資料庫實務

109-1四資工二A 4080E007 蔡德龍

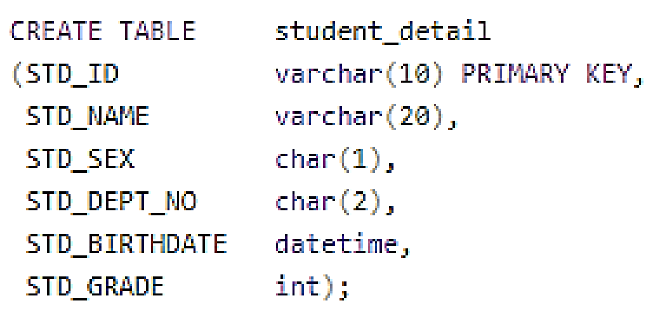
目录

contents

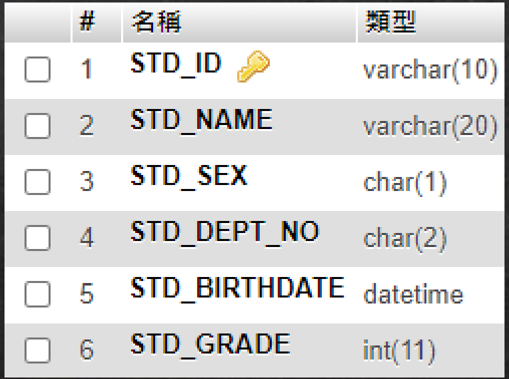
CREATE TABLE

CREATE TABLE ：

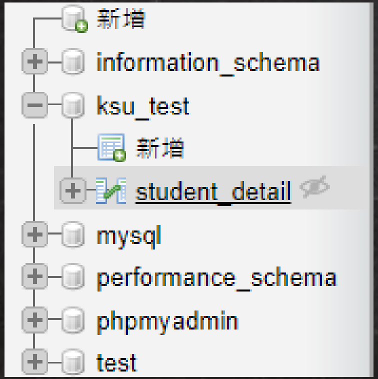
CREATE TABLE “表格名”
(“欄位 1” “欄位 1 資料種類”,
 “欄位 2” “欄位 2 資料種類",
 ...);



```
CREATE TABLE      student_detail
(STD_ID           varchar(10) PRIMARY KEY,
 STD_NAME         varchar(20),
 STD_SEX          char(1),
 STD_DEPT_NO      char(2),
 STD_BIRTHDATE    datetime,
 STD_GRADE        int);
```



	#	名稱	類型
□	1	STD_ID 🔑	varchar(10)
□	2	STD_NAME	varchar(20)
□	3	STD_SEX	char(1)
□	4	STD_DEPT_NO	char(2)
□	5	STD_BIRTHDATE	datetime
□	6	STD_GRADE	int(11)

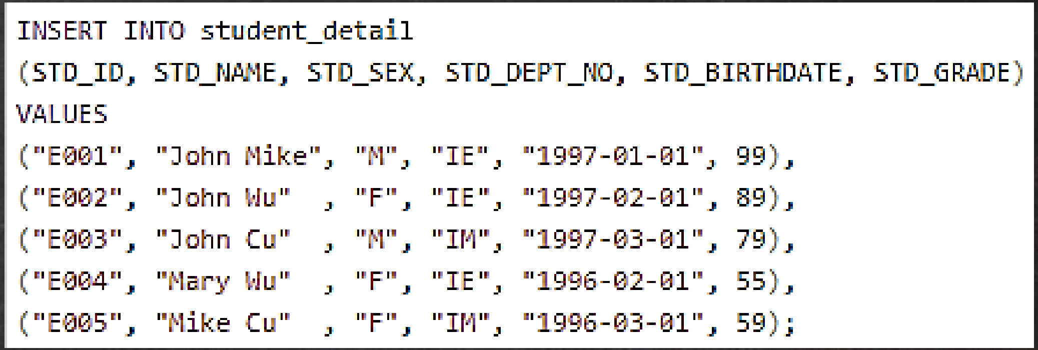


建立資料表

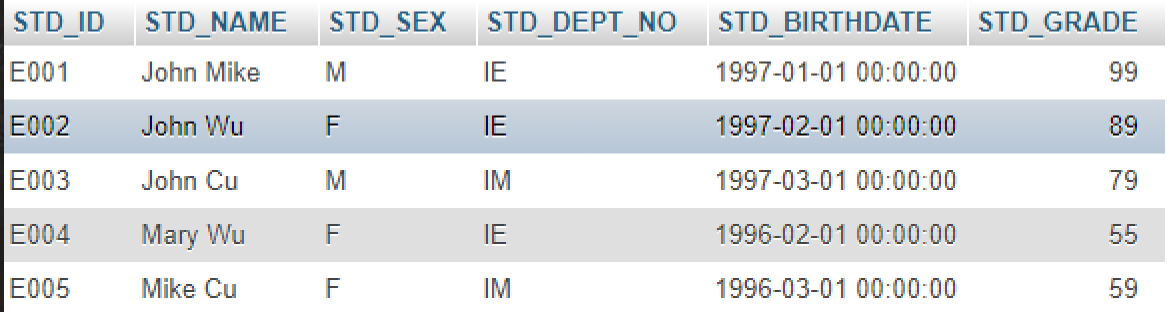
INSERT TABLE

INSERT TABLE：

INSERT INTO "表格名"
("欄位1", "欄位2", ...)
VALUES ("值1", "值2", ...);



```
INSERT INTO student_detail
(STD_ID, STD_NAME, STD_SEX, STD_DEPT_NO, STD_BIRTHDATE, STD_GRADE)
VALUES
("E001", "John Mike", "M", "IE", "1997-01-01", 99),
("E002", "John Wu"  , "F", "IE", "1997-02-01", 89),
("E003", "John Cu"  , "M", "IM", "1997-03-01", 79),
("E004", "Mary Wu"  , "F", "IE", "1996-02-01", 55),
("E005", "Mike Cu"  , "F", "IM", "1996-03-01", 59);
```



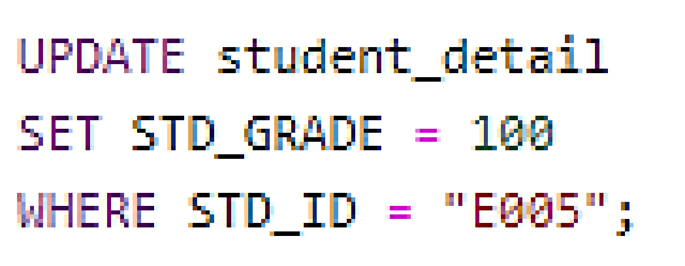
STD_ID	STD_NAME	STD_SEX	STD_DEPT_NO	STD_BIRTHDATE	STD_GRADE
E001	John Mike	M	IE	1997-01-01 00:00:00	99
E002	John Wu	F	IE	1997-02-01 00:00:00	89
E003	John Cu	M	IM	1997-03-01 00:00:00	79
E004	Mary Wu	F	IE	1996-02-01 00:00:00	55
E005	Mike Cu	F	IM	1996-03-01 00:00:00	59

新增資料

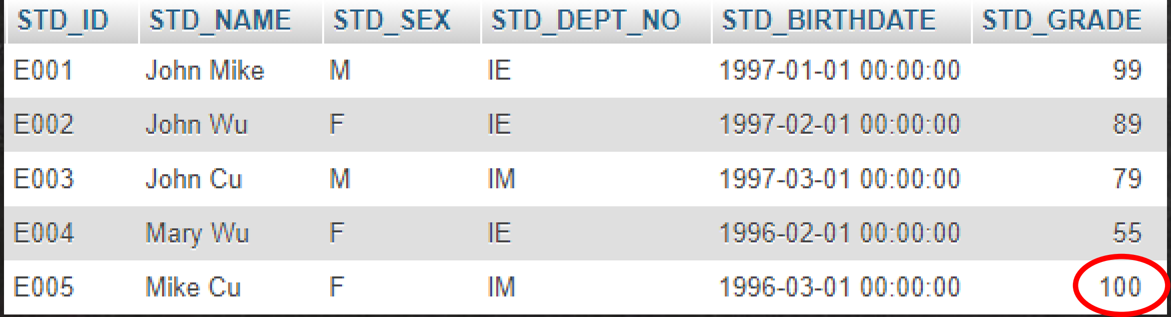
UPDATE

UPDATE：

```
UPDATE "表格名"
SET "欄位1" = [新值]
WHERE "條件";
```



```
UPDATE student_detail
SET STD_GRADE = 100
WHERE STD_ID = "E005";
```



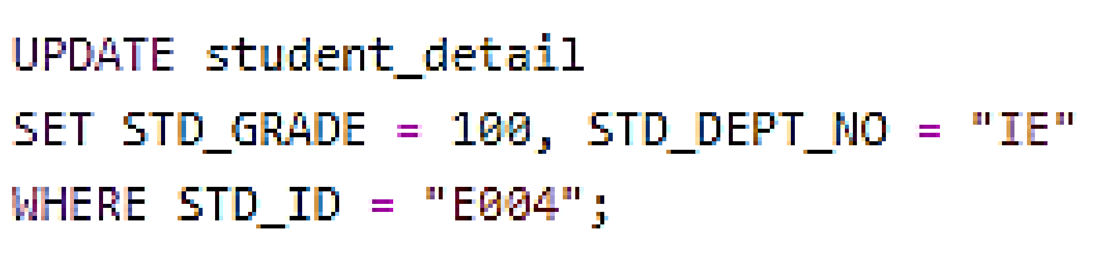
STD_ID	STD_NAME	STD_SEX	STD_DEPT_NO	STD_BIRTHDATE	STD_GRADE
E001	John Mike	M	IE	1997-01-01 00:00:00	99
E002	John Wu	F	IE	1997-02-01 00:00:00	89
E003	John Cu	M	IM	1997-03-01 00:00:00	79
E004	Mary Wu	F	IE	1996-02-01 00:00:00	55
E005	Mike Cu	F	IM	1996-03-01 00:00:00	100

更新資料

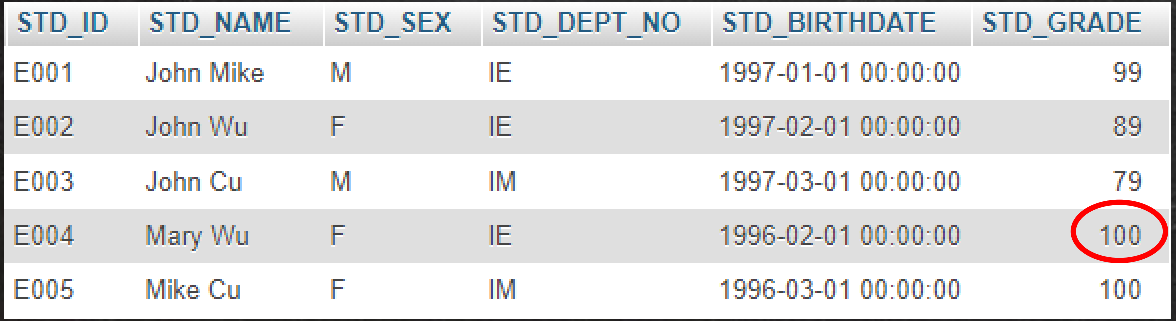
UPDATE

UPDATE：

```
UPDATE "表格名"
SET "欄位1" = [新值]
WHERE "條件";
```



```
UPDATE student_detail
SET STD_GRADE = 100, STD_DEPT_NO = "IE"
WHERE STD_ID = "E004";
```



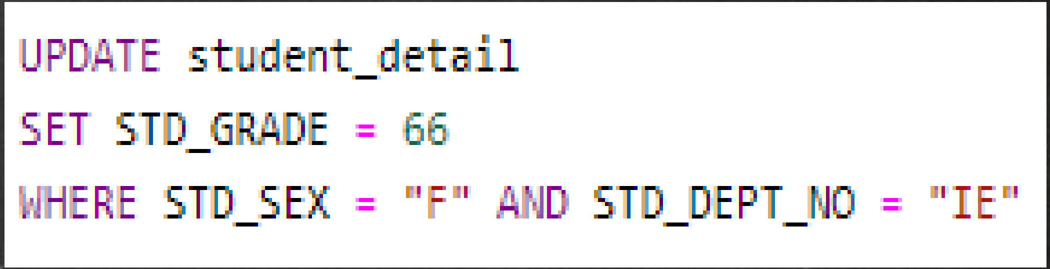
STD_ID	STD_NAME	STD_SEX	STD_DEPT_NO	STD_BIRTHDATE	STD_GRADE
E001	John Mike	M	IE	1997-01-01 00:00:00	99
E002	John Wu	F	IE	1997-02-01 00:00:00	89
E003	John Cu	M	IM	1997-03-01 00:00:00	79
E004	Mary Wu	F	IE	1996-02-01 00:00:00	100
E005	Mike Cu	F	IM	1996-03-01 00:00:00	100

更新資料

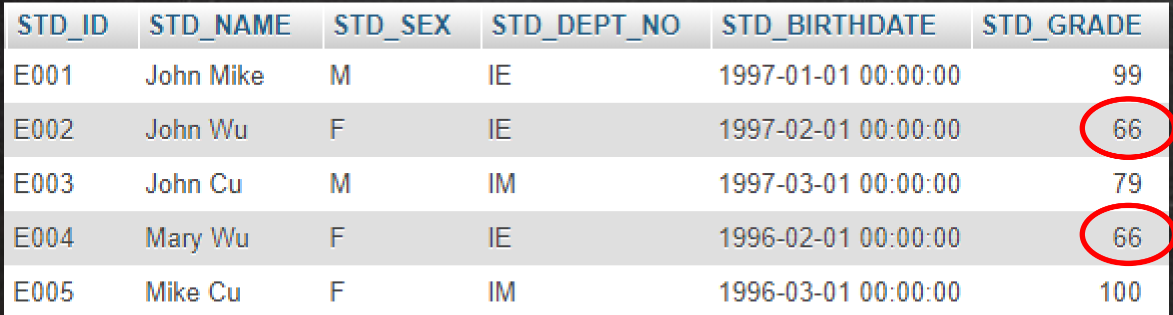
UPDATE

UPDATE：

```
UPDATE "表格名"
SET "欄位1" = [新值]
WHERE "條件";
```



```
UPDATE student_detail
SET STD_GRADE = 66
WHERE STD_SEX = "F" AND STD_DEPT_NO = "IE"
```



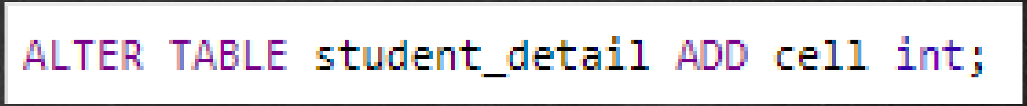
STD_ID	STD_NAME	STD_SEX	STD_DEPT_NO	STD_BIRTHDATE	STD_GRADE
E001	John Mike	M	IE	1997-01-01 00:00:00	99
E002	John Wu	F	IE	1997-02-01 00:00:00	66
E003	John Cu	M	IM	1997-03-01 00:00:00	79
E004	Mary Wu	F	IE	1996-02-01 00:00:00	66
E005	Mike Cu	F	IM	1996-03-01 00:00:00	100

更新資料

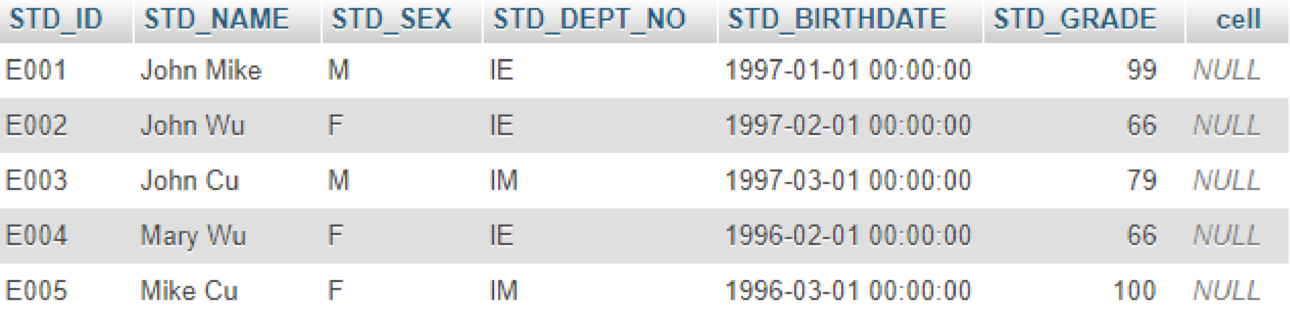
ALTER TABLE

ALTER TABLE：

ALTER TABLE "table_name"
[改變方式];



```
ALTER TABLE student_detail ADD cell int;
```



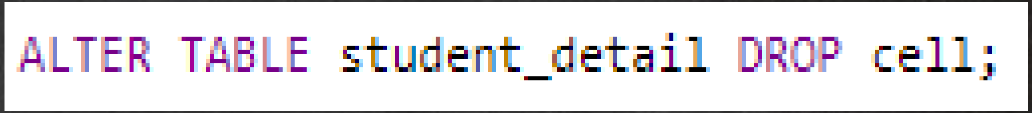
STD_ID	STD_NAME	STD_SEX	STD_DEPT_NO	STD_BIRTHDATE	STD_GRADE	cell
E001	John Mike	M	IE	1997-01-01 00:00:00	99	*NULL*
E002	John Wu	F	IE	1997-02-01 00:00:00	66	*NULL*
E003	John Cu	M	IM	1997-03-01 00:00:00	79	*NULL*
E004	Mary Wu	F	IE	1996-02-01 00:00:00	66	*NULL*
E005	Mike Cu	F	IM	1996-03-01 00:00:00	100	*NULL*

新增欄位

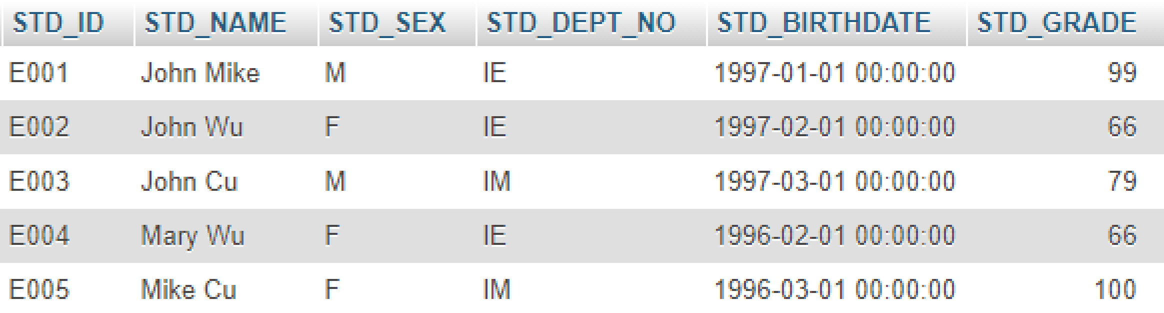
ALTER TABLE

ALTER TABLE：

ALTER TABLE "table_name"
[改變方式];



```
ALTER TABLE student_detail DROP cell;
```



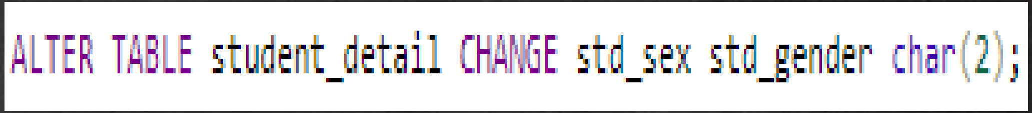
STD_ID	STD_NAME	STD_SEX	STD_DEPT_NO	STD_BIRTHDATE	STD_GRADE
E001	John Mike	M	IE	1997-01-01 00:00:00	99
E002	John Wu	F	IE	1997-02-01 00:00:00	66
E003	John Cu	M	IM	1997-03-01 00:00:00	79
E004	Mary Wu	F	IE	1996-02-01 00:00:00	66
E005	Mike Cu	F	IM	1996-03-01 00:00:00	100

刪除欄位

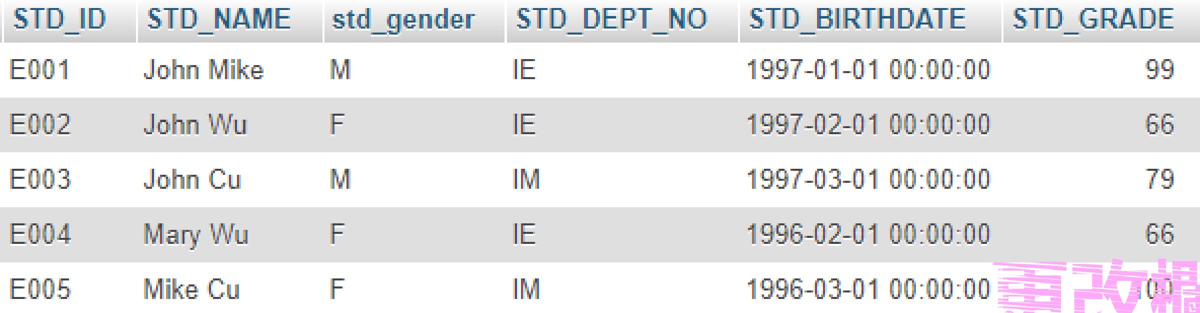
ALTER TABLE

ALTER TABLE：

ALTER TABLE "table_name"
[改變方式];



```
ALTER TABLE student_detail CHANGE std_sex std_gender char(2);
```



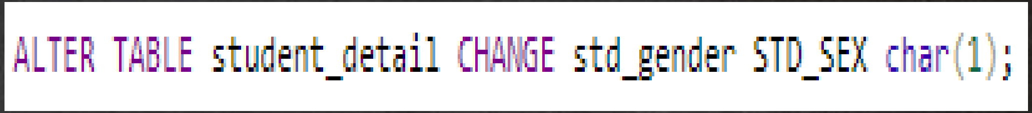
STD_ID	STD_NAME	std_gender	STD_DEPT_NO	STD_BIRTHDATE	STD_GRADE
E001	John Mike	M	IE	1997-01-01 00:00:00	99
E002	John Wu	F	IE	1997-02-01 00:00:00	66
E003	John Cu	M	IM	1997-03-01 00:00:00	79
E004	Mary Wu	F	IE	1996-02-01 00:00:00	66
E005	Mike Cu	F	IM	1996-03-01 00:00:00	[illegible]

更改欄位名稱 與 型別

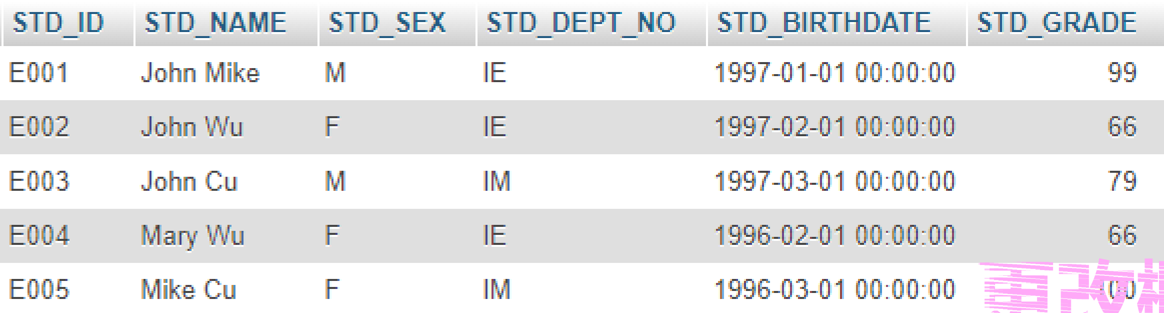
ALTER TABLE

ALTER TABLE：

ALTER TABLE "table_name"
[改變方式];



```
ALTER TABLE student_detail CHANGE std_gender STD_SEX char(1);
```



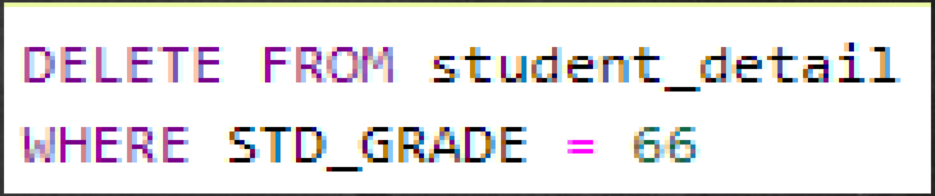
STD_ID	STD_NAME	STD_SEX	STD_DEPT_NO	STD_BIRTHDATE	STD_GRADE
E001	John Mike	M	IE	1997-01-01 00:00:00	99
E002	John Wu	F	IE	1997-02-01 00:00:00	66
E003	John Cu	M	IM	1997-03-01 00:00:00	79
E004	Mary Wu	F	IE	1996-02-01 00:00:00	66
E005	Mike Cu	F	IM	1996-03-01 00:00:00	00

更改欄位名稱 與 型別

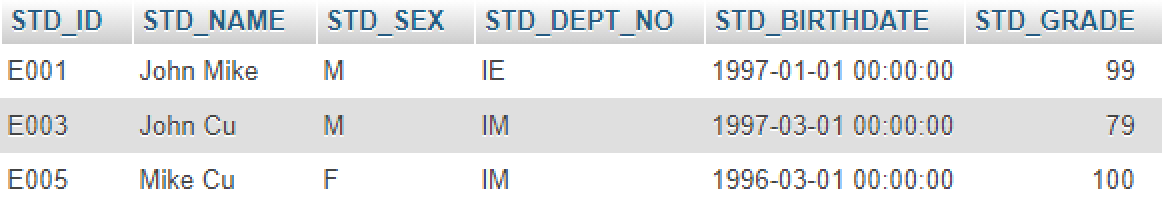
DELETE FROM

DELETE FROM：

DELETE FROM "表格名"
WHERE "條件";



```
DELETE FROM student_detail
WHERE STD_GRADE = 66
```



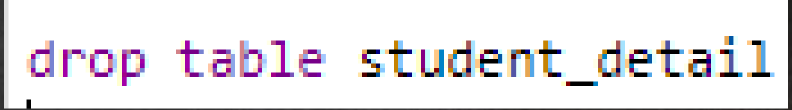
STD_ID	STD_NAME	STD_SEX	STD_DEPT_NO	STD_BIRTHDATE	STD_GRADE
E001	John Mike	M	IE	1997-01-01 00:00:00	99
E003	John Cu	M	IM	1997-03-01 00:00:00	79
E005	Mike Cu	F	IM	1996-03-01 00:00:00	100

刪除資料

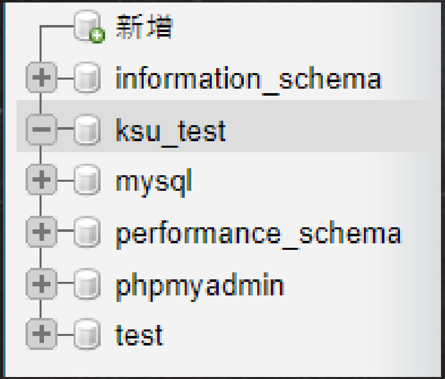
DROP TABLE

DROP TABLE：

DROP TABLE "表格名";



```
drop table student_detail
```

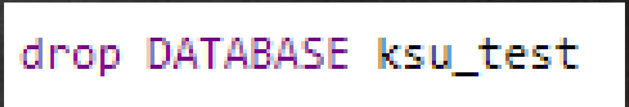


刪除資料表

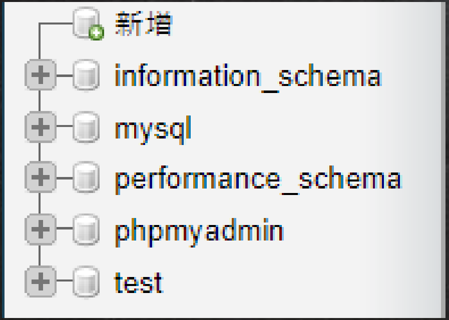
DROP TABLE

DROP TABLE：

DROP TABLE "表格名";



```
drop DATABASE ksu_test
```



新增
information_schema
mysql
performance_schema
phpmyadmin
test

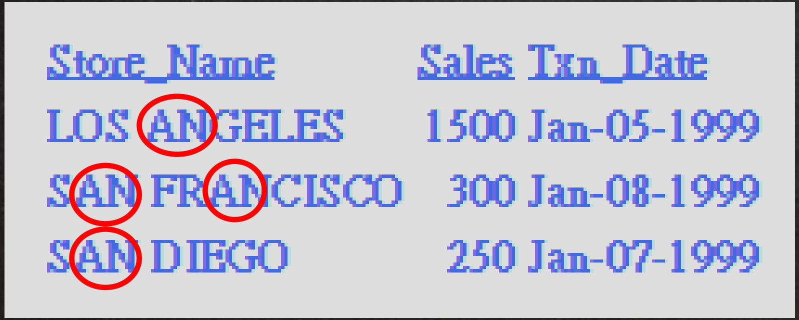
刪除資料庫

LIKE

LIKE：

```
SELECT "欄位名"
FROM "表格名"
WHERE "欄位名" LIKE {模式};
```

```
SELECT *
FROM Store_Information
WHERE store_name LIKE '%AN%';
```



Store_Name	Sales	Txn_Date
LOS ANGELES	1500	Jan-05-1999
SAN FRANCISCO	300	Jan-08-1999
SAN DIEGO	250	Jan-07-1999

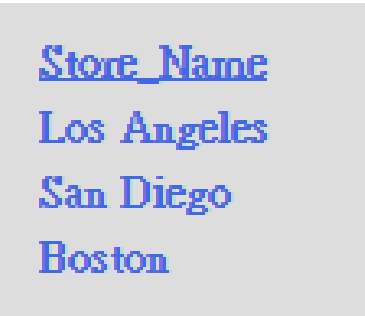
模糊查詢

DISTINCT

DISTINCT ：

```
SELECT DISTINCT "欄位名"
FROM "表格名";
```

```
SELECT DISTINCT Store_Name FROM Store_Information;
```



Store_Name
Los Angeles
San Diego
Boston

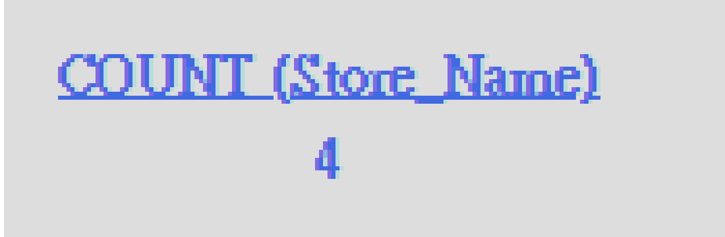
不重複資料

COUNT

COUNT：

```
SELECT COUNT("欄位名")
FROM "表格名";
```

```
SELECT COUNT (Store_Name)
FROM Store_Information
WHERE Store_Name IS NOT NULL;
```

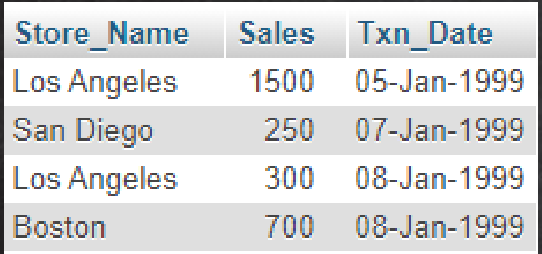


COUNT (Store_Name)
4

計算表格內有多少筆資料

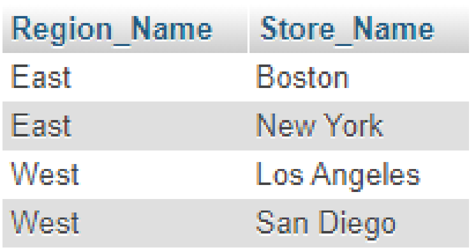
SQL表格連結(Join)

Store_Information

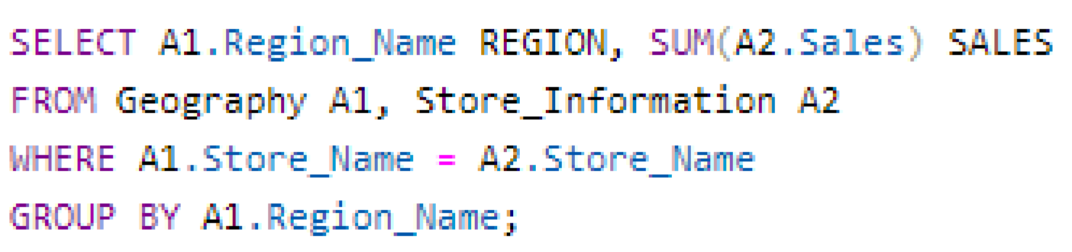


Store_Name	Sales	Txn_Date
Los Angeles	1500	05-Jan-1999
San Diego	250	07-Jan-1999
Los Angeles	300	08-Jan-1999
Boston	700	08-Jan-1999

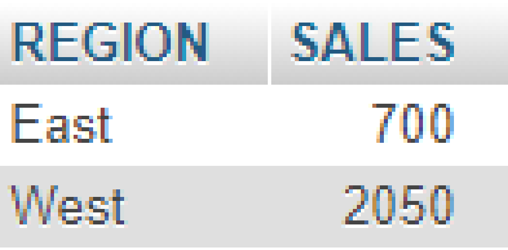
Geography



Region_Name	Store_Name
East	Boston
East	New York
West	Los Angeles
West	San Diego



```
SELECT A1.Region_Name REGION, SUM(A2.Sales) SALES
FROM Geography A1, Store_Information A2
WHERE A1.Store_Name = A2.Store_Name
GROUP BY A1.Region_Name;
```



REGION	SALES
East	700
West	2050

表格連結

連結兩個以上的資料表格

※ 連結的欄位：

◎ 同屬性 ◎ 同型別，同長度

```
SELECT ksu_std_name, ksu_std_department, dept_name
FROM
ksu_std_table as a, dept_detail as b
WHERE
a.ksu_std_department = b.dept_id
```

新增簡稱，資料多時方便使用

連結兩個以上的資料表格

※ 連結的欄位：

◎ 同屬性 ◎ 同型別，同長度

```
SELECT ksu_std_name, ksu_std_department, dept_name
FROM
ksu_std_table, dept_detail
WHERE
ksu_std_department = dept_id
```

去掉簡稱，依樣可以執行

混淆 簡稱(一)

※ 連結的欄位：

◎ 同屬性 ◎ 同型別，同長度

```
SELECT ksu_std_name, a.ksu_std_department,
dept_name
FROM
ksu_std_table as a, dept_detail as b
WHERE
a.ksu_std_department = b.ksu_std_department
```

混淆 簡稱(二)

※ 連結的欄位：

◎ 同屬性 ◎ 同型別，同長度

```
SELECT
    a.ksu_std_name,
    a.ksu_std_department,
    b.dept_name
FROM
    ksu_std_table as a,
    dept_detail as b
WHERE
    a.ksu_std_department = b.ksu_std_department
```

計算各系別欄位(個數)

※ 連結的欄位：

◎ 同屬性 ◎ 同型別，同長度

```
SELECT ksu_std_department, COUNT(*) as 個數
FROM `ksu_std_table`
WHERE ksu_std_department = "IE" OR
ksu_std_department = "IM"
GROUP BY ksu_std_department
```

計算各系別欄位(個數)

計算各系別欄位(個數) - 簡數

※ 連結的欄位：

◎ 同屬性 ◎ 同型別，同長度

```
SELECT a.ksu_std_department, COUNT(*) as 個數
FROM ksu_std_table as a, dept_detail as b
WHERE a.ksu_std_department = b.ksu_std_department
GROUP BY a.ksu_std_department
```

計算各系別欄位(個數) - 簡數

PHP連接SQL

```
<?php
 $name=str_replace("'","''",$_REQUEST['name']);
 $id=str_replace ("'","''",$_REQUEST['id']);
 $db_host = "localhost";
 $db_name = "ksu_database";
 $db_table = "ksu_cstd_table";
 $db_user = "root";
 $db_password = "";
 $conn = mysqli_connect($db_host, $db_user, $db_password);
 if(empty($conn)){
   print  mysqli_error ($conn);
   die ("無法對資料庫連線！" );
   exit;
 }
 if(!mysqli_select_db( $conn, $db_name)){
   die("資料庫不存在!");
   exit;
 }  mysqli_set_charset($conn,'utf8');
```

```
$result = mysqli_query($conn,"SELECT * FROM ksu_std_table where
ksu_std_name = '$name' or  ksu_std_id = '$id'");
 echo "<table border='1'>
 <tr>
   <th>ksu_std_name</th>  <th>ksu_std_id</th> <th>ksu_std_age</th>
 </tr>";
 while($row = mysqli_fetch_array($result))
 {
   echo "<tr>";
   echo "<td>" . $row['ksu_std_name'] . "</td>";
   echo "<td>" . $row['ksu_std_id'] .   "</td>";
   echo "<td>" . $row['ksu_std_age'] .  "</td>";
   echo "</tr>";
 }
 echo "</table>";
 echo "records found!";
?>
<form enctype="multipart/form-data"  method="post"
action="ksu_select.html">
<input type="submit" name="sub" value="返回"/>
</form>
```

PHP網頁程式碼 - 連結PHP

```
<!doctype html>

<html lang="zh_tw">

<head>

<meta charset="utf-8">

<title>Hello</title>

</head>

<body>

<h3> ksu select operation </h3>


<form enctype="multipart/form-data" method="post"
action="ksu_select.php">

學生姓名:<input type="text" name="name" size="10"/><br/>

學生學號:<input type="text" name="id" size="10"/><br/>

<input type="submit" name="sub" value="查詢"/>

</form>

</body>

</html>
```

Frontend+Backend

前端程式 ：ksu_select.html
給使用者看，且與後端程式連結

後端程式 ：ksu_select.php
在伺服器上運作，且與資料庫連結的程式，因SQL被植入php中，事實上與資料庫連結的程式是SQL.

資料庫：MySQL
Server(伺服器)：Apache

高階程式語言：C#、C++、PHP、Java、Ruby、Python、Deliphi、Android Java ......
植入SQL：SQLine、DBMS ......

PHP放置位置
C:\xampp\htdocs
http://localhost
http://localhost/ksu_select.html

Frontend+Backend