**What is thread starvation?**

 Thread starvation occurs when a thread is perpetually denied access to resources it needs to proceed with its execution. This can happen when other threads are constantly given priority, preventing the starved thread from making progress.

**Docs Oracle description:**

***Starvation***describes a situation where a thread is unable to gain regular access to shared resources and is unable to make progress. This happens when shared resources are made unavailable for long periods by **"greedy"** threads.

**It could happen in the following situations:**

* **Priority Scheduling**: If a system uses priority-based scheduling, lower-priority threads may be starved if higher-priority threads continuously occupy the CPU.
* **Resource Contention**: If a thread is waiting for a resource that is constantly being used by other threads, it may never get a chance to acquire it.

**What is DeadLock?**

Deadlock is a situation where two or more threads are blocked forever, each waiting for the other to release a resource. This creates a cycle of dependencies that prevents any of the threads from proceeding.

**It could happen when:**

* **Circular Wait:** Threads form a circular chain where each thread holds a resource that the next thread in the chain needs.
* **Hold and Wait:** Threads hold onto resources while waiting for other resources to become available.

Deadlock results in a complete halt of the affected threads, leading to system unresponsiveness and potential application failure.

**What is LiveLock?**

 Livelock is a situation where two or more threads continuously change their state in response to each other without making any progress. Unlike deadlock, where threads are blocked, livelock threads are active but unable to complete their tasks.

**Consequences**: Livelock can lead to high CPU usage and unresponsiveness, as threads are busy but not productive.