### Database Systems CSE 414

Lectures 5: Grouping & Aggregation

CSE 414 - Spring 2016

### **Announcements**

- · HW2 is out
  - due next Tuesday 11pm
  - same format as HW1
  - uses joins, aggregation, grouping
- WQ2 due Sunday 11pm

CSE 414 - Spring 2016

2

### Outline

- · Last time:
  - outer joins
  - how to aggregate over all rows
- Grouping & aggregations (6.4.3 6.4.6)

CSE 414 - Spring 2016

3

### Aggregation

Purchase(product, price, quantity)

Find number of bagels sold for more than \$1

SELECT Sum(quantity) as TotalSold

FROM Purchase

WHERE price > 1 and product = 'bagel'

CSE 414 - Spring 2017

## **Grouping and Aggregation**

Purchase(product, price, quantity)

Find number sold for more than \$1 for each product

SELECT product, Sum(quantity)
FROM Purchase
WHERE price > 1
GROUP BY product

Let's see what this means...

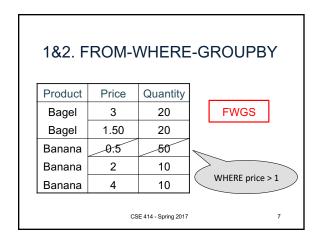
CSE 414 - Spring 2017

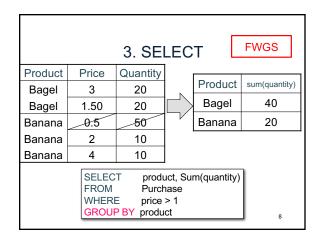
## **Grouping and Aggregation**

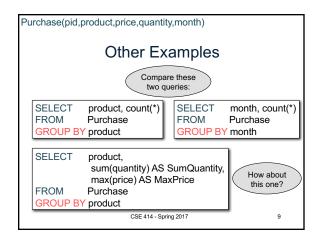
- 1. Compute the FROM and WHERE clauses.
- 2. Group by the attributes in the GROUP BY
- 3. Compute the SELECT clause: grouped attributes and aggregates.

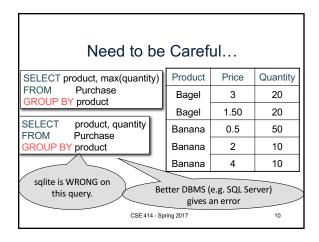
**FWGS** 

CSE 414 - Spring 2017



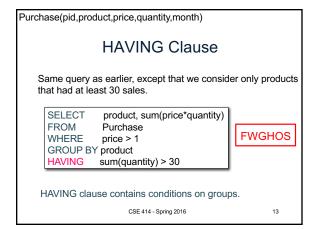


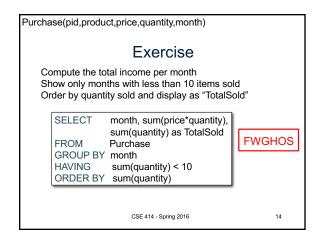










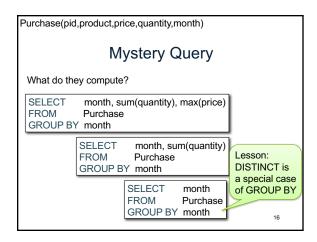


### WHERE vs HAVING

- · WHERE condition is applied to individual rows
  - The rows may or may not contribute to the aggregate
  - No aggregates allowed here
- · HAVING condition is applied to the entire group
  - Entire group is returned, or not at all
  - May use aggregate functions in the group

CSE 414 - Spring 2016

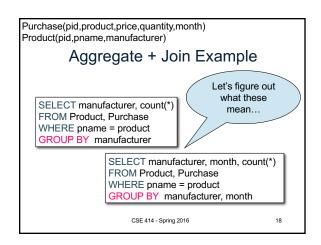
15



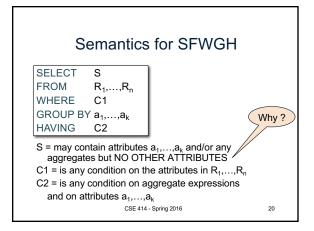
## Aggregates and Joins

```
create table Product(
   pid int primary key,
   pname varchar(15),
   manufacturer varchar(15));

insert into product values(1, 'bagel', 'Sunshine Co.');
insert into product values(2, 'banana', 'BusyHands');
insert into product values(3, 'gizmo', 'GizmoWorks');
insert into product values(4, 'gadget', 'BusyHands');
insert into product values(5, 'powerGizmo', 'PowerWorks');
CSE414-Spring 2016
17
```



## Nested Loop Semantics for SFW SELECT x1.a1, x2.a2, ... xm.am FROM R1 as x1, R2 as x2, ... Rm as xm WHERE Cond for x1 in R1: for x2 in R2: ... for xm in Rm: if Cond(x1, x2...): output(x1.a1, x2.a2, ... xm.am)



### Semantics for SFWGH

SELECT S
FROM R<sub>1</sub>,...,R<sub>n</sub>
WHERE C1
GROUP BY a<sub>1</sub>,...,a<sub>k</sub>
HAVING C2

### Evaluation steps:

- 1. Evaluate FROM-WHERE using Nested Loop Semantics
- 2. Group by the attributes a<sub>1</sub>,...,a<sub>k</sub>
- 3. Apply condition C2 to each group (may have aggregates)
- 4. Compute aggregates in S and return the result

  CSE 414 Spring 2016

  21

### Semantics for SFWGH

SELECT S FROM R<sub>1</sub>,...,R<sub>n</sub> WHERE C1 GROUP BY a<sub>1</sub>,...,a<sub>k</sub> HAVING C2

Execution order:

Evaluation steps:

- 1. Evaluate FROM-WHERE using Nested Loop Semantics
- 2. Group by the attributes a<sub>1</sub>,...,a<sub>k</sub>
- 3. Apply condition C2 to each group (may have aggregates)
- 4. Compute aggregates in S and return the result

  CSE 414 Spring 2016

Purchase(pid,product,price,quantity,month)
Product(pid,pname,manufacturer)

Aggregate + Join Example

SELECT manufacturer, count(\*)
FROM Product, Purchase
WHERE pname = product
GROUP BY manufacturer

SELECT manufacturer, month, count(\*)
FROM Product, Purchase
WHERE pname = product
GROUP BY manufacturer, month, count(\*)
FROM Product, Purchase
WHERE pname = product
GROUP BY manufacturer, month

## Empty Groups In the result of a group by query, there is one row per group in the result No group can be empty! In particular, count(\*) is never 0 SELECT manufacturer, count(\*) FROM Product, Purchase WHERE pname = product GROUP BY manufacturer

# Empty Group Solution: Outer Join SELECT manufacturer, count(quantity) FROM Product LEFT OUTER JOIN Purchase ON pname = product GROUP BY manufacturer Why not count(\*)? CSE 414-Spring 2016 25

Purchase(pid,product,price,quantity,month)
Product(pid,pname,manufacturer)

Exercise:

Find all manufacturers with more than 10 items sold.
Return manufacturer name and number of items sold.

SELECT manufacturer, sum(quantity)
FROM Product, Purchase
WHERE pname = product
GROUP BY manufacturer
HAVING sum(quantity) > 10

CSE 414 - Spring 2016

Purchase(pid,product,price,quantity,month)
Product(pid,pname,manufacturer)

### Exercise:

Find all manufacturers with more than 1 distinct product sold Return the name of the manufacturer and number of distinct products sold

SELECT manufacturer, count(distinct product)

FROM Product, Purchase

WHERE pname = product

GROUP BY manufacturer

HAVING count(distinct product) > 1

CSE 414 - Spring 2016

Purchase(pid,product,price,quantity,month)
Product(pid,pname,manufacturer)

### Exercise:

Find all products with more than 2 purchases Return the name of the product and max price it was sold

SELECT pname max(price) FROM Product, Purchase

WHERE pname = product GROUP BY pname

HAVING COUNT(\*) > 2

CSE 414 - Spring 2016

28

Purchase(pid,product,price,quantity,month)
Product(pid,pname,manufacturer)

### Exercise:

Find all manufacturers with at least 5 purchases in one month Return manufacturer name, month, and number of items sold

SELECT manufacturer, month, sum(quantity)

FROM Product, Purchase

GROUP BY manufacturer, month

HAVING count(\*) > 2

CSE 414 - Spring 2016

29

27