UBC Department of Computer Science
Undergraduate Events
More details @ https://my.cs.ubc.ca/students/development/events

Realtor.com Technology Meet-up

Wed. Mar. 7th, 2018
5:30pm-8:00pm
BrainStation

Hoosuite's Celebration of International Women's Day

Thurs. Mar. 8th, 2019
9:00am – 10:00am
https://www.facebook.com/HootsuiteLife/

CA Technologies Info Session & Tech Talk: The Future of APIs

Tues. Mar. 13th, 2018
5:30pm-7:00pm
DMP 301

Tesla Info Session

Wed. Mar. 14th, 2018
5:30pm-7:30pm
Scarfe Room 100

CSSS Olympics

Fri. Mar. 16th, 2018
5:00pm – 7:00pm
Pacific Poke Lounge

Goldman Sachs Info Session

Fri. Mar. 23rd, 2018
12:30pm – 1:45pm
David Lam 005 & 009

Deloitte Days

Thurs. Mar. 29th, 2018
11:00am – 4:00pm
Jack Poole Hall

CPSC 304 – March 6, 2018 Administrative notes

- Reminder: tutorial this week
- Reminder: tutorial due Friday
- Project feedback will be coming – it will be brief. Primary goal was to make sure people had things planned
- Reminder: Midterm #2: March 8@7pm in CIRS 1250 (same room as last time)
  - Closed notes, closed book, closed neighbour
  - Bring ID
  - Will cover FDs through the end of Datalog
  - Thursday's lecture will be review

Now where were we…

- We'd just started SQL
- In particular, we'd been looking at queries of the basic form

SELECT
FROM
WHERE

So how does a typical SQL query relate to relational algebra then?

SQL:

SELECT $A_1, A_2, \ldots, A_n$
FROM $r_1, r_2, \ldots, r_m$
WHERE P

Is approximately equal to

Relational algebra

$$\pi_{A1, A2, \ldots, An}(\sigma_P (r_1 \times r_2 \times \ldots \times r_m))$$

Difference? Duplicates.

Remove them? Distinct

Using DISTINCT

- Find the names of actors who have been in at least one movie

```
SELECT   DISTINCT Name
FROM     StarsIn S, MovieStar M
WHERE    S.StarID = M.StarID
```

- Would removing DISTINCT from this query make a difference?

Clicker question: distinction

Consider the relation: Scores(Team, Opponent, RunsFor, RunsAgainst) and the query:

```
SELECT DISTINCT Team,
               RunsFor
FROM    Scores
```

Which is true:

A. 1 appears once
B. 5 appears twice
C. 6 appears 4 times
D. All are true
E. None are true

Team	Opponent	Runs For	Runs Against
Dragons	Tigers	5	3
Carp	Swallows	4	6
Bay Stars	Giants	2	1
Marines	Hawks	5	3
Ham Fighters	Buffaloes	1	6
Lions	Golden Eagles	8	12
Tigers	Dragons	3	5
Swallows	Carp	6	4
Giants	Bay Stars	1	2
Hawks	Marines	3	5
Buffaloes	Ham Fighters	6	1
Golden Eagles	Lions	12	8

Clicker question: distinction

Consider the relation: Scores(Team, Opponent, RunsFor, RunsAgainst) and the query:

```
SELECT DISTINCT Team,
               RunsFor
FROM    Scores
```

Which is true:

A. 1 appears once
B. 5 appears twice Correct
C. 6 appears 4 times
D. All are true
E. None are true

Team	Opponent	Runs For	Runs Against
Dragons	Tigers	5	3
Carp	Swallows	4	6
Bay Stars	Giants	2	1
Marines	Hawks	5	3
Ham Fighters	Buffaloes	1	6
Lions	Golden Eagles	8	12
Tigers	Dragons	3	5
Swallows	Carp	6	4
Giants	Bay Stars	1	2
Hawks	Marines	3	5
Buffaloes	Ham Fighters	6	1
Golden Eagles	Lions	12	8

clickerdistinction.sql

Join Example

- Find the names of all movie stars who have been in a movie

Join Example

- Find the names of all movie stars who have been in a movie

```
SELECT Name
FROM StarsIn S, MovieStar M
WHERE S.StarID = M.StarID
```

Is this totally correct?

StarID	Name	Gender
1	Harrison Ford	Male
2	Vivian Leigh	Female
3	Judy Garland	Female

MovieID	StarID	Character
1	1	Han Solo
4	1	Indiana Jones
2	2	Scarlett O'Hara
3	3	Dorothy Gale

Harrison Ford will appear twice

Join Example

- Find the names of all movie stars who have been in a movie

```
SELECT Name
FROM StarsIn S, MovieStar M
WHERE S.StarID = M.StarID
```

Is this totally correct?

```
SELECT DISTINCT Name
FROM StarsIn S, MovieStar M
WHERE S.StarID = M.StarID
```

What if two movie stars had the same name?

- What if I run the following query?

```
SELECT DISTINCT StarID, Name
FROM StarsIn S, MovieStar M
WHERE S.StarID = M.StarID
```

Error: Column StarID is ambiguous

Renaming Attributes in Result

- SQL allows renaming relations and attributes using the **as** clause:

  old-name **as** *new-name*

- Example: Find the title of movies and all the characters in them, and rename "Character" to "Character1"

```
SELECT    Title, Character AS Character1
FROM      StarsIn S, Movie M
WHERE     M.MovieID = S.MovieID
```

Congratulations: You know select-project-join queries

- Very common subset to talk about
  - You saw it in tutorial
- Can do many (but not all) useful things

SQL is *declarative*, not procedural
how do we know? Lets see what
procedural would look like…

Conceptual Procedural Evaluation Strategy

1. Compute the cross-product of *relation-list*.
2. Discard resulting tuples if they fail *qualifications*.
3. Delete attributes that are not in *target-list*.
4. If DISTINCT is specified, eliminate duplicate rows.

Example of Conceptual Procedural Evaluation

SELECT Name

FROM MovieStar M, StarsIn S

WHERE S.StarID = M.StarID AND MovieID = 276

join selection

MovieStar **X** StarsIn

(StarID)	Name	Gender	MovieID	(StarID)	Character
1273	Nathalie Portman	Female	272	1269	Leigh Anne Touhy
1273	Nathalie Portman	Female	273	1270	Mary
1273	Nathalie Portman	Female	274	1271	King George VI
1273	Nathalie Portman	Female	276	1273	Nina Sayers
…	…	…	…	…	…

New Students Example

- Class(name,meets_at,room,fid)
- Student(snum,sname,major,standing,age)
- Enrolled(snum,cname)
- Faculty(fid,fname,deptid)

Class Table

Name	Meets_at	Room	FID
Data Structures	MWF 10	R128	489456522
Database Systems	MWF 12:30-1:45	1320 DCL	142519864
Operating System Design	TuTh 12-1:20	20 AVW	489456522
Archaeology of the Incas	MWF 3-4:15	R128	248965255
Aviation Accident Investigation	TuTh 1-2:50	Q3	011564812
Air Quality Engineering	TuTh 10:30-11:45	R15	011564812
Introductory Latin	MWF 3-4:15	R12	248965255
American Political Parties	TuTh 2-3:15	20 AVW	619023588
Social Cognition	Tu 6:30-8:40	R15	159542516
Perception	MTuWTh 3	Q3	489221823
Multivariate Analysis	TuTh 2-3:15	R15	090873519
Patent Law	F 1-2:50	R128	090873519
Urban Economics	MWF 11	20 AVW	489221823
Organic Chemistry	TuTh 12:30-1:45	R12	489221823
Marketing Research	MW 10-11:15	1320 DCL	489221823
Seminar in American Art	M 4	R15	489221823
Orbital Mechanics	MWF 8 1320	DCL	011564812
Dairy Herd Management	TuTh 12:30-1:45	R128	356187925
Communication Networks	MW 9:30-10:45	20 AVW	141582651
Optical Electronics	TuTh 12:30-1:45	R15	254099823
Intoduction to Math	TuTh 8-9:30	R128	489221823

Student Table

SNUM	SNAME	MAJOR	ST	AGE
51135593	Maria White	English	SR	21
60839453	Charles Harris	Architecture	SR	22
99354543	Susan Martin	Law	JR	20
112348546	Joseph Thompson	Computer Science	SO	19
115987938	Christopher Garcia	Computer Science	JR	20
132977562	Angela Martinez	History	SR	20
269734834	Thomas Robinson	Psychology	SO	18
280158572	Margaret Clark	Animal Science	FR	18
301221823	Juan Rodriguez	Psychology	JR	20
318548912	Dorthy Lewis	Finance	FR	18
320874981	Daniel Lee	Electrical Engineering	FR	17
322654189	Lisa Walker	Computer Science	SO	17
348121549	Paul Hall	Computer Science	JR	18
351565322	Nancy Allen	Accounting	JR	19
451519864	Mark Young	Finance	FR	18
455798411	Luis Hernandez	Electrical Engineering	FR	17
462156489	Donald King	Mechanical Engineering	SO	19
550156548	George Wright	Education	SR	21
552455318	Ana Lopez	Computer Engineering	SR	19
556784565	Kenneth Hill	Civil Engineering	SR	21
567354612	Karen Scott	Computer Engineering	FR	18
573284895	Steven Green	Kinesiology	SO	19
574489456	Betty Adams	Economics	JR	20
578875478	Edward Baker	Veterinary Medicine	SR	21

Enrolled Table

SNUM	CNAME
112348546	Database Systems
115987938	Database Systems
348121549	Database Systems
322654189	Database Systems
552455318	Database Systems
455798411	Operating System Design
552455318	Operating System Design
567354612	Operating System Design
112348546	Operating System Design
115987938	Operating System Design
322654189	Operating System Design
567354612	Data Structures
552455318	Communication Networks
455798411	Optical Electronics
455798411	Organic Chemistry
301221823	Perception
301221823	Social Cognition
301221823	American Political Parties
556784565	Air Quality Engineering
99354543	Patent Law
574489456	Urban Economics

Faculty Table

FID	FNAME	DEPTID
142519864	I. Teach	20
242518965	James Smith	68
141582651	Mary Johnson	20
011564812	John Williams	68
254099823	Patricia Jones	68
356187925	Robert Brown	12
489456522	Linda Davis	20
287321212	Michael Miller	12
248965255	Barbara Wilson	12
159542516	William Moore	33
090873519	Elizabeth Taylor	11
486512566	David Anderson	20
619023588	Jennifer Thomas	11
489221823	Richard Jackson	33
548977562	Ulysses Teach	20

What kinds of queries can you answer so far?

- Find the names of all classes taught by Elizabeth Taylor

- Find the student ids of those who have taken a course named “Database Systems”

What kinds of queries can you answer so far?

- Find the names of all classes taught by Elizabeth Taylor

```
SELECT name
FROM Faculty f, class c
WHERE f.fid = c.fid and fname = 'Elizabeth Taylor'
```

Do we need distinct? A. Yes. B. No

- Find the student ids of those who have taken a course named “Database Systems”

```
SELECT snum
FROM enrolled e
WHERE cname = 'Database Systems'
```

Do we need distinct? A. Yes. B. No

What kinds of queries can you answer so far?

- Find the departments that have at least one faculty member

What kinds of queries can you answer so far?

- Find the departments that have at least one faculty member

```
SELECT DISTINCT deptid
FROM faculty
```

What kinds of queries can you answer so far?

Find the departments that have more than one faculty member (express not equal by "<>")

```
SELECT DISTINCT f1.deptid
FROM faculty f1, faculty f2
WHERE f1.fid <>f2.fid AND
F1.deptid = f2.deptid
```

That is why renaming is important

f1

fid	fname	Deptid
90873519	Elizabeth Taylor	11
619023588	Jennifer Thomas	11
…	…	…

f2

fid	fname	Deptid
90873519	Elizabeth Taylor	11
619023588	Jennifer Thomas	11
…	…	…

A good example for using the same table twice in a query

Do I need Distinct?

String comparisons

What are the student ids of those who have taken a course with “Database” in the name?

A string walks into a bar…

```
SELECT DISTINCT snum
FROM   enrolled
WHERE cname LIKE '%Database%'
```

- LIKE is used for string matching:
  - '_' stands for any one character and
  - '%' stands for 0 or more arbitrary characters.
- SQL supports string operations such as
  - concatenation (using "||")
  - converting from upper to lower case (and vice versa)
  - finding string length, extracting substrings, etc.

Do I need distinct?
A = yes B no

Ordering of Tuples

- List in alphabetic order the names of actors who were in a movie in 1939

```
SELECT distinct Name
FROM   Movie, StarsIn, MovieStar
WHERE Movie.MovieID = StarsIn.MovieID and
   StarsIn.StarID = MovieStar.StarID and year = 1939
ORDER BY Name
```

Order is specified by:

- **desc** for descending order
- **asc** for ascending order (default)
- E.g. **order by** *Name* **desc**

Clicker question: sorting

- Relation R has schema R(a,b,c). In the result of the query
  SELECT a, b, c
  FROM R
  ORDER BY c DESC, b ASC;
- What condition must a tuple *t* satisfy so that *t* **necessarily precedes** the tuple (5,5,5)? Identify one such tuple from the list below.

A. (3,6,3)
B. (1,5,5)
C. (5,5,6)
D. All of the above
E. None of the above

clickerorder.sql and clickerorder2.sql produce different ordering for 7,5,5 vs. 1,5,5

Clicker question: sorting

- Relation R has schema R(a,b,c). In the result of the query
  SELECT a, b, c
  FROM R
  ORDER BY c DESC, b ASC;
- What condition must a tuple *t* satisfy so that *t* **necessarily precedes** the tuple (5,5,5)? Identify one such tuple from the list below.

A. (3,6,3) — 3 < 5
B. (1,5,5) — Not specified
C. (5,5,6) — Right
D. All of the above
E. None of the above