**Build Expectation**

**Task**

Implement a Python decorator that should take whatever the decorated function returns, and writes it to a file in a new line.

For the sake of this problem, let us assume that the decorated functions always return a string.

**Constraints**

The decorator should be named ***log\_message***and should write to the file ***/tmp/decorator\_logs.txt***.

**Example usage:**

@log\_message

**def** **a\_function\_that\_returns\_a\_string**():

**return** "A string"

@log\_message

**def** **a\_function\_that\_returns\_a\_string\_with\_newline**(s):

**return** "{}\n".format(s)

@log\_message

**def** **a\_function\_that\_returns\_another\_string**(string=""):

**return** "Another string"

On running in python shell,

>>> a\_function\_that\_returns\_a\_string()

>>> a\_function\_that\_returns\_a\_string\_with\_newline("Newline String")

>>> a\_function\_that\_returns\_another\_string(string="Another String")

Should write the string to the file.

$ cat /tmp/decorator\_logs.txt

A string

Newline string

Another string