More Examples on SQL, Relational Algebra, and Tuple Relational Calculus Queries

Last Lecture:

- More Examples in TRC
- Safe Calculus Expressions
- Domain Relational Calculus (DRC)

Today:

- More Examples in TRC, SQL and RA

University Example

Professors (pid, pname, dept, ext.)

Students (sid, sname, major-dept, year)

Courses (cid, cname, dept, credithours)

Enrollment (sem-year, sid, cid, grade)

Teach (pid, cid, sem-year, class_size)

Professors have id's (pid), names (pname), dept that they work (dept), and a telephone extension (ext).

Similary, Students have id, name, major-dept and year (i.e, freshman, sophomore, etc).

Attributes of Courses and Enrollment are self explanatory. Attributes "dept" in relations Professors, and Courses, and attribute "major-dept" in relation Students have the same domain, and have values like "CS", "CE", "EE", "ME", etc.

Attribute "sem-year" has values like "Spring2017", "Fall2016", etc.

Assume that cid's are unique, i.e. if there are multiple sections of a course, each section has a unique cid.

Queries: Express in RA, TRC, SQL

Professors (pid, pname, dept, ext.)

Students (sid, sname, major-dept, year)

Courses (cid, cname, dept, credithours)

Enrollment (sem-year, sid, cid, grade)

Teach (pid, cid, sem-year, class_size)

Express the queries below using RA and in TRC and SQL.

1. Find cid and cname of courses that are offered by “CS” department that are taught by professors who are from another department in “Fall2015”.
2. Find sid’s, names and major-dept of students who enrolled in a course that is taught by a professor Smart.
3. Find pid and names of professors who teach **no** courses in “Fall2015”.

Professors (pid, pname, dept, ext.)
Students (sid, sname, major-dept, year)
Courses (cid, cname, dept, credithours)
Enrollment (sem-year, sid, cid, grade)
Teach (pid, cid, sem-year, class_size)

Find cid and cname of courses that are offered by “CS” department that are taught by professors who are from another department in “Fall2015”.

- RA

$$\Pi_{cid,\ cname}(\sigma_{dept = \text{'CS'}}(Courses) \bowtie_{Courses.cid = d.cid} \rho_{d}(\sigma_{dept <> \text{'CS'} \wedge sem\text{-}year=\text{'Fall2015'}}(Professors \bowtie Teach)))$$

- TRC

$$\{t \mid (\exists s)\ (s \in Courses \wedge t[cid] = s[cid] \wedge t[cname] = s[cname] \wedge s[dept] = \text{“CS”} \wedge (\exists u)\ (u \in Teach \wedge s[cid] = u[cid] \wedge u[sem\text{-}year] = \text{“Fall2015”}) \wedge (\exists v)\ (v \in Professors \wedge v[pid] = u[pid] \wedge v[dept] <> \text{“CS”})\}$$

- SQL

```
Select t.cid, cname from Professors p, Teach t, Courses c
Where p.dept <> 'CS' and p.pid = t.pid and t.sem-year = 'Fall2015' and c.cid = t.cid and c.dept = 'CS'
```

Professors (<u>pid</u>, pname, dept, ext.)
Students (<u>sid</u>, sname, major-dept, year)
Courses (<u>cid</u>, cname, dept, credithours)
Enrollment (<u>sem-year, sid, cid,</u> grade)
Teach (<u>pid, cid, sem-year,</u> class_size)

Find sid's, names and major-dept of students who enrolled in a course that is taught by a professor Smart.

- RA $\prod_{sid,\ sname,\ major\text{-}dept}$((Enrollment $\bowtie$ Students) $\bowtie$ ($\sigma_{pname=\ 'Smart'}$ (Professors $\bowtie$Teach)))

- TRC

$$\{t \mid (\exists s)\ (s \in Students \wedge t[sid] = s[sid] \wedge t[sname] = s[sname]$$
$$\wedge$$
$$t[\text{major-}dept] = s[\text{major-dept}]$$
$$\wedge\ (\exists u)\ (u \in Enrollments \wedge s[sid] = u[sid])$$
$$\wedge\ (\exists v)\ (v \in Teach \wedge u[cid] = v[cid]\}$$
$$\wedge\ (\exists w)\ (w \in Professors \wedge v[pid] = u[pid] \wedge$$
$$v[pname] = \text{“Smart”})\ \}$$

- SQL

Select s.sid, s.sname, s.major-dept from Professors p, Teach t, Students s, Enrollment e
Where p.pname='Smart' and p.pid = t.pid and t.cid = e.cid and e.sid = s.sid

Professors (pid, pname, dept, ext.)
Students (sid, sname, major-dept, year)
Courses (cid, cname, dept, credithours)
Enrollment (sem-year, sid, cid, grade)
Teach (pid, cid, sem-year, class_size)

Find pid and names of professors who teach **no** courses in “Fall2015”.

- RA $\prod_{pid,\ pname}$ (Professors) - $\prod_{pid,\ pname}$ ($\sigma_{sem\text{-}year=\ 'Fall2015'}$ (Professor $\bowtie$ Teach)))

- TRC $\{t \mid (\exists s)\ (s \in Professors \wedge t\ [pid] = s\ [pid\ ] \wedge t\ [pname] = s\ [pname]$
  $\wedge\ \neg\ (\exists u)\ (u \in Teach \wedge u\ [pid] = s\ [pid] \wedge u\ [sem\text{-}year] =$ ‘Fall2015’$)\ \}$

  $\{t \mid (\exists s)\ (s \in Professors \wedge t\ [pid] = s\ [pid\ ] \wedge t\ [pname] = s\ [pname]$
  $\wedge\ (\forall\ u)\ (u \in Teach \wedge u\ [pid] = s\ [pid] \Rightarrow u\ [sem\text{-}year] <>$ ‘Fall2015’$)\ \}$

- SQL Select pid, pname from Professors p where
  not exists (select pid from Teach t where p.pid = t.pid and sem-year = ‘Fall2015’)

Queries: Express in RA, TRC, SQL

Professors (pid, pname, dept, ext.)

Students (sid, sname, major-dept, year)

Courses (cid, cname, dept, credithours)

Enrollment (sem-year, sid, cid, grade)

Teach (pid, cid, sem-year, class_size)

4. Find pid and names of professors who teach **only** courses offered by “CS” department.
5. Find pnames and pids of professors who teach **every** course offered by “CS” department.
6. Find sid’s of students who enroll in “Spring 2017” **every** 3 credit hr course offered by “CS” department.
7. Find cid’s and names of courses in which **every** student majoring in “CS” enrolled in “Spring 2017”

Professors (pid, pname, dept, ext.)
Students (sid, sname, major-dept, year)
Courses (cid, cname, dept, credithours)
Enrollment (sem-year, sid, cid, grade)
Teach (pid, cid, sem-year, class_size)

Find pid and names of professors who teach **only** courses offered by "CS" department.

- RA

$\prod_{pid,\ pname}$ (Professors $\bowtie$ Teach $\bowtie$ ($\sigma_{dept='CS'}$(Courses)) - Professors $\bowtie$ Teach $\bowtie$ ($\sigma_{dept<>'CS'}$(Courses)))

- TRC

$\{t \mid (\exists s)\ (s \in Professors \wedge t[pid] = s[pid] \wedge t[pname] = s[pname]$
$\wedge\ (\exists u)\ (u \in Teach \wedge u[pid] = s[pid]$
$\wedge\ (\forall v)\ (v \in Courses \wedge u[cid] = v[cid] \Rightarrow v[dept] = \text{'CS'})\ )\ )\ \}$

- SQL

```
Select p1.pid, p1.pname from Professors p1, Teach t1, Courses c1 where p1.pid = t1.pid and
t1.cid = c1.cid and c1.dept = 'CS' and not exists (
     select t2.pid from Teach t2, Courses c2 where p1.pid = t2.pid and t2.cid = c2.cid and
c2.dept <> 'CS')
```

Professors (pid, pname, dept, ext.)
Students (sid, sname, major-dept, year)
Courses (cid, cname, dept, credithours)
Enrollment (sem-year, sid, cid, grade)
Teach (pid, cid, sem-year, class_size)

Find pid and names of professors who teach **only** courses offered by "CS" department.

- RA

$\prod_{pid,\ pname}$ (Professors $\bowtie$ Teach $\bowtie$ ($\sigma_{dept='CS'}$(Courses)) - Professors $\bowtie$ Teach $\bowtie$ ($\sigma_{dept<>'CS'}$(Courses)))

- TRC

$\{t \mid (\exists s)\ (s \in Professors \wedge t[pid] = s[pid] \wedge t[pname] = s[pname]$
$\wedge\ (\exists u)\ (u \in Teach \wedge u[pid] = s[pid]$
$\wedge\ (\forall v)\ (v \in Courses \wedge u[cid] = v[cid] \Rightarrow v[dept] = \text{'CS'})\ )\ )\ \}$

- SQL

```
Select p1.pid, p1.pname from Professors p1, Teach t1, Courses c1 where p1.pid = t1.pid and
t1.cid = c1.cid and c1.dept = 'CS' and not exists (
     select t2.pid from Teach t2, Courses c2 where p1.pid = t2.pid and t2.cid = c2.cid and
c2.dept <> 'CS')
```

Professors (pid, pname, dept, ext.)
Students (sid, sname, major-dept, year)
Courses (cid, cname, dept, credithours)
Enrollment (sem-year, sid, cid, grade)
Teach (pid, cid, sem-year, class_size)

Find pnames and pids of professors who teach **every** course offered by "CS" department.

- RA

$$\Pi_{pid,\ pname}((\text{Professors} \bowtie \text{Teach}) \ / \ \Pi_{cid}(\sigma_{dept=\ 'CS'}(\text{Courses}))\ )$$

- TRC

$$\{t \mid (\exists s)\ (s \in Professors \wedge t\ [pid] = s\ [pid\ ] \wedge t\ [pname] = s\ [pname]$$
$$\wedge\ (\forall\ u)\ (u \in Courses \wedge u\ [dept] = 'CS' \Rightarrow (\exists v)\ (v \in Teach \wedge s\ [pid] = v\ [pid] \wedge v\ [cid] = u\ [cid])\ )\ \}$$

- SQL

```
Select pid, pname from Professors p where not exists (
      (select c.cid from Courses c where c.cname = 'CS') except
      (select t.cid from Teach t where t.pid = p.pid)
)
```

Professors (pid, pname, dept, ext.)
Students (sid, sname, major-dept, year)
Courses (cid, cname, dept, credithours)
Enrollment (sem-year, sid, cid, grade)
Teach (pid, cid, sem-year, class_size)

Find sid's of students who enroll in "Spring 2017" **every** 3 credit hr course offered by "CS" department.

- RA

  $\prod_{sid,cid}$((Students $\bowtie$ $\sigma_{sem\text{-}year= \text{'Spring2017'}}$(Enrollment)) / $\prod_{cid}$ ($\sigma_{dept= \text{'CS'} \wedge credithours=3}$ (Courses)))

- TRC

  $\{t \mid (\exists s)\ (s \in Students \wedge t\,[sid] = s\,[sid\,]$
  $\wedge\ (\forall\, u)\ (u \in Courses \wedge u\,[dept] = \text{'CS'} \wedge u\,[credithours] = 3 \Rightarrow$
  $(\exists w)\ (w \in Enrollment \wedge s\,[sid] = w\,[sid] \wedge w\,[cid] = u\,[cid] \wedge w$
  $[sem\text{-}year] = \text{'Spring2017'})\ )\ \}$

- SQL

```
Select sid from Students s where not exists (
      (select c.cid from Courses c where c.dept = 'CS' and c.credithours = 3) except
      (select e.cid from Enrollment e where e.sid = s.sid and sem-year =
'Spring2017')
)
```

Professors (pid, pname, dept, ext.)
Students (sid, sname, major-dept, year)
Courses (cid, cname, dept, credithours)
Enrollment (sem-year, sid, cid, grade)
Teach (pid, cid, sem-year, class_size)

Find cid's and names of courses in which **every** student majoring in "CS" enrolled in "Spring 2017"

- RA

  $\prod_{cid,cname,sid}$(Courses $\bowtie$ ($\sigma_{sem\text{-}year= \text{'Spring2017'}}$ (Enrollment)))/ $\prod_{sid}$ ($\sigma_{major\text{-}dept= \text{'CS'}}$ (Students))

- TRC

  $\{t \mid (\exists s)\ (s \in Course \wedge t[cid] = s[cid] \wedge t[cname] = s[cname] \wedge (\forall u)\ (u \in Students \wedge u[major\text{-}dept] = \text{'CS'} \Rightarrow (\exists v)\ (v \in Enrollment \wedge u[sid] = v[sid] \wedge v[sem\text{-}year] = \text{'Spring2017'} \wedge s[cid] = v[cid])\ )\ \}$

- SQL

```
Select cid,cname from Courses c where not exists (
      (select s.sid from Students s where s.major-dept='CS') except
      (select e.sid from Enrollment e where e.cid = c.cid and e.sem-year =
'Spring2017')
)
```

Queries: Express in SQL

Instructor (id, name, dept-name, salary)

Students (id, name, dept, year)

Courses (course_id, title, dept_name, credits)

Enrollment (semester, year, student_id, course_id, grade)

Teaches (id, course_id, sec_id, semester, year)

1. For each instructor, find the number of courses that they teach, and list the id, name, department of the instructor, and the number of courses taught.
2. For each department, find the name and id of instructors who teach the maximum number of courses, and list the dept., id and name of each such instructor.
3. For each department, find the name and id of instructors who teach the maximum number of courses, and list the dept., id, name of each such instructor, together with the number of courses the instructor teaches.

Instructor (id, name, dept-name, salary)
Students (id, name, dept, year)
Courses (course_id, title, dept_name, credits)
Enrollment (semester, year, student_id, course_id, grade)
Teaches (id, course_id, sec_id, semester, year)

For each instructor, find the number of courses that they teach, and list the id, name, department of the instructor, and the number of courses taught.

- SQL

```
Select p1.dept_name, p1.id, p1.name, count(distinct course_id)
From Teaches t1, Instructor p1
Where t1.id = p1.id
Group by p1.id, p1.name, p1.dept_name
```

Dept_name	ID	Name	Count
Elec. Eng.	98345	Kim	1
History	32343	El Said	1
Biology	76766	Crick	2
Finance	12121	Wu	1
Comp. Sci.	10101	Srinivasan	3
Physics	22222	Einstein	1
Music	15151	Mozart	1
Comp. Sci.	45565	Katz	2
Comp. Sci.	83821	Brandt	2

Instructor (id, name, dept-name, salary)
Students (id, name, dept, year)
Courses (course_id, title, dept_name, credits)
Enrollment (semester, year, student_id, course_id, grade)
Teaches (id, course_id, sec_id, semester, year)

For each department, find the name and pid of professors who teach the maximum number of courses, and list the dept., pid and pname of each such professor.

- SQL

```
(Select p1.dept_name, p1.id, p1.name
From Teaches t1, Instructor p1
Where t1.id = p1.id)
EXCEPT
(Select tct1.dept_name, tct1.id, tct1.name From
    (Select p2.dept_name, t2.id, p2.name, count(distinct t2.course_id) as ct
    From Teaches t2, Instructor p2
    Where t2.id = p2.id
    Group by p2.dept_name, t2.id, p2.name) as tct1,
    (Select p3.dept_name, t3.id, p3.name, count(distinct t3.course_id) as ct
    From Teaches t3, Instructor p3
    Where t3.id = p3.id Group by p3.dept_name, t3.id, p3.name) as tct2
Where tct1.dept_name = tct2.dept_name and tct1.ct < tct2.ct)
```

Dept_name	ID	Name
Elec. Eng.	98345	Kim
History	32343	El Said
Biology	76766	Crick
Finance	12121	Wu
Comp. Sci.	10101	Srinivasan
Physics	22222	Einstein
Music	15151	Mozart

Instructor (id, name, dept-name, salary)
Students (id, name, dept, year)
Courses (course_id, title, dept_name, credits)
Enrollment (semester, year, student_id, course_id, grade)
Teaches (id, course_id, sec_id, semester, year)

For each department, find the name and pid of professors who teach the maximum number of courses, and list the dept., pid, pname of each such professor, together with the number of courses professor teaches.

- SQL

```
(Select p1.dept_name, p1.id, p1.name, count(distinct t1.course_id)
From Teaches t1, Instructor p1
Where t1.id = p1.id
Group by p1.dept_name, p1.id, p1.name)
EXCEPT
(Select tct1.dept_name, tct1.id, tct1.name, tct1.ct From
    (Select p2.dept_name, t2.id, p2.name, count(distinct t2.course_id) as ct
            From Teaches t2, Instructor p2
    Where t2.id = p2.id
    Group by p2.dept_name, t2.id, p2.name) as tct1,
    (Select p3.dept_name, t3.id, p3.name, count(distinct t3.course_id) as ct
            From Teaches t3, Instructor p3
    Where t3.id = p3.id Group by p3.dept_name, t3.id, p3.name) as tct2
Where tct1.dept_name = tct2.dept_name and tct1.ct < tct2.ct)
```

Dept_name	ID	Name	Count
Elec. Eng.	98345	Kim	1
History	32343	El Said	1
Biology	76766	Crick	2
Finance	12121	Wu	1
Comp. Sci.	10101	Srinivasan	3
Physics	22222	Einstein	1
Music	15151	Mozart	1