CGS 2545: Database Concepts
Spring 2012

SQL In-class Exercises – Part 1 Basic SQL
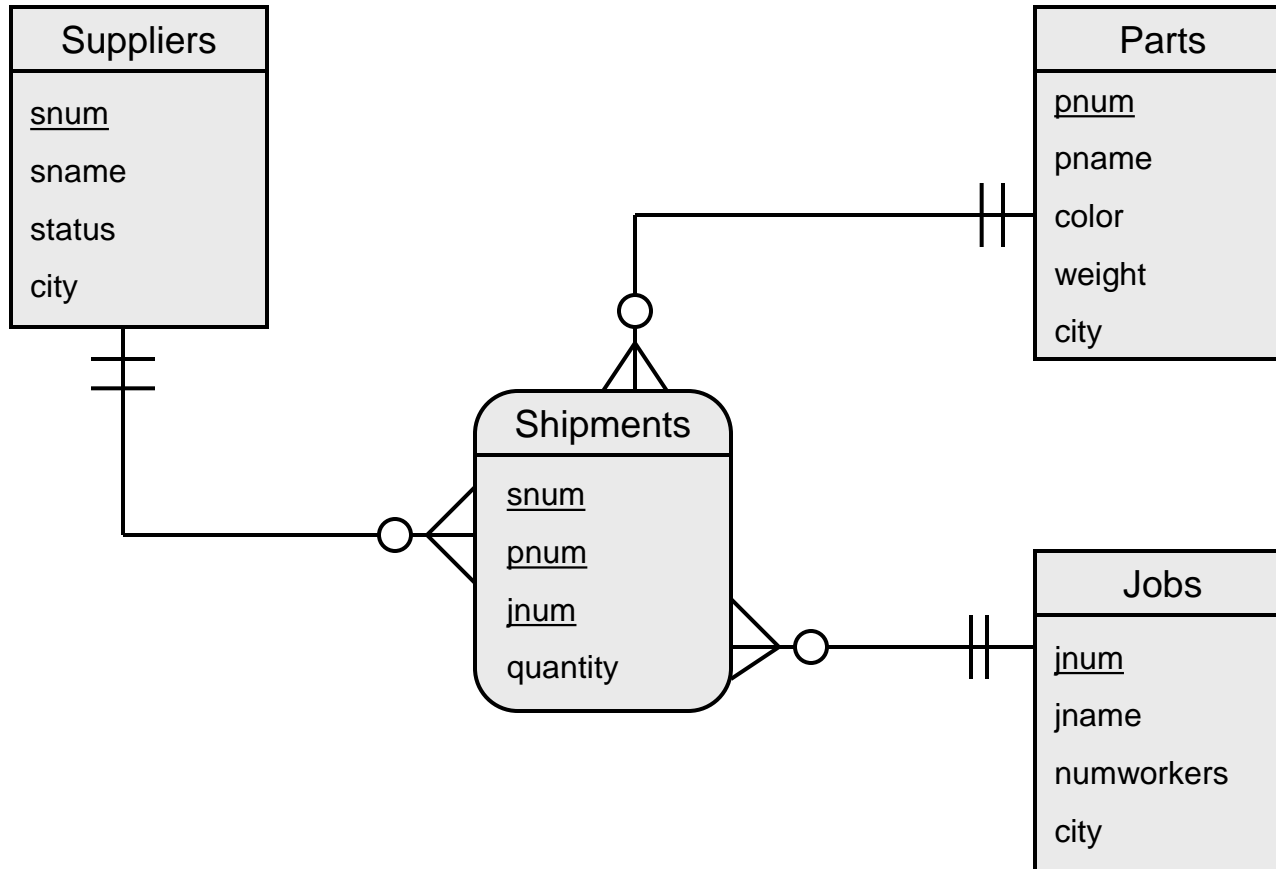
Instructor :   Dr. Mark Llewellyn
                    markl@cs.ucf.edu
                    HEC 236, 407-823-2790
        http://www.cs.ucf.edu/courses/cgs2545/spr2012

Department of Electrical Engineering and Computer Science
Computer Science Division
University of Central Florida

SQL In Class Exercises

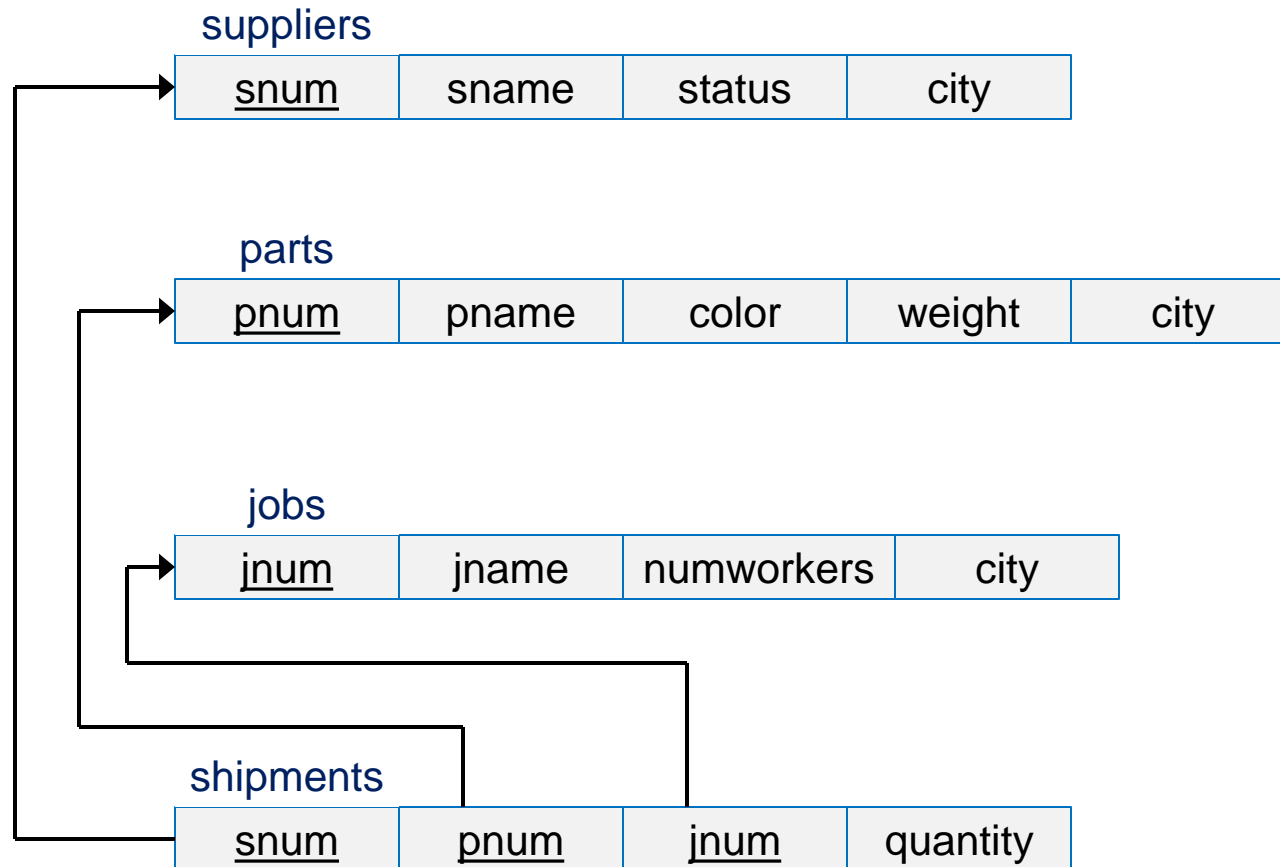- Use the following database scheme for problems 1-9 in this exercise.

Suppliers
snum
sname
status
city

Parts
pnum
pname
color
weight
city

Shipments
snum
pnum
jnum
quantity

Jobs
jnum
jname
numworkers
city

- Develop SQL expressions for each of the following queries:

SQL In Class Exercises

- The schema version of the database for problems 1-9.

suppliers
snum	sname	status	city

parts
pnum	pname	color	weight	city

jobs
jnum	jname	numworkers	city

shipments
snum	pnum	jnum	quantity

1. Create the table definition for the suppliers schema. Do not allow a supplier's information to be deleted if they have a shipment.

<u>Solution</u>

```
CREATE TABLE suppliers

(snum VARCHAR2(5) NOT NULL,

 sname VARCHAR2(25),

 status INTEGER,

 city VARCHAR2(20),

 CONSTRAINT supplier_PK PRIMARY KEY (snum)

    ON UPDATE RESTRICT);
```

2. Assuming that the tables for the parts and jobs were created in a similar fashion to that of the suppliers table on the previous page, create the table definition for the shipments schema.

Solution

```
CREATE TABLE shipments

(snum VARCHAR2(5) NOT NULL,

 pnum VARCHAR2(5) NOT NULL,

 jnum VARCHAR2(5) NOT NULL,

 quantity INTEGER,

 CONSTRAINT ship_PK PRIMARY KEY (snum, pnum, jnum),

 CONSTRAINT ship_FK1 FOREIGN KEY (snum) REFERENCES suppliers(snum),

 CONSTRAINT ship_FK2 FOREIGN KEY (pnum) REFERENCES parts(pnum),

 CONSTRAINT ship_FK3 FOREIGN KEY (jnum) REFERENCES jobs(jnum) );
```

3. Insert a new supplier's information into the suppliers table.

Solution

```
INSERT INTO suppliers VALUES
("S1","Kristy",14,"Orlando);
```

4. Delete from the shipment table every row where the quantity is less than 10.

<u>Solution</u>

```
DELETE FROM shipments

WHERE quantity < 10;
```

5. Update the suppliers table by modifying the status of every supplier whose current status is 10 by increasing the status by 5.

<u>Solution</u>

```
UPDATE suppliers

   SET status = status + 5

   WHERE status = 10;
```

6. Update the parts table by modifying the weight of part number 6 to its current weight + 20.

Solution

```
UPDATE parts

    SET weight = weight + 20

    WHERE pnum = 6;
```

7. Modify the data in the parts table so that every part that was blue is now colored green.

Solution

```
UPDATE parts

    SET color = "green"

    WHERE color = "blue";
```

8. List only the names of those suppliers who are located in Orlando.

Solution

```
SELECT sname

FROM suppliers

WHERE city = "Orlando";
```

9. List the part number for every part that is shipped by more than one supplier.

Solution

```
SELECT pnum

FROM shipments

GROUP BY pnum

HAVING COUNT (snum) > 1;
```

WHERE clause restricts by rows

HAVING clause restricts by groups

10. Find the average weight of all parts.

Solution

```
SELECT AVG(weight)

FROM parts;
```

11. For each part list the part number and the total quantity in which that part is shipped and order the results in descending order of the total quantity shipped. Name the total quantity shipped in the result as totalShipped.

Solution

```
SELECT pnum, SUM(quantity) AS totalShipped

FROM shipments

GROUP BY pnum

ORDER BY SUM(quantity) DESC;
```

12. List the supplier number and the total quantity of parts that supplier ships and group the results by supplier number in descending order of the total quantity supplied.

Solution

```
SELECT snum, sum(shipments.quantity)

     AS totalShipped

FROM shipments

GROUP BY snum

ORDER BY sum(quantity) DESC;
```