Tutorial 4 – Nested SQL

CSC343 - Introduction to Databases
Fall 2008

TA: Lei Jiang

Nested Query

- **Question**

Student (snum: integer, sname: string, major: string, level: string, age: integer)
Class (name: string, meets_at: string, room: string, fid: integer)
Enrolled (snum: integer, cname: string)
Faculty (fid: integer, fname: string, deptid: integer)

  - (E5.1.2) Q8: Find the ***age*** of the oldest student who is enrolled in a course taught by I. Teach.

- **The way of thinking**
  - Divide and Conquer
    - Find the set S1 of students who are enrolled in a course taught by I. Teach
    - Find the age of the oldest student in a set S1 of students

- **Answer**

Q8:

```
SELECT MAX(S.age)
FROM Student S
WHERE S.snum IN (
                    SELECT E.snum
                    FROM Class C, Enrolled E, Faculty F
                    WHERE E.cname = C.name AND C.fid = F.fid
                            AND F.fname = 'I.Teach'
)
```

Without nested query

```
SELECT MAX(S.age)
FROM Student S, Class C,
        Enrolled E, Faculty F
WHERE S.snum = E.snum
        AND E.cname = C.name
        AND C.fid = F.fid
        AND F.fname = 'I.Teach'
```

Nested Query

- **Question**

Student (snum: integer, sname: string, major: string, level: string, age: integer)
Class (name: string, meets_at: string, room: string, fid: integer)
Enrolled (snum: integer, cname: string)
Faculty (fid: integer, fname: string, deptid: integer)

  - (E5.1.3) Q9: Find the names of all classes that either meet in room BA1180 or have 2 or more students enrolled.

- **The way of thinking**
  - Divide and Conquer
    - Find the set A of classes that meet in room BA1180
    - **Find the set B of classes that have 2 or more students enrolled**
    - Return A union B

- **Answer**

Without Nested Query:
SELECT C.name
FROM Class C
WHERE C.room = 'BA1180'
UNION
SELECT E.cname
FROM Enrolled E
GROUP BY E.cname
HAVING COUNT (*) >= 2)

Q9:
SELECT C.name
FROM Class C
WHERE C.room = 'BA1180'
OR C.name IN (**SELECT E.cname**
FROM Enrolled E
GROUP BY E.cname
HAVING COUNT (*) >= 2)

Nested Query

Student (snum: integer, sname: string, major: string, level: string, age: integer)
Class (name: string, meets_at: string, room: string, fid: integer)
Enrolled (snum: integer, cname: string)
Faculty (fid: integer, fname: string, deptid: integer)

- **Question**
  - (E5.1.5) Q10: Find the names of faculty members *who teach in every room* in which some class is taught..
- **The way of thinking**
  - Rephrase (change "for all / every …" into "there is no …"):
    - Q10: Find the names of faculty members for whom there is no such room that some class is taught in that room and he/she has not taught a class in that room.
  - Divide and Conquer
    - Find the names of faculty members for whom there is no such room that …
    - some class is taught in that room but he/she has not taught a class in that room
      - Find the set A of all the rooms in which some class is taugh
      - **Find the set B of all the rooms in which he/she has taught a class**
      - Take the set difference A - B
- **Answer**

Q10:
SELECT F.fname
FROM Faculty F
WHERE NOT EXISTS ((SELECT C.room FROM Class C)
EXCEPT
(**SELECT C1.room FROM Class C1 WHERE C1.fid = F.fid**))

Nested Query

- **Question**
  - Given following database instance, answer Q10.

Student

snum	sname	major	level	age
101	Helen	CS	JR	19
102	Charles	CS	SR	21
103	Andy	CS	GR	25
104	Bob	CS	SR	23
105	Zorba	CS	GR	31

Enrolled

snum	cname
101	CSC343
101	CSC443
101	ECE300
102	CSC343
102	ECE201
103	CSC343
103	CSC443
103	ECE300
103	ECE201
105	CSC343

Class

name	meets_at	room	fid
CSC343	W1	BA1180	201
CSC443	T2	BA1170	202
ECE300	M1	BA1180	203
ECE201	F12	BA1190	203
CSC165	R3	BA1170	203

Faculty

fid	fname	deptid
201	S. Jackson	301
202	M. Shanks	301
203	I. Teach	302

Q10:

```
SELECT F.fname                                  (1)
FROM Faculty F
WHERE NOT EXISTS (( SELECT C.room               (2)
                    FROM Class C )
                    EXCEPT
                   (SELECT C1.room              (3)
                    FROM Class C1
                    WHERE C1.fid = F.fid ))
```

- **Answer**

I. Teach

Nested Query

- **Question**

Student (snum: integer, sname: string, major: string, level: string, age: integer)
Class (name: string, meets_at: string, room: string, fid: integer)
Enrolled (snum: integer, cname: string)
Faculty (fid: integer, fname: string, deptid: integer)

  - (E5.1.6) Q11: Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five.

- **The way of thinking**
  - Divide and Conquer
    - **For each faculty member, find the combined enrolment *r* of the courses he/she teaches**
    - Print the name of that faculty member if *r* is less than 5.

- **Answer**

Q11:

```
SELECT F.fname
FROM Faculty F
WHERE 5 > (SELECT COUNT(E.snum)
           FROM Class C, Enrolled E
           WHERE C.name = E.cname
           AND C.fid = F.fid)
```

Each student may appear in the Enrolled table many time. Why do not we add the keyword 'DISTINCT' when we count student numbers here?

Nested Query

- **Question**

Student (snum: integer, sname: string, major: string, level: string, age: integer)
Class (name: string, meets_at: string, room: string, fid: integer)
Enrolled (snum: integer, cname: string)
Faculty (fid: integer, fname: string, deptid: integer)

  - (E5.1.10) Q12: Find the names of students enrolled in the maximum number of classes.
- **The way of thinking**
  - Rephrase (change "maximum …" into "greater than or equal to all others …")
    - Find the names of students who enrolled the number of classes greater than or equal to all other students did.
  - Divide and Conquer
    - Find the names of students who is …
    - enrolled the number of classes greater than or equal to …
    - **the number of classes all other students enrolled.**
- **Answer**

```
Q12:
SELECT DISTINCT S.sname
FROM Student S
WHERE S.snum IN (SELECT E.snum
                 FROM Enrolled E
                 GROUP BY E.snum
                 HAVING COUNT (*) >= ALL (SELECT COUNT (*)
                                          FROM Enrolled E2
                                          GROUP BY E2.snum ))
```


Nested Query

- **Question**

Student (snum: integer, sname: string, major: string, level: string, age: integer)
Class (name: string, meets_at: string, room: string, fid: integer)
Enrolled (snum: integer, cname: string)
Faculty (fid: integer, fname: string, deptid: integer)

  - (E5.1.11) Q13: Find the names of students not enrolled in any class.
- **The way of thinking**
  - Divide and Conquer
    - **Find the set A of students who have enrolled in some class**
    - Print the names of students who is not in A
- **Answer**

```
Q13:
SELECT DISTINCT S.sname
FROM Student S
WHERE S.snum NOT IN (SELECT E.snum
                     FROM Enrolled E )
```

- Tutorial notes are also available at: http://www.cs.utoronto.ca/~leijiang/ta/343/08f/index.html