Set #5: SQL SELECT: sub-queries
(Chapter 2 in Kroenke)

Recap: (partial)

- SELECT [DISTINCT] column_name(s)
  FROM table_names
  WHERE conditions_for_rows
  GROUP BY column_name(s)
  HAVING conditions_for_groups
  ORDER BY column_name(s) [ASC/DESC]
Today’s Agenda

- Sub-queries

Sub-queries

- Aggregate Operators Limitations
  - Return only one row
  - Not in WHERE clause
Select age of oldest student

- SELECT MAX (Age)
  FROM Students S

Select oldest students and their age

- SELECT S.SName, MAX (Age)
  FROM Students S
  (Illegal!

- SELECT S.SName, S.Age
  FROM Students S
  WHERE S.AGE = (SELECT MAX(Age)
                 FROM Students)
  (Correct!

  Sub-query)
Select students with age higher than average

- SELECT *
  FROM Students
  WHERE Age > AVG(Age)

- SELECT *
  FROM Students
  WHERE Age > (SELECT AVG(Age)
                FROM Students)

Subqueries

- SELECT *
  FROM Students
  WHERE Age > (SELECT AVG(Age)
               FROM Students)

  Second select is a subquery (or nested query)
  You can have subqueries in FROM or HAVING clause also
Subqueries in FROM Clause

- Find name of students enrolled in both ‘IT360’ and ‘IT334’

```sql
SELECT FName + ' ' + LName AS StudentName
FROM Students, (SELECT Alpha
FROM Enroll
WHERE CourseID = 'IT360'
AND Alpha IN
(SELECT Alpha
FROM Enroll
WHERE CourseID = 'IT334')
) AS ResultAlphaTable
WHERE Students.Alpha = ResultAlphaTable.Alpha
```

Subqueries Exercise

- Students(Alpha, LName, FName, Class, Age)
- Enroll(Alpha, CourseID, Semester, Grade)

1. Find alpha for students enrolled in both ‘IT360’ and ‘IT334’
2. Find name of students enrolled in both ‘IT360’ and ‘IT334’
Class Exercise

- Students(Alpha, LName, FName, Class, Age)
- Enroll(Alpha, CourseID, Semester, Grade)

- Find the name of students enrolled in ‘IT360’
  - Usual way
  - Use subqueries

Class Exercise

- What does this query compute:
- SELECT FName, LName
  FROM Students S, Enroll E1, Enroll E2
  WHERE S.Alpha = E1.Alpha
    AND S.Alpha = E2.Alpha
    AND E1.CourseID = ‘IT360’
    AND E2.CourseID = ‘IT344’
SELECT Summary

- SELECT [DISTINCT] column_name(s) | aggregate_expr
  FROM table_name(s)
  WHERE row_conditions
  GROUP BY grouping_columns
  HAVING group_conditions
  ORDER BY column_name(s) [ASC/DESC]