

LAB 5 – Advanced Data Retrieval Command

1. พื้นฐานการเชื่อมตาราง

หากเราต้องการเรียกคุณข้อมูลจาก field ต่างๆ ซึ่งอยู่คนละตาราง เราจะต้องใช้วิธีที่เรียกว่า การเชื่อมตาราง (Join Table) ซึ่งโดยพื้นฐานการเชื่อมตารางมาจากการหาผลคูณทางคณิตศาสตร์ที่เรียกว่า ผลคูณคาร์ทีเซียน (Cartesian Product)

ตัวอย่างที่ 1

```
select Employee.Name, Department.Name from Employee, Department;
```

Employee.name	Department.name
Somchai	Finance
Somchai	Research and Development
Somchai	Marketing
Wichit	Finance
Wichit	Research and Development
Wichit	Marketing
Somjai	Finance
Somjai	Research and Development
Somjai	Marketing
Aphitsit	Finance
Aphitsit	Research and Development
Aphitsit	Marketing
Yongyut	Finance
Yongyut	Research and Development
Yongyut	Marketing
Somjit	Finance
Somjit	Research and Development
Somjit	Marketing

ตัวอย่างที่ 2

จากผลลัพธ์ที่ได้จากตัวอย่างที่ 1 จะเห็นว่าเป็นการจับคู่ จากแต่ละ record ในตาราง Employee ไปยังทุกๆ record ในตาราง Department แต่อย่างไรก็ตาม หากเราต้องการแสดงผลเฉพาะการจับคู่ระหว่างพนักงานในตาราง Employee กับ แผนกที่พนักงานเหล่านี้สังกัดอยู่ ในตาราง Department จริงๆ เราสามารถใช้คำสั่งเงื่อนไขเพิ่มเข้าไปได้ดังนี้

```
select Employee.Name, Department.Name  
from Employee, Department  
where Employee.Department_ID=Department.Department_ID ;
```

```
mysql> select Employee.name, Department.name
-> from Employee, Department
-> where Employee.Department_ID=Department.Department_ID ;
+-----+-----+
| name | name |
+-----+-----+
| Somchai | Research and Development |
| Wichit | Finance |
| Somjai | Research and Development |
| Aphitsit | Marketing |
| Yongyut | Research and Development |
| Somjit | Research and Development |
+-----+-----+
```

ตัวอย่างที่ 3

```
select Assignment.Client_ID, Employee.Name
from Assignment, Employee
where Assignment.Employee_ID= Employee.Employee_ID;
```

```
mysql> select Assignment.Client_ID, Employee.Name
-> from Assignment, Employee
-> where Assignment.Employee_ID= Employee.Employee_ID;
+-----+-----+
| Client_ID | Name |
+-----+-----+
| 1 | Somchai |
| 2 | Wichit |
+-----+-----+
```

สรุปหลักการที่สำคัญของการเชื่อมตารางกือ ตาราง 2 ตารางที่เชื่อมกัน ได้จะต้องมี field หนึ่งที่ปรากฏอยู่ในทั้ง 2 ตารางนั้น และใช้ field ดังกล่าวกำหนดเงื่อนไขให้มีค่าเท่ากัน

2. การกำหนดชื่อย่อของตารางในการเชื่อมตาราง

เพื่อการเขียนคำสั่งที่สั้นและง่ายกว่าเดิม เราสามารถกำหนดชื่อย่อของตารางได้ โดยในคำสั่ง SQL ที่ส่วนของ from ให้ตั้งชื่อย่อ หลังชื่อเดิมของตาราง และการใช้งานให้ใช้ชื่อย่อที่ตั้งขึ้นแทนชื่อเดิมได้เลย

ตัวอย่างที่ 4

```
select E.Name, D.Name
from Employee E, Department D
where E.Department_ID=D.Department_ID;
```

```
mysql> select E.Name, D.Name
-> from Employee E, Department D
-> where E.Department_ID=D.Department_ID;
+-----+-----+
| Name | Name |
+-----+-----+
| Somchai | Research and Development |
| Wichit | Finance |
| Somjai | Research and Development |
| Aphitsit | Marketing |
| Yongyut | Research and Development |
| Somjit | Research and Development |
+-----+-----+
```

ตัวอย่างที่ 5

```
select A.Client_ID, E.Name
from Assignment A, Employee E
where A.Employee_ID= E.Employee_ID;
```

```
mysql> select A.Client_ID, E.Name
-> from Assignment A, Employee E
-> where A.Employee_ID= E.Employee_ID;
+-----+-----+
| Client_ID | Name |
+-----+-----+
| 1 | Somchai |
| 2 | Wichit |
+-----+-----+
```

3. การเชื่อมตารางมากกว่า 2 ตาราง

การเรียกดูข้อมูลจาก field ต่างๆ มาจากตารางมากกว่า 2 ตารางขึ้นไป ก็ให้ใช้หลักการเดียวกับการเชื่อมตาราง 2 ตาราง คือ หา field ที่ปรากฏอยู่ทั้ง 2 ตาราง กำหนดเงื่อนไขให้มีค่าเท่ากัน และ ใช้ ตระกูล and เชื่อมคู่ตารางอื่นให้ครบถ้วนหมด

ตัวอย่างที่ 6

```
select C.Name as Client_Name, E.Name as Employee_Name
from Employee E, Client C, Assignment A
where E.Employee_ID= A.Employee_ID
and A.Client_ID=C.Client_ID;
```

```
mysql> select C.Name as Client_Name, E.Name as Employee_Name
-> from Employee E, Client C, Assignment A
-> where E.Employee_ID= A.Employee_ID
-> and A.Client_ID=C.Client_ID;
+-----+-----+
| Client_Name | Employee_Name |
+-----+-----+
| ABC Company Limited | Somchai |
| Imperial Industry | Wichit |
+-----+-----+
```

ตัวอย่างที่ 7

```
select C.Name as Client_Name, E.Name as Employee_Name, D.Name as Department_Name
from Employee E, Client C, Assignment A, Department D
where E.Employee_ID= A.Employee_ID
and A.Client_ID=C.Client_ID
and E.Department_ID=D.Department_ID;
```

```
mysql> select C.Name as Client_Name, E.Name as Employee_Name, D.Name as Department_Name
-> from Employee E, Client C, Assignment A, Department D
-> where E.Employee_ID= A.Employee_ID
-> and A.Client_ID=C.Client_ID
-> and E.Department_ID=D.Department_ID;
+-----+-----+-----+
| Client_Name | Employee_Name | Department_Name |
+-----+-----+-----+
| ABC Company Limited | Somchai | Research and Development |
| Imperial Industry | Wichit | Finance |
+-----+-----+-----+
```

ตัวอย่างที่ 8

```
select E.Name as Employee_Name, D.Name as Department_Name
from Employee E, Client C, Assignment A, Department D
where E.Employee_ID= A.Employee_ID
and A.Client_ID=C.Client_ID
and E.Department_ID=D.Department_ID
and C.Name='ABC Company Limited';
```

```
mysql> select E.Name as Employee_Name, D.Name as Department_Name
-> from Employee E, Client C, Assignment A, Department D
-> where E.Employee_ID= A.Employee_ID
-> and A.Client_ID=C.Client_ID
-> and E.Department_ID=D.Department_ID
-> and C.Name='ABC Company Limited';
+-----+-----+
| Employee_Name | Department_Name |
+-----+-----+
| Somchai | Research and Development |
+-----+-----+
```

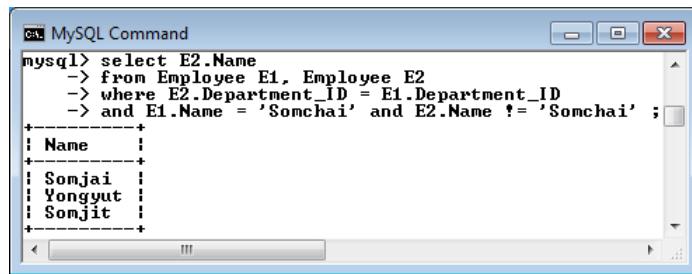
4. การเชื่อมตารางเดิม

การเชื่อมตารางนั้น ไม่เพียงแต่เชื่อมตารางที่ต่างกันเท่านั้น เรายังสามารถทำการเชื่อมตารางเดียวกันได้ ซึ่งหลักการก็เหมือนกับการเชื่อมตาราง 2 ตาราง หรือมากกว่า 2 ตาราง ดังที่กล่าวมาก่อนหน้านี้

ตัวอย่างที่ 9

ตัวอย่างนี้เป็นการแสดงชื่อพนักงานคนที่ทำงานแผนกเดียวกับพนักงานที่ชื่อ Somchai

```
select E2.Name  
from Employee E1, Employee E2  
where E2.Department_ID = E1.Department_ID  
and E1.Name = 'Somchai' and E2.Name != 'Somchai' ;
```



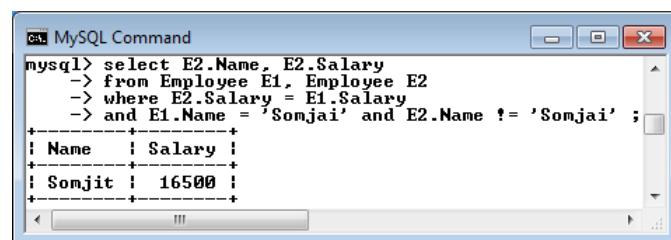
The screenshot shows the MySQL Command window with the following output:

```
mysql> select E2.Name  
-> from Employee E1, Employee E2  
-> where E2.Department_ID = E1.Department_ID  
-> and E1.Name = 'Somchai' and E2.Name != 'Somchai' ;  
+-----+  
| Name |  
+-----+  
| Somjai |  
| Yongyut |  
| Somjit |  
+-----+
```

ตัวอย่างที่ 10

ตัวอย่างนี้เป็นการแสดงชื่อพนักงานและเงินเดือนของคนที่มีเงินเดือนเท่ากับพนักงานที่ชื่อ Somjai

```
select E2.Name, E2.Salary  
from Employee E1, Employee E2  
where E2.Salary = E1.Salary  
and E1.Name = 'Somjai' and E2.Name != 'Somjai' ;
```



The screenshot shows the MySQL Command window with the following output:

```
mysql> select E2.Name, E2.Salary  
-> from Employee E1, Employee E2  
-> where E2.Salary = E1.Salary  
-> and E1.Name = 'Somjai' and E2.Name != 'Somjai' ;  
+-----+-----+  
| Name | Salary |  
+-----+-----+  
| Somjit | 16500 |  
+-----+-----+
```

ชื่อ-นามสกุล.....รหัสนิสิต.....เลขที่.....

5. การเชื่อมตารางทางซ้าย (Left Join) และ ทางขวา (Right Join)

การเชื่อมตารางทางซ้าย (Left Join) เป็นคำสั่งที่ใช้สำหรับการระบุเงื่อนไขการเลือกข้อมูลในตาราง โดยเงื่อนไขการทำ Left Join จะทำการเลือกข้อมูลหลัก (ตารางทางซ้ายของคำสั่ง Join) เป็นตัวตั้งและเลือกข้อมูลรอง (ตารางทางขวาของคำสั่ง Join) ซึ่งมีค่าคีย์ที่ตรงกัน โดยถ้าข้อมูลจากตารางทางซ้ายไม่ตรงเงื่อนไขจะแสดงผลเป็น Null

การเชื่อมตารางทางขวา (Right Join) เป็นคำสั่งที่ใช้สำหรับการระบุเงื่อนไขการเลือกข้อมูลในตาราง โดยเงื่อนไขการทำ Right Join จะทำการเลือกข้อมูลหลัก (ตารางทางขวาของคำสั่ง Join) เป็นตัวตั้งและเลือกข้อมูลรอง (ตารางทางซ้ายของคำสั่ง Join) ซึ่งมีค่าคีย์ที่ตรงกัน โดยถ้าข้อมูลจากตารางทางซ้ายไม่ตรงเงื่อนไขจะแสดงผลเป็น Null

รูปแบบของคำสั่งแสดงได้ดังนี้

```
select[Table_Name1].Field1, [Table_Name2].Field1, ...
from [Table_Name1] left join [Table_Name2]
on [Table_Name1].Field = [Table_Name2].Field
```

```
select[Table_Name1].Field1, [Table_Name2].Field1, ...
from [Table_Name1] right join [Table_Name2]
on [Table_Name1].Field = [Table_Name2].Field
```

โดยที่

Table_Name1, Table_Name2 คือ ชื่อตารางที่ 1 และ ตารางที่ 2

Filed1, Field2, ... คือ ชื่อ filed1, filed2, ...

Field คือ ชื่อ field ที่เป็นคีย์ที่ใช้ในการ join

ตัวอย่างที่ 11

```
select *
from Employee left join Assignment
on Employee.Employee_ID = Assignment.Employee_ID;
```

Table: Employee (ใช้เป็นตัวตั้ง)

Employee_ID	Name	Job	Salary	Department_ID
1111	Somchai	Programmer	15000	128
2222	Wichit	DBA	13500	42
3333	Somjai	Programmer	16500	128
4444	Aphitsit	System Administrator	12000	130
4445	Yongyut	Programmer	20000	128
4446	Somjit	Programmer	16500	128

Table: Assignment

Client_ID	Employee_ID	Work_Date	Hours
1	1111	2009-04-27	8.5
2	2222	2009-05-26	7

```
mysql> select *
-> from Employee left join Assignment
-> on Employee.Employee_ID = Assignment.Employee_ID;
```

Employee_ID	Name	Job	Salary	Department_ID	Client_ID	Employee_ID	Work_Date	Hours
1111	Somchai	Programmer	15000	128	1	1111	2009-04-27	8.5
2222	Wichit	DBA	13500	42	2	2222	2009-05-26	7
3333	Somjai	Programmer	16500	128	NULL	NULL	NULL	NULL
4444	Aphitsit	System Administrator	12000	130	NULL	NULL	NULL	NULL
4445	Yongyut	Programmer	20000	128	NULL	NULL	NULL	NULL
4446	Somjit	Programmer	16500	128	NULL	NULL	NULL	NULL

ตัวอย่างที่ 12

```
select *
from Employee right join Assignment
on Employee.Employee_ID = Assignment.Employee_ID;
```

Table: Assignment (ใช้เป็นตัวตั้ง)

Client_ID	Employee_ID	Work_Date	Hours
1	1111	2009-04-27	8.5
2	2222	2009-05-26	7

Table: Employee

Employee_ID	Name	Job	Salary	Department_ID
1111	Somchai	Programmer	15000	128
2222	Wichit	DBA	13500	42
3333	Somjai	Programmer	16500	128
4444	Aphitsit	System Administrator	12000	130
4445	Yongyut	Programmer	20000	128
4446	Somjit	Programmer	16500	128

```
mysql> select *
-> from Employee right join Assignment
-> on Employee.Employee_ID = Assignment.Employee_ID;
```

Employee_ID	Name	Job	Salary	Department_ID	Client_ID	Employee_ID	Work_Date	Hours
1111	Somchai	Programmer	15000	128	1	1111	2009-04-27	8.5
2222	Wichit	DBA	13500	42	2	2222	2009-05-26	7

ตัวอย่างที่ 13

ตัวอย่างนี้เป็นการแสดงชื่อพนักงานที่ไม่ได้รับมอบหมายให้ดูแลลูกค้าเลย โดยนำผลลัพธ์ในตัวอย่างข้อ 11 มาพิจารณาอีกรัง จะได้คำสั่งดังนี้

```
select Employee.Name  
from Employee left join Assignment  
on Employee.Employee_ID = Assignment.Employee_ID  
where Assignment.Client_ID is Null;
```

The screenshot shows the MySQL Command window. The query is:

```
mysql> select Employee.Name  
-> from Employee left join Assignment  
-> on Employee.Employee_ID = Assignment.Employee_ID  
-> where Assignment.Client_ID is Null;
```

The result set is:

Name
Somjai
Aphitsit
Yongyut
Somjut

5. การเรียกดูข้อมูลแบบย่อ

เป็นลักษณะของชุดคำสั่งที่ใช้ในการเรียกดูข้อมูลหนึ่งภายในอีกชุดของการเรียกดูข้อมูลหนึ่ง ปกติจะเรียกว่าเป็นการเรียกดูข้อมูลแบบย่อ (subqueries หรือ nested queries)

ตัวอย่างที่ 14

ตัวอย่างนี้เป็นการแสดงชื่อพนักงานที่ยังไม่ได้รับมอบหมายให้ดูแลลูกค้า

```
select Name from Employee  
where Employee_ID not in (select Employee_ID from Assignment);
```

The screenshot shows the MySQL Command window. The query is:

```
mysql> select Name from Employee  
-> where Employee_ID not in (select Employee_ID from Assignment);
```

The result set is:

Name
Somjai
Aphitsit
Yongyut
Somjut

ตัวอย่างที่ 15

ตัวอย่างนี้เป็นการแสดงชื่อพนักงานที่มีเงินเดือนมากที่สุด

```
select Name from Employee
```

```
where Salary = (select max(Salary) from Employee);
```

```
mysql> select Name from Employee  
-> where Salary = (select max(Salary) from Employee);  
+-----+  
| Name |  
+-----+  
| Yongyut |  
+-----+
```

ตัวอย่างที่ 16

ตัวอย่างนี้เป็นการแสดงชื่อพนักงานที่ได้รับมอบหมายให้ดูแลลูกค้า และ พนักงานคนนั้นทำงานในตำแหน่ง Programmer ด้วย

```
select E2.Name from Employee E2
```

```
where E2.Employee_ID in (
```

```
    select E1.Employee_ID from Assignment A, Employee E1
```

```
    where A.Employee_ID = E1.Employee_ID
```

```
    and E1.Job = 'Programmer'
```

```
);
```

```
mysql> select E2.Name from Employee E2  
-> where E2.Employee_ID in (  
->     select E1.Employee_ID from Assignment A, Employee E1  
->     where A.Employee_ID = E1.Employee_ID  
->     and E1.Job = 'Programmer'  
-> );  
+-----+  
| Name |  
+-----+  
| Somchai |  
+-----+
```

แบบฝึกหัด

1. จงแสดงคำสั่ง SQL เพื่อใช้เชื่อมตาราง Client และ Assignment ให้ได้ผลลัพธ์ดังนี้

Client_ID	Name	Client_ID	Employee_ID	Work_date
1	ABC Company Limited	1	1111	2009-04-27
2	Imperial Industry	1	1111	2009-04-27
1	ABC Company Limited	2	2222	2009-05-26
2	Imperial Industry	2	2222	2009-05-26

2. จากข้อ 1 จงแสดงคำสั่ง SQL เพื่อใช้ในการแสดงผลเฉพาะ record ที่มี Client_ID ตรงกับของตาราง Client และ Assignment ที่นำมาเชื่อมกันให้ได้ผลลัพธ์ดังนี้

Client_ID	Name	Client_ID	Employee_ID	Work_date
1	ABC Company Limited	1	1111	2009-04-27
2	Imperial Industry	2	2222	2009-05-26

3. จงแสดงคำสั่ง SQL เพื่อใช้เชื่อมตาราง Employee และ Employee_Skill ให้ได้ผลลัพธ์ดังนี้

Employee_ID	Name	Job	Salary	Department_ID	Skill
1111	Somchai	Programmer	15000	128	C
1111	Somchai	Programmer	15000	128	Java
1111	Somchai	Programmer	15000	128	Perl
2222	Wichit	DBA	13500	42	DB2
3333	Somjai	Programmer	16500	128	Java
3333	Somjai	Programmer	16500	128	VB
4444	Aphitsit	System Administrator	12000	130	Linux
4444	Aphitsit	System Administrator	12000	130	NT
4445	Yongyut	Programmer	20000	128	PHP
4446	Somjit	Programmer	16500	128	JSP

ชื่อ-นามสกุล.....รหัสนิสิต.....เลขที่.....

4. จงแสดงคำสั่ง SQL เพื่อเชื่อมตาราง Client และ Assignment ให้ได้ผลลัพธ์ดังนี้ โดยคำสั่ง SQL ที่ใช้จะต้องกำหนดชื่อเบื้องต้นของตารางแทนการใช้ชื่อเต็มของตาราง (ให้เปลี่ยนชื่อตาราง Client เป็น C และชื่อตาราง Assignment เป็น A)

Name	Address	Employee_ID	Work_Date	Hours
ABC Company Limited	1 Silom Rd.	1111	2009-04-27	8.5
Imperial Industry	100 Samutprakran	2222	2009-05-26	7

5. จงแสดงคำสั่ง SQL เพื่อใช้เชื่อมตาราง Employee , Employee_Skill และ Department ให้ได้ผลลัพธ์ดังนี้

Name	Job	Salary	Department Name	Skill
Aphitsit	System Administrator	12000	Marketing	Linux
Aphitsit	System Administrator	12000	Marketing	NT
Somchai	Programmer	15000	Research and Development	C
Somchai	Programmer	15000	Research and Development	Java
Somchai	Programmer	15000	Research and Development	Perl
Somjai	Programmer	16500	Research and Development	Java
Somjai	Programmer	16500	Research and Development	UB
Somjit	Programmer	16500	Research and Development	JSP
Wichit	DBA	13500	Finance	DB2
Yongyut	Programmer	20000	Research and Development	PHP

6. จงแสดงคำสั่ง SQL เพื่อ แสดงรายละเอียดของพนักงานคนที่มีอาชีพเดียวกันกับพนักงานที่ชื่อ Yongyut ให้ได้ผลลัพธ์ดังนี้

Name
Somchai
Somjai
Somjit

ชื่อ-นามสกุล..... รหัสนิสิต..... ลิขที่.....

7. จงแสดงคำสั่ง SQL เพื่อเชื่อมตารางให้ได้ผลลัพธ์ดังต่อไปนี้

Client_ID	Employee_ID	Work_Date	Hours	Employee_ID	Skill
1	1111	2009-04-27	8.5	1111	C
1	1111	2009-04-27	8.5	1111	Java
1	1111	2009-04-27	8.5	1111	Perl
2	2222	2009-05-26	?	2222	DB2
NULL	NULL	NULL	NULL	3333	Java
NULL	NULL	NULL	NULL	3333	VB
NULL	NULL	NULL	NULL	4444	Linux
NULL	NULL	NULL	NULL	4444	NT
NULL	NULL	NULL	NULL	4445	PHP
NULL	NULL	NULL	NULL	4446	JSP

8. จงแสดงคำสั่ง SQL เพื่อเชื่อมตารางให้ได้ผลลัพธ์ดังต่อไปนี้

Employee_ID	Skill	Client_ID	Employee_ID	Work_Date	Hours
1111	C	1	1111	2009-04-27	8.5
1111	Java	1	1111	2009-04-27	8.5
1111	Perl	1	1111	2009-04-27	8.5
2222	DB2	2	2222	2009-05-26	?
3333	Java	NULL	NULL	NULL	NULL
3333	VB	NULL	NULL	NULL	NULL
4444	Linux	NULL	NULL	NULL	NULL
4444	NT	NULL	NULL	NULL	NULL
4445	PHP	NULL	NULL	NULL	NULL
4446	JSP	NULL	NULL	NULL	NULL

9. จงแสดงคำสั่ง SQL เพื่อ แสดงชื่อพนักงานที่มีเงินเดือนมากกว่าเงินเดือนเฉลี่ยของพนักงานทั้งหมด (โดยคำสั่ง SQL ที่ใช้ จะต้องมีการใช้ subqueries ด้วย)

Name	Salary
Somjai	16500
Yongyut	20000
Somjit	16500

10. จงแสดงคำสั่ง SQL เพื่อแสดงชื่อพนักงานที่ยังไม่ได้รับมอบหมายให้คูณแล้วก็และ พนักงานคนนั้นทำงานในตำแหน่ง System Administrator ด้วย (โดยคำสั่ง SQL ที่ใช้ จะต้องมีการใช้ subqueries ด้วย)

Employee_ID	Name	Job	Salary	Department_ID
4444	Aphitsit	System Administrator	12000	130