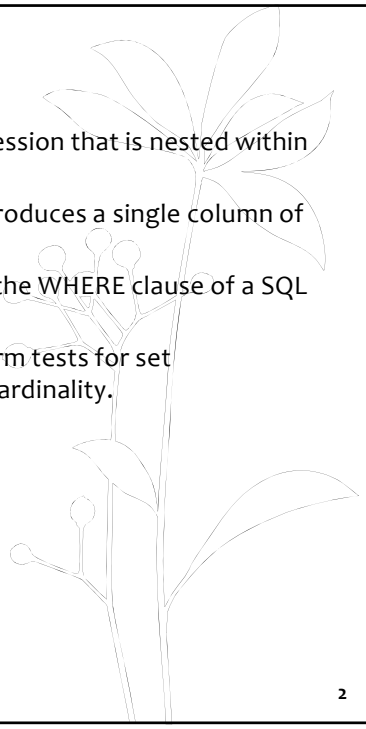
CSCD 327: RELATIONAL DATABASE SYSTEMS

SUBQUERIES

Instructor: Dr. Dan Li

Nested Subqueries

- A **subquery** is a **select-from-where** expression that is nested within another query.
- In the most common uses, a subquery produces a single column of data as its query results.
- Subqueries are most frequently used in the WHERE clause of a SQL statement.
- A common use of subqueries is to perform tests for set membership, set comparisons, and set cardinality.



2

Set Membership

- Find courses offered in Fall 2009 and in Spring 2010

This provides a way to implement INTERSECT.

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

This provides a way to implement EXCEPT.

3

Example Query

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

This provides a way to avoid using JOIN.

4

Set Comparison (=, <>, <, <=, >, >=)

- Find names of instructors with salary greater than that of some (at least one) instructor in the Biology department.

- Same query using **> some** clause

By default, the comparison using > means greater than some, so the keyword **some** can be omitted.
Any is the same as **some**.

5

Definition of Some Clause

- F <comp> **some** $r \Leftrightarrow \exists\ t \in r$ such that (F <comp> t)
  Where <comp> can be: $<,\ \leq,\ >,\ =,\ \neq$

(5 < **some** | 0 / 5 / 6) = true (read: 5 < some tuple in the relation)

(5 < **some** | 0 / 5) = false

(5 = **some** | 0 / 5) = true

(5 ≠ **some** | 0 / 5) = true (since 0 ≠ 5)

(= **some**) is the same as **in**
However, (≠ **some**) is not the same as **not in**

6

Example Query

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

7

Definition of all Clause

- F <comp> **all** $r \Leftrightarrow \forall\ t \in r$ (F <comp> t)

(5 < **all** {0, 5, 6}) = false

(5 < **all** {6, 10}) = true

(5 = **all** {4, 5}) = false

(5 ≠ **all** {4, 6}) = true (since 5 ≠ 4 and 5 ≠ 6)

(≠ **all**) is the same as **not in**
However, (= **all**) is not the same as **in**

8

Existence Test

- The **exists** construct returns the value **true** if the argument subquery is nonempty.
- **exists** $r \Leftrightarrow r \neq \emptyset$
- **not exists** $r \Leftrightarrow r = \emptyset$

9

Correlation Variables

- Yet another way of specifying the query "Find all courses taught in both the Fall 2009 semester and in the Spring 2010 semester"

This provides another way to implement INTERSECT.

- **Correlated subquery**
- **Correlation name** or **correlation variable, *course_id* in this example.**

10

Not Exists

- Find all students who have taken all courses offered in the Biology department.

How to implement division in SQL?

- Note that $X - Y = \emptyset \Leftrightarrow X \subseteq Y$
- *Note:* Cannot write this query using = **all** and its variants

11

Subqueries in the FROM Clause

- SQL allows a subquery expression to be used in the **from** clause
- Find the average instructors' salaries of those departments where the average salary is greater than $42,000.

dept_name	avg_salary
Biology	72000.000000
Comp. Sci.	77333.333333
Elec. Eng.	80000.000000
Finance	85000.000000
History	61000.000000
Physics	91000.000000

- Note that we do not need to use the **having** clause in the above solution

12

Subqueries in the FROM Clause (Cont.)

- And yet another way to write it: **lateral** clause

```
select name, salary, avg_salary
from instructor I1,
        lateral (select avg(salary) as avg_salary
                 from instructor I2
                 where I2.dept_name= I1.dept_name);
```

- Lateral clause permits later part of the **from** clause (after the lateral keyword) to access correlation variables from the earlier part.
- Note: lateral is part of the SQL standard, but is not supported on many database systems (e.g. MySQL); some databases such as SQL Server offer alternative syntax

13

Subqueries in the HAVING Clause

- *List the salespeople whose average order amount for products manufactured by ACI is at least as big as that salesperson's overall average order size.*

14

With Clause

- The **with** clause provides a way of defining a temporary table whose definition is available only to the query in which the **with** clause occurs.
- Find all departments with the maximum budget

Temporary tables are just like regular tables, except they exist only for the current session, and are dropped when the session ends.

NOTE: **MySQL doesn't support WITH clause. Use "Create Temporary Table" instead.**

15

Scalar Subquery

- Scalar subquery is one which is used where a single value is expected
- Find the budget and the total salary for each department

Scalar subquery in SELECT clause

- Runtime error if subquery returns more than one result tuple

16