**Task 1: Decorators in Python**

1. Creating a Simple Decorator

1. Create a simple function called my\_function that prints "Function is called".
2. Create a decorator function called my\_decorator that prints "Decorator had modified my\_function output" and override printing of decorated function.

Solution:  
"""

Task 1 solution

"""

def my\_function\_decorator(fun):

    """

    Decorater to modify the output of my\_function()

    """

    def inner():

        print("Decorator has modified my function output")

        result = fun()

        return result

    return inner

@my\_function\_decorator

def my\_function():

    """

    My function that returns a string when called

    """

    return "Function is called"

RESULT = my\_function()

print(RESULT)

Output: