**Functions**

**Function:** Functions in Python are defined using the **def** keyword, followed by the function name, parentheses () for parameters, and a colon:. The body of the function is indented.

A function is a block of organized, reusable code that performs a specific task.

It is defined as a block of code which can used to perform an action.

Functions will be executed whenever we make a function call. And we can call the function for ‘n’ number of times.

**Syntax:**

**def function\_name (parameters):**

**statements**

**return value**

**There are some main functions in this they are:**

1. Built-in Functions
2. User-defined Functions
3. Lambda Functions
4. Recursive Functions
5. Higher-order Functions

**Built-in Function:**

The functions which comes along with python software are known as Built-in-function or it is also known as pre-defined functions.

Ex: input( ), id( ), type( ),… etc.

text = "sravya"  
print(len(text))

Output: 6

**User-defined Function:**

This function which are developed by the programmers explicity according to their business requirements.

functions that you define using the **def** keyword. These functions can accept parameters, perform operations, and return values.

**Lambda Function:**

Lambda functions are small, anonymous functions that are defined using the lambda keyword. They are typically used for short, simple operations that do not require a full function definition.

It is used to create anonymous (nameless) functions. These functions are also called lambda functions.

**Syntax:**

**lambda arguments: expression**

add = lambda x, y: x + y  
result = add(27, 31)  
print(result)

Output: 58

**Recursive Function:**

It is one that calls by itself method it is known as recursive function. It must have basis part and recursive part.

def factorial(n):  
 if n == 0 or n == 1:  
 return 1  
 else:  
 return n \* factorial(n-1)  
result = factorial(8)  
print("factorial of 8:", result)

Output:

factorial of 8: 40320

**Higher-order Functions**

These are functions that can take other functions as arguments or return them as results. Common examples are map ( ), filter(), and reduce().

Ex:

def square(x):

return x \* x

numbers = [1, 2, 3, 4, 5]

squared\_numbers = list(map(square, numbers))

print(squared\_numbers)

**Output:** [1, 4, 9, 16, 25]