

Lecture Notes – Truncate_Drop Table

Section 1: Lecture Summary

The lecture covers the **TRUNCATE** and **DROP** commands in DDL. **TRUNCATE** clears all data from an existing table without deleting the table structure. **DROP** removes specific elements like columns or entire tables from the database.

Section 2: Key Concepts and Explanations

TRUNCATE is a direct command used to remove all rows from a table, leaving the table empty but intact. It does not require **ALTER** and is applied as **TRUNCATE TABLE table_name**. This is useful for quickly clearing data but must be used cautiously to avoid data loss.

DROP is versatile and requires **ALTER TABLE** for dropping columns, such as **ALTER TABLE table_name DROP COLUMN column_name**. To drop an entire table, use **DROP TABLE table_name**, which completely removes the table from the database. **DROP** can also target views or indexes. These operations are permanent and part of DDL, used sparingly after initial table setup.

Section 3: Example Code and Use Cases

Using the **companyDB** schema:

TRUNCATE example to clear all data from **Employees** table:

```
TRUNCATE TABLE Employees;
```

DROP column example to remove **JobTitle** from **Employees**:

```
ALTER TABLE Employees DROP COLUMN JobTitle;
```

DROP table example to delete **Employees** entirely:

```
DROP TABLE Employees;
```

Section 4: Key Takeaways

TRUNCATE empties table data; DROP removes columns or tables. Use DDL commands like these once during setup and handle carefully to prevent data loss or impacts on views and programs. Practice on new databases, not production ones.