Lecture Notes – NaturalJoin

Section 1: Lecture Summary

Natural join retrieves data from multiple tables by combining them based on common columns. It joins tables like Employees and Departments using **DeptID**, keeping only one copy of the common column in the result, and can extend to multiple tables like Projects.

Section 2: Key Concepts and Explanations

Natural join automatically joins tables on all common columns, such as **DeptID** in Employees and Departments, or **DeptID** between Departments and Projects. It combines all columns from both tables but eliminates duplicate common columns, producing a single table with matched rows only. Table aliases like **e** for Employees and **d** for Departments simplify queries when specifying columns, especially for common ones like **DeptID**; unique columns like **Location** do not need aliases. Joins reconstruct decomposed tables from normalization, avoiding redundancy while retrieving complete data.

Section 3: Example Code and Use Cases

```
SELECT * FROM Employees NATURAL JOIN Departments;
```

Combines Employees and Departments on **DeptID**, returning **EmpID**, **FirstName**, **LastName**, **JobTitle**, **HireDate**, **Salary**, **DeptName**, **Location** (one **DeptID** copy).

```
SELECT * FROM Employees NATURAL JOIN Departments NATURAL JOIN Projects;
```

Joins Employees, Departments (on **DeptID**), and Projects (on **DeptID**), including **ProjectID** and **ProjectName** for employees' projects.

```
SELECT e.EmpID, e.FirstName, d.DeptName, d.Location
FROM Employees e NATURAL JOIN Departments d;
```

Uses aliases to select specific columns after natural join on **DeptID**.

Section 4: Key Takeaways

Natural join uses common columns like **DeptID** to merge tables without duplicates. Aliases clarify column sources in multi-table queries. Essential for reconstructing normalized data from companyDB tables.