Joins

Lecture 6
Subsections 4.3.2, 5.1.5 - 5.1.6

Robb T. Koether

Hampden-Sydney College

Mon, Jan 27, 2014

Outline

Implicit Joins

- We can join two or more relations together in the `FROM` clause simply by listing them.
- MySQL will create a tuple *for every combination* of tuples, taken one from each relation.
- For example, if we join relations R_1, R_2, and R_3 and they have 5, 9, and 16 tuples, respectively, then the join will have $5 \times 9 \times 16 = 720$ tuples.

Implicit Joins

- To keep the numbers small, we will switch from the `company` database to a smaller database, `hardware`.
- Tables:
  - catalog – Lists all items for sale
  - orders – Lists all orders made by customers
  - customers – Lists all customers
  - cust_orders – Matches customers with orders

Implicit Joins

The Hardware Database

cat_no	item	price
1234	Hammer	12.98
4567	Wrench	5.49
6789	Pliers	9.98
2345	Sander	5.99
3456	Drill	15.79

catalog

order_no	cat_no	quant
222	1234	2
444	8765	3
555	4567	2
555	6789	1
777	5432	1
888	1234	1
222	4567	2

orders

order_no	cust_no
222	33333
444	98765
555	33333
777	98765
888	54321

cust_orders

Implicit Joins

The Hardware Database

cust_no	name	street	city	state	zip
54321	Joe Smith	123 Main St.	Farmville	VA	23901
33333	Lisa Jackson	444 Broadway	New York	NY	10004
98765	Bob Jones	999 Ocean Dr.	Los Angeles	CA	90230
88888	Betty Mason	3456 Woodland Pl.	Seattle	WA	98103

customers

Implicit Joins

- We can join the `catalog`, `orders`, `customers`, `cust_orders` tables together.
- There are 5 catalog items, 7 order entries, 4 customers, and 5 cust_order tuples, so this table has 700 tuples.
- It matches every catalog item with every order and with every customer and with every customer/order entry.
- This is probably not what we wanted.

Implicit Joins

`catalog` Joined with `orders`

```
SELECT * FROM catalog, orders;
+--------+--------+-------+----------+--------+-------+
| cat_no | item   | price | order_no | cat_no | quant |
+--------+--------+-------+----------+--------+-------+
| 1234   | Hammer | 12.98 |      222 | 1234   |     2 |
| 4567   | Wrench |  5.49 |      222 | 1234   |     2 |
| 6789   | Pliers |  9.98 |      222 | 1234   |     2 |
| 2345   | Sander |  5.99 |      222 | 1234   |     2 |
| 3456   | Drill  | 15.79 |      222 | 1234   |     2 |
| 1234   | Hammer | 12.98 |      444 | 8765   |     3 |
| 4567   | Wrench |  5.49 |      444 | 8765   |     3 |
| 6789   | Pliers |  9.98 |      444 | 8765   |     3 |
| 2345   | Sander |  5.99 |      444 | 8765   |     3 |
| 3456   | Drill  | 15.79 |      444 | 8765   |     3 |
| 1234   | Hammer | 12.98 |      555 | 4567   |     2 |
| 4567   | Wrench |  5.49 |      555 | 4567   |     2 |
| 6789   | Pliers |  9.98 |      555 | 4567   |     2 |
| 2345   | Sander |  5.99 |      555 | 4567   |     2 |
| 3456   | Drill  | 15.79 |      555 | 4567   |     2 |
| 1234   | Hammer | 12.98 |      555 | 6789   |     1 |
| 4567   | Wrench |  5.49 |      555 | 6789   |     1 |
| 6789   | Pliers |  9.98 |      555 | 6789   |     1 |
| 2345   | Sander |  5.99 |      555 | 6789   |     1 |
| 3456   | Drill  | 15.79 |      555 | 6789   |     1 |
| 1234   | Hammer | 12.98 |      777 | 5432   |     1 |
| 4567   | Wrench |  5.49 |      777 | 5432   |     1 |
   :        :        :          :      :          :
| 3456   | Drill  | 15.79 |      222 | 4567   |     2 |
+--------+--------+-------+----------+--------+-------+
```

Selections

- To avoid this undesirable outcome, we use the `WHERE` clause to match tuples that belong together.
- The catalog's `cat_no` and the order's `cat_no` should agree.
- We could use the `WHERE` clause as follows.

```
WHERE catalog.cat_no = orders.cat_no
```

- Note that the attribute had to be associated with the table. Why?

Implicit Joins

`catalog` Joined with `orders`

```
SELECT * FROM catalog, orders
WHERE catalog.cat_no = orders.cat_no;
```

cat_no	item	price	order_no	cat_no	quant
1234	Hammer	12.98	222	1234	2
1234	Hammer	12.98	888	1234	1
4567	Wrench	5.49	555	4567	2
4567	Wrench	5.49	222	4567	2
6789	Pliers	9.98	555	6789	1

Implicit Joins

`catalog` Joined with `orders`

```
SELECT * FROM catalog AS c, orders AS o
WHERE c.cat_no = o.cat_no;
+--------+--------+-------+----------+--------+-------+
| cat_no | item   | price | order_no | cat_no | quant |
+--------+--------+-------+----------+--------+-------+
| 1234   | Hammer | 12.98 |      222 | 1234   |     2 |
| 1234   | Hammer | 12.98 |      888 | 1234   |     1 |
| 4567   | Wrench |  5.49 |      555 | 4567   |     2 |
| 4567   | Wrench |  5.49 |      222 | 4567   |     2 |
| 6789   | Pliers |  9.98 |      555 | 6789   |     1 |
+--------+--------+-------+----------+--------+-------+
```

- In such circumstances, it is convenient to give the tables aliases.
- Note that `cat_no` appears twice in the output.

Outline

The JOIN Operator

- There is a JOIN operator that will join tables in various ways, depending on the kind of join used.
  - Join
  - Natural join
  - Left join
  - Right join

Join

The `JOIN` Operator

`table_name` `JOIN` *`table_name`*

- The `JOIN` operator is the same as the comma operator that we saw earlier.
- It pairs every tuple from the first table with every tuple from the second table.
- This is the same as the *Cartesian product* of the two tables.

Join

Join on Catalog Number

```
SELECT *
FROM catalog JOIN orders
ON catalog.cat_no = orders.cat_no;
+--------+--------+-------+----------+--------+-------+
| cat_no | item   | price | order_no | cat_no | quant |
+--------+--------+-------+----------+--------+-------+
| 1234   | Hammer | 12.98 |      222 | 1234   |     2 |
| 1234   | Hammer | 12.98 |      888 | 1234   |     1 |
| 4567   | Wrench |  5.49 |      222 | 4567   |     2 |
| 4567   | Wrench |  5.49 |      555 | 4567   |     2 |
| 6789   | Pliers |  9.98 |      555 | 6789   |     1 |
+--------+--------+-------+----------+--------+-------+
```

- We can use the `ON` clause (or `WHERE` clause) to match tuples on specified attributes.

Join

- To get the table that we want, we need to
  - Eliminate the duplicate `cat_no` column.
  - Add a `cost` column (`price` × `quant`).
  - List attributes in order `order_no`, `cat_no`, `item`, `price`, `quant`, and `cost`.
  - Arrange the orders by order number and by catalog number within orders.

Join

The Desired Table

```
SELECT order_no, catalog.cat_no, item, price,
quant, price*quant AS cost
FROM catalog JOIN orders
ON catalog.cat_no = orders.cat_no
ORDER by order_no, catalog.cat_no;
```

order_no	cat_no	item	price	quant	cost
222	1234	Hammer	12.98	2	25.96
222	4567	Wrench	5.49	2	10.98
555	4567	Wrench	5.49	2	10.98
555	6789	Pliers	9.98	1	9.98
888	1234	Hammer	12.98	1	12.98

- Note that `cat_no` had to be qualified by its table name to avoid ambiguity.

Outline

Join

- The `NATURAL JOIN` does not use the ON clause.
- The natural join matches tuples based on their attribute names.
- Furthermore, the matching attributes will appear *only once* in the output.
- This shows the benefit of using the same attribute name for attributes that represent the same thing.

Join

Natural Join of `catalog` and `orders`

```
SELECT * FROM catalog NATURAL JOIN orders;
+--------+--------+-------+----------+-------+
| cat_no | item   | price | order_no | quant |
+--------+--------+-------+----------+-------+
| 1234   | Hammer | 12.98 |      222 |     2 |
| 1234   | Hammer | 12.98 |      888 |     1 |
| 4567   | Wrench |  5.49 |      555 |     2 |
| 4567   | Wrench |  5.49 |      222 |     2 |
| 6789   | Pliers |  9.98 |      555 |     1 |
+--------+--------+-------+----------+-------+
```

Join

- Now we can more easily get the table that we wanted, by using the natural join.

Join

Natural Join

```
SELECT order_no, cat_no, item, price, quant,
price*quant AS cost
FROM catalog NATURAL JOIN orders
ORDER by order_no, cat_no;
```

```
+----------+--------+--------+-------+-------+-------+
| order_no | cat_no | item   | price | quant | cost  |
+----------+--------+--------+-------+-------+-------+
|      222 | 1234   | Hammer | 12.98 |     2 | 25.96 |
|      222 | 4567   | Wrench |  5.49 |     2 | 10.98 |
|      555 | 4567   | Wrench |  5.49 |     2 | 10.98 |
|      555 | 6789   | Pliers |  9.98 |     1 |  9.98 |
|      888 | 1234   | Hammer | 12.98 |     1 | 12.98 |
+----------+--------+--------+-------+-------+-------+
```

Join

Natural Join

```
SELECT order_no, cust_no, name, cat_no, item, price, quant, price*quant AS cost
FROM catalog NATURAL JOIN orders NATURAL JOIN cust_orders NATURAL JOIN customers
ORDER BY order_no, cat_no;
+----------+---------+--------------+--------+--------+-------+-------+-------+
| order_no | cust_no | name         | cat_no | item   | price | quant | cost  |
+----------+---------+--------------+--------+--------+-------+-------+-------+
|      222 | 33333   | Lisa Jackson | 1234   | Hammer | 12.98 |     2 | 25.96 |
|      222 | 33333   | Lisa Jackson | 4567   | Wrench |  5.49 |     2 | 10.98 |
|      555 | 33333   | Lisa Jackson | 4567   | Wrench |  5.49 |     2 | 10.98 |
|      555 | 33333   | Lisa Jackson | 6789   | Pliers |  9.98 |     1 |  9.98 |
|      888 | 54321   | Joe Smith    | 1234   | Hammer | 12.98 |     1 | 12.98 |
+----------+---------+--------------+--------+--------+-------+-------+-------+
```

- We can “natural join” more than two tables.

Natural Join

- By the way, what happened to orders 444 and 777?

Outline

Left Join

- The left join:
  - For every tuple in the left table that matches a tuple in the right table, the tuples are joined.
  - For every tuple in the left table that does not match a tuple in the right table, the left tuple is joined with NULL values.
- The `LEFT JOIN` operator requires the `ON` clause.

Left Join

LEFT JOIN Example

```
SELECT * FROM catalog LEFT JOIN orders
ON catalog.cat_no = orders.cat_no;
+--------+--------+-------+----------+--------+-------+
| cat_no | item   | price | order_no | cat_no | quant |
+--------+--------+-------+----------+--------+-------+
| 1234   | Hammer | 12.98 |      222 | 1234   |     2 |
| 1234   | Hammer | 12.98 |      888 | 1234   |     1 |
| 4567   | Wrench |  5.49 |      222 | 4567   |     2 |
| 4567   | Wrench |  5.49 |      555 | 4567   |     2 |
| 6789   | Pliers |  9.98 |      555 | 6789   |     1 |
| 2345   | Sander |  5.99 |     NULL | NULL   |  NULL |
| 3456   | Drill  | 15.79 |     NULL | NULL   |  NULL |
+--------+--------+-------+----------+--------+-------+
```

Right Join

- The `right join` is exactly like the left join, except that the roles of the tables are reversed.
  - For every tuple in the right table that matches a tuple in the left table, the tuples are joined.
  - For every tuple in the right table that does not match a tuple in the left table, the right tuple is joined with NULL values.
- The `RIGHT JOIN` operator requires the `ON` clause.

Right Join

RIGHT JOIN Example

```
SELECT * FROM catalog RIGHT JOIN orders
ON catalog.cat_no = orders.cat_no;
+--------+--------+-------+----------+--------+-------+
| cat_no | item   | price | order_no | cat_no | quant |
+--------+--------+-------+----------+--------+-------+
| 4567   | Wrench |  5.49 |      222 | 4567   |     2 |
| 1234   | Hammer | 12.98 |      222 | 1234   |     2 |
| NULL   | NULL   |  NULL |      444 | 8765   |     3 |
| 4567   | Wrench |  5.49 |      555 | 4567   |     2 |
| 6789   | Pliers |  9.98 |      555 | 6789   |     1 |
| NULL   | NULL   |  NULL |      777 | 5432   |     1 |
| 1234   | Hammer | 12.98 |      888 | 1234   |     1 |
+--------+--------+-------+----------+--------+-------+
```

Right Join

Reverse the Order of the Tables

```
SELECT * FROM orders LEFT JOIN catalog
ON catalog.cat_no = orders.cat_no;
+----------+--------+-------+--------+--------+-------+
| order_no | cat_no | quant | cat_no | item   | price |
+----------+--------+-------+--------+--------+-------+
|      222 | 4567   |     2 | 4567   | Wrench |  5.49 |
|      222 | 1234   |     2 | 1234   | Hammer | 12.98 |
|      444 | 8765   |     3 | NULL   | NULL   |  NULL |
|      555 | 4567   |     2 | 4567   | Wrench |  5.49 |
|      555 | 6789   |     1 | 6789   | Pliers |  9.98 |
|      777 | 5432   |     1 | NULL   | NULL   |  NULL |
|      888 | 1234   |     1 | 1234   | Hammer | 12.98 |
+----------+--------+-------+--------+--------+-------+
```

Outline

Querying Joins

- Write a query that will do the following.
  - Display the order number, catalog number, item, unit price, quantity, and cost for Joe Smith.
  - Display the names of all customers who have ordered Wrenches.
  - Display the names of all customers who ordered the same item as some other customer.

Outline

Assignment

Assignment

- Read Subsections 4.3.2, 5.1.5 - 5.1.6.