**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:

def my\_decorator(func):  
 def inner():  
 rs = func()  
 return "Decorator had modified my function output"  
 return inner  
  
@my\_decorator  
def my\_function():  
 return "Function is called"  
  
res = my\_function()  
print(res)

Result:

C:\Users\WRisaana\pythonProject\_createpkg\Scripts\python.exe C:\Users\WRisaana\PycharmProjects\pythonProject\_createpkg\Exercises\Ex1Decorators.py

Decorator had modified my function output

Process finished with exit code 0