MySQL簡易教程

1. 此教程無包含安裝過程，請自行安裝
2. 教程內容使用MySQL Workbench，但與使用MySQL Shell編輯、下指令方式無異

建功高中物理科
cow3690m@tmail.hc.edu.tw

呂昀修

目錄

Chap. 7
多個表格的建立與連結

前情提要：過去打的指令請先不要刪除，後面會發現原因xd

建功高中物理科
cow3690m@tmail.hc.edu.tw

呂昀修

Chap. 7 實際建立並串聯所有表格

Employee

emp_id	name	birthday	gender	salary	branch_id	buddy_id
206	Alice	1999/10/8	F	50000	1	NULL
207	Bob	1985/9/16	M	30000	2	NULL
208	Herry	2000/12/19	M	35000	3	NULL
209	Eve	1997/1/22	F	47000	1	206
210	John	1990/11/10	M	68000	3	208

Branch

branch_id	name	manager_id
1	研發	206
2	行政	207
3	資訊	208

Client

client_id	client_name	phone
400	Apple	3412
401	Banana	55688
402	Candy	4022
403	Dino	34157
404	ETE	8864

Works_with

emp_id	client_id	total_sales
206	400	70000
207	401	24000
208	400	10000
208	403	24000
210	404	88000

Chap. 7 實際建立並串聯所有表格

Employee

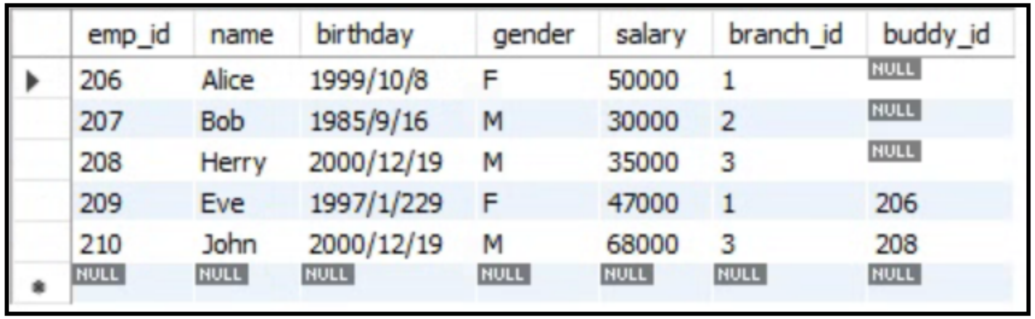
emp_id	name	birthday	gender	salary	branch_id	buddy_id
206	Alice	1999/10/8	F	50000	1	NULL
207	Bob	1985/9/16	M	30000	2	NULL
208	Herry	2000/12/19	M	35000	3	NULL
209	Eve	1997/1/22	F	47000	1	206
210	John	1990/11/10	M	68000	3	208

在設立foreign key前，必須先有這行資料記得先

```
alter table `employee` add `branch_id` int;
```

再

```
alter table `employee`
add foreign key(`branch_id`)
references `branch`(`branch_id`)
on delete set null;
```



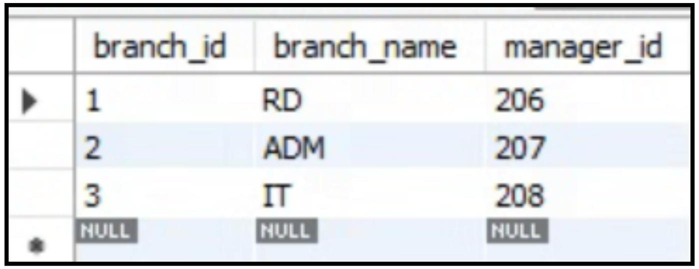
	emp_id	name	birthday	gender	salary	branch_id	buddy_id
▶	206	Alice	1999/10/8	F	50000	1	NULL
	207	Bob	1985/9/16	M	30000	2	NULL
	208	Herry	2000/12/19	M	35000	3	NULL
	209	Eve	1997/1/229	F	47000	1	206
	210	John	2000/12/19	M	68000	3	208
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

新增buddy_id欄位、新增資料請自行練習

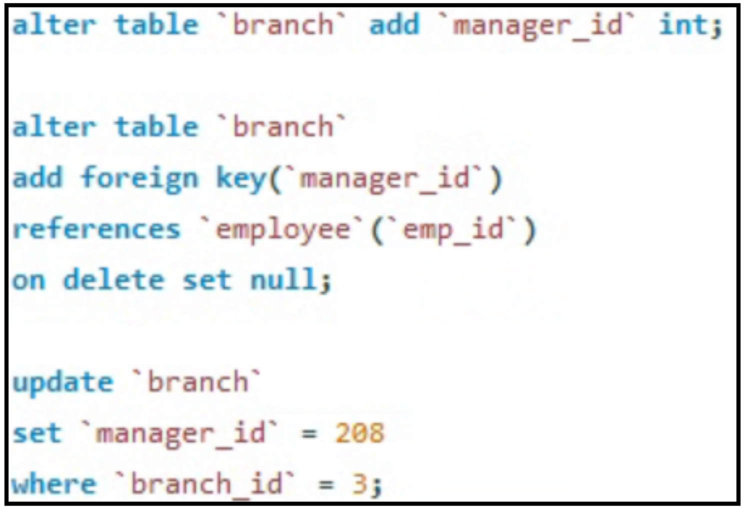
Chap. 7 實際建立並串聯所有表格

Branch

branch_id	name	manager_id
1	研發	206
2	行政	207
3	資訊	208



	branch_id	branch_name	manager_id
▶	1	RD	206
	2	ADM	207
	3	IT	208
*	NULL	NULL	NULL

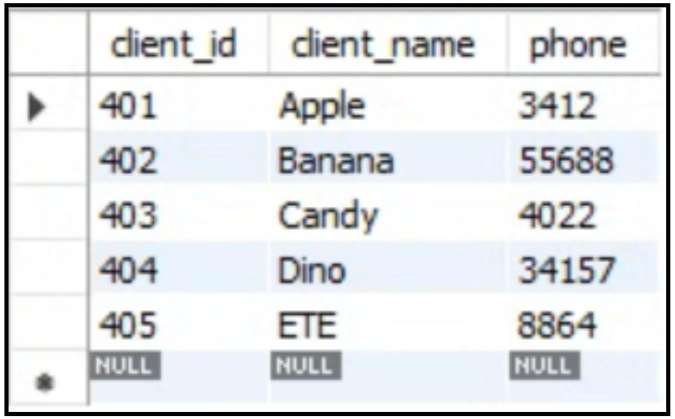


```
alter table `branch` add `manager_id` int;

alter table `branch`
add foreign key(`manager_id`)
references `employee`(`emp_id`)
on delete set null;

update `branch`
set `manager_id` = 208
where `branch_id` = 3;
```

Chap. 7 實際建立並串聯所有表格

Client

client_id	client_name	phone
400	Apple	3412
401	Banana	55688
402	Candy	4022
403	Dino	34157
404	ETE	8864

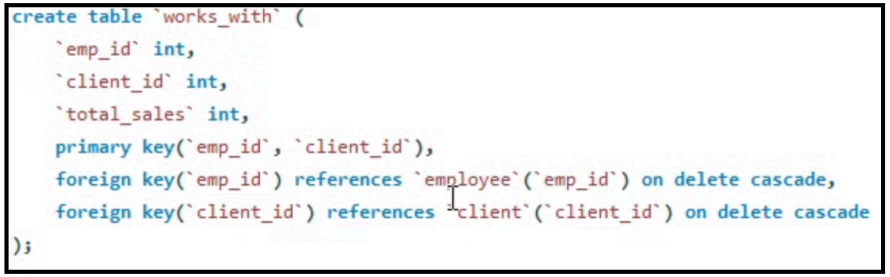


	client_id	client_name	phone
▶	401	Apple	3412
	402	Banana	55688
	403	Candy	4022
	404	Dino	34157
	405	ETE	8864
*	NULL	NULL	NULL

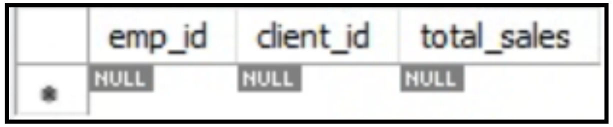
Chap. 7 實際建立並串聯所有表格

Works_with

emp_id	client_id	total_sales
206	400	70000
207	401	24000
208	400	10000
208	403	24000
210	404	88000



```
create table `works_with` (
    `emp_id` int,
    `client_id` int,
    `total_sales` int,
    primary key(`emp_id`, `client_id`),
    foreign key(`emp_id`) references `employee`(`emp_id`) on delete cascade,
    foreign key(`client_id`) references `client`(`client_id`) on delete cascade
);
```



	emp_id	client_id	total_sales
*	NULL	NULL	NULL

恭喜，此時你會發現你不能插入資料到works_with裡面！

（Error code: 1452 Cannot add or update a child row: a foreign key constraint fails）

原因是在連結外鍵時，是不會把原資料庫當中的資料抓取過來，連結外鍵後新增的資料才會自動抓取

所以我們現在把所有的表格都刪除後重來一次！

Chap. 7 實際建立並串聯所有表格

Employee

emp_id	name	birthday	gender	salary	branch_id	buddy_id

Branch

branch_id	name	manager_id

Client

client_id	client_name	phone

Works_with

emp_id	client_id	total_sales

1. 先建立好所有全空的表格，並且設定好相互的關聯性

```
show databases;
use `first_data`;
show tables;

create table `employee` (
`emp_id` int primary key auto_increment,
    `name` varchar(20) not null,
    `birthday` varchar(20),
    `gender` varchar(20),
    `salary` int,
    `branch_id` int,
    `buddy_id` int
);
create table `branch` (
`branch_id` int primary key auto_increment,
    `branch_name` varchar(20),
    `manager_id` int
);
alter table `employee`
add foreign key(`buddy_id`)
references `employee`(`emp_id`)
on delete set null;

alter table `employee`
add foreign key(`branch_id`)
references `branch`(`branch_id`)
on delete set null;

alter table `branch`
add foreign key(`manager_id`)
references `employee`(`emp_id`)
on delete set null;

create table `client` (
`client_id` int primary key,
    `client_name` varchar(20),
    `phone` varchar(20)
);
create table `works_with` (
`emp_id` int,
    `client_id` int,
    `total_sales` int,
    primary key(`emp_id`, `client_id`),
    foreign key(`emp_id`) references `employee`(`emp_id`) on delete cascade,
    foreign key(`client_id`) references `client`(`client_id`) on delete cascade
);
```

Chap. 7 實際建立並串聯所有表格

1. 先建立好所有全空的表格，並且設定好相互的關聯性
2. 開始插入資料

```
insert into `branch` values(1, 'RD', null);
insert into `branch`(`branch_name`, `manager_id`) values('ADM', null);
insert into `branch`(`branch_name`, `manager_id`) values('IT', null);


insert into `employee` values(206, 'Alice', '1999/10/8', 'F', 50000, 1, null);
insert into `employee`(`name`, `birthday`, `gender`, `salary`, `branch_id`, `buddy_id`) values('Bob', '1985/9/16', 'M', 30000, 2, null);
insert into `employee`(`name`, `birthday`, `gender`, `salary`, `branch_id`, `buddy_id`) values('Herry', '2000/12/19', 'M', 35000, 3, null);
insert into `employee`(`name`, `birthday`, `gender`, `salary`, `branch_id`, `buddy_id`) values('Eve', '1997/1/229', 'F', 47000, 1, 206);
insert into `employee`(`name`, `birthday`, `gender`, `salary`, `branch_id`, `buddy_id`) values('John', '2000/12/19', 'M', 68000, 3, 208);

update `branch`
set `manager_id` = 206
where `branch_id` = 1;
update `branch`
set `manager_id` = 207
where `branch_id` = 2;
update `branch`
set `manager_id` = 208
where `branch_id` = 3;

insert into `client` values(400, 'Apple', '3412');
insert into `client` values(401, 'Banana', '55688');
insert into `client` values(402, 'Candy', '4022');
insert into `client` values(403, 'Dino', '34157');
insert into `client` values(404, 'ETE', '8864');

insert into `works_with` values(206, 400, 70000);
insert into `works_with` values(207, 401, 24000);
insert into `works_with` values(208, 400, 10000);
insert into `works_with` values(208, 403, 24000);
insert into `works_with` values(210, 404, 88000);
```

Chap. 7 實際建立並串聯所有表格

Employee

emp_id	name	birthday	gender	salary	branch_id	buddy_id
206	Alice	1999/10/8	F	50000	1	NULL
207	Bob	1985/9/16	M	30000	2	NULL
208	Herry	2000/12/19	M	35000	3	NULL
209	Eve	1997/1/22	F	47000	1	206
210	John	1990/11/10	M	68000	3	208

Branch

branch_id	name	manager_id
1	研發	206
2	行政	207
3	資訊	208

Client

client_id	client_name	phone
400	Apple	3412
401	Banana	55688
402	Candy	4022
403	Dino	34157
404	ETE	8864

Works_with

emp_id	client_id	total_sales
206	**400**	70000
207	**401**	24000
208	**400**	10000
208	**403**	24000
210	**404**	88000

Chap. 8
取得資料

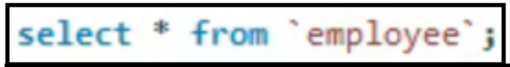
建功高中物理科

cow3690m@tmail.hc.edu.tw

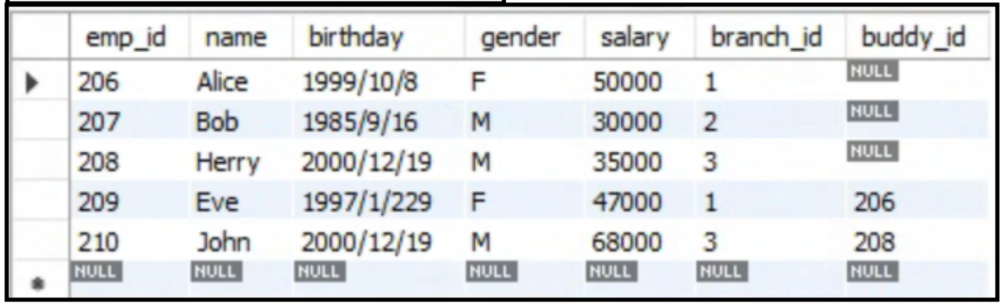
呂昀修

Chap. 8 取得資料（select）

最簡易的就是取得該表格所有資料：
select * from `table_name`;



```
select * from `employee`;
```



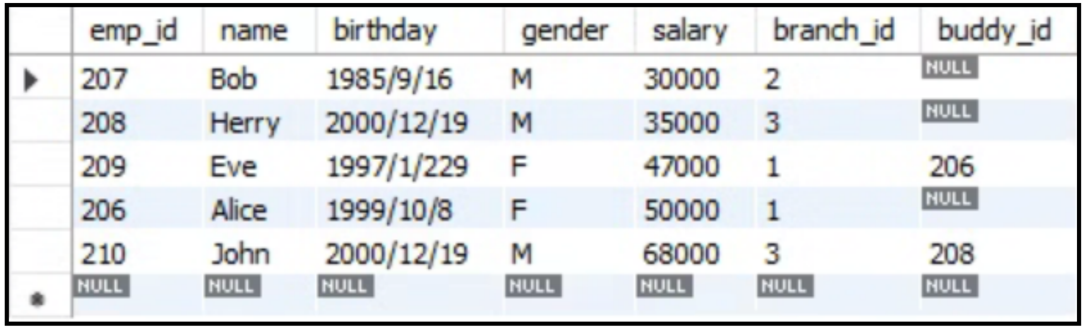
	emp_id	name	birthday	gender	salary	branch_id	buddy_id
▶	206	Alice	1999/10/8	F	50000	1	NULL
	207	Bob	1985/9/16	M	30000	2	NULL
	208	Herry	2000/12/19	M	35000	3	NULL
	209	Eve	1997/1/229	F	47000	1	206
	210	John	2000/12/19	M	68000	3	208
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Chap. 8 取得資料（order by）

依照某條件取得資料：

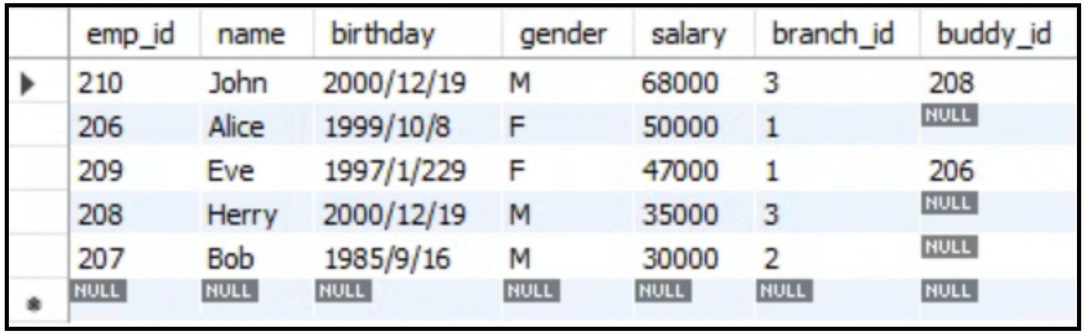
例如：依照薪水低到高取得員工資料：

```
select * from `employee` order by `salary`;
```



	emp_id	name	birthday	gender	salary	branch_id	buddy_id
▶	207	Bob	1985/9/16	M	30000	2	NULL
	208	Herry	2000/12/19	M	35000	3	NULL
	209	Eve	1997/1/229	F	47000	1	206
	206	Alice	1999/10/8	F	50000	1	NULL
	210	John	2000/12/19	M	68000	3	208
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
select * from `employee` order by `salary` desc;
```



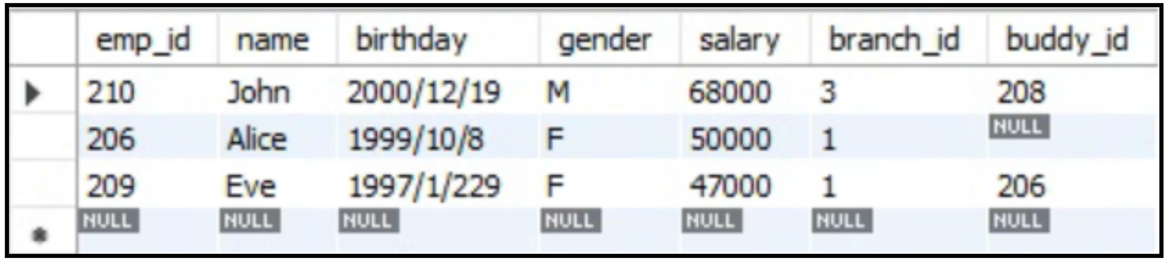
	emp_id	name	birthday	gender	salary	branch_id	buddy_id
▶	210	John	2000/12/19	M	68000	3	208
	206	Alice	1999/10/8	F	50000	1	NULL
	209	Eve	1997/1/229	F	47000	1	206
	208	Herry	2000/12/19	M	35000	3	NULL
	207	Bob	1985/9/16	M	30000	2	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Chap. 8 取得資料（limit）

依照某條件取得資料：

例如：取得薪水前三高的員工名單：

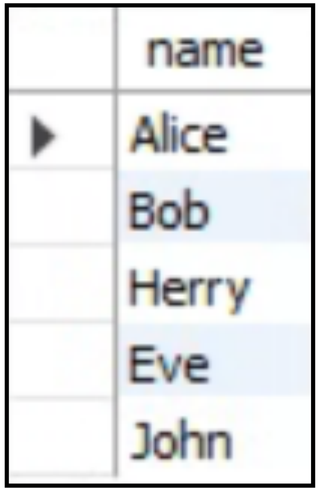
select * from `employee` order by `salary` desc limit 3;



	emp_id	name	birthday	gender	salary	branch_id	buddy_id
▶	210	John	2000/12/19	M	68000	3	208
	206	Alice	1999/10/8	F	50000	1	NULL
	209	Eve	1997/1/229	F	47000	1	206
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

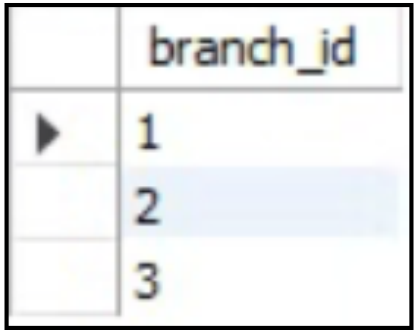
Chap. 8 取得資料（distinct）

取得某欄位數據：
例如：取得所有員工的名字
select `name` from `employee`;



	name
▶	Alice
	Bob
	Herry
	Eve
	John

取得某欄位不重複數據：
例如：從員工表格中取得部門id，並且不要重複
select distinct `branch_id` from `employee`;



	branch_id
▶	1
	2
	3

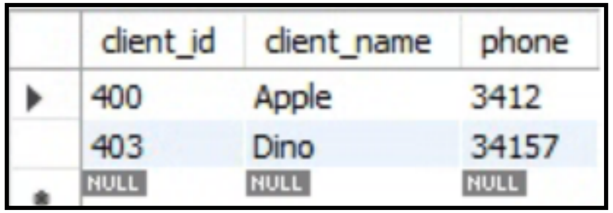
Chap. 8 取得資料（count/avg/sum/max/min/like）

取得員工人數：
select count(*) from `employee`;

取得生日晚於2000/1/1的女員工人數：
select count(*) from `employee` where `birthday` > ‘2000/1/1’ and `gender` = ‘F’;

取得員工的薪水XXX：
select avg(`salary`) from `employee`;
select sum(`salary`) from `employee`;
select max(`salary`) from `employee`;
select min(`salary`) from `employee`;

取得電話開頭為34的客戶：
select * from `client` where `phone` like ’34%’;



	client_id	client_name	phone
▶	400	Apple	3412
	403	Dino	34157
*	NULL	NULL	NULL

取得生日在12月的員工名稱：
select `name` from `employee` where `birthday` like ‘%/12/%’;
也可以用一個底線「_」代表前面有一個字元
select `name` from `employee` where `birthday` like ‘_____12%’;（這邊有五個底線，因為XXXX/12/XX）

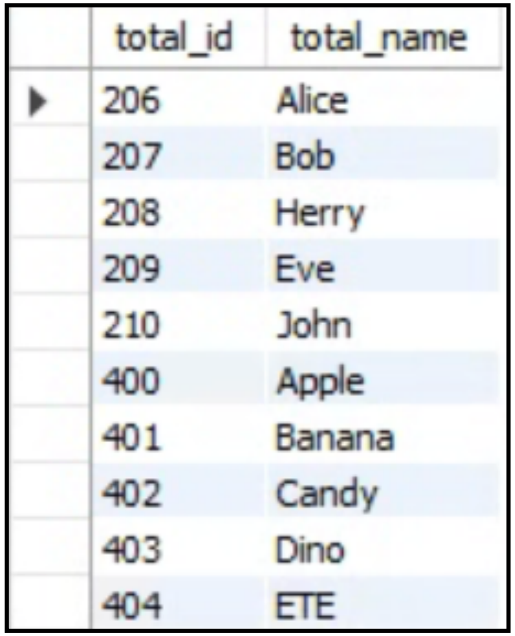
Chap. 8 取得資料（union）

取得員工+客戶名稱

```
select `name` from `employee`
union
select `client_name` from `client`;
```

取得全部名稱及id後，將欄位的header name改掉

```
select `emp_id` as `total_id`, `name` as `total_name`
from `employee`
union
select `client_name` from `client`;
```

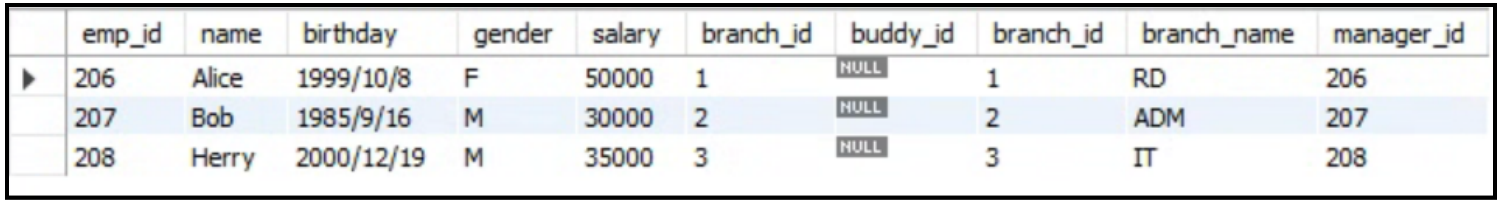


	total_id	total_name
▶	206	Alice
	207	Bob
	208	Herry
	209	Eve
	210	John
	400	Apple
	401	Banana
	402	Candy
	403	Dino
	404	ETE

Chap. 8 取得資料（join）

取得部經理資料

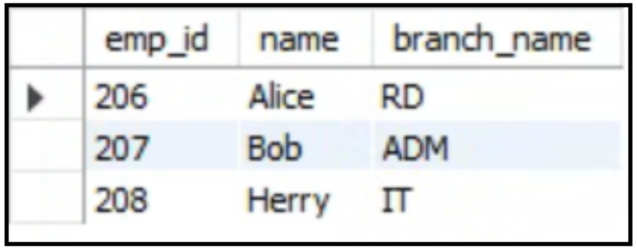
```
select * from `employee` join `branch` on `emp_id` = `manager_id`;
```



	emp_id	name	birthday	gender	salary	branch_id	buddy_id	branch_id	branch_name	manager_id
▶	206	Alice	1999/10/8	F	50000	1	NULL	1	RD	206
	207	Bob	1985/9/16	M	30000	2	NULL	2	ADM	207
	208	Herry	2000/12/19	M	35000	3	NULL	3	IT	208

也可以

```
select `employee`.`emp_id`, `employee`.`name`, `branch`.`branch_name`
from `employee` join `branch`
on `employee`.`emp_id` = `branch`.` manager_id`;
```



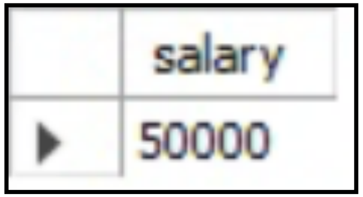
	emp_id	name	branch_name
▶	206	Alice	RD
	207	Bob	ADM
	208	Herry	IT

Chap. 8 取得資料（子查詢subquery）

若我想要查詢：

某員工的buddy_id的薪水資料

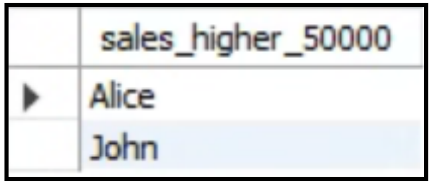
```
select `salary`
from `employee`
where `emp_id` = (
    select `buddy_id`
    from `employee`
    where `emp_id` = 209
);
```



salary
50000

找出對單一客戶銷售額超過50000的員工名稱

```
select `name` as `sales_higher_50000` from `employee`
where `emp_id` in(
    select `emp_id` from `works_with`
    where `total_sales` > 50000
);
```



sales_higher_50000
Alice
John

Chap. 9
python connect MySQL

建功高中物理科
cow3690m@tmail.hc.edu.tw

呂昀修

Chap. 9 python connect MySQL

必須要先安裝mysql

```
python -m pip install mysql-connector-python

import mysql.connector
connection = mysql.connector.connection(
    host=‘localhost’,
    port=‘3306’,
    user=‘root’,
    password=‘0000’, # 這是我的密碼
    database=‘first_data’ # 預設要連線的database
)
cursor = connection.cursor()

cursor.execute(“show databases;”)
for x in cursor:
    print(x)

cursor.close() # 結尾必要
connection.close() # 結尾必要
```

如果今天你是在terminal當中向SQL query
每次print cursor後，cursor的記憶會清空
若要重複使用cursor.execute()後的結果
務必設定變數！