**1. Difference between NULL and 0?**

| **NULL** | **0** |
| --- | --- |
| Represents **unknown/missing** value | Numeric value (zero) |
| Not equal to anything (even itself) | Defined and comparable |
| Can't use arithmetic directly (e.g., NULL + 1 = NULL) | Can do math (0 + 1 = 1) |

**Example:**

SELECT \* FROM employees WHERE salary = NULL; -- Won’t work

SELECT \* FROM employees WHERE salary = 0; -- Finds exact 0 values

**2. What is a DEFAULT constraint?**

The DEFAULT constraint sets a **predefined value** when no value is given during INSERT.

**Syntax:**

CREATE TABLE employees (

id INT,

status VARCHAR(20) DEFAULT 'Active'

);

**Usage:**

INSERT INTO employees (id) VALUES (1);

-- status will be 'Active' automatically

**3. How does IS NULL work?**

You **can’t use = to check NULL**. Instead, use IS NULL or IS NOT NULL.

**Example:**

SELECT \* FROM books WHERE author\_id IS NULL;

**4. How do you update multiple rows?**

Use UPDATE with a WHERE clause that matches multiple rows.

**Example:**

UPDATE employees

SET status = 'Inactive'

WHERE department = 'HR';

All employees in HR will be updated.

**5. Can we insert partial values?**

Yes, if:

* **Other columns allow NULL or have DEFAULT values**
* You **specify columns** explicitly

**Example:**

INSERT INTO employees (id, name) VALUES (1, 'Tarun');

Columns not mentioned will be NULL or defaulted.

**6. What happens if a NOT NULL field is left empty?**

An error occurs:

INSERT INTO students (id, name) VALUES (1, NULL); -- if name is NOT NULL

**7. How do you rollback a deletion?**

Only possible **if not committed** yet.

**Example:**

DELETE FROM books WHERE book\_id = 10;

ROLLBACK; -- Will undo the delete if not committed yet

After COMMIT, rollback won’t work.

**8. Can we insert values into specific columns only?**

Yes. Specify the column names.

**Example:**

INSERT INTO books (title, author\_id) VALUES ('SQL Mastery', 3);

**9. How to insert values using SELECT?**

Useful to copy data from one table to another.

**Syntax:**

INSERT INTO archive\_books (book\_id, title)

SELECT book\_id, title FROM books

WHERE published\_year < 2000;

**10. What is ON DELETE CASCADE?**

Used in **foreign key constraints** to auto-delete child rows if the parent is deleted.

**Example:**

CREATE TABLE orders (

order\_id INT PRIMARY KEY,

customer\_id INT,

FOREIGN KEY (customer\_id) REFERENCES customers(customer\_id) ON DELETE CASCADE

);

If a customer is deleted, all their orders are also deleted.

**11.** **What is ON DELETE SET NULL?**

ON DELETE SET NULL is a **referential action** used in a **foreign key constraint**.  
It **automatically sets the foreign key column to NULL** in the child table when the **referenced row** in the parent table is **deleted**.

**Example:**

CREATE TABLE employees (

emp\_id INT PRIMARY KEY,

manager\_id INT,

FOREIGN KEY (manager\_id) REFERENCES employees(emp\_id)

ON DELETE SET NULL

);

If a manager is deleted from the employees table (parent)

Then all employees reporting to that manager will have manager\_id = NULL.