CS105 Lab 3: Developing Queries

Brian Borucki – bborucki@bu.edu

Aggregates & Queries

- Three cases when you'll want aggregates:
  - You just want a numeric value
    - SELECT AGG(attribute)
      FROM <table>

    - SELECT MIN(runtime)
      FROM Movie
      WHERE rating = 'PG';

Aggregates & Queries

- You're using GROUP BY, assign an aggregate value to each group

  -
    ```
    SELECT <attribute 1>, AGG(attribute 2)
    FROM <table>
    GROUP BY <attribute 1>
    ```
- You're using GROUP BY and want to use an aggregate to check for a condition

  -
    ```
    SELECT <attribute 1>
    FROM <table>
    GROUP BY <attribute 1>
    HAVING AGG() >= x
    ```

Aggregates & Queries

- Why not something like:
  - SELECT <attribute 1>, AGG(attribute 2)
    FROM <table>
- Aggregates return a single value, SELECT may return with more than one row
- Not defined how the DBMS will assign this value to each row
- Some DBMS's are ok with it, others aren't, but no guarantee your results will make sense

Subqueries

- Subqueries must function correctly as queries on their own
- When we are solving a problem composed of sub-problems, we need subqueries
  - Looking for all R rated movie(s) with the shortest run time... but what is the shortest run time?
  - Need a query to find the shortest run time before we can figure out what movie(s) correspond to it

Example

```
SELECT student, credit_status
FROM Enrolled
WHERE course IN (SELECT course
                 FROM Enrolled
                 GROUP BY course
                 HAVING COUNT(*)>1);
```

- What does this query do?

Example

-
  ```
  SELECT student, credit_status
  FROM Enrolled
  WHERE course IN (SELECT course
                   FROM Enrolled
                   GROUP BY course
                   HAVING COUNT(*)>1);
  ```
- What does this query do?
- What does the sub-query do?

Example - Subquery

-
  ```
  SELECT course
                 FROM Enrolled
                 GROUP BY course
                 HAVING COUNT(*)>1;
  ```
- Grouping each course together
- For each group, counting how many rows are in that group
- If that group has a count > 1, select it's course attribute

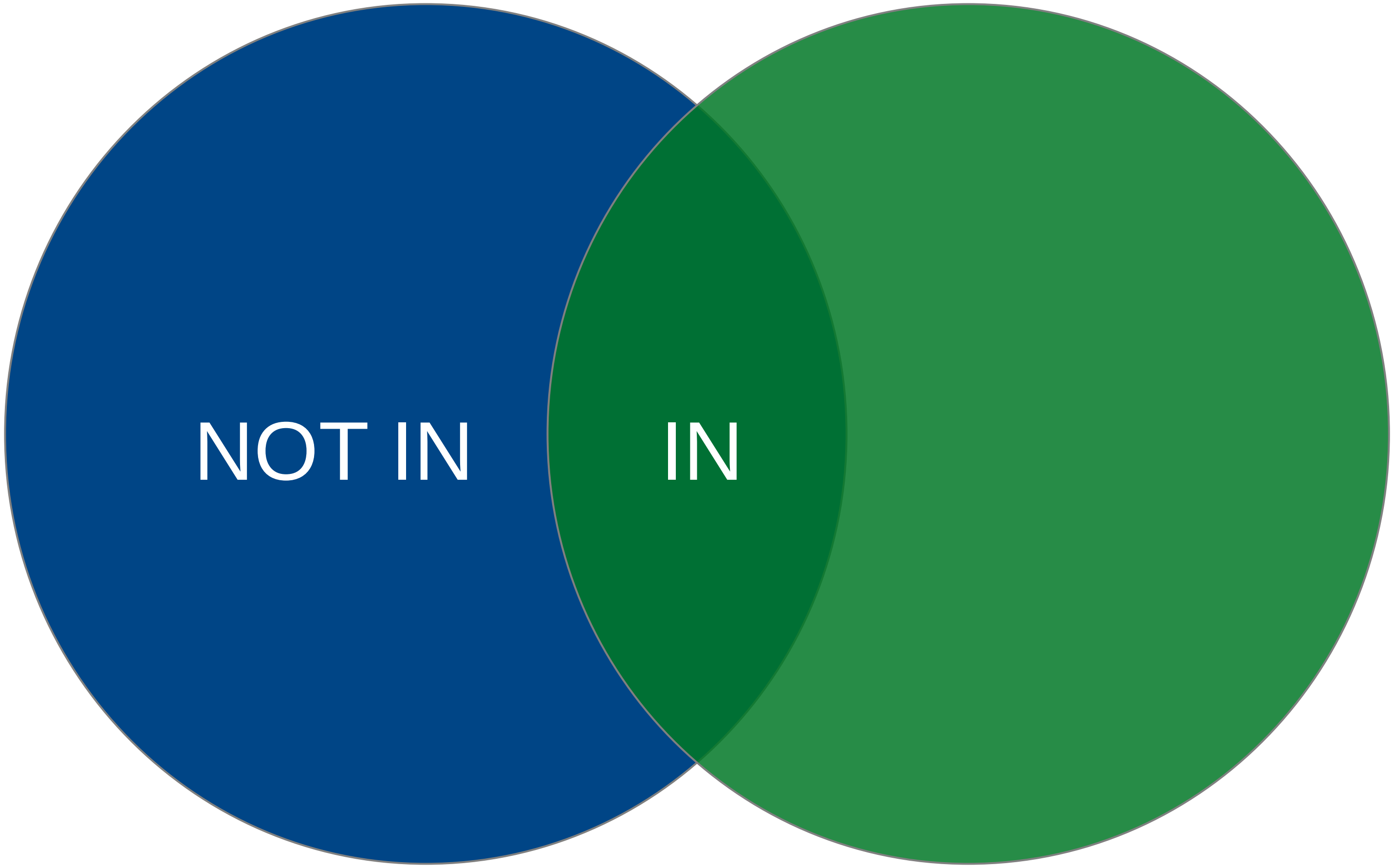
Example

```
SELECT student, credit_status
 FROM Enrolled
 WHERE course IN (SELECT course
         FROM Enrolled
         GROUP BY course
         HAVING COUNT(*)>1);
```

Example

- SELECT student, credit_status
  FROM Enrolled
  WHERE course IN (a column of the names of
  courses that have more than 1 student);
- What does this query do?
- First need to understand the IN keyword

IN Keyword



WHERE <attribute> [NOT] IN (set)

Set may be the result of a subquery, or a set of values

WHERE credit_status IN ('ugrad', 'grad')

Example

- SELECT student, credit_status
  FROM Enrolled
  WHERE course IN (a column of the names of
  courses that have more than 1 student)
- What does this query do?
- Returns the student ID credit status of every student that is in enrolled in a course with more than one student.

Rules of Thumb

- Start with the FROM clause
- If you need more than one table, determine the necessary join conditions
- Determine if a GROUP BY clause is needed
- Determine any other conditions that are needed
- Fill in the rest of the query: SELECT, ORDER BY

Example

- Find the number of students for each major.
  - How many tables?
    - FROM MajorsIn
  - Using GROUP BY?
    - Yes, on what?
    - GROUP BY dept
  - Other conditions?
  - What do we put in SELECT?
    - SELECT dept, COUNT(*)

Example

- Find the number of students for each major.

- Putting it all together we get:

-
  ```
  SELECT dept, COUNT(*)
  FROM MajorsIn
  GROUP BY (dept);
  ```

Inner Joins

- When the information we want appears in all of the tables

- SELECT *
  FROM MajorsIn, Enrolled
  WHERE MajorsIn.Student = Enrolled.student;

- Student appear only if their ID is in BOTH MajorsIn and Enrolled

Outer Joins

- When the information we want appears in at least the left table

- SELECT *
  FROM Enrolled LEFT OUTER JOIN MajorsIn
  ON Enrolled.student = MajorsIn.student

- Student appear if their ID is in Enrolled whether or not their ID appears in MajorsIn

This Lab

- Pair up with a partner or two
- Please make sure your name gets on the attendance sheet
- Please make sure you submit the lab on websubmit
- Let me know if you have any questions!