

```

1 # Function
2
3 """
4 FUNCTION:
5     --> A process for executing a task.
6     --> It can accept input and return an output.
7     --> Useful for executing similar procedures over and over.
8     --> Reusable Code.
9     --> Stay DRY - Don't Repeat Yourself!
10    --> Clean Up and Prevent Code Duplication.
11    --> "Abstract Away" code for other users:
12        --> Imagine if you had to rewrite the "print()" function for every
program you wrote.
13    """
14
15    """
16 Defining Functions
17    """
18
19 def greet():
20     print("Hi")
21
22    """
23 Executing Function
24    """
25 greet()
26 greet()
27
28    """
29 Returning a Values From Function
30     --> return keyword:
31         --> Exits The Function.
32         --> Outputs Whatever Value is Placed After The return Keyword.
33    """
34 def print_square_of_7():
35     return 7 ** 2
36
37 square_of_7 = print_square_of_7()
38 print(square_of_7)
39
40    """
41 Parameters
42     --> Supports Default Parameters Also.
43    """
44 def add(x, y):
45     return x + y
46
47 def square(x):
48     return x * x
49
50 print(add(22, 11))
51 print(square(64))
52
53 # Default Parameters
54 def cube(x = 2):
55     return x ** 3
56
57 print(cube(5))
58 print(cube())
59

```

```
60 """
61 Parameters v/s Arguments:
62     --> A parameter is a variable in a method definition.
63     --> When a method is called, the arguments are the data you pass into the
method's parameters.
64     --> Parameter is a variable in the declaration of function.
65     --> In Above Example: 'x' is a parameter and data for x is argument.
66 """
67
68 """
69 Keyword Arguments
70 """
71 def full_name(first, last):
72     return f"Your Name is {first} {last}"
73
74 print(full_name(first="Abhishek", last="Baghel"))
75 print(full_name(last="Baghel", first="Dylan"))
```