**East West University**

**Department of Computer Science and Engineering**

**CSE 302: LAB 05 (Handout)**

**Course Instructor: Mahmuda Rawnak Jahan**

## Introducing Nested Subqueries and Outer Joins in SQL

1. **Course (course\_id, title, dept\_name, credits)**
2. **Section (course\_id, sec\_id, semester, year, building, room\_number, time\_slot\_id)**
3. **Teaches (ID, course\_id, sec\_id, semester, year)**
4. **Instructor (ID, name, dept\_name, salary)**
5. **Student (ID, name, dept\_name, tot\_cred)**
6. **Takes (ID, course\_id, sec\_id, semester, year, grade)**
7. **Department (dept\_name, building, budget)**

Subqueries in the WHERE clause

**A. IN / NOT IN**

* **Find courses offered in Fall 2009 and in Spring 2010**

SELECT DISTINCT course\_id

FROM Section

WHERE semester = 'Fall' AND year = 2009

AND course\_id IN ( SELECT course\_id

FROM Section

WHERE semester = 'Spring' AND year = 2010

);

* **Find courses offered in Fall 2009 but not in Spring 2010**

SELECT DISTINCT course\_id

FROM Section WHERE semester = 'Fall' AND year = 2009

AND course\_id NOT IN (

SELECT course\_id

FROM Section

WHERE semester = 'Spring' AND year = 2010

);

* **Find the total number of (distinct) students who have taken course sections taught by the instructor with ID 10101**

SELECT COUNT(DISTINCT student\_id) AS total\_students

FROM takes

WHERE course\_id IN ( SELECT course\_id FROM section

WHERE instructor\_id = 10101 );

**B. SOME / ALL**

* **Find names of instructors with salaries greater than that of some instructor in the Biology department**

SELECT name FROM instructor WHERE salary > SOME ( SELECT salary FROM instructor WHERE dept\_name = 'Biology');

* **Find the names of all instructors whose salary is greater than the salary of all instructors in the Biology department**

SELECT name FROM instructor WHERE salary > ALL ( SELECT salary FROM instructor WHERE dept\_name = 'Biology' );

**C. EXISTS/NOT EXISTS**

* **Find all courses taught in both the Fall 2009 semester and in the Spring 2010 semester**

SELECT DISTINCT course\_id FROM section s1

WHERE semester = 'Fall' AND year = 2009 AND

EXISTS ( SELECT 1 FROM section s2 WHERE s1.course\_id = s2.course\_id AND semester = 'Spring' AND year = 2010 );

* **Find all courses taught in Fall 2009 semester but not in the Spring 2010 semester**  
  SELECT DISTINCT course\_id FROM section s1

WHERE semester = 'Fall' AND year = 2009 AND NOT EXISTS ( SELECT 1 FROM section s2 WHERE s1.course\_id = s2.course\_id AND semester = 'Spring' AND year = 2010 );

* **Find all students who have taken all courses offered in the Biology department**  
  SELECT ID FROM Student s

WHERE NOT EXISTS ( SELECT course\_id FROM Course

WHERE dept\_name = 'Biology'

AND course\_id NOT IN ( SELECT course\_id FROM Takes

WHERE ID = s.ID

)

);

**➔ Subqueries in the FROM clause**

* **Find the average instructors’ salaries of those departments where the average salary is greater than $42,000**

SELECT dept\_name, AVG(salary) AS avg\_salary

FROM Instructor

GROUP BY dept\_name

HAVING AVG(salary) > 42000;

**➔ Complex Queries using WITH clause**

* **Find all departments with the maximum budget**

WITH DeptBudget AS (

SELECT dept\_name, MAX(budget) AS max\_budget

FROM Department )

SELECT dept\_name

FROM DeptBudget

WHERE budget = max\_budget;

* **Find all departments where the total salary is greater than the average of the total salary at all departments**  
  WITH DeptSalary AS (

SELECT dept\_name, SUM(salary) AS total\_salary

FROM Instructor

GROUP BY dept\_name

)

SELECT dept\_name

FROM DeptSalary

WHERE total\_salary > (SELECT AVG(total\_salary) FROM DeptSalary);

**➔ Subqueries in the SELECT clause (Scalar Subquery)**

* **Find number of instructors for each department**

SELECT dept\_name,

(SELECT COUNT(\*)

FROM Instructor i

WHERE i.dept\_name = d.dept\_name) AS num\_instructors

FROM Department d;

**➔ Performing Outer Joins**

* **Left Outer Join Example**  
  SELECT d.dept\_name, i.name

FROM Department d

LEFT OUTER JOIN Instructor i ON d.dept\_name = i.dept\_name;

* **Right Outer Join Example**  
    
  SELECT i.name, d.dept\_name

FROM Instructor i

RIGHT OUTER JOIN Department d ON i.dept\_name = d.dept\_name;

* **Full Outer Join Example**  
    
  SELECT d.dept\_name, i.name

FROM Department d

FULL OUTER JOIN Instructor i ON d.dept\_name = i.dept\_name;

* **Find the number of instructors for each department, including departments with no instructor**SELECT d.dept\_name, COUNT(i.ID) AS num\_instructors

FROM Department d

LEFT OUTER JOIN Instructor i ON d.dept\_name = i.dept\_name

GROUP BY d.dept\_name;