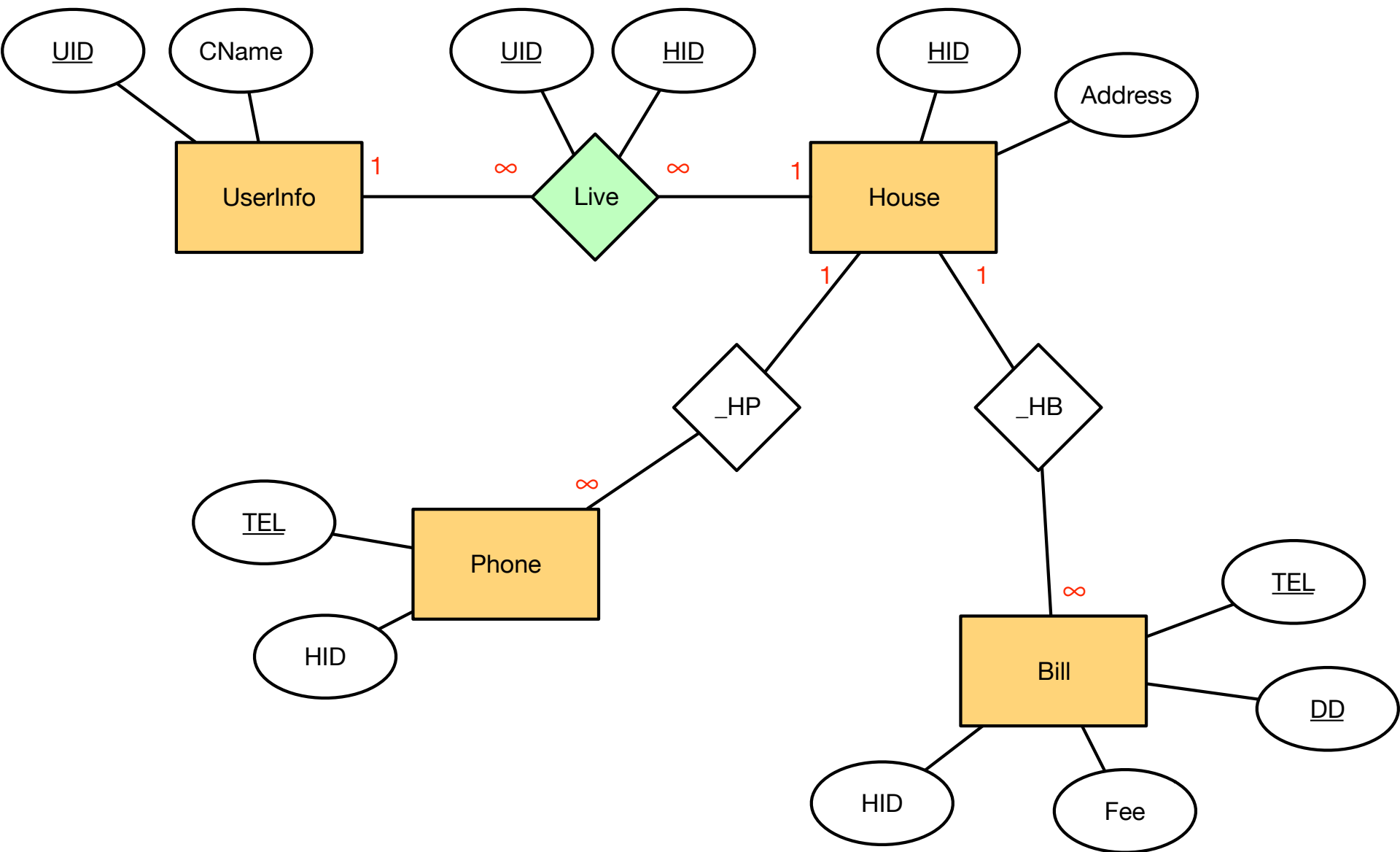


SQL Command

朱克剛



SELECT基本語法

SELECT 欄位

FROM 資料表

WHERE 條件

沒有條件時，
WHERE 可不寫

SELECT 單一資料表

顯示 UserInfo 資料表中的所有資料

```
SELECT *  
FROM userinfo
```

顯示 UserInfo 資料表中的個別欄位

```
SELECT uid, cname  
FROM userinfo
```

LIKE

需要所有姓李的基本資料

```
SELECT *  
FROM userinfo  
WHERE cname LIKE '李%'
```

SQL command 的字串
為單引號

列出所有姓李的以及身份證字號後四碼為1234

```
SELECT *  
FROM userinfo  
WHERE cname LIKE '李%' AND uid LIKE '%1234'
```

=, >, <, >=, <=, <>

列出電話費超過500元的資料

```
SELECT *  
FROM bill  
WHERE fee > 500
```

列出電話費在500~1000元的資料

```
SELECT *  
FROM bill  
WHERE fee >= 500 AND fee < 1000
```

BETWEEN AND

列出電話費1000-2000元的資料

```
SELECT *  
FROM bill  
WHERE fee BETWEEN 500 AND 1000
```



包含500與
1000

IN、NOT IN

列出王大明與李大媽的基本資料

```
SELECT *  
FROM userinfo  
WHERE cname IN ('王大明', '李大媽')
```

||

```
SELECT *  
FROM userinfo  
WHERE cname = '王大明' OR cname = '李大媽'
```


排序 ORDER BY

將電話號碼由小到大排序

```
SELECT *  
FROM phone  
ORDER BY tel
```

將電話號碼由大到小排序

```
SELECT *  
FROM phone  
ORDER BY tel DESC
```

MYSQL按照中文筆畫數排序

MySQL預設編碼為utf8，若要按照中文筆畫數排序，轉成big5再排即可

```
select *  
from userinfo  
order by convert(cname using big5)
```

IS NULL 、 IS NOT NULL

列出使用者姓名為輸入的資料

```
SELECT *  
FROM userinfo  
WHERE cname IS NULL OR cname = "
```

在 MS-SQL 中，空字串不會轉成NULL

ORACLE沒有空字串

Oracle 沒有空字串

- Oracle會將空字串內容自動轉換成 NULL

UID	CNAME
101	<NULL>
102	

填 NULL

填空字串

```
SELECT *  
FROM table  
WHERE cname = ''
```

MS-SQL 會顯示 102 那一筆
Oracle 則查無資料

函數 count()

列出UserInfo資料表中有多少筆資料

```
SELECT count(*)  
FROM userinfo
```

關連線處理 - JOIN

四種類型

- **INNER JOIN**
 - 關連線左右兩邊均有資料
- **LEFT OUTER JOIN**
 - 關連線左側資料較多
- **RIGHT OUTER JOIN**
 - 關連線右側資料較多
- **CROSS JOIN**
 - 未設定關連而形成交叉對應

INNER JOIN

列出身份證字號、姓名、住址、電話

```
SELECT
    userinfo.uid,
    cname,
    address,
    tel
FROM
    userinfo, live, house, phone
WHERE
    userinfo.uid = live.uid AND
    live.hid = house.hid AND
    house.hid = phone.hid
```

OUTER JOIN

列出身份證字號、姓名、住址、電話

```
SELECT
    userinfo.uid,
    cname,
    address,
    tel
FROM
    userinfo LEFT JOIN live
        ON userinfo.uid = live.uid
    LEFT JOIN house
        ON live.hid = house.hid
    LEFT JOIN phone
        ON house.hid = phone.hid
```


別名

列出身份證字號、姓名、住址、電話

```
SELECT
    a.uid AS "身份證字號",
    cname AS "姓名",
    address AS "住址",
    tel AS "電話"
FROM
    userinfo AS a, live AS b, house AS c, phone AS d
WHERE
    a.uid = b.uid AND
    b.hid = c.hid AND
    c.hid = d.hid
ORDER BY a.uid
```

AS 可省略. 別名除了單一個純英文字外, 前後必須以雙引號夾住

Oracle 不可打 AS
MS-SQL 必須打 AS

別名 - MySQL

列出身份證字號、姓名、住址、電話

```
SELECT
    a.uid AS '身份證字號',
    cname AS '姓名',
    address AS '住址',
    tel AS '電話'
FROM
    userinfo AS a, live AS b, house AS c, phone AS d
WHERE
    a.uid = b.uid AND
    b.hid = c.hid AND
    c.hid = d.hid
ORDER BY a.uid
```

群組 GROUP BY

列出每支電話的總費用

```
SELECT tel, sum(fee)
FROM bill
GROUP BY tel
```

不重複資料 DISTINCT

列出所有的姓氏

```
SELECT DISTINCT left(cname, 1)
FROM userinfo
```

補充 left(), right(), substring()

列出每個姓氏有幾筆資料

```
SELECT lastname, count(*) AS n
FROM (
    SELECT left(cname, 1) as lastname
    FROM userinfo
) AS a
GROUP BY lastname
```

GROUP BY 陷阱

列出每支電話哪一個月電話費最高

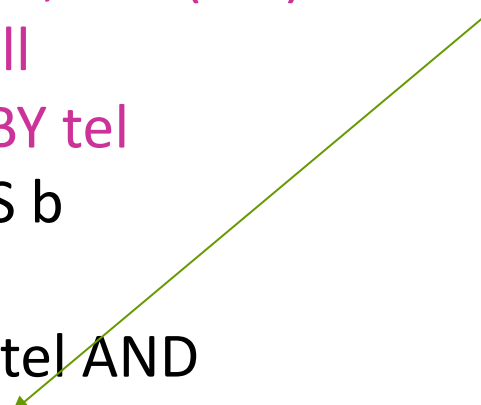
```
SELECT tel, max(fee), dd  
FROM bill  
GROUP BY tel, dd
```

如果有兩個月的電話費都同樣是最高時，查詢結果便會錯誤

正解

列出每支電話哪一個月電話費最高

```
SELECT a.tel, max_fee, dd
FROM (
    SELECT tel, max(fee) AS max_fee
    FROM bill
    GROUP BY tel
) AS a, bill AS b
WHERE
    a.tel = b.tel AND
    a.max_fee = b.fee
```



練習

請列出每個人有多少支電話？

請列出每個人有多少地方住？

UNION ALL

目的：合併兩個一模一樣的查詢結果

```
SELECT * FROM userinfo WHERE uid = 'A01'  
UNION ALL  
SELECT 'TTT', '測試員'
```


TOP

請列出繳費金額最高的前三支電話號碼

```
SELECT TOP 3 *  
FROM bill  
ORDER BY fee DESC
```

請列出前百分之50的會員資料

```
SELECT TOP 50 PERCENT *  
FROM userinfo
```

LIMIT (MySQL)

請列出繳費金額最高的前三支電話號碼

```
SELECT *  
FROM bill  
ORDER BY fee DESC  
LIMIT 3
```

HAVING

列出平均繳費金額超過300元的電話資料

```
SELECT tel, avg(fee)
FROM bill
GROUP BY tel
HAVING avg(fee) > 300
```



```
SELECT * FROM (
    SELECT tel, avg(fee) AS avg_fee
    FROM bill
    GROUP BY tel
) AS a
WHERE avg_fee > 300
```

練習

請列出電話帳單金額總額最高的電話相關資料

視觀表 View

建立一個住在台北市民眾資料的視觀表

```
CREATE VIEW v_userinfo_taipei AS
  SELECT userinfo.uid, cname, address
  FROM userinfo, live, house
  WHERE
    userinfo.uid = live.uid AND
    live.hid = house.hid AND
    house.address LIKE '台北市%'
```

視觀表用法

查詢目前居住在台北市的人口有多少人

```
SELECT count(*)  
FROM v_userinfo_taipei
```

INSERT INTO

插入一筆新的資料到 userinfo 資料表

```
INSERT INTO userinfo VALUES (  
    'A03',  
    '王小毛'  
)
```

```
INSERT INTO userinfo (uid) VALUES (  
    'A04'  
)
```

複製資料 (1)

將台北市民眾資料複製到另外一個資料表
(`new_table`必須先存在)

```
INSERT INTO new_table (uid, cname)
  SELECT uid, cname
  FROM v_userinfo_taipei
```


複製資料 (2)

新的資料表不需要事先存在

SQL Server

```
select * into new_table from userinfo
```

MySQL

```
create new_table select * from userinfo
```

UPDATE

更新資料表中所有資料

```
UPDATE userinfo SET cname = NULL
```

更新特定資料

- 將 A03 的姓名改為孫小毛，身份證字號改為B01

```
UPDATE userinfo SET  
    cname = '孫小毛',  
    uid = 'B01'  
WHERE uid = 'A03'
```

DELETE

刪除所有電話帳單資料

```
DELETE FROM bill  
TRUNCATE TABLE bill
```

刪除孫小毛資料

```
DELETE FROM userinfo  
WHERE uid = 'B01'
```

COMMIT & ROLLBACK

BEGIN TRANSACTION

- 宣告交易開始

COMMIT

- 確認交易成功

ROLLBACK

- 交易失敗，恢復交易前狀態

範例：

```
BEGIN TRANSACTION
```

```
    DELETE FROM userinfo WHERE uid = 'A01'
```

```
ROLLBACK
```

```
// A01 資料還在
```

COMMIT & ROLLBACK - MySQL

START TRANSACTION

- 宣告交易開始

COMMIT

- 確認交易成功

ROLLBACK

- 交易失敗，恢復交易前狀態

範例：

```
START TRANSACTION;  
    DELETE FROM userinfo WHERE uid = 'A01';  
ROLLBACK;  
// A01 資料還在
```