

* Recursion in Python :-

- A function that calls itself is a recursive function.
- Recursion is the process of defining something in terms of itself.

```
def recurse():  
    ...  
    recurse()  
    ...  
  
recurse()
```

(Recursive function in Python)

* Example of recursive function

Program to compute factorial of a number using recursion.

Factorial of a number is the product of all the integers from 1 to that number.
for example $6!$ ($6 \times 5 \times 4 \times 3 \times 2 \times 1 = 720$)

Example

```
def fact(n):
```

```
    if n == 1:
```

```
        return 1
```

```
    else:
```

```
        return(n * fact(n-1))
```

```
# function call
```

```
num = 5
```

```
print ("The factorial of", num, "is", fact(num))
```

Output

The factorial of 5 is 120.

* Advantages of Recursion:-

- Recursive functions divide the problem into simple sub-problems.
- Recursive functions make the code look clean and elegant.
- Sequence generation is simpler in recursion such as fibonacci series as compared to that of loops.

* Disadvantages of Recursion:

- Recursive code is hard to develop and debug.
- Difficult to understand.
- Recursive calls are expensive, as they take up a lot of memory and time.