SQL Notes – Transactions, DCL & Constraints

1. Transactions in SQL

A transaction is a sequence of SQL operations executed as a single logical unit. Transactions ensure data integrity using COMMIT and ROLLBACK.

Key Commands:

- **START TRANSACTION** – Begins a transaction
- **INSERT** – Inserts data temporarily
- **COMMIT** – Saves changes permanently
- **ROLLBACK** – Undoes uncommitted changes

Example:

```
START TRANSACTION;
INSERT INTO student (stud_name, standard, age) VALUES ('Tejas', 8, 13);
ROLLBACK;
```

2. SAVEPOINT

SAVEPOINT allows partial rollback inside a transaction. It is optional and names are not case-sensitive.

Example:

```
START TRANSACTION;
INSERT INTO student VALUES (...);
SAVEPOINT sp1;
INSERT INTO student VALUES (...);
ROLLBACK TO sp1;
COMMIT;
```

3. DCL (Data Control Language)

DCL commands control access and permissions in the database.

- **GRANT** – Gives permissions to users
- **REVOKE** – Removes permissions from users

Example:

```
GRANT SELECT, INSERT ON student TO tejas;
REVOKE INSERT ON student FROM tejas;
```

4. SQL Constraints

Constraints enforce rules on table data to maintain integrity.

- **PRIMARY KEY** – Unique and not NULL
- **FOREIGN KEY** – Enforces referential integrity
- **UNIQUE** – Prevents duplicate values
- **NOT NULL** – Disallows NULL values
- **CHECK** – Applies conditions
- **DEFAULT** – Sets default values

ON DELETE CASCADE:

Automatically deletes child records when the parent record is deleted.

```
FOREIGN KEY (student_id) REFERENCES student(id) ON DELETE CASCADE;
```