SQL Data Manipulation Language (DML) Questions

1. Let R=(A, B, C), S=(C, D, E) be two relational schema. Let q and r be relations (i.e., tables) on schema R; and s be a relation (i.e., a table) on schema S. Convert the following relational algebra queries to SQL.

(i) $q - r$

(ii) $\Pi_{A, C}(r) \bowtie \Pi_{C, D}(s)$

Solution:

(i)

```
SELECT * FROM q

EXCEPT

SELECT * FROM r;
```

(ii)

```
SELECT r.A, r.C, s.D

FROM r, s

WHERE r.C = s.C;
```

2. Consider the following schema containing airport flight information. Primary Keys are in bold.

FLIGHTS(**flno:integer**, from:string, to:string, distance:integer, departs:time, arrives:time)

AIRCRAFT(**aid:integer**, aname:string, cruisingrange:integer)

CERTIFIED(**eid:integer**, **aid:integer**)

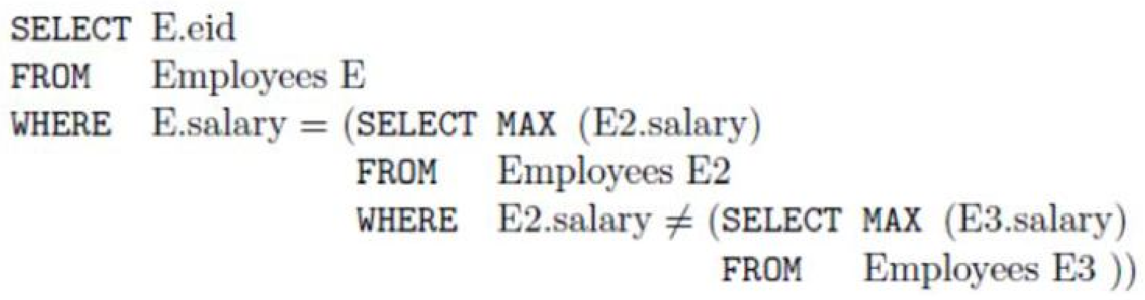
EMPLOYEES(**eid:integer**, ename:string, salary:integer)

Note that the Employees relation describes pilots and other kinds of employees as well; every pilot is certified for some aircraft (otherwise, he or she would not qualify as a pilot), and only pilots are certified to fly.

Give an SQL expression for the following query. Your solution should be only one SQL statement.

Find the eids of employees who make the second highest salary.

Solution:



```
SELECT E.eid
FROM   Employees E
WHERE  E.salary = (SELECT MAX (E2.salary)
                   FROM   Employees E2
                   WHERE  E2.salary ≠ (SELECT MAX (E3.salary)
                                       FROM   Employees E3 ))
```

3. Consider the following schema representing a database (primary keys are underlined).

PRODUCT(<u>model</u>, manufacturer, type)

PC(<u>model</u>, speed, ram, hd, price)

LAPTOP(<u>model</u>, speed, ram, hd, screen, price)

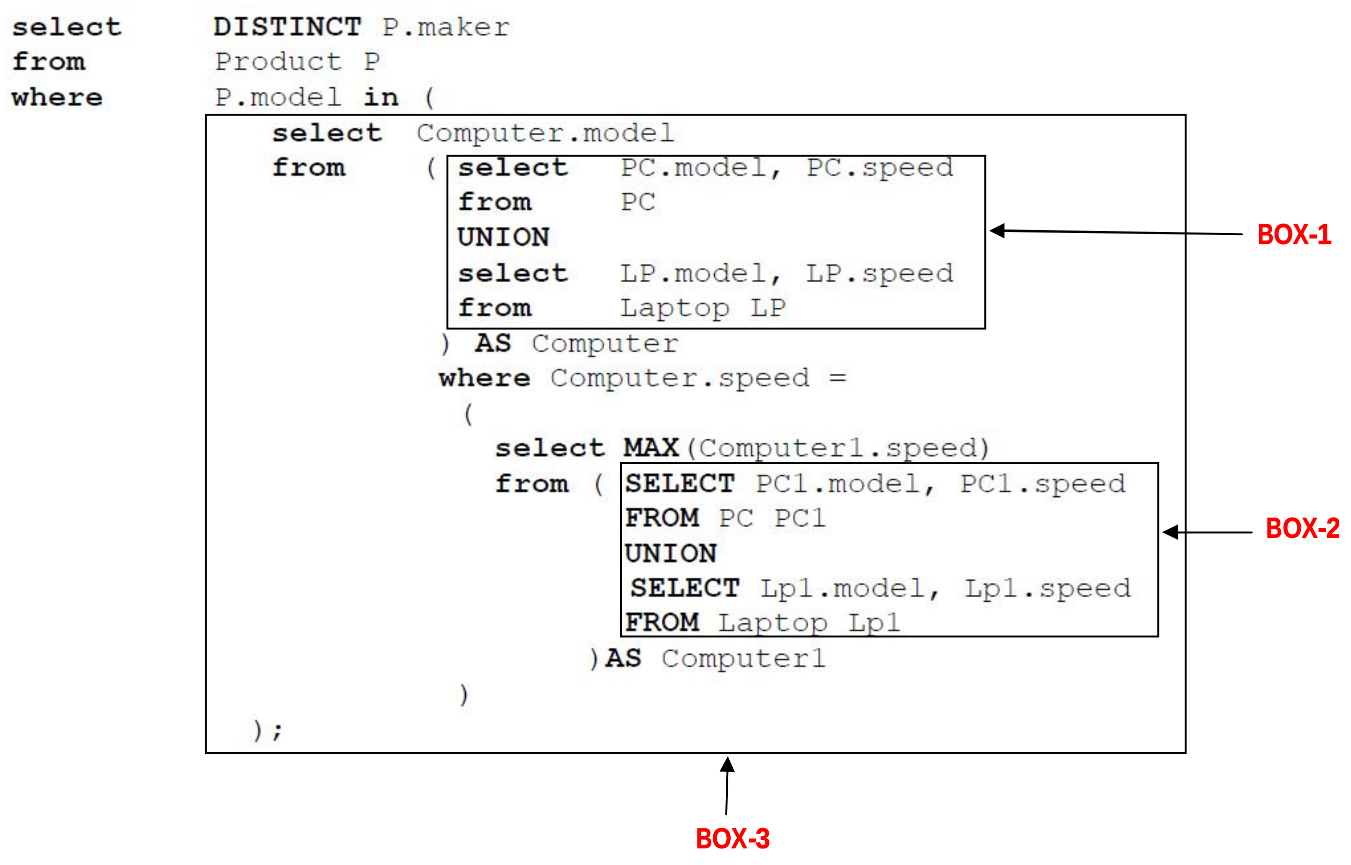
PRINTER(<u>model</u>, color, type, price)

A PRODUCT is either a PC, a LAPTOP or a PRINTER and must have a tuple in the corresponding table. There is a foreign key constraint on the model of PCs, Laptops and Printers referencing the primary key model of PRODUCT.

Express the following queries in SQL. Your solution should be only one SQL statement.

Find the manufacturer(s) of computer (PC or laptop) with the highest available speed.

Solution:



```
select      DISTINCT P.maker
from        Product P
where       P.model in (
        select  Computer.model
        from      ( select    PC.model, PC.speed
                    from      PC
                    UNION
                    select    LP.model, LP.speed
                    from      Laptop LP
                  ) AS Computer
                  where Computer.speed =
                   (
                      select MAX(Computer1.speed)
                      from ( SELECT PC1.model, PC1.speed
                             FROM PC PC1
                             UNION
                              SELECT Lp1.model, Lp1.speed
                             FROM Laptop Lp1
                           )AS Computer1
                   )
    );
```

Practice Questions **SQL (Week-9 Oct 15)** **CZ2007**

During the revision, I came up with one question:
what is the difference between operators IN and Some?
In my perspective, they both determine whether the tuple or rec in contained in relation.

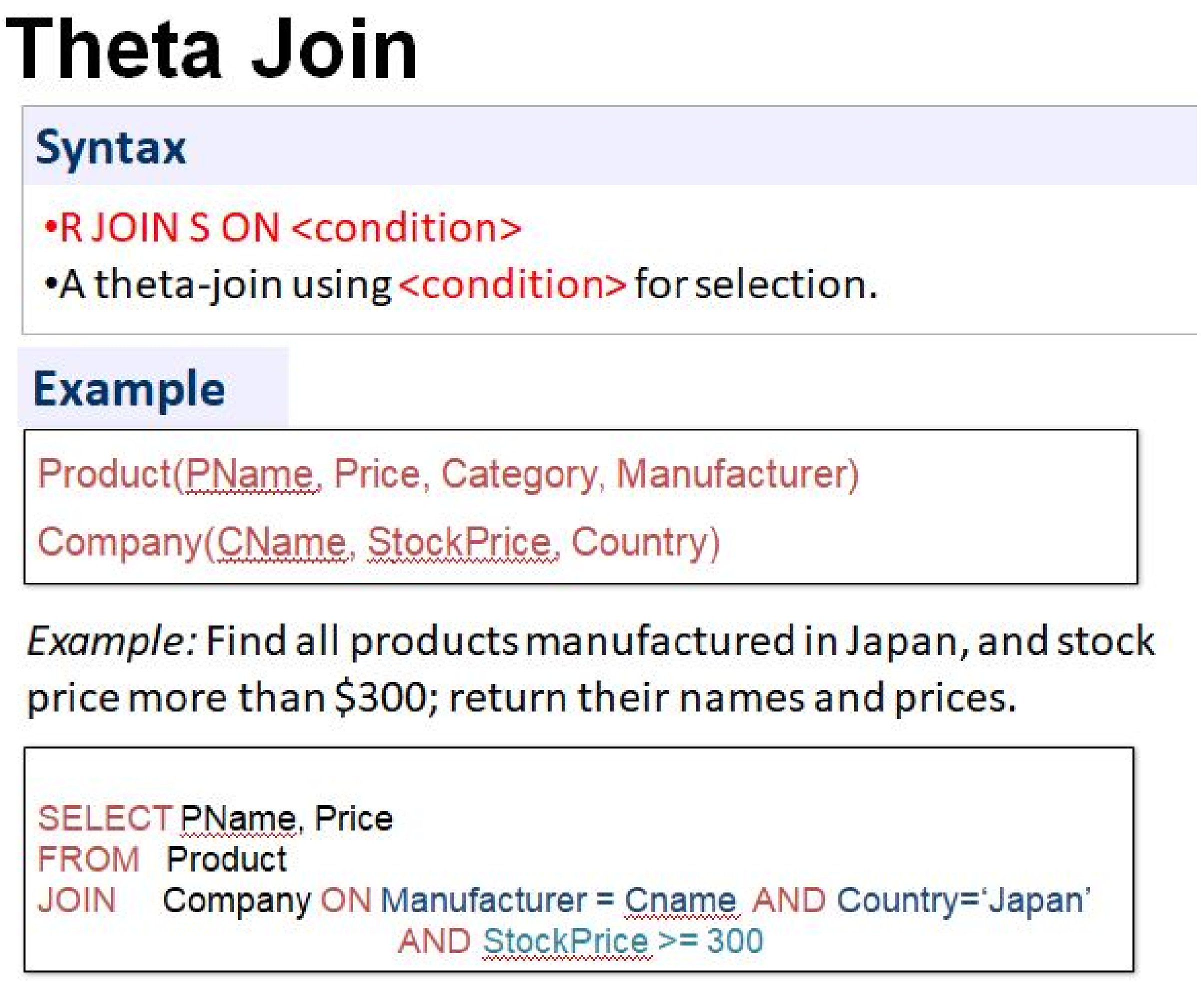
“In” is same as “SOME=”.

Sorry for sending the second email. Is it that SOME support comparison operation > or <, but not IN? e.g. price>SOME(....)

SOME can also support comparison as you mentioned.

And one more question, Inner Join and Theta Join, do they have difference in SQL? Since Theta Join is Crossing product of all the tables, then all the attributes seem to be maintained, it sounds like it is the same as Inner Join.

In theta join, you can specify extra join condition, such as >, <, etc.



Theta Join

Syntax

- R JOIN S ON <condition>
- A theta-join using <condition> for selection.

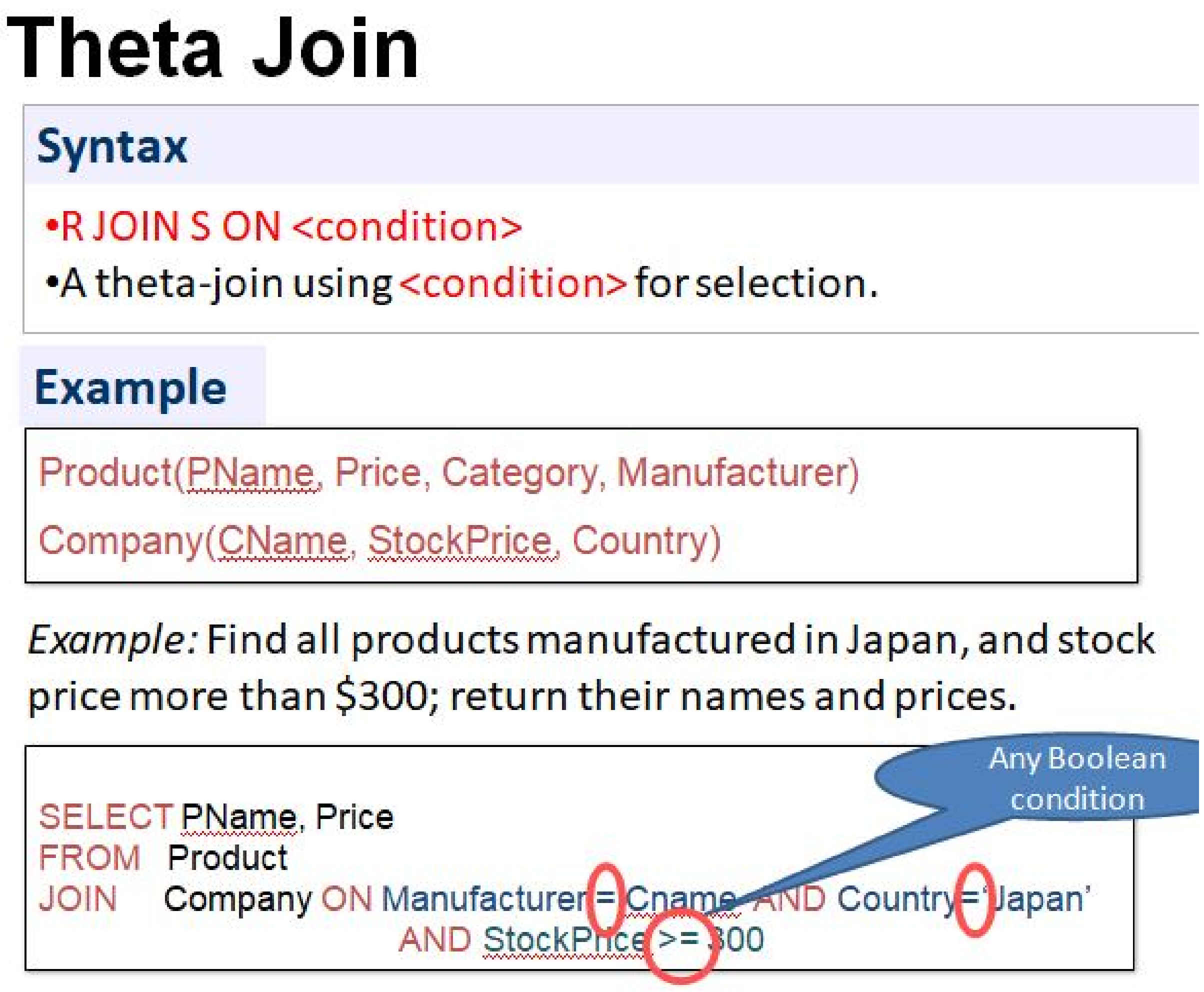
Example

Product(PName, Price, Category, Manufacturer)

Company(CName, StockPrice, Country)

Example: Find all products manufactured in Japan, and stock price more than $300; return their names and prices.

```
SELECT PName, Price
FROM   Product
JOIN    Company ON Manufacturer = Cname  AND Country='Japan'
                      AND StockPrice >= 300
```



Theta Join

Syntax

- R JOIN S ON <condition>
- A theta-join using <condition> for selection.

Example

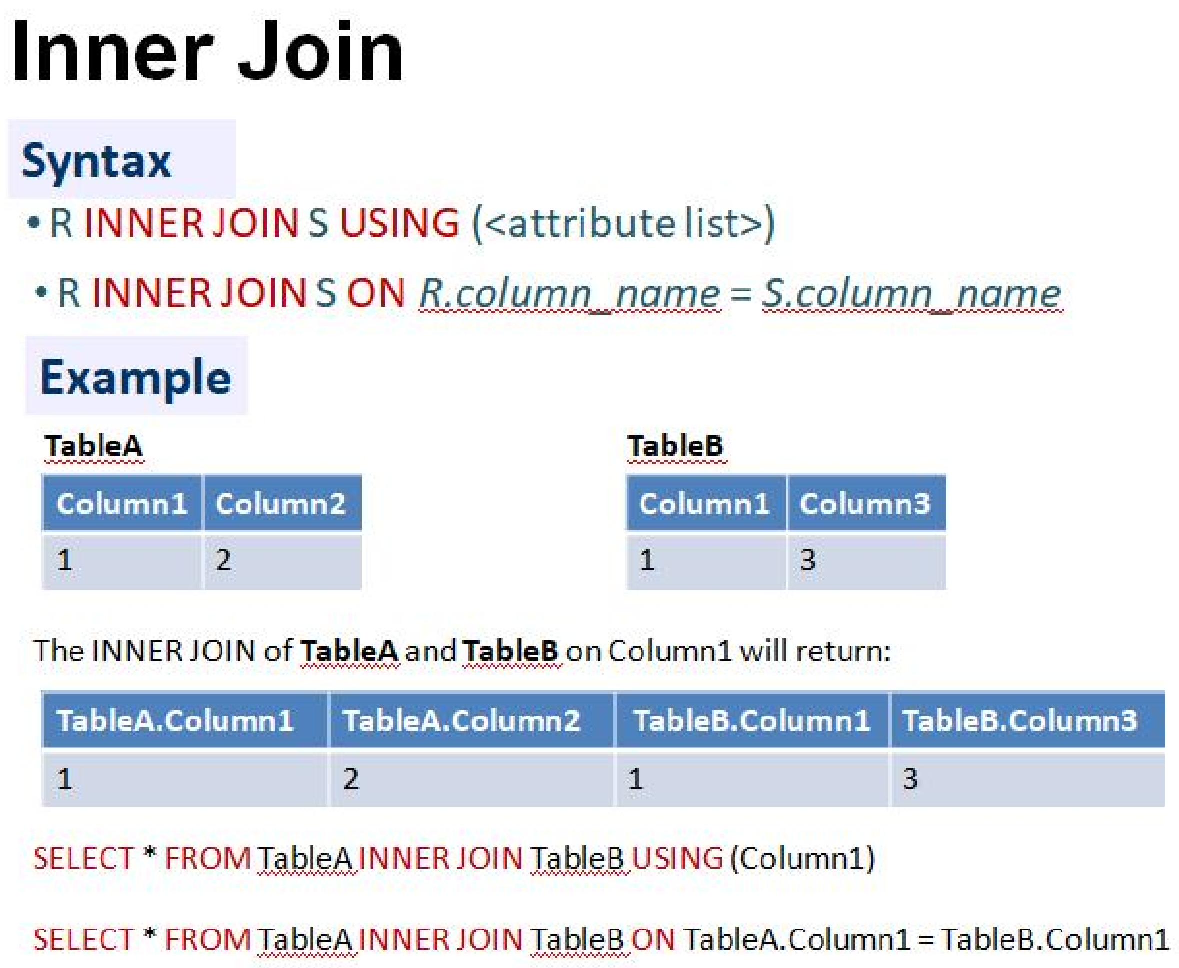
Product(PName, Price, Category, Manufacturer)

Company(CName, StockPrice, Country)

Example: Find all products manufactured in Japan, and stock price more than $300; return their names and prices.

```
SELECT PName, Price
FROM   Product
JOIN   Company ON Manufacturer=Cname AND Country='Japan'
                  AND StockPrice >= 300
```

Any Boolean condition



Inner Join

Syntax

- R INNER JOIN S USING (<attribute list>)
- R INNER JOIN S ON *R.column_name* = *S.column_name*

Example

TableA

Column1	Column2
1	2

TableB

Column1	Column3
1	3

The INNER JOIN of **TableA** and **TableB** on Column1 will return:

TableA.Column1	TableA.Column2	TableB.Column1	TableB.Column3
1	2	1	3

```
SELECT * FROM TableA INNER JOIN TableB USING (Column1)

SELECT * FROM TableA INNER JOIN TableB ON TableA.Column1 = TableB.Column1
```

---

1. Does it matter in which order we put the clauses? Example putting HAVING before WHERE or putting WHERE before GROUP BY.

Yes, it matters. Follow the order as in the lecture slide.
