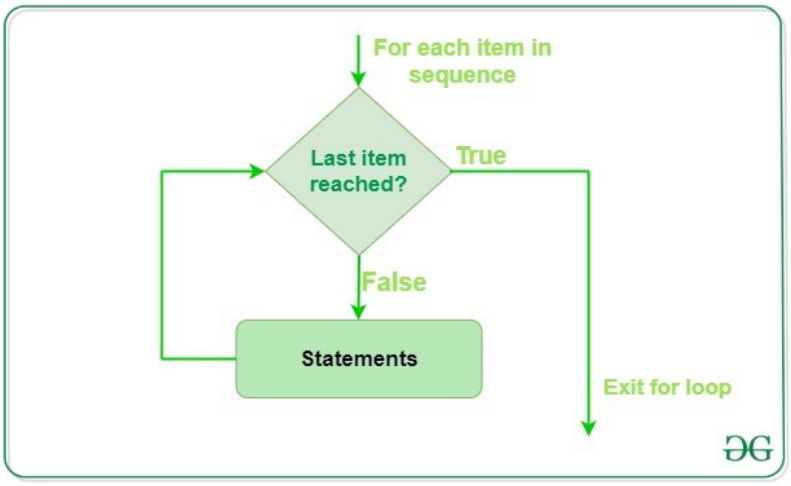
```
range(10)
```

loops

- Python programming language provides following types of loops to handle looping requirements.
- Python provides three ways for executing the loops. While all the ways provide similar basic functionality, they differ in their syntax and condition checking time.

for Loop:

- For loops are used for sequential traversal. For example: traversing a list or string or array etc.
- In Python, there is no C style for loop, i.e., for (i=0; i<n; i++).
- There is “for in” loop which is similar to for each loop in other languages.



In [ ]:

```
for i in range(10):
    print(i)
```

In [ ]:

```
for i in range(10):
    print(i, end = ' ')
```

In [ ]:

```
for i in range(2,5):
    print(i)
```

In [ ]:

```
for i in range(2,11,2):
    print(i)
```

In [ ]:

```
Range = eval(input('Please enter a range: '))
for i in range(Range):
    num = eval(input('Please specify a number to be squared: '))
    print("The Squared Value of the given number is:", num*num)
print('The Loop is done now!!')
```

Random Numbers in Python

- Python defines a set of functions that are used to generate or manipulate random numbers.
- This particular type of functions are used in a lot of games, lotteries or any application requiring random number generation.

In [ ]:

```
from random import randint
```

In [ ]:

```
import random
```

In [ ]:

```
a = randint(1,10)
Guess = eval(input('Please Guess a number:'))
if a == Guess:
    print('You are a Genius')
else:
    print('Try again!')
```

In [ ]:

```
Guess = eval(input('Please provide your Temperature:'))
if Guess<10:
    print('Very Cold, get a jacket!')
elif Guess >10:
    print('You are in a better place')
else:
    print('Good Job!')
```

In [ ]:

```
print('**************')
print('**************')
print('**************')
print('**************')
```

In [ ]:

```
print('**************')
print('*            *')
print('*            *')
print('**************')
```