

PRACTICE PROBLEM SET #1

CSC 261/461 (Database Systems), Spring 2018,
University of Rochester

Problem 1: Set Operations

Relation R			
A	B	C	D
4	3	2	1
1	4	6	10
11	12	4	9
6	11	9	34
8	3	11	12

Relation S			
A	B	C	D
13	9	10	2
10	11	14	19
1	4	6	10
6	11	9	34

Suppose we have the two tables (or relations) R and S : instances of these relations are shown above.

- What is the union of R and S ($R \cup S$) ?
- What is the intersection of R and S ($R \cap S$) ?
- What is their Cartesian product ($R \times S$) ?

Note: We will cover union and intersection on Monday

1 Problem 1: (solution)

Table 1: Tables (Solution of problem 1)

$R \cup S$			
A	B	C	D
4	3	2	1
1	4	6	10
11	12	4	9
6	11	9	34
8	3	11	12
13	9	10	2
10	11	14	19

$R \cap S$			
A	B	C	D
1	4	6	10
6	11	9	34

$R \times S$							
R.A	R.B	R.C	R.D	S.A	S.B	S.C	S.D
4	3	2	1	13	9	10	2
1	4	6	10	13	9	10	2
11	12	4	9	13	9	10	2
6	11	9	34	13	9	10	2
8	3	11	12	13	9	10	2
4	3	2	1	10	11	14	19
1	4	6	10	10	11	14	19
11	12	4	9	10	11	14	19
6	11	9	34	10	11	14	19
8	3	11	12	10	11	14	19
4	3	2	1	1	4	6	10
1	4	6	10	1	4	6	10
11	12	4	9	1	4	6	10
6	11	9	34	1	4	6	10
8	3	11	12	1	4	6	10
4	3	2	1	6	11	9	34
1	4	6	10	6	11	9	34
11	12	4	9	6	11	9	34
6	11	9	34	6	11	9	34
8	3	11	12	6	11	9	34

Table 2: Tables (Product, PC and Printer)

Sample data for relation <i>Product</i>		Sample data for relation <i>PC</i>					Sample data for relation <i>Printer</i>			
manufacturer	model	model	speed	ram	rd	price	model	color	type	price
HP	1001	1001	700	64	48xCD	799	3001	true	ink-jet	231
Canon	1002	1002	1500	128	12xDVD	2499	3002	true	ink-jet	267
Toshiba	2005	2005	600	64	12xDVD	2399	3003	false	laser	390
HP	2006	2006	800	96	15xCD	2999	3004	true	ink-jet	439
Microsoft	2007	2007	850	128	40xDVD	3099	3005	true	bubble	200
Canon	2008	2008	600	64	12xCD	1249	3006	true	laser	1999
Apple	2009	2009	750	256	13xDVD	2599	3007	false	laser	350
Sony	3001									
Sony	3002									
Nokia	3003									
Sony	3004									
Toshiba	3005									
HP	3006									
Camon	3007									

Problem 2: Relational Database and Schema

Refer Table 2. The Product table (or relation) stores the manufacturer and model number (primary key). The PC table contains a subset of the model numbers provided in the Product table. For each model number, the speed (of the processor, in megahertz), the amount of RAM (in megabytes), the speed and type of the removable disk (CD or DVD), and the price is provided. The Printer relation records another set of models, and stores whether the printer produces color output (true/false), the process type (laser, ink-jet, or bubble), and the price.

Write down the **schema** for these three relations.

Problem 2: (Solution)

Product (manufacturer, model)
 PC (model, speed, ram, rd, price)
 Printer (model, color, type, price)

Problem 3: Structured Query Language (SQL)

For the relations *PC*, *Printer* and *Product* write SQL query for the following questions:

1. Create tables for these relations. Identify the primary key and foreign keys for each table.
2. Populate these tables with the data provided in section 1
3. Find out all the PC models which have a speed of at least 800?
4. Find the model numbers of all color laser printers.

5. Find the model number and price of all printers made by manufacturer Canon.
6. Find all manufacturers producing printers that cost more than \$1000
7. List all the Computers with DVD player sorted by their Speed. those pairs of PC models that have both the same speed
8. Find all the distinct manufacturers that produces both, computers worth less than \$1000 and More than \$2000
9. Update the price of all PC by \$100
10. Delete all black & white printers

Problem 3: (Solution)

There are many solutions to these questions. We are just providing you one.

1. CREATE TABLE product(
 - model INT,
 - manufacturer VARCHAR(30) NOT NULL,
 - PRIMARY KEY(model)
);

 CREATE TABLE pc (
 - model INT,
 - speed INT,
 - ram INT,
 - rd VARCHAR(10) NOT NULL,
 - price DECIMAL (18, 2),
 - PRIMARY KEY(model),
 - FOREIGN KEY (model) REFERENCES PRODUCT(model)
);

 CREATE TABLE printer (
 - model INT,
 - color BOOLEAN NOT NULL,
 - type VARCHAR(10) NOT NULL,
 - price DECIMAL (18, 2),
 - PRIMARY KEY(model),
 - FOREIGN KEY (model) REFERENCES PRODUCT(model)
);
2. INSERT INTO table_nameVALUES (value1,value2,value3,...);

3. SELECT model FROM pc WHERE speed >= 800
4. SELECT model FROM printer WHERE color = true
5. SELECT printer.model, printer.price FROM product,printer where printer.model = product.model and product.manufacturer='canon'
6. SELECT product.manufacture FROM product,printer where printer.model = product.model and printer.price > 1000
7. SELECT * FROM pc WHERE rd LIKE '%DVD' ORDER BY speed;
8. SELECT manufacturer FROM product WHERE model IN (SELECT model FROM pc WHERE price < 1000) and model IN (SELECT model FROM pc WHERE price > 2000)
9. UPDATE pc SET price = price + 100
10. DELETE FROM printer WHERE color = false

Problem 4: Table Relationship

This schema is same as the one used during class.

Product (pname, price, category, manufacturer)

Company (cname, stockPrice, country)

In the product table, foreign key **manufacturer** refers to **cname**. Find all categories of products that are made in China.

Problem 4: (Solution)

```
SELECT Product.category FROM Product,Company WHERE Product.manufacturer = Company.cname and Company.country = "China"
```

Problem 5: Create Table based on Table Info

SQLite (we are using for Jupyter notebook) supports a nifty command:

```
pragma table_info(tablename);
```

The equivalent in MySQL is:

```
DESCRIBE tablename;
```

This produces all the related info of a table. The following table is retrieved from the command:

```
pragma table_info(Product);
```

Write a sql command to create table Product based on this metainfo.

Note: Give attention to default values, NOT NULL, primary key and datatype

cid	name	type	notnull	dflt_value	pk
0	pname	varchar(20)	0	None	1
1	price	money	0	None	0
2	category	varchar	0	None	0
3	manufacturer	varchar	1	None	0
4	rating	float	0	3	0

Problem 5: (Solution)

```
CREATE TABLE Product (  
pname VARCHAR(20),  
price FLOAT,  
category VARCHAR(30),  
manufacturer VARCHAR(30) NOT NULL,  
rating FLOAT DEFAULT 3,  
PRIMARY KEY( pname )  
);
```