Experiment 10

To demonstrate the use of **COMMIT, SAVEPOINT, and ROLLBACK** commands on a sample table in Oracle 12c.

1. **Table Creation**
   - Create a table named `bank` with the following structure:
     - `bankname` → `VARCHAR2(50)`
     - `headoffice` → `VARCHAR2(50)`
     - `branch` → `VARCHAR2(50)`
     - `branchcode` → `VARCHAR2(10)` (Primary Key)

2. **Insert Records (before Savepoint)**
   - Insert the following rows into the table:

bankname	headoffice	branch	branchcode
SBI	Mumbai	Kochi	B001
SBI	Mumbai	Chennai	B002
HDFC	Mumbai	Bengaluru	B003

3. **Insert More Records (after Savepoint)**
   - Insert the row:
   - `ICICI, Mumbai, Delhi, B004`
   - Create another savepoint named `sp2`.

4. **Perform Updates and Deletes**
   - Update the branch of bankcode `B004` from *Delhi* to *Hyderabad*.
   - Delete the record with branchcode `B003`.
   - Display all rows with `SELECT * FROM bank;`

5. **Rollback to Savepoints**
   - Rollback to `sp2` and display the table.
     - Observe whether B003 and B004 changes are restored.
   - Rollback further to `sp1` and display the table.
     - Observe what happens to the B004 record.

6. **Commit the Transaction**
    - Commit the changes after rolling back to `sp1`.
    - Try to perform a `ROLLBACK` again.
    - Check whether the committed state remains unchanged.

Expected Outputs to Verify

- After step 4: B003 deleted, B004 updated to Hyderabad.
- After rollback to sp2: 4 rows, B004 back to Delhi.
- After rollback to sp1: only 3 rows remain (no B004).
- After commit: rollback has no effect.