

MYSQL 程式設計

朱克剛

變數宣告

- Session 等級變數
- 變數前要加「@」
 - `set @n = 1`
 - `select @n := value from ...`
- 區域等級變數，用在 store procedure 或 function 中
- 變數前不加「@」
 - `declare n Int default null`
 - `select value into n from ...`

Store Procedure

DELIMITER \$\$

create procedure test(n int)

begin

if n > 0 then

select '> 0' as answer;

elseif n = 0 then

select '= 0' as answer;

else

select '< 0' as answer;

end if;

end \$\$

DELIMITER ;

呼叫方式
call test(20)

參數類型

■ IN / INOUT / OUT

DELIMITER \$\$

```
create procedure swap(inout a int, inout b int)
```

```
begin
```

```
    declare tmp int;
```

```
    set tmp = a;
```

```
    set a = b;
```

```
    set b = tmp;
```

```
end $$
```

DELIMITER ;

```
set @a = 10;
```

```
set @b = 20;
```

```
select @a as 'before' union all select @b;
```

```
call swap(@a, @b);
```

```
select @a as 'after' union all select @b
```

While 迴圈

■ $1 + 2 + \dots + n$

```
DELIMITER $$
create procedure acc(n int)
begin
    declare sum int default 0;
    declare i int default 0;
    while i <= n do
        set sum = sum + i;
        set i = i + 1;
    end while;
    select sum as answer;
end
DELIMITER ;
```

Loop 迴圈

- 執行結果為：「1,2,4,5,6,7,8,9,」

```
begin
  declare i int default 1;
  declare str varchar(200) default "";

  label: loop
    if i = 10 then
      leave label;
    end if;

    if i = 3 then
      set i = i + 1;
      iterate label;
    end if;

    set str = concat(str, i, ',');
    set i = i + 1;
  end loop;

  select str as answer;
end
```

Function

- 有傳回值
- 不可單純使用 select 將查詢結果丟出 function 外

DELIMITER \$\$

```
create function myadd(a float, b float) returns float
begin
    return a + b;
end $$
```

DELIMITER ;

呼叫方式

```
select myadd(5, 3)
```

```
set @n = myadd(5, 3)
```

CURSOR

Cursor

- 對每一筆資料作最細微的控制
 - 可以單獨處理每一筆資料
 - 例如：將阿拉伯數字轉成大寫國字
 - 1 -> 壹元
 - 203 -> 貳佰零叁元

建立、開啟與關閉 Cursor

建立 → `DECLARE c CURSOR FOR select fee from bill`

開啟 → `OPEN c`

...

關閉 → `CLOSE c`

範例：將電話費用加總

DELIMITER \$\$

create function totalfee() returns int

begin

declare isDone bool default false;

declare c_fee int;

declare total int default 0;

declare c cursor for select fee from Bill;

declare **continue handler for not found** set isDone = true;

open c;

fetch c into c_fee;

while **!isDone** do

set total = total + c_fee;

fetch c into c_fee;

end while;

close c;

return total;

end \$\$

DELIMITER ;

cursor

cursor

cursor

fee
250
300
100

錯誤處理

錯誤發生就離開

```
delimiter $$  
create procedure pro_name()  
begin  
    declare exit handler for sqlexception select 'ERROR';  
    update userinfo set cname = '李小妹' where uid = 'T01';  
    insert into userinfo (uid) values ('A01');  
end $$  
delimiter ;
```

離開前有機會作一些事情，例如 log

PK 重複

預存程序中使用交易機制

```
delimiter $$
```

```
create procedure pro_name()
```

```
begin
```

```
    declare flag bool default false;
```

```
    declare continue handler for sqlexception set flag = true;
```

```
    start transaction;
```

```
        update userinfo set cname = '李小妹' where uid = 'T01';
```

```
        insert into userinfo (uid) values ('A01');
```

```
        if flag then
```

```
            rollback;
```

```
            select 'FAIL: rollback';
```

```
        else
```

```
            commit;
```

```
            select 'SUCCESS: commit';
```

```
        end if;
```

```
end $$
```

```
delimiter ;
```