Lecture Notes – Delete Table

Section 1: Lecture Summary

The lecture covers the **DELETE** clause as a DML command for removing specific rows from a table using a **WHERE** condition. It demonstrates deleting single rows by primary key and multiple rows based on a salary threshold, contrasting **DELETE** (for selective removal) with **TRUNCATE** (DDL for all rows).

Section 2: Key Concepts and Explanations

DELETE syntax: `DELETE FROM table_name WHERE condition;`. The **WHERE** clause is essential to specify rows for deletion; omitting it deletes all rows. Examples include deleting by **EmpID** (e.g., employee 5) or by **Salary** (e.g., less than 60000).
DELETE targets specific rows, while **TRUNCATE** removes all data from the table as a DDL operation.

Section 3: Example Code and Use Cases

Using **companyDB.Employees** table:

Delete single employee by **EmpID**:

```
DELETE FROM Employees WHERE EmpID = 5;
```

Delete multiple employees by **Salary**:

```
DELETE FROM Employees WHERE Salary < 60000;
```

These queries remove rows matching the **WHERE** condition from **Employees**, leaving higher-salary records (e.g., employees 2 and 4 if their **Salary** exceeds 60000).

Section 4: Key Takeaways

- Always use **WHERE** with **DELETE** to avoid removing all rows.

- **DELETE** is DML for conditional row removal; **TRUNCATE** is DDL for full table clearance.

- Practice on a test table to avoid altering production data like **companyDB**.