**Assignment 6**

1. Write a Python program to create a decorator that logs the arguments and return value of a function.

**2.** Write a Python program to create a decorator function to measure the execution time of a function.

**3.** Write a Python program to create a decorator to convert the return value of a function to a specified data type.

**4.** Write a Python program that implements a decorator to cache the result of a function.

**5.** Write a Python program that implements a decorator to validate function arguments based on a given condition.

**6.** Write a Python program that implements a decorator to retry a function multiple times in case of failure.

**7.** Write a Python program that implements a decorator to enforce rate limits on a function.

**8.** Write a Python program that implements a decorator to add logging functionality to a function.

**9.** Write a Python program that implements a decorator to handle exceptions raised by a function and provide a default response.

**10.** Write a Python program that implements a decorator to enforce type checking on the arguments of a function.

**11.** Write a Python program that implements a decorator to measure the memory usage of a function.

**12.** Write a Python program that implements a decorator to provide caching with expiration time for a function.