Carnegie Mellon Univ. Dept. of Computer Science 15-415 - Database Applications

C. Faloutsos Rel. model - SQL part1

Carnegie Mellon

General Overview - rel. model

- Formal query languages
 - rel algebra and calculi
- · Commercial query languages
 - -SQL
 - QBE, (QUEL)

Carnegie Mellon

15-415 - C. Faloutsos

Overview - detailed - SQL

- DML
 - select, from, where, renaming
 - set operations
 - ordering
 - aggregate functions
 - nested subqueries
- other parts: DDL, embedded SQL, auth etc

Carnegie Mellor

15-415 - C. Faloutso

DML

General form

select a1, a2, ... an

from r1, r2, ... rm

where P

[order by]

[group by ...]

[having ...]

Carnegie Mellon

5-415 - C. Faloutsos

Reminder: our Mini-U db STUDENT CLASS Name Address <u>c-id</u> c-name units 15-413 s.e. 2 123 smith main str 234 jones forbes ave 15-412 o.s. TAKES SSN c-id gr 123 15-413 A grade 234 15-413 B Carnegie Mellon 15-415 - C. Faloutsos

DML - eg:

find the ssn(s) of everybody called "smith"

select ssn

from student

where name="smith"

Carnegie Mellon

15-415 - C. Faloutsos

DML - observation

General form

select a1, a2, ... an

from r1, r2, ... rm

where P

equivalent rel. algebra query?

Carnegie Mello

15-415 - C. Faloutsos

DML - observation

General form

select a1, a2, ... an

from r1, r2, ... rm

where P

 $\pi_{{}_{a1,a2,...an}}(\sigma_{{}_{P}}(r1\times r2\times...\times rm))$

Carnegie Mellon

15-415 - C. Faloutsos

DML - observation

General form

select distinct a1, a2, ... an from r1, r2, ... rm where P $\pi_{a1,a2,...an}(\sigma_p(r1\times r2\times...\times rm))$

Carnegie Mellon

15-415 - C. Faloutsos

select clause

select [distinct | all] name

from student

where address="main"

Carnegie Mellon

15-415 - C. Faloutsos

where clause

find ssn(s) of all "smith"s on "main"

select ssn

from student

where address="main" and

name = "smith"

Carnegie Mellon

15-415 - C. Faloutsos

where clause

- boolean operators (and or not ...)
- comparison operators (<, >, =, ...)
- and more...

Carnegie Mellon

15-415 - C. Faloutsos

12

CMU - 15-415

What about strings?

find student ssns who live on "main" (st or str or street - ie., "main st" or "main str" ...)

15-415 - C. Faloutsos

What about strings?

find student ssns who live on "main" (st or str or street)

select ssn

from student

where address like "main%"

%: variable-length don't care

_: single-character don't care

15-415 - C. Faloutsos

from clause

find names of people taking 15-415

STUDENT		CLASS		
Ssn Name	Address	c-id	c-name	units
123 smith	main str	15-413	s.e.	2
234 jones	forbes ave	15-412	o.s.	2

TAKES
SSN c-id grade 123 15-413 A 234 15-413 B

Carnegie Mellon

15-415 - C. Faloutsos

from clause

find names of people taking 15-415

select name

from student, takes

where ???

Carnegie Mellon

15-415 - C. Faloutsos

from clause

find names of people taking 15-415

select name

from student, takes

where student.ssn = takes.ssn and

takes.c-id = "15-415"

Carnegie Mellon

15-415 - C. Faloutsos

renaming - tuple variables

find names of people taking 15-415

select name

 ${\bf from}\ our Very Own Student,\ student Taking Classes$

where ourVeryOwnStudent.ssn =

student Taking Classes.ssn

and studentTakingClasses.c-id = "15-415"

Carnegie Mellon

15-415 - C. Faloutsos

18

CMU - 15-415 3

renaming - tuple variables

find names of people taking 15-415
select name
from ourVeryOwnStudent as S,
studentTakingClasses as T
where S.ssn =T.ssn
and T.c-id = "15-415"

Carnegie Mellon

15-415 - C. Faloutsos

renaming - self-join

• self -joins: find Tom's grandparent(s)

D-id C-id Mary Tom Peter Mary Peter Peter	PC			PC	
Peter Mary Peter Mary	p-id	c-id		p-id	c-id
	Mary	Tom	_ /	Mary	(Tom)
	Peter	Mary		Peter	Mary
John Tom John (Tom)	John	Tom		John	(Tom)

Carnegie Mel

15-415 - C. Faloutsos

renaming - self-join

find grandparents of "Tom" (PC(p-id, c-id))
select gp.p-id
from PC as gp, PC
where gp.c-id= PC.p-id
and PC.c-id = "Tom"

Carnegie Mellon

15-415 - C. Faloutsos

renaming - theta join

find course names with more units than 15-415

select c1.c-name

from class as c1, class as c2

where c1.units > c2.units

and c2.c-id = "15-415"

Carnegie Mellon 15-415 - C. Faloutsos 22

renaming - theta join

find course names with more units than 15-415

select c1.c-name

from class as c1, class as c2

where c1.units > c2.units

and c2.c-id = "15-415"

Carnegie Mellon

15-415 - C. Faloutsos

find course names with more units than 15-415

select c1.name

from class as c1, class as c2 where c1.units > c2.units

and c2.c-id = "15-415"

 $\{t \mid \exists c 1 \in CLASS \mid \exists c 2 \in CLASS \mid c \mid c \mid [c - id] = 15 - 415 \land$

 $c \ 2[units] > c \ 1[units] \land$

t[c - name] = c2[c - name]

Carnegie Mellon

15-415 - C. Faloutsos

24

CMU - 15-415 4

Overview - detailed - SQL

- DML
 - select, from, where
 - set operations
 - ordering
- aggregate functions
- nested subqueries
- other parts: DDL, embedded SQL, auth etc

Carnegie Mellon

15-415 - C. Faloutsos

set operations

find ssn of people taking both 15-415 and 15-413

TAKES		
SSN	c-id	grade
123	15-413	Α
234	15-413	В

arnegie Mellon 15-415 - C. Faloutsos

set operations

find ssn of people taking both 15-415 and 15- $\,$

413

select ssn

from takes

where c-id="15-415" and

.: c-id="15-413".

Carnegie Mellor

- C. Faloutsos

set operations

find ssn of people taking both 15-415 and 15-

(select ssn from takes where c-id="15-415")

intersect

(select ssn from takes where c-id="15-413")

other ops: union, except

Carnegie Mellon 15-415 - C. Faloutsos 28

Overview - detailed - SQL

- DML
 - select, from, where
 - set operations
 - ordering
 - aggregate functions
 - nested subqueries
- other parts: DDL, embedded SQL, auth etc

Carnegie Mellon

15-415 - C. Faloutsos

Ordering

find student records, sorted in name order

select *

from student

where

Carnegie Mellon

15-415 - C. Faloutsos

30

CMU - 15-415 5

Ordering

find student records, sorted in name order

select *

from student

order by name asc

asc is the default

15-415 - C. Faloutsos

Ordering

find student records, sorted in name order; break ties by reverse ssn

select *

from student

order by name, ssn desc

15-415 - C. Faloutsos

Overview - detailed - SQL

- DML
 - select, from, where
 - set operations
 - ordering
 - aggregate functions
 - nested subqueries
- other parts: DDL, embedded SQL, auth etc

Aggregate functions

find avg grade, across all students

select ??

from takes

SSN	c-id	grade
123	15-413	4
234	15-413	3

Carnegie Mellon

Aggregate functions

find avg grade, across all students

select avg(grade) from takes

 SSN
 c-id
 grade

 123
 15-413
 4
 234 15-413

35

- result: a single number
- Which other functions?

Carnegie Mellon

15-415 - C. Faloutsos

Aggregate functions

• A: sum count min max (std)

Carnegie Mellon

15-415 - C. Faloutsos

36



find total number of enrollments select count(*)

from takes

SSN c-id grade 123 15-413 234 15-413

15-415 - C. Faloutson

Aggregate functions

find total number of students in 15-415

select count(*)

from takes

where c-id="15-415"

SN_	c-id	grade
123	15-413	4
234	15-413	3

15-415 - C. Faloutso

Aggregate functions

find total number of students in each course select count(*)

from takes

where ???

SSN c-id grade 123 15-413 234 15-413

Aggregate functions

find total number of students in each course

select c-id, count(*)

from takes

group by c-id

SSN c-id grade 123 15-413 234 15-413

c-id count 15-413

Aggregate functions

find total number of students in each course select c-id, count(*)

from takes

group by c-id order by c-id

SSN c-id grade 123 15-413 234 15-413

41

<u>c-id</u> count 15-413 2

Carnegie Mellon

15-415 - C. Faloutsos

Aggregate functions

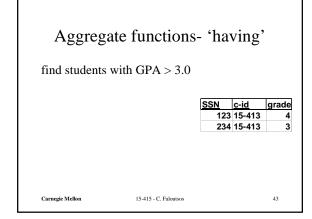
find total number of students in each course, and sort by count, decreasing

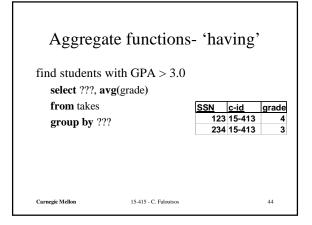
select c-id, count(*) as pop SSN c-id grade 123 15-413 from takes 234 15-413 group by c-id order by pop desc

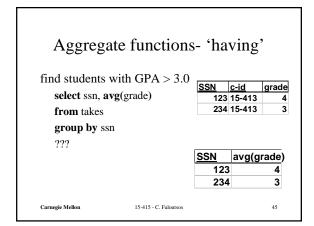
Carnegie Mellon

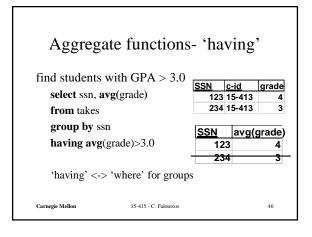
15-415 - C. Faloutsos

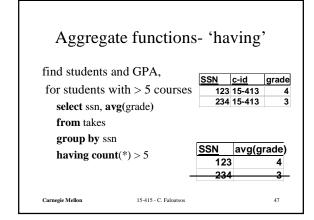
42











Overview - detailed - SQL

• DML

- select, from, where

- set operations

- ordering

- aggregate functions

- nested subqueries

• other parts: DDL, embedded SQL, auth etc