HW #3

- The SQL parts of this homework allow for, if you prefer, using MongoDB instead.

Part甲

- **Two queries are said to be equivalent if they give the same answer for every instance of the database.**
- **Consider the following database:**

government (name, rank, salary)

politician (name, party, gender)

donation (name, organization, amount)

- **For each of the following five pairs of SQL queries, determine whether they are equivalent. If they are not, give a database instance as a counter-example. Otherwise, just state that they are equivalent.**

HW #3 (2)

```
甲1A.   select name
        from donation A
        where not exists
          (select * from donation B
           where name = 'Campbell'
           and not exists
             (select * from donation C
              where C.organization = B.organization
              and C.name = A.name))
```

```
甲1B.   select name
        from donation A
        where not exists
          (select * from donation
           where name = 'Campbell'
           and organization not in
             (select organization
              from donation B
              where B.name = A.name))
```

HW #3 (3)

甲2A.

```
(select name
 from government
 where salary >= 100000)
 union
(select name
 from government
 where salary < 100000)
```

甲2B.

```
select name
from government
```

HW #3 (4)

甲3A.

```
(select name
 from government
 where rank = ‘minister’)
 union
(select name from donation
 where amount >= 100000)
```

甲3B.

```
select government.name
from government, donation
where government.name = donation.name
          and (rank = ‘minister’ or amount >= 100000)
```

HW #3 (5)

```
甲4A.  select name
       from politician
       where name not in
           (select name from government)

甲4B.  select name
       from politician
       where not exists
           (select *
            from government
            where government.name = politician.name)
```

HW #3 (6)

甲5A.
```
select name
from politician
where name not in
      (select name from government)
```

甲5B.
```
select politician.name
from politician, government
where not politician.name = government.name
```

Sample Test Dataset (甲)

government

name	rank	salary
Albert	minister	150000
Bobbie	clerk	50000
Don	clerk	null

politician

name	party	gender
Albert	Republic	male
Charlie	Democrat	male

donation

name	organization	amount
Charlie	American Red Cross	150000
Charlie	National AIDS Fund	80000
Charlie	UNICEF	80000
Don	NineMillion	50000
Don	American Red Cross	60000
Campbell	American Red Cross	70000
Campbell	National AIDS Fund	60000
Mike	NineMillion	90000

HW #3 (7)

Part乙

- Consider the following collection of relation schemes:

  professor (profname, deptname)

  department (deptname, building)

  committee (commname, profname)

- Write MySQL-compatible SQL queries that solve each of the following problems:

HW #3 (8)

乙1: Find all the professors who are in any one of the committees that professor Piper is in.

乙2: Find all the professors who are in at least all those committees that professor Piper is in.

乙3: Find all the professors who have not offices in any of those buildings that professor Piper has offices in.

Sample Test Dataset (乙)

professor

profname	deptname
Piper	Computer Science
James	Computer Science
George	Computer Science
William	Electrical Engineering
Matthew	Electrical Engineering
Oliver	Mechanical Engineering
Lewis	Mechanical Engineering

department

deptname	building
Computer Science	ICICS/CS
Electrical Engineering	KAIS
Mechanical Engineering	CEME

committee

commname	profname
Operation	James
Operation	William
Communication	James
Communication	Piper
Communication	Oliver
Communication	Lewis
Teaching	James
Teaching	Piper
Teaching	Matthew
Teaching	Lewis
Graduate Admissions	William
Graduate Admissions	George
Computing	Matthew

HW #3 (9)

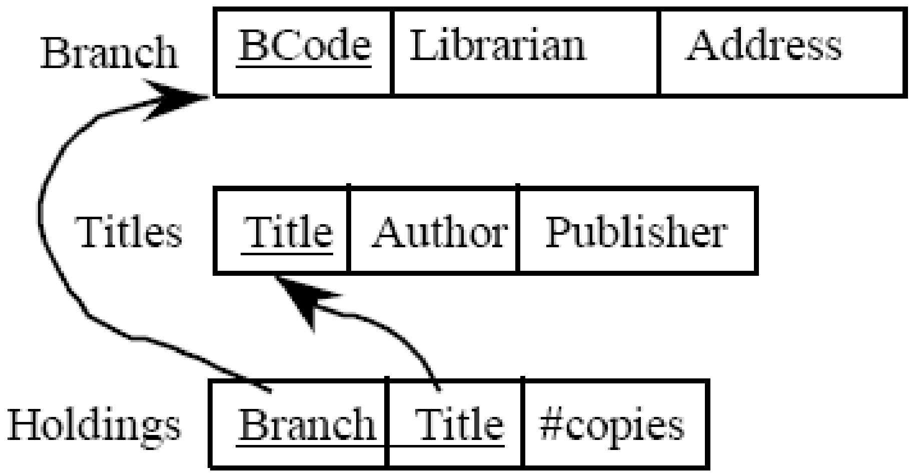
Part丙

- Consider the following relational database schema. It is intended to represent the holdings of a multi-branch library. A sample database instance is also given.
- Write SQL statements to retrieve the following information:

丙1. The names of all library books published by Macmillan.

丙2. Branches that hold any books by Ann Brown.

丙3. The total number of books held at each branch.



BCode	Librarian	Address
B1	John Smith	2 Anglesea Rd
B2	Mary Jones	34 Pearse St
B3	Francis Owens	Grange X

Title	Author	Publisher
Susannah	Ann Brown	Macmillan
How to Fish	Amy Fly	Stop Press
A History of Dublin	David Little	Wiley
Computers	Blaise Pascal	Applewoods
The Wife	Ann Brown	Macmillan

Branch	Title	#copies
B1	Susannah	3
B1	How to	2
B1	A hist	1
B2	How to	4
B2	Computers	2
B2	The Wife	3
B3	A hist ..	1
B3	Computers	4
B3	Susannah	3
B3	The Wife	1

HW #3 (10)

Part丁

- The parts of this problem are multiple choice單選. In each of the below questions, consider the three SQL queries listed, and circle one of the listed options. Assume there are no NULL values in any of the tables. The queries will all involve either IN, ALL, ANY (recall, equivalent to "SOME"), or EXISTS.

丁1. SELECT S.A FROM S WHERE S.B NOT IN (SELECT R.B FROM R WHERE R.B > 5);

HW #3 (11)

SELECT S.A FROM S WHERE S.B <> ALL (SELECT R.B FROM R WHERE R.B > 5);

SELECT S.A FROM S WHERE NOT EXISTS (SELECT R.B FROM R WHERE R.B > 5 AND R.B = S.B);

(A) The first query is not equivalent to any of the other queries.

(B) The first query is equivalent to the second query only.

(C) The first query is equivalent to the third query only.

(D) All three queries are equivalent.

HW #3 (12)

丁2. SELECT S.A FROM S WHERE S.B <= ANY (SELECT R.B FROM R WHERE R.B > 5);

SELECT S.A FROM S WHERE S.B <= ALL (SELECT R.B FROM R WHERE R.B > 5);

SELECT S.A FROM S WHERE EXISTS (SELECT R.B FROM R WHERE R.B > 5 AND R.B >= S.B);

(A) The first query is not equivalent to any of the other queries.

(B) The first query is equivalent to the second query only.

HW #3 (13)

(C) The first query is equivalent to the third query only.

(D) All three queries are equivalent.

丁3. SELECT S.A FROM S WHERE S.B IN (SELECT R.B FROM R WHERE R.B > 5);

SELECT S.A FROM S WHERE S.B = ANY (SELECT R.B FROM R WHERE R.B > 5);

SELECT S.A FROM S WHERE S.B >= ALL (SELECT R.B FROM R WHERE R.B > 5);

HW #3 (14)

(A) The first query is not equivalent to any of the other queries.

(B) The first query is equivalent to the second query only.

(C) The first query is equivalent to the third query only.

(D) All three queries are equivalent.

HW #3 (15)

Part戊

- Given the following schema, write SQL statements that would achieve the following goals:

Manufacturer (name, country, phone)

Product (manu_name, model, style)

Desktop (model, speed, RAM, HD, list_price)

Laptop (model, speed, RAM, HD, screen, list_price)

戊1. Find the average HD size of the Desktop PCs.

戊2. Find the average price of laptops with a speed of at least 3.0.

HW #3 (16)

戊3. Find the average price of desktop and laptops made by Dell.

戊4. Find, for each different price, the average speed of a PC.

戊5. Find the manufacturers that make at least 3 different models of desktop PCs.

戊6. Find for each manufacturer that makes desktop the maximum speed of a desktop.

戊7. Find the for each speed of desktop PC above 2.5, the average hard-disk size.

戊8. Find for each manufacturer, the average speed of its laptops.

HW #3 (17)

戊9. Find the average hard-disk size of a desktop PC for all those manufacturers that make laptops.

戊10. Delete all desktop PCs with less than 400GB of HD.

戊11. Using 2 insert statements, insert the following data in the DB: desktop PC model 1500 is made by Acer, has speed 3.1, RAM 2048, HD 300, and sells for $799.

戊12. Delete all laptops made by a manufacturer that does **not** make PCs.

戊13. For each PC, double the amount of HD and add 2048 to the amount of RAM.

For each laptop made by Dell, add one inch to the screen size and subtract $200 from the price.

HW #3 (18)

Part己

- You are given the following relational schema.

  Books(bid:integer, btitle:string, author:string,
  year:integer, price:integer)

  Orders(cid:integer, bid:integer, quantity:integer)

  Customers(cid:integer, cname:string, zipcode:string)

- The meaning of attributes is as follows:
- **bid**: book unique id, **btitle**: book title, **author**: name of book author, **year**: book publication year, **price**: unit price per copy;

HW #3 (19)

- **quantity**: number of book copies purchased with an order;
- **cid**: unique customer identifier, **cname**: customer name, **zipcode**: customer address zipcode.

- Write SQL queries for the following:

己1. Find the titles of books that were ordered only in quantities of at least 100.

己2. Find the authors of books that cost at most $40 and were ordered from zipcode 12345.

HW #3 (20)

- 3. Find the names of customers who ordered some book published in year 2000 and also ordered at least 10 copies of some book that costs more than $100.
- 4. Find the authors of books for which there are at least two orders placed.
- 5. Find the titles of books ordered by those customers who are the only registered customers in their particular zipcode area (i.e., there is no other customer with the same zipcode in the **Customers** table).
- 6. Find the authors of the books that were ordered only from zipcode 02125.

HW #3 (21)

己7. Find the zipcodes of customers who ordered at least 10 copies (in a single order) of a book written by an author whose name starts with “Cod”.

己8. [Group By進階指令] For each customer who ordered at least 5 distinct books (regardless of publication year), find the price of the most expensive book published in 1990 which was ordered by that customer. In the output, the costumer should be listed by name.

己9. Find the title(s) of the book(s) that were ordered from every zipcode present in the customers table.

HW #3 (22)

己10. [Group By進階指令] Find the total dollar amount of purchases for every customer in zipcode 02125; list customer name in the output along with the amount.

己11. [Group By進階指令] Find the zipcode(s) that generated the highest revenue for the store (i.e., the largest combined dollar amount for orders originating in that zipcode).