5.

1) COMMIT;

2) DELETE FROM EMP WHERE EMP\_ID = 101;

ROLLBACK;

3) SAVEPOINT BeforeSalaryUpdate;

4)  **Atomicity**:

* This property ensures that all operations within a transaction are completed successfully. If any part of the transaction fails, the entire transaction is rolled back, and the database remains unchanged.
* In simple terms, "all or nothing."

 **Consistency**:

* A transaction must bring the database from one valid state to another. After a transaction, all data must be in a consistent state, adhering to all defined rules, constraints, and relationships.
* For example, if a rule specifies that an employee's salary cannot be negative, the database should not allow such a transaction.

 **Isolation**:

* Transactions should be isolated from each other, meaning the intermediate state of a transaction is invisible to other transactions. Even if multiple transactions occur simultaneously, each should appear as though it is the only transaction being processed.
* This prevents data inconsistencies caused by concurrent access.

 **Durability**:

* Once a transaction is committed, it becomes permanent. Even if there is a system failure after the commit, the changes made by the transaction are preserved and cannot be lost.
* This ensures the reliability of the database in the face of power failures or crashes.