1. Implement a TCP based server program in Java to provide a resource to two clients using the concept of Mutual Exclusion.

[Note: The problem which mutual exclusion addresses is a problem of resource sharing: how can a software system control multiple processes' access to a shared resource, when each process needs exclusive control of that resource while doing its work? The mutual-exclusion solution to this problem makes the shared resource available only while the process is in a specific code segment called the critical section. It controls access to the shared resource by controlling each process' execution of that part of its program where the resource would be used.]

There has to be a program called Critical Section. When one client is accessing that if there is a request from the second client the server program should deny it, and allow the access only after the first releases it.

1. Implement a TCP based server program in Java to deal the issue of a client performing the two transactions simultaneously using serializability.

[Note: Serializability is the major correctness criterion for concurrent transactions' executions.]

Using server program, handle the client performing two transactions to ensure serializability.