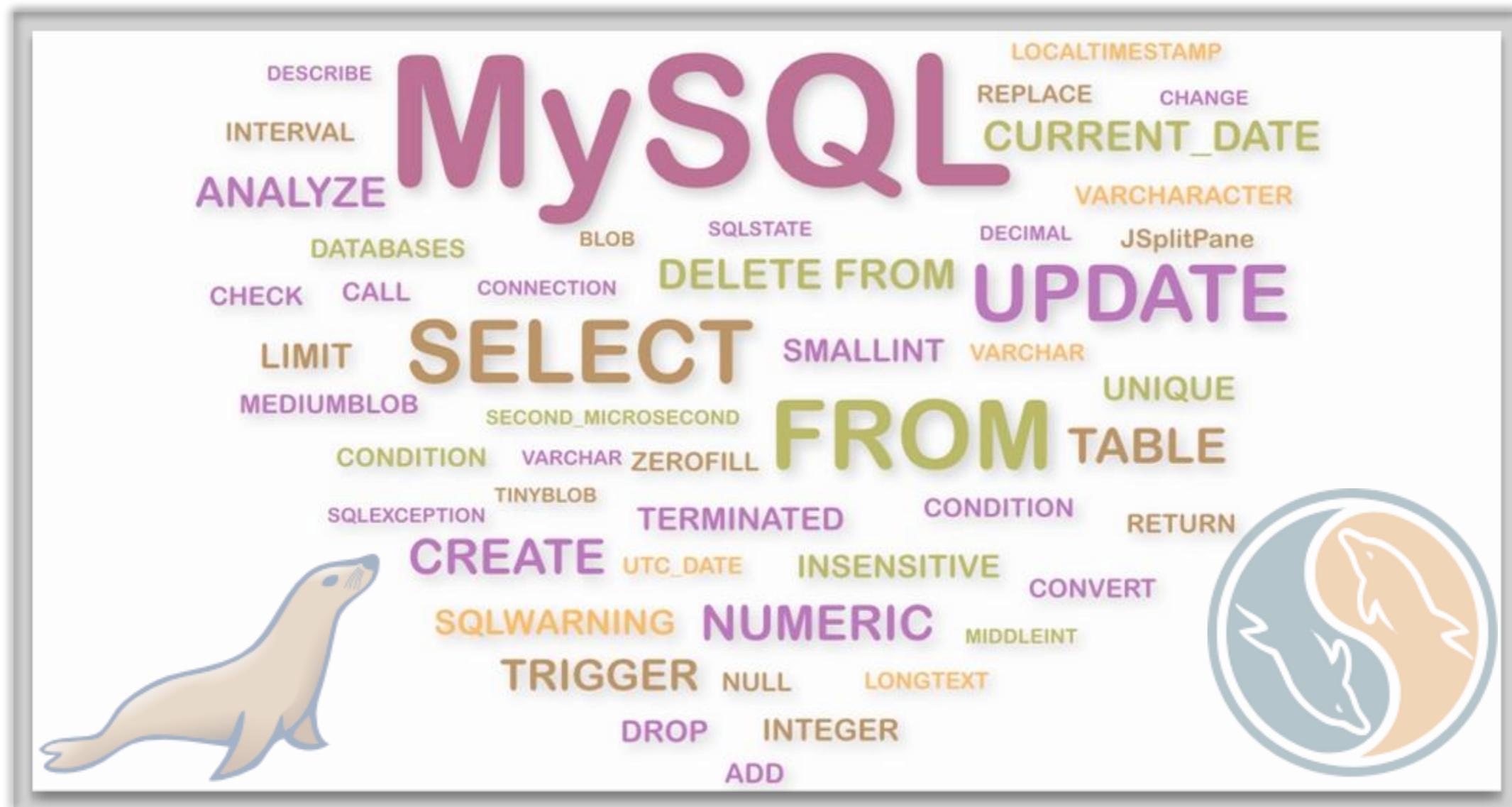


# MySQL

## 新手入門 超級手冊

第二版

適用  
MySQL 8.x 與  
MariaDB 10.x



## 第八章 表格

# 表格

- 建立表格
- 修改與刪除表格
- 查詢表格資訊

# 建立表格

- 表格屬性
- 字串欄位屬性
- 數值欄位屬性
- 通用欄位屬性
- TIMESTAMP欄位型態與預設值
- 使用其它表格建立一個新表格
- 建立暫存表格

# 建立表格

決定這個表格所需要的欄位與索引，每一個欄位也要決定名稱、型態與其它屬性

使用「CREATE TABLE」敘述建立一個新的表格

```
CREATE TABLE city (
    ID int(11) NOT NULL auto_increment,
    Name char(35) NOT NULL default '',
    CountryCode char(3) NOT NULL default '',
    District char(20) NOT NULL default '',
    Population int(11) NOT NULL default '0',
    PRIMARY KEY (ID)
) AUTO_INCREMENT=4080 DEFAULT CHARSET=latin1
```

在建立表格的時候設定表格的屬性

# 建立表格

使用「CREATE TABLE」敘述

CREATE TABLE 表格名稱  
(欄位定義, ...)

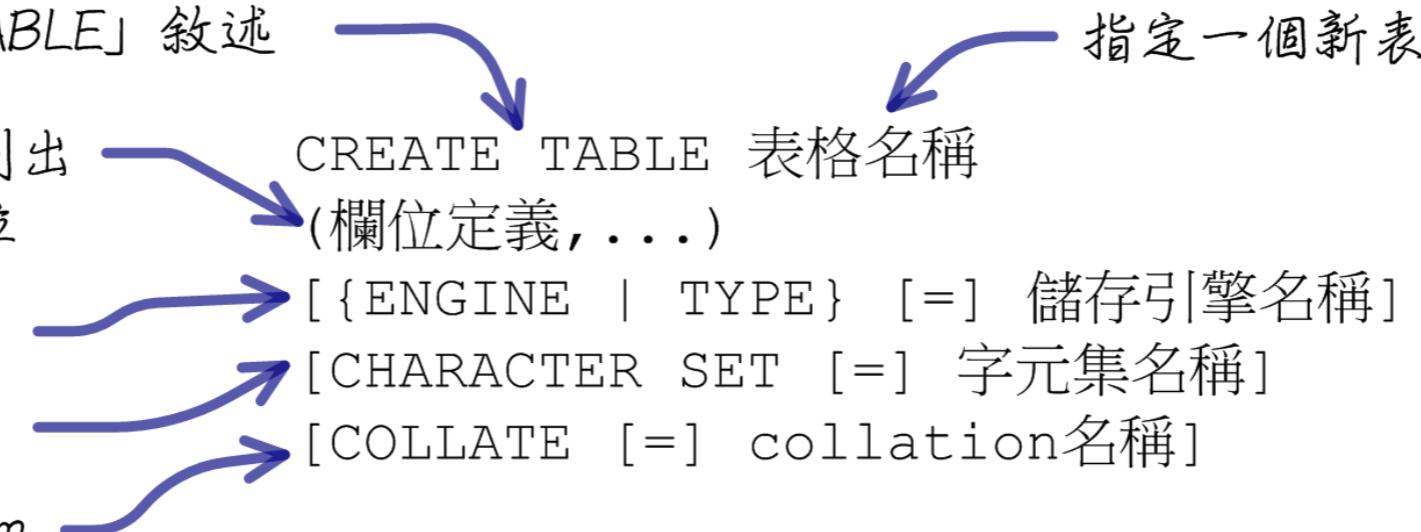
在左右括號中列出所有需要的欄位

指定儲存引擎

指定字元集

指定collation

指定一個新表格的名稱



# 建立表格

自己取一個欄位的名稱 → 欄位名稱 欄位型態 [欄位屬性], ...

根據欄位要儲存的資料,  
選擇一個最合適的型態 ↑

可以依照需要為欄位  
加入額外的屬性設定

使用逗號隔開每  
一個欄位的設定

# 表格屬性

使用「ENGINE」或「TYPE」  
指定表格使用的儲存引擎

使用「CHARACTER SET」  
或「CHARSET」指定表格  
預設的字元集

使用「COLLATE」  
指定collation

CREATE TABLE 表格名稱  
(欄位定義, ...)

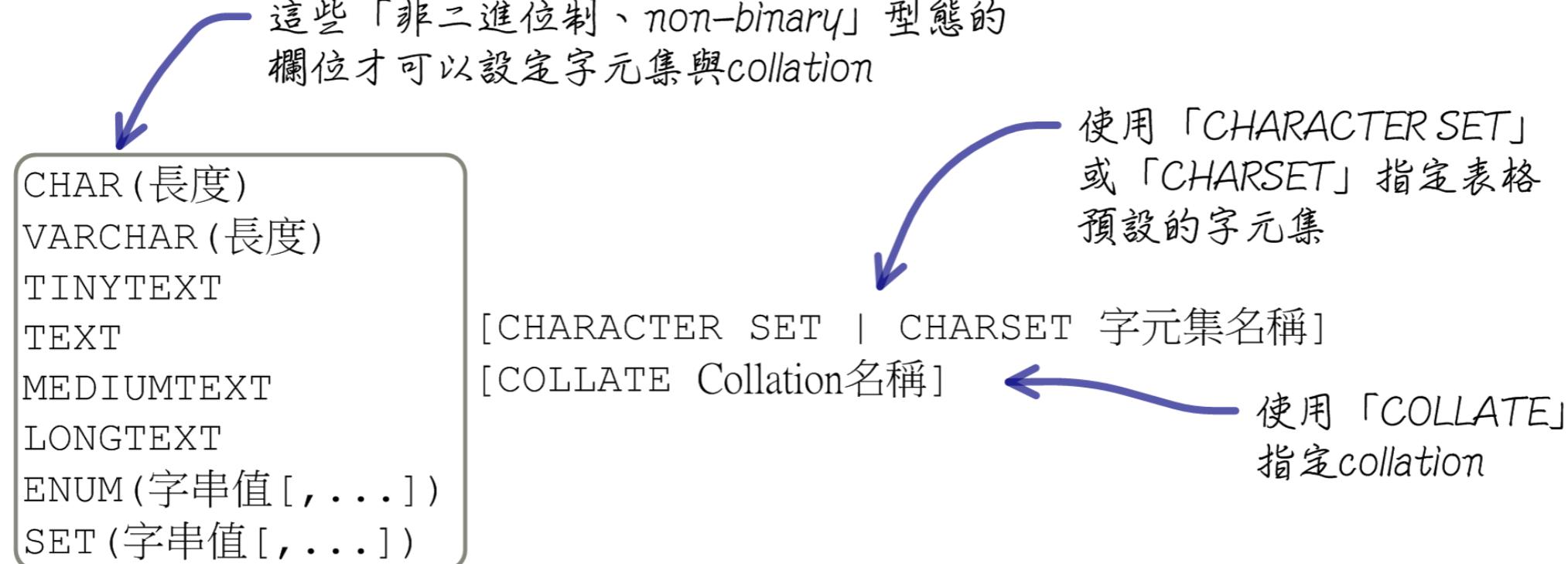
[ {ENGINE | TYPE} [=] 儲存引擎名稱 ]

[ CHARACTER SET | CHARSET [=] 字元集名稱 ]

[ COLLATE [=] collation名稱 ]

這些設定的「=」可以省略

# 字符串欄位屬性

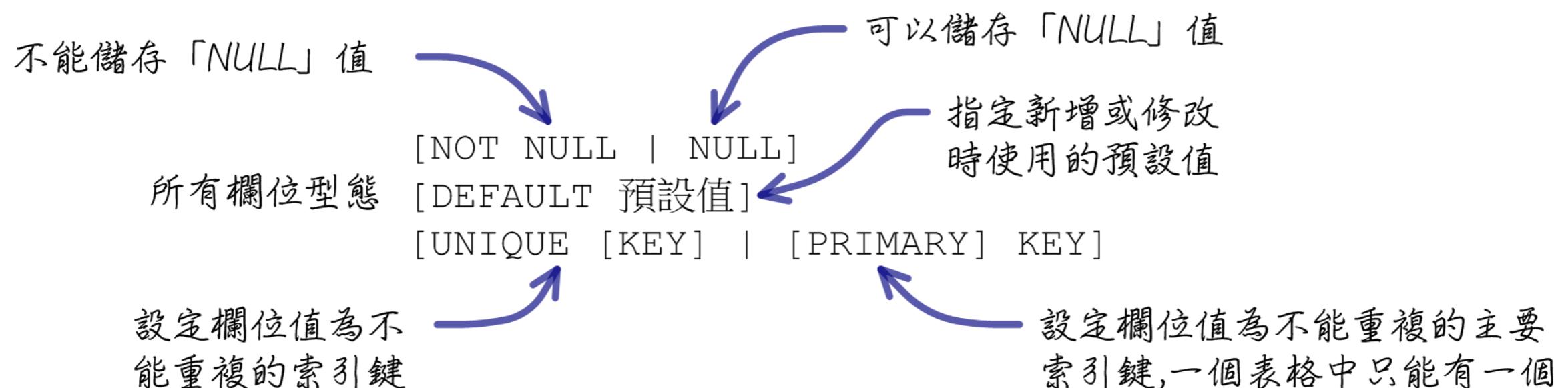


# 數值欄位屬性

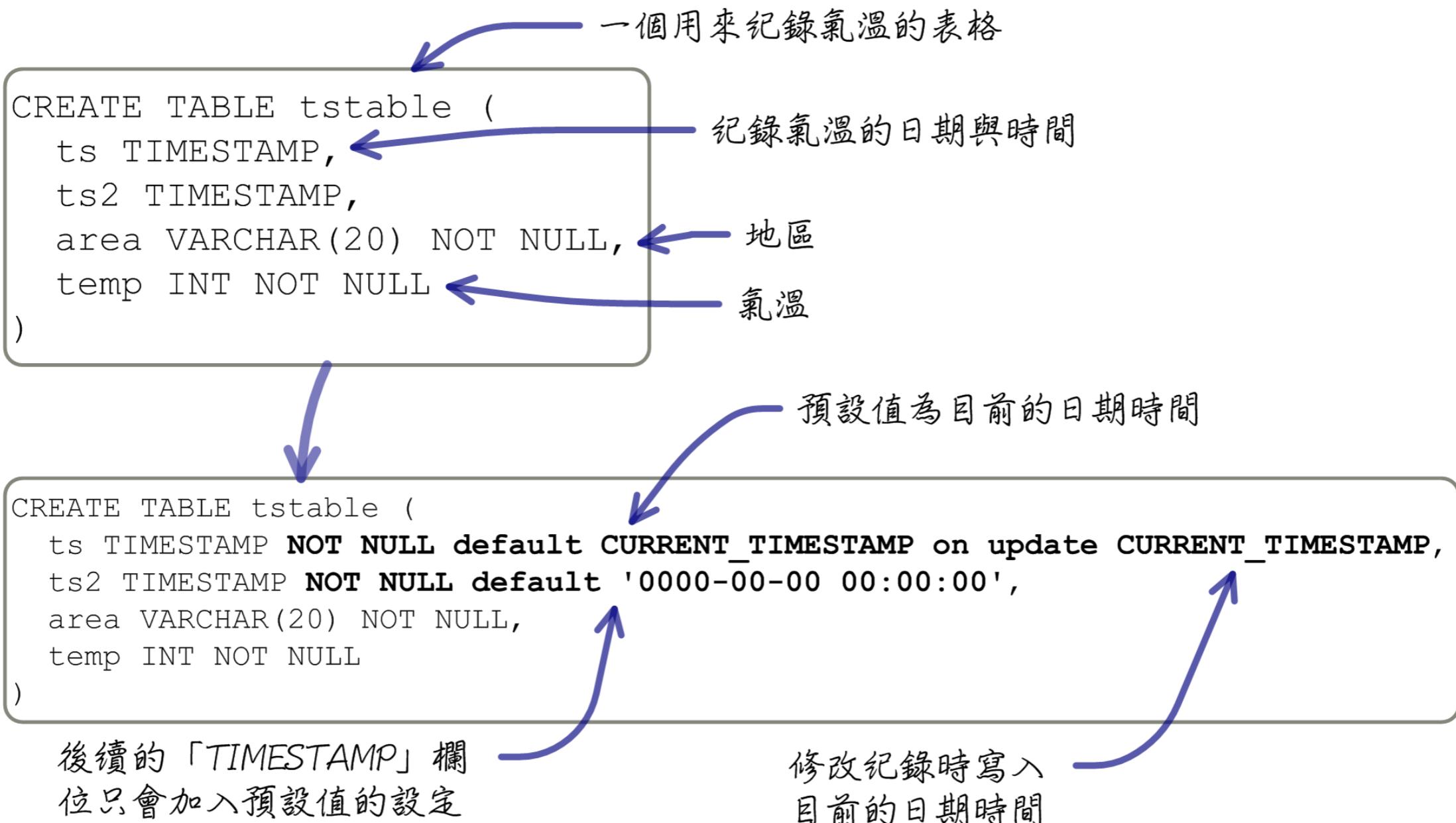
TINYINT [ (長度) ]  
SMALLINT [ (長度) ]  
MEDIUMINT [ (長度) ]  
INT [ (長度) ]  
INTEGER [ (長度) ]  
BIGINT [ (長度) ]  
REAL [ (長度, 小數位數) ]  
DOUBLE [ (長度, 小數位數) ]  
FLOAT [ (長度, 小數位數) ]  
DECIMAL (長度, 小數位數)  
NUMERIC (長度, 小數位數)

[UNSIGNED] 不能儲存負數  
[ZEROFILL] 在左側根據長度的設定填滿「0」  
[AUTO\_INCREMENT] 設定為自動累計欄位, 「DECIMAL」和「NUMERIC」不適用, 其它都可以

# 通用欄位屬性



# TIMESTAMP欄位型態與預設值



# TIMESTAMP欄位型態與預設值

- 「NOT NULL」不允許你儲存「NULL」值。
- 「DEFAULT CURRENT\_TIMESTAMP」設定預設值為目前的日期時間。在所有欄位型態中，只有「TIMESTAMP」可以使用「CURRENT\_TIMESTAMP」指定預設值。其它的欄位型態，在指定預設值只能是一個明確的值。
- 「ON UPDATE」可以指定在修改紀錄的時候，MySQL自動幫你填入的資料。

# 使用其它表格建立一個新表格

```
CREATE TABLE [IF NOT EXISTS] 表格名稱
[(欄位定義, . . .)]
[表格屬性]
一個查詢敘述
```

← 使用一樣的建立表格語法

← 使用這個查詢回傳的結果來建立  
表格,包含查詢的欄位與紀錄

# 使用其它表格建立一個新表格

- MySQL 使用查詢結果的欄位名稱與型態來建立新的表格。
- 如果沒有指定儲存引擎、字元集或collation的話，建立的新表格使用資料庫預設的儲存引擎、字元集與collation。
- 欄位的索引與「AUTO\_INCREMENT」設定都會被忽略。
  -

# 建立暫存表格

CREATE [TEMPORARY] TABLE [IF NOT EXISTS] 表格名稱  
...

加入「TEMPORARY」關鍵字



# 建立暫存表格

- TEMPORARY表格是每一個用戶端專屬的表格，用戶端離線後，MySQL就會自動刪除這些表格。
- 因為TEMPORARY表格是用戶端專屬的表格，其它用戶端不能使用，所以不同的用戶端，使用同樣名稱建立TEMPORARY表格也沒有關係。
- TEMPORARY表格名稱可以跟資料庫中的表格名稱一樣，不過在TEMPORARY表格存在的時候，資料庫中的表格會被隱藏起來。
- 只能使用「ALTER TABLE」修改TEMPORARY表格名稱，不可以使用「RENAME TABLE」修改TEMPORARY表格名稱。

# 修改與刪除表格

- 增加欄位
- 修改欄位
- 刪除欄位
- 修改表格名稱
- 刪除表格

# 修改與刪除表格

ALTER TABLE 表格名稱 修改定義 [, . . . ]



加入修改表格的設定

# 增加欄位

ADD [COLUMN] 欄位定義 [FIRST | AFTER 欄位名稱]  
ADD [COLUMN] (欄位定義[, . . . ])

↑ 增加一個新欄位, 還可以指定新欄位的位置

↑ 可以增加多個新欄位

# 修改欄位

CHANGE [COLUMN] 舊欄位名稱 新的欄位定義 [FIRST | AFTER 欄位名稱]  
MODIFY [COLUMN] 欄位定義 [FIRST | AFTER 欄位名稱]

可以修改指定欄位的名稱, 定義與位置

可以修改指定欄位的定義與位置

# 刪除欄位

DROP [COLUMN] 欄位名稱



指定一個要刪除的欄位名稱

# 修改表格名稱

使用「ALTER TABLE」修改表格名稱

```
ALTER TABLE 舊表格名稱 RENAME [TO] 新表格名稱
```

```
RENAME TABLE 舊表格名稱 TO 新表格名稱 [, ...]
```

使用「RENAME TABLE」修改表格名稱，  
這種語法可以一次修改多個表格名稱

# 刪除表格

如果表格存在的話就刪除它；如果不存在的話也不會發生錯誤

```
DROP TABLE [IF EXISTS] 表格名稱 [, ...]
```

指定要刪除的表格名稱

# 查詢表格資訊

```
SHOW TABLES FROM world
```

查詢資料庫中  
有哪些表格

指定一個資料庫名稱

Tables_in_world
city
country
countrylanguage

# 查詢表格資訊

```
SELECT TABLE_SCHEMA, TABLE_NAME, ENGINE, TABLE_ROWS,  
       AUTO_INCREMENT, TABLE_COLLATION  
FROM   information_schema.TABLES  
WHERE  TABLE_SCHEMA = 'world'
```



TABLE_SCHEMA	TABLE_NAME	ENGINE	TABLE_ROWS	AUTO_INCREMENT	TABLE_COLLATION
world	city	MyISAM	4079	4080	latin1_swedish_ci
world	country	MyISAM	239	NULL	latin1_swedish_ci
world	countrylanguage	MyISAM	984	NULL	latin1_swedish_ci