

Lecture Notes – Alter Table

Section 1: Lecture Summary

This lecture covers the **ALTER TABLE** command, a DDL (Data Definition Language) statement used to modify existing table structures. The focus is on adding new columns to tables and modifying column properties such as data types, sizes, and constraints. The lecture demonstrates practical examples using the `employees_two` table, showing how to add a `job_title` column and subsequently modify it to include a `NOT NULL` constraint.

Section 2: Key Concepts and Explanations

ALTER TABLE Command

The `ALTER TABLE` statement allows you to modify table definitions after creation. This is useful when business requirements change or when you need to refine your database schema.

Adding Columns

When adding a column, you must specify the column name, data type, and size. The basic syntax is:

```
ALTER TABLE table_name ADD COLUMN column_name data_type(size);
```

When a new column is added without constraints, it allows `NULL` values by default, meaning existing rows will have `NULL` in that column.

Modifying Columns

The MODIFY keyword allows you to change an existing column's properties, including its data type, size, or constraints. When modifying a column, you must provide the complete definition including data type, size, and any constraints.

The syntax is:

```
ALTER TABLE table_name MODIFY column_name data_type(size) constraint;
```

****NOT NULL Constraint****

Adding a NOT NULL constraint makes a column mandatory. This ensures that every record must have a value in that column. In organizational contexts, this is critical for fields like job titles, where every employee must have a designated role.

Section 3: Example Code and Use Cases

Using the companyDB schema, here are practical examples:

****Adding a New Column****

```
ALTER TABLE Employees ADD COLUMN JobTitle VARCHAR(50);
```

This adds a JobTitle column to the Employees table with a maximum length of 50 characters. Since no constraint is specified, NULL values are permitted.

****Modifying a Column with Constraints****

```
ALTER TABLE Employees MODIFY JobTitle VARCHAR(100) NOT NULL;
```

This modifies the existing JobTitle column by increasing the size to 100 characters and adding a NOT NULL constraint, ensuring every employee record must have a job title assigned.

****Practical Scenario****

If you need to track additional employee information:

```
ALTER TABLE Employees ADD COLUMN HireDate DATE;
```

Then modify it later:

```
ALTER TABLE Employees MODIFY HireDate DATE NOT NULL;
```

This ensures that all employee records capture when they were hired.

Section 4: Key Takeaways

When adding columns, always specify the data type and size to define the column structure properly. When modifying columns, provide the complete column definition including data type, size, and any constraints you want to enforce. Use NOT NULL constraints for mandatory fields that should never be empty. The ALTER TABLE command is essential for maintaining flexible database schemas that can adapt to changing business requirements without losing existing data.