Python Multithreading & Multiprocessing – 10 Practice Questions

Multithreading

1. Write a Python program to create **two threads**, where one thread prints numbers from 1 to 10 and the other prints numbers from 11 to 20.
2. Create a multithreaded program in which **one thread prints even numbers** and another thread **prints odd numbers** up to 50.
3. Write a program using **multithreading** to calculate the **square and cube** of a given number simultaneously.
4. Create multiple threads that **increment a shared variable**. Use a **Lock** to prevent race conditions.
5. Write a multithreaded program that simulates **downloading multiple files** at the same time.

Multiprocessing

6. Write a Python program using **multiprocessing** to create **two processes** that print their **process IDs**.
7. Create a multiprocessing program where **one process calculates the sum** of numbers from 1 to 100 and **another process calculates the factorial** of a given number.
8. Write a program using multiprocessing to **find prime numbers** in a given range.
9. Create a multiprocessing program that demonstrates **inter-process communication** using a Queue.
10. Write a program to compare the **execution time** of a task using **multithreading** and **multiprocessing**.

Thread Pool

1. Write a Python program using **ThreadPoolExecutor** to **download multiple web pages** concurrently.
2. Create a program using a **thread pool** that executes multiple I/O-bound tasks and prints the thread name for each task.

Process Pool

1. Write a Python program using **ProcessPoolExecutor** to calculate the **factorial of multiple numbers** in parallel.
2. Create a program using a **process pool** to perform a **CPU-intensive task** and display the process ID for each task.

Web Scraping (Using Threads / Pools)

1. Write a Python program that performs **web scraping** on multiple URLs **concurrently** using:

- Multithreading **or**
- ThreadPoolExecutor
  Extract and display the **page titles** from each website.
