Parameter Passing in Function Call   [1/5]

```python
#Parameter Passing  Case 1 Numeric Data

def test_function(t_num):
        print("         Now, inside test_function: ")
        print("         t_num    =", t_num,        "             id = ",id(t_num))

        t_num = 4

        print("\n        Still, Inside test_function after changing values: ")
        print("         t_num    =", t_num,        "             id = ",id(t_num))


def my_function():

        test_num = 10

        print("\n **** Before calling test_function: ")
        print("test_num    =", test_num,        "             id = ",id(test_num))

        print("\n **** Calling test_function *****")
        test_function(test_num)

        print("\n **** After calling test_function:   ")
        print("test_num    =", test_num,        "             id = ",id(test_num))


my_function()
```

```
 **** Before calling test_function:
test_num    = 10           id =  1635596192

 **** Calling test_function *****
        Now, inside test_function:
    t_num    = 10           id =  1635596192

        Still, Inside test_function after changing values:
    t_num    = 4           id =  1635596096

 **** After calling test_function:
test_num    = 10           id =  1635596192
>>> |
```

```python
#Parameter Passing  Case 2:  Passing String

def test_function(t_string):
        print("         Now, inside test_function: ")
        print("t_string =", t_string, "                id = ",id(t_string))

        t_string = "Japan"

        print("\n       Still, Inside test_function after changing values: ")
        print("t_string =", t_string, "                id = ",id(t_string))


def my_function():

        test_string = "Korea"

        print("\n **** Before calling test_function: ")
        print("test_string =", test_string, "                id = ",id(test_string))

        print("\n **** Calling test_function *****")
        test_function(test_string)

        print("\n **** After calling test_function:  ")
        print("test_string =", test_string, "                id = ",id(test_string))


my_function()
```

```
**** Before calling test_function:
test_string = Korea              id =  54036576

**** Calling test_function *****
        Now, inside test_function:
t_string = Korea                 id =  54036576

        Still, Inside test_function after changing values:
t_string = Japan                 id =  54036544

**** After calling test_function:
test_string = Korea              id =  54036576
>>>
```

Parameter Passing in Function Call   [3/5]

```python
#Parameter Passing  Case 3:  Passing Boolean

def test_function(t_bool):
        print("          Now, inside test_function: ")
        print("t_bool   =", t_bool, "                   id = ",id(t_bool))

        t_bool = False

        print("\n        Still, Inside test_function after changing values: ")
        print("t_bool   =", t_bool, "                   id = ",id(t_bool))


def my_function():

        test_bool = True

        print("\n **** Before calling test_function: ")
        print("test_bool   =", test_bool, "                   id = ",id(test_bool))

        print("\n **** Calling test_function *****")
        test_function(test_bool)

        print("\n **** After calling test_function:  ")
        print("test_bool   =", test_bool, "                   id = ",id(test_bool))


my_function()
```

```
 **** Before calling test_function:
test_bool   = True                    id =   1635414864

 **** Calling test_function *****
        Now, inside test_function:
t_bool    = True                    id =   1635414864

        Still, Inside test_function after changing values:
t_bool    = False                   id =   1635414880

 **** After calling test_function:
test_bool   = True                    id =   1635414864
>>>
```

```python
#Parameter Passing  Case 4:  Passing List

def test_function(t_list):
        print("         Now, inside test_function: ")
        print("    t_list   =", t_list,            " id = ", id(t_list))

        t_list.pop(3)

        print("\n      Still, Inside test_function after changing values: ")
        print("    t_list   =", t_list,            " id = ", id(t_list))


def my_function():

        test_list = [23, 12, 9, 7]

        print("\n **** Before calling test_function: ")
        print("test_list   =", test_list,           " id = ",id(test_list))

        print("\n **** Calling test_function *****")
        test_function(test_list)

        print("\n **** After calling test_function:   ")
        print("test_list   =", test_list,            " id = ",id(test_list))

my_function()
```

```
**** Before calling test_function:
test_list   = [23, 12, 9, 7]    id = 62083840

**** Calling test_function *****
        Now, inside test_function:
   t_list    = [23, 12, 9, 7]   id = 62083840

     Still, Inside test_function after changing values:
   t_list    = [23, 12, 9]   id =  62083840

**** After calling test_function:
test_list   = [23, 12, 9]        id =  62083840
>>>
```

```python
#Parameter Passing  Case 5:  Passing Object

class FooClass():
        def __init__(self):
            self.test_var = 0

def test_function(t_object):
        print("     Now, inside test_function: ")
        print("        t_object.test_var =", t_object.test_var, "    id = ",id(t_object.test_var))

        t_object.test_var = 10000

        print("\n    Still, Inside test_function after changing values: ")
        print("        t_object.test_var =", t_object.test_var, " id = ",id(t_object.test_var))


def my_function():

        test_object = FooClass()
        test_object.test_var = 100

        print("\n **** Before calling test_function: ")
        print("test_object.test_var =", test_object.test_var, "    id = ",id(test_object.test_var))

        print("\n **** Calling test_function *****")
        test_function(test_object)

        print("\n **** After calling test_function:  ")
        print("test_object.test_var =", test_object.test_var, "    id = ",id(test_objec

my_function()
```

```
**** Before calling test_function:
test_object.test_var = 100   id =  1635597632

**** Calling test_function *****
    Now, inside test_function:
    t_object.test_var = 100      id =  1635597632

    Still, Inside test_function after changing values:
    t_object.test_var = 10000  id =  51365328

**** After calling test_function:
test_object.test_var = 10000     id =  51365328
```