# *CodroidHub Summer Training*

# Title:-

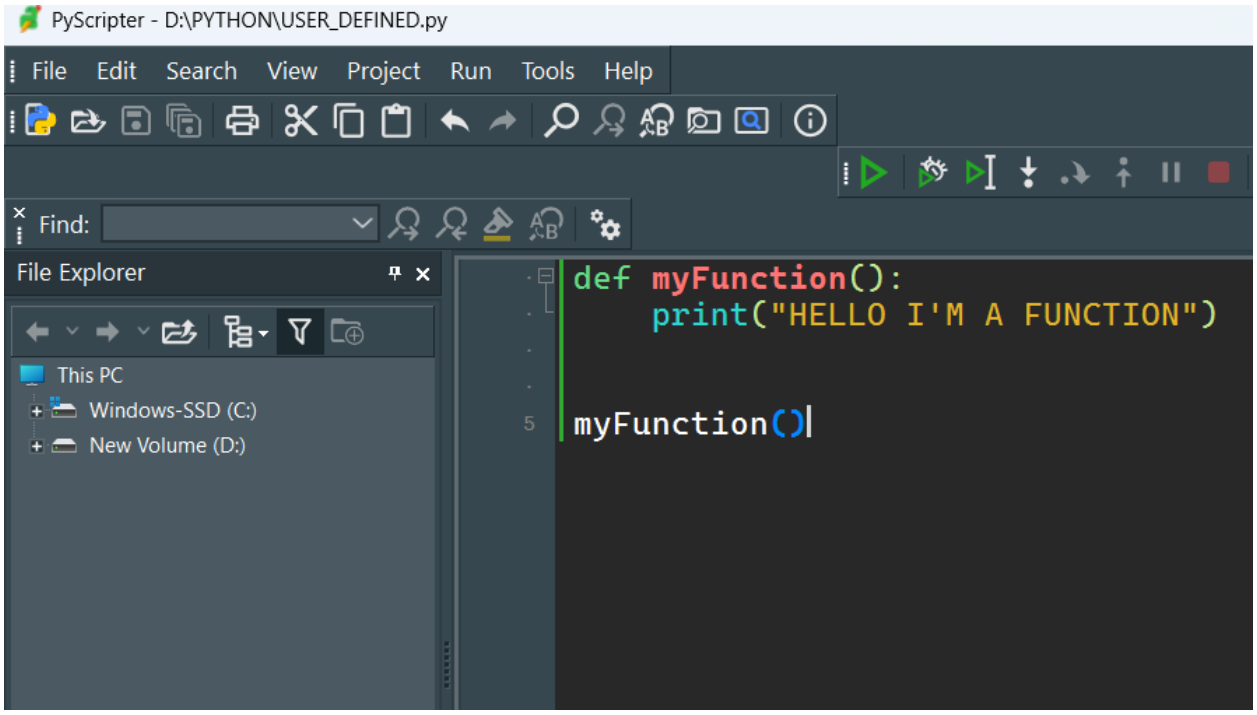
**FUNCTIONS in Python**
**PRE-DEFINED FUNCTIONS in Python**
**USER-DEFINED FUNCTIONS in Python**
**FUNCTION WITH PARAMETERS in Python**
**DEFAULT ARGUEMENT FUNCTION in Python**
**RECURSIVE FUNCTION in Python**

## *FUNCTIONS IN PYTHON*

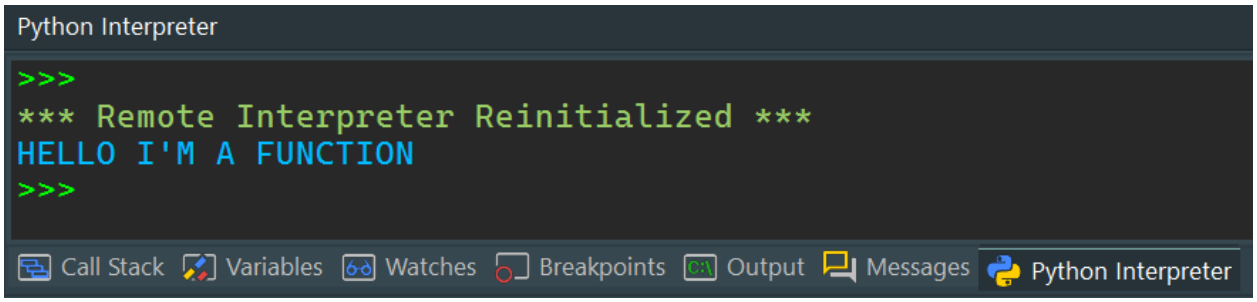
*A function is a block of code which only runs when it is called. You can pass data, known as parameters, into a function. A function can return data as a result.*

***Creating a Function***
*In Python a function is defined using the def keyword.*
***Calling a Function***
*To call a function, use the function name followed by parenthesis.*

*EXAMPLE:*




*OUTPUT:*




# *PRE-DEFINED FUNCTIONS IN PYTHON*

*A predefined function is a function that has already been written in the programming language and can be used by the programmer. A function will return a value that can be stored in a variable or sometimes in a conditional statement*
*There are many pre-defined or built in functions in python. Total there are 68 built in functions in python programming. For example: print(), list(), abs(), ascii(),all(), any(), max(), min() & many more.*

*EXAMPLE:*

PyScripter - D:\PYTHON\PRE-DEFINED_FUNCTION.py

```
#PRE-DEFIND FUNCTION
print("WELCOME TO FUNCTION!")

print(max(5,7,2))
print(min(5,7,2))
c=(3,4,5,7)
print("SUM=",sum(c))
print("MAX=",max(c))
print("MIN=",min(c))

print(abs(5.3))

a=(1,2,3,4,4,5,6)

print(a.count(4))
```

*OUTPUT:*

Python Interpreter

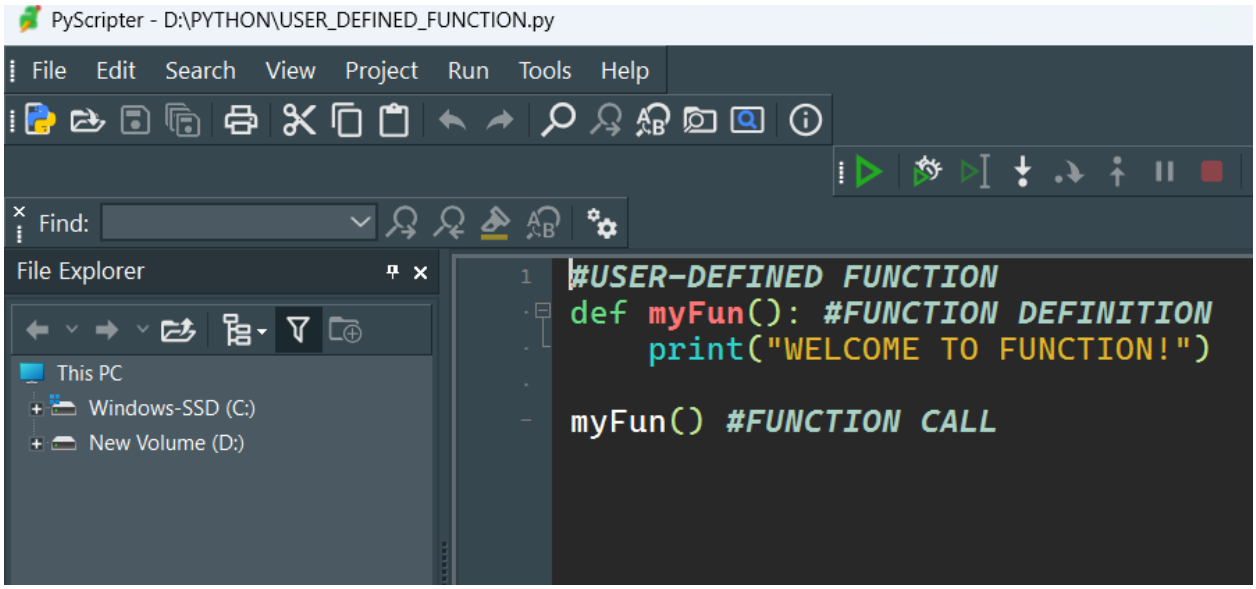
```
*** Remote Interpreter Reinitialized ***
WELCOME TO FUNCTION!
7
2
SUM= 19
MAX= 7
MIN= 3
5.3
2
>>>
```

# *USER-DEFINED FUNCTIONS IN PYTHON*

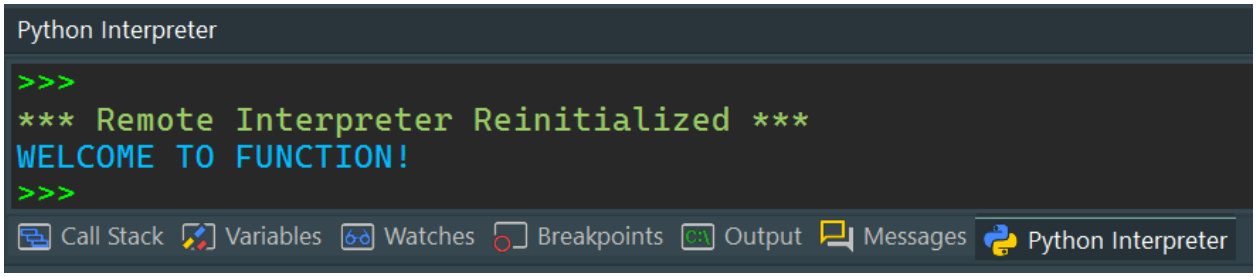
*All the functions that are written by any of us come under the category of user-defined functions. Below are the steps for writing user-defined functions in Python.*

- *In Python, a def keyword is used to declare user-defined functions.*
- *An indented block of statements follows the function name and arguments which contains the body of the function.*

*EXAMPLE:*




*OUTPUT:*




# *FUNCTION WITH PARAMETERS IN PYTHON*

*If you have experience in C/C++ or Java then you must be thinking about the return type of the function and data type of arguments. That is possible in Python as well .*

*EXAMPLE:*

PyScripter - D:\PYTHON\function_with_parameter.py

```python
#FUNCTION WITH PARAMETER
def myFun(message):
    print(message)

myFun("welcome")

def find_sum(a,b):

    print(a+b)

find_sum(9,7)

def sum(x,y):
    return x+y

x=int(input("ENTER FIRST NUMBER ="))
y=int(input("ENTER SECOND NUMBER="))
z=sum(x,y)
print("sum=",z)
```

*OUTPUT:*

Python Interpreter

```
*** Remote Interpreter Reinitialized ***
welcome
16
ENTER FIRST NUMBER =23
ENTER SECOND NUMBER=17
sum= 40
```

*Let us make a program to calculate simple interest using user defined function.*

*EXAMPLE:*

PyScripter - D:\PYTHON\simple_interest.py

```python
#SIMPLE INTEREST
def SI(p,r,t):
    return p*r*t/100

p=int(input("enter the principal amount="))
r=int(input("enter the rate of interest="))
t=int(input("enter the amount of time(IN YEARS)="))
si=SI(p,r,t)
print("SIMPLE INTEREST=",si)
```

*OUTPUT:*

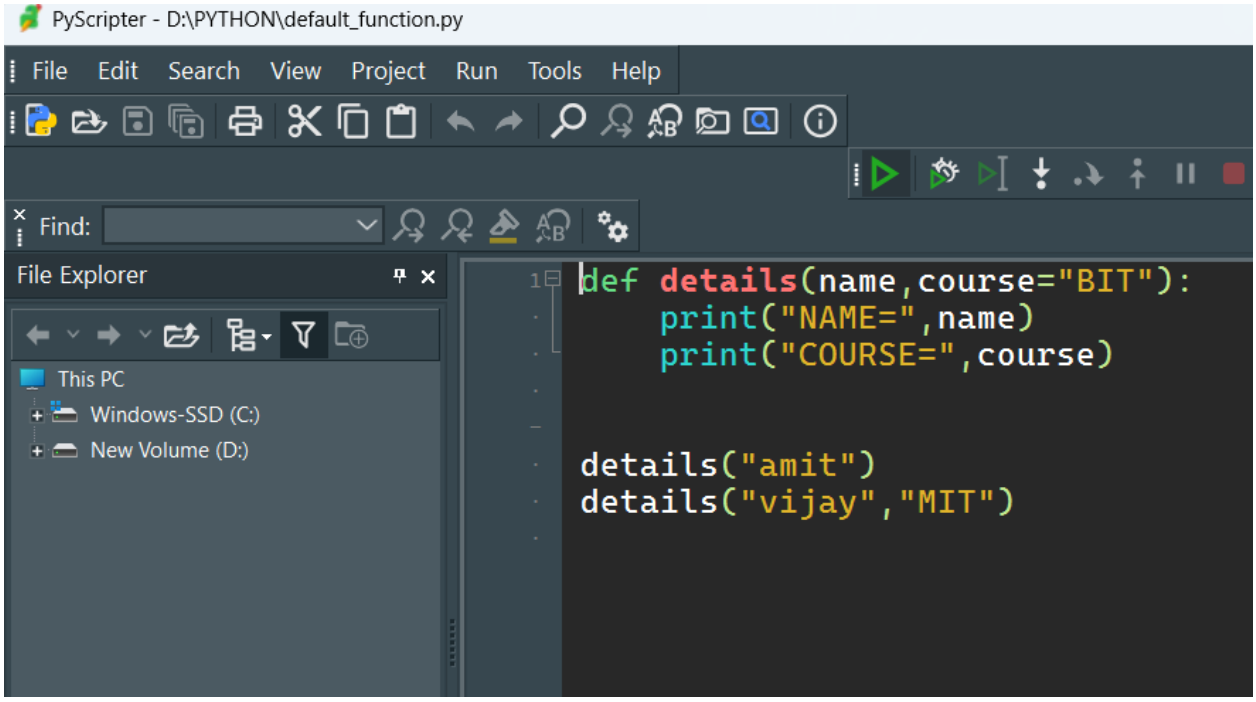
Python Interpreter

```
>>>
*** Remote Interpreter Reinitialized ***
enter the principal amount=200
enter the rate of interest=10
enter the amount of time=2
SIMPLE INTEREST= 40.0
>>>
```

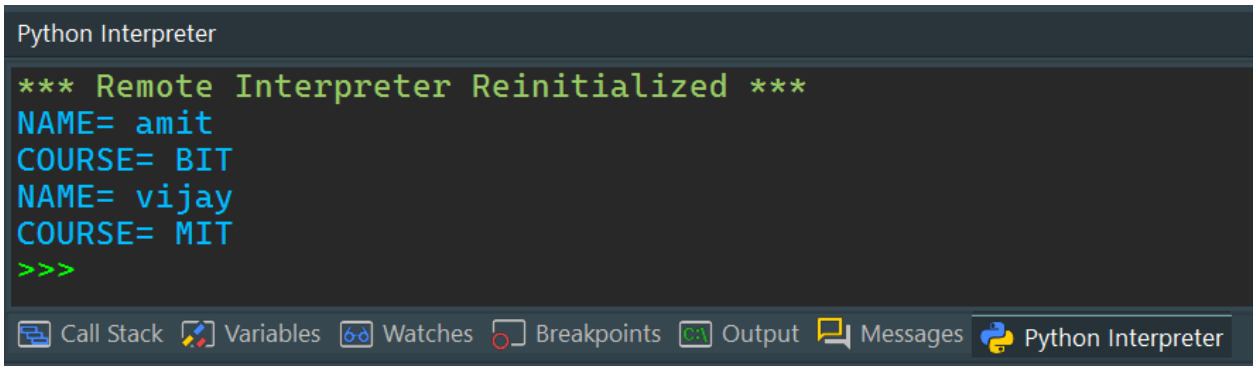
## *DEFAULT ARGUMENT FUNCTION IN PYTHON*

*Python allows function arguments to have default values. If the function is called without the argument, the argument gets its default value.*
*Python has a different way of representing syntax and default values for function arguments. Default values indicate that the function argument will take that value if no argument value is passed during the function call. The default value is assigned by using the assignment(=) operator of the form keyword name=value.*

*EXAMPLE:*




*OUTPUT:*




# *RECURSIVE FUNCTION IN PYTHON*

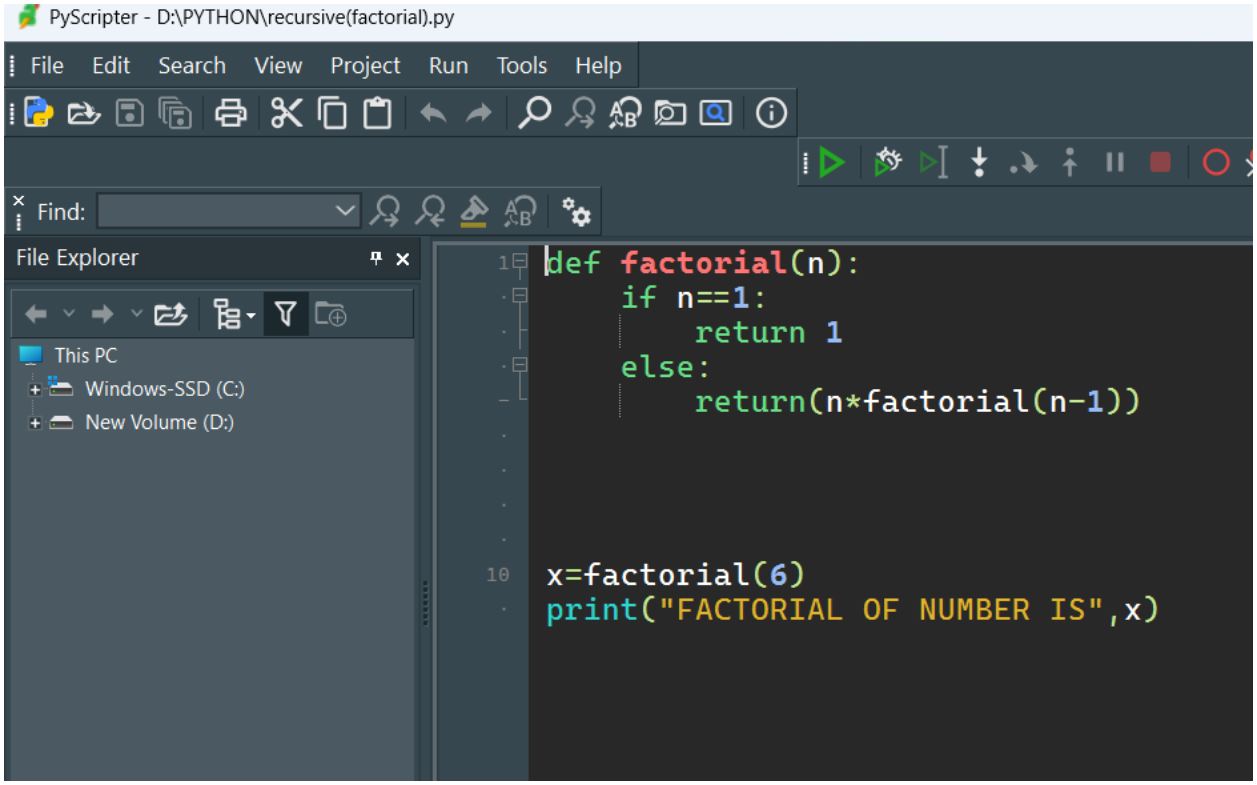
*The term Recursion can be defined as the process of defining something in terms of itself. In simple words, it is a process in which a function calls itself directly or indirectly.*

***Advantages of using recursion***

- *A complicated function can be split down into smaller sub-problems utilizing recursion.*
- *Sequence creation is simpler through recursion than utilizing any nested iteration.*
- *Recursive functions render the code look simple and effective.*

*Let us solve a problem of finding a factorial of a number using recursion.*

*EXAMPLE:*




*OUTPUT:*