**9.What is formal arguments in python?**

**Objective:**

* To see formal arguments in python.

**Process:**

* **Positional**

Positional arguments are the arguments passed to a function in correct positional order.

The number of arguments in the function call should match exactly with the function definition.

* **Keyword**

Keyword arguments are related to the function calls.

When you use keyword arguments in a function call, the caller identifies the arguments by the parameter name.

* **Default**

A default argument is an argument that assumes a default value if a value is not provided in the function call for that argument.

* **Variable length**

An asterisk (\*) is placed before the variable name that holds the values of all non keyword variable arguments.

**Input:**

* Sample code that executes all concepts of formal arguments in python3.

**Output:**

* All formal arguments examples.

**Source code:**

#Positional arguments

def city(str1,str2):

str3=str1+str2

return str3

print("\*\*\*\*\*\*\*\*\*\*\*\*\*Positional arguments output\n")

print(city("Tamil","Nadu"))

#keyword arguments

def employee(name="sathish",salary=5000):

return name,salary

print("\*\*\*\*\*\*\*\*\*\*\*\*\*keyword arguments output\n")

print(employee(salary=8000,name="kumar"))

#default arguments

def points(x=10,y=5,z=7):

return x,y,z

print("\*\*\*\*\*\*\*\*\*\*\*\*\*default arguments output\n")

print(points(10,2,3))

print(points(2,10,1))

print(points(2,3))

print(points(x=5,y=4))

#Variable length arguments

def varlength(\*names):

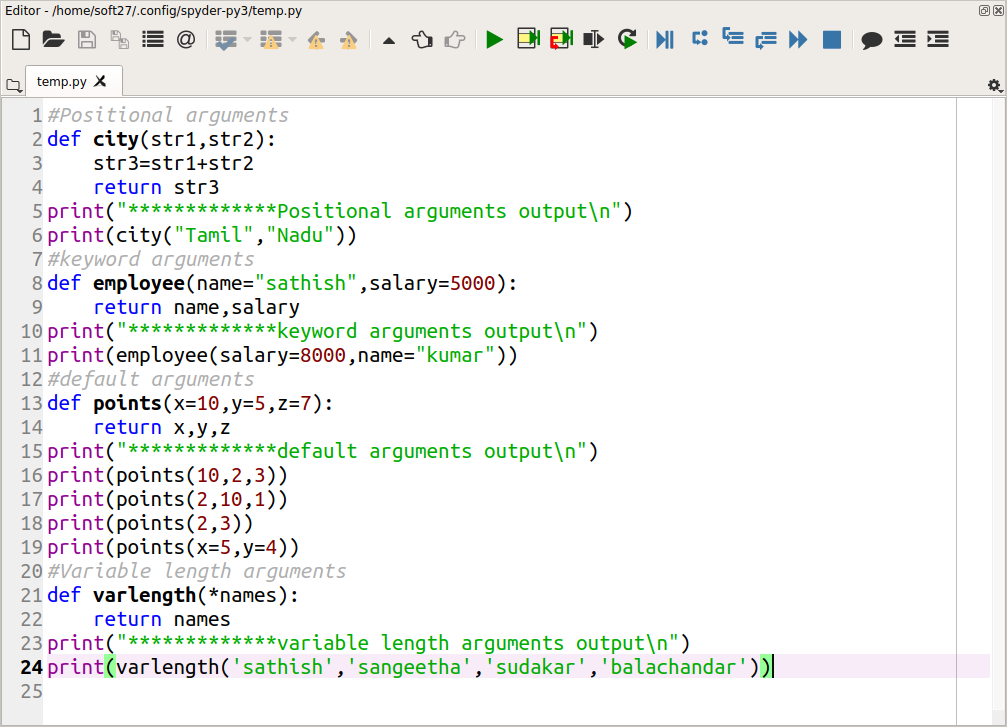
return names

print("\*\*\*\*\*\*\*\*\*\*\*\*\*variable length arguments output\n")

print(varlength('sathish','sangeetha','sudakar',

'balachandar'))

**Screen shot:**

****

****