Today

- Group By (cont)
- Subqueries (intro)

Assignments

- HW5 due
- HW 6 out (due Tues)
**Group By attributes**

What can appear in a SELECT clause under GROUP BY?

- the grouping attributes (attributes in GROUP BY clause)
- or aggregate operators (which are applied to the group)

This query is not legal

```
SELECT name
FROM customer
GROUP BY salesperson;
```

Q: Why not?

- name is not a grouping attribute, and is not an aggregate
- there could be many names for each group!

Multiple attributes can appear in a GROUP BY

- Same rule applies: No two rows in a group have different values for attributes in the GROUP BY

What can appear in a HAVING clause?

- Comparisons to only those attributes in the GROUP BY clause
- Comparisons to expressions (like aggregates) that result in a single value for the group

HAVING clause without a GROUP BY

- Treats result of FROM and WHERE as a single group

Q: How does this differ to a WHERE clause?

- WHERE applies to each row, not the whole group
Example Tables

Customer(number, name, address, c_rating, c_amount, c_balance, salesperson)

Salesperson(number, name, address, office)

<table>
<thead>
<tr>
<th>number</th>
<th>name</th>
<th>...</th>
<th>salesperson</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>mary</td>
<td>...</td>
<td>5</td>
</tr>
<tr>
<td>102</td>
<td>john</td>
<td>...</td>
<td>8</td>
</tr>
<tr>
<td>103</td>
<td>dave</td>
<td>...</td>
<td>NULL</td>
</tr>
<tr>
<td>106</td>
<td>sam</td>
<td>...</td>
<td>5</td>
</tr>
<tr>
<td>107</td>
<td>oliver</td>
<td>...</td>
<td>5</td>
</tr>
<tr>
<td>109</td>
<td>susan</td>
<td>...</td>
<td>2</td>
</tr>
<tr>
<td>110</td>
<td>luis</td>
<td>...</td>
<td>8</td>
</tr>
</tbody>
</table>

- where Customer.salesperson is a FK to Salesperson.number
Subqueries

A subquery is a “nested” query

• Primarily used within WHERE and FROM clauses
• Can also be used in SELECT and HAVING clauses
• We’ll primarily look at the WHERE case

```sql
SELECT c1.number, c1.name
FROM customer c1
WHERE c1.c_rating = (SELECT MAX(c2.rating)
FROM customer c2);
```

• the subquery is the inner query
• the rest is the outer query

Q: What does the inner query return?
   – a single value (the max credit rating)

Q: What does the outer query return?
   – name and number of customers with the highest credit ratings
How could we evaluate this query?

1. start with the FROM clause in the outer query
2. take a row from the customer table
3. check if the row satisfies the WHERE clause
   – which involves evaluating the inner query
4. if so, output the number and name
5. etc.

- In this case the inner query always returns the same result
  – and so only needs to be evaluated once!
- The inner query is **NOT correlated**
  – it does not use any attributes from the outer query (c1)
WHERE subquery comparators

Single-valued comparisons

```
SELECT c1.number, c1.name
FROM customer c1
WHERE c1.c_rating = (SELECT MAX(c2.rating)
                   FROM customer c2);
```

- The comparator can be any of the six standard ones:
  
  `<`, `<=`, `=`, `!=`, `>=`, `>`

- For these comparators, the inner query must return a single value!

ANY/ALL comparisons

```
SELECT s.number, s.name
FROM salesperson s
WHERE s.name = ANY (SELECT c.name FROM customer c);
```

- For ANY
  
  - subquery can return more than one answer
  - the expression (here =) must be true for at least one subquery answer
  - “SOME” is equivalent to ANY

Q: What does this query return? Can it be written without a subquery?

  * Salespersons that are also customers
  * Yes!

Q: Rewrite this query ...

- For ALL, expression must be true for every subquery answer
More examples

```
SELECT s.name
FROM salesperson s
WHERE s.number = ANY (SELECT c.salesperson
                        FROM customer c
                        WHERE c.c_rating = 3);
```

Q: What does this query return?

– Salespeople that have at least one customer with a credit rating of 3

Q: Can this query be rewritten without subqueries? If so, rewrite it.

```
SELECT s.name
FROM salesperson s
WHERE s.number = ALL (SELECT c.salesperson
                       FROM customer c
                       WHERE c.c_rating = 3);
```

Q: What does this query return?

– Salespeople that have all the customers with a credit rating of 3

Q: Can this be rewritten w/out subqueries (given SQL we’ve seen)? Try it.