**Implement a 2PC for distributed transaction management.**

**Description:**

A two-phase commit is a standardized protocol that ensures that a database commit is implementing in the situation where a commit operation must be broken into two separate parts.

In database management, saving data changes is known as a commit and undoing changes is known as a rollback. Both can be achieved easily using transaction logging when a single server is involved, but when the data is spread across geographically-diverse servers in distributed computing (i.e., each server being an independent entity with separate log records), the process can become more tricky.

A special object, known as a coordinator, is required in a distributed transaction. As its name implies, the coordinator arranges activities and synchronization between distributed servers. The two-phase commit is implemented as follows:

Phase 1 - Each server that needs to commit data writes its data records to the log. If a server is unsuccessful, it responds with a failure message. If successful, the server replies with an OK message.

Phase 2 - This phase begins after all participants respond OK. Then, the coordinator sends a signal to each server with commit instructions. After committing, each writes the commit as part of its log record for reference and sends the coordinator a message that its commit has been successfully implemented. If a server fails, the coordinator sends instructions to all servers to roll back the transaction. After the servers roll back, each sends feedback that this has been completed.

Step 1: Open command prompt and run DBConnector code (for Database Connectivity)

Step 2: Open another command prompt and run TestDB code (To run the database and Table content)

Step 3: Open another command prompt and run Server code (Server will enter waiting state)

Step 4: Open another command prompt and run Client code (Client gets connected to server and server state changes)

Step 5: Open another command prompt and run Client code again (Second Client gets connected now)

Step 6: Now click on prepared in both clients

Step 7: Enter SQL statement and click on Execute in one of the client (or both clients with different data)

Step 8: Run TestDB again to see the updates in the table.

Sample Output:

