Week 10
Quiz Chapter 9

Group 26:
John O’Leary, Tom Postler
1. Which type of join should be used to include no-matching tuples from both tables in the result?

a. Inner join
b. Left outer join
c. Right outer join
d. Full outer join
1. Which type of join should be used to include no-matching tuples from both tables in the result?

d. Full outer join

( next page for diagrams )
A) Inner join returns only matching tuples.

B/C) Left/Right outer join returns matching tuples and no-matching tuples from the left/right table, respectively.

D) Full outer join returns everything.
2. Which clauses the SELECT statement can have nested queries:

a. FROM
b. WHERE
c. HAVING
d. All of the above
2. Which clauses the SELECT statement can have nested queries:

d. All of the above

According to p. 325 in the text, nested queries typically appear in the WHERE or HAVING clauses, but are also allowed in the FROM clause.
3. Which relational operation is very difficult to represent in SQL without nested queries:

a. Outer Join
b. Cartesian product
c. Division
d. Set difference
3. Which relational operation is very difficult to represent in SQL without nested queries:

**c. Division**

- Because Division compares *all* or *every*, instead of *any* (for instance, list students in *every* club), we almost always need a nested query to find out how many elements make up *all* elements.
- Continuing the 'students in every club' example:
  - `SELECT StdNo`
  - `FROM StdClub`
  - `GROUP BY StdNo`
  - `HAVING COUNT(*) = ( SELECT COUNT(*) FROM Club)`
4. Type II nested queries are not used for join problems because

a. Repeated execution makes those inefficient for this purpose

b. They can not represent join operation

c. Type I queries are adequate to represent join problems

d. All of the above
4. Type II nested queries are not used for join problems because

a. Repeated execution makes those inefficient for this purpose

Type II nested queries execute one time for each row of the outer query.
5. How many rows are there in the “student” table, if “select count(name) from student where gpa > 3” returns the number M and “select count(*) from student where gpa <= 3” returns the number N:
   a. M + N
   b. >= M + N
   c. <= M + N
   d. None of the above
5. How many rows are there in the “student” table, if “select count(name) from student where gpa > 3” returns the number M and “select count(*) from student where gpa <= 3” returns the number N:

b. \( \geq M + N \)

There are two assumptions:
- Every student has a name
- Not every student has a GPA

Then all students that have a GPA are \( M + N \), yet there can still be more students than this.
6. Explain a situation when a full outer join is useful.

A full outer join is often useful when you wish to inspect two tables to see which rows are orphaned.

For example, see the next slide. Upon inspection, you can easily see that there is a Faculty that does not teach an Offering, and there is an Offering without a Faculty assigned to it.
### Faculty

<table>
<thead>
<tr>
<th>FacNo</th>
<th>FacName</th>
</tr>
</thead>
<tbody>
<tr>
<td>111-11-1111</td>
<td>joe</td>
</tr>
<tr>
<td>222-22-2222</td>
<td>sue</td>
</tr>
<tr>
<td>333-33-3333</td>
<td>sara</td>
</tr>
</tbody>
</table>

### Offering

<table>
<thead>
<tr>
<th>OfferNo</th>
<th>FacNo</th>
</tr>
</thead>
<tbody>
<tr>
<td>1111</td>
<td>111-11-1111</td>
</tr>
<tr>
<td>2222</td>
<td>222-22-2222</td>
</tr>
<tr>
<td>3333</td>
<td>111-11-1111</td>
</tr>
<tr>
<td>4444</td>
<td></td>
</tr>
</tbody>
</table>

### Outer Join of Offering and Faculty

<table>
<thead>
<tr>
<th>FacNo</th>
<th>FacName</th>
<th>OfferNo</th>
</tr>
</thead>
<tbody>
<tr>
<td>111-11-1111</td>
<td>joe</td>
<td>1111</td>
</tr>
<tr>
<td>222-22-2222</td>
<td>sue</td>
<td>2222</td>
</tr>
<tr>
<td>111-11-1111</td>
<td>joe</td>
<td>3333</td>
</tr>
<tr>
<td>333-33-3333</td>
<td>sara</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>4444</td>
</tr>
</tbody>
</table>