



澳門城市大學  
Universidade da Cidade de Macau  
City University of Macau

# 設計計算工作室2

## MySQL操作\_單表查詢

主講人 | 澳門城市大學  
City University of Macau

周帥  
Shuai Zhou

# 目錄

- 簡單查詢
- 按條件查詢
- 高級查詢
- 為表和字段取別名



# 簡單查詢



# SELECT語句

SELECT [DISTINCT] \* | 字段1,字段2,...

FROM 表名

[WHERE 條件表達式1]

[GROUP BY 字段名 [HAVING 條件表達式2]]

[ORDER BY 字段名 [ASC|DESC]]

[LIMIT [OFFSET] 記錄數];



# 課上作業

說明：把接下來的18道題的答案、截圖粘貼到HW07模版.docx。

○ 文中所有XXX都為你名字每個字拼音的首字母



# 查詢所有字段——例子

‘ ’不等於空值  
Null才是空值

- 建立數據庫 xxx\_db\_w7
- 在xxx\_db\_w7中建立表student, 表的結構結下：

字段/ 屬性/ 列名	id	name	grade	gender
說明	編號	姓名	成績	性別(m=男/f=女)
數據類型	INT	VARCHAR(20)	FLOAT	CHAR(1)
約束	主鍵 自動增加AUTO_INCREMENT	非空		

- 添加右側的數據後執行：

- SELECT id, name, gender, grade FROM student;
- SELECT id, grade, gender, name FROM student;
- SELECT \* FROM student;

grade	name	gender
40	Amy	m
100	Bob	m
90	Cathy	m
88	David	f
66	Emily	f
86	Flora	m
92	Grace	m
90	Hugh	空值

以上3個SELECT有何異同？(answer#1,無需截圖)



# 查詢指定字段——例子

- SELECT name, gender FROM student; (截圖2a)
- SELECT gender, name FROM student; (截圖2b)

以上2個SELECT執行後，它們有何異同？(answer#2)



# 按條件查詢





# 帶關係運算符的查詢

SELECT 字段1,字段2,...

FROM 表名

WHERE 條件表達式;

## 關係運算符

=

<>

!=

<

<=

>

>=



# 帶關係運算符的查詢——例子

查詢student表中id值為4的學生的編號和姓名：

- `SELECT id, name FROM student WHERE id=4;`

查詢student表中姓名是Cathy的學生的姓名和性別：

- `SELECT name, gender FROM student WHERE name='Cathy';`

**參考上述例子寫出以下SQL語句：**

**查詢student表中成績高於80分的學生姓名 (answer#3)， 並把結果截圖保存(截圖3)**



# 帶IN關鍵字的查詢

- SELECT \* | 字段1,字段2,...
- FROM 表名
- WHERE 字段名 [NOT] IN (元素1, 元素2, ...);



# 帶IN關鍵字的查詢——例子

查詢student表中id值為1,2,3的學生的編號、成績、姓名和性別：

- `SELECT id, grade, name, gender FROM student WHERE id IN (1,2,3);`

**參考上述例子寫出以下SQL語句：查詢student表中id不為1,2,3的記錄(answer#4)，並把查詢結果截圖保存(截圖4)**



# 帶 BETWEEN AND 關鍵字的查詢

BETWEEN AND 用於判斷某個字段值是否在指定範圍內。

SELECT \* | 字段1, 字段2, ...

FROM 表名

WHERE 字段名 [NOT] BETWEEN 值1 AND 值2;



# 帶BETWEEN AND關鍵字的查詢——例子

查詢student表中id值在2至5之間的學生姓名和id:

- `SELECT id, name FROM student WHERE id between 2 AND 5;`

**參考上述例子寫出以下SQL語句：查詢student表中id不在2至5之間的學生的所有資料(answer#5)， 並把查詢結果截圖保存(截圖5)**



# 空值查詢

```
SELECT * | 字段1, 字段2, ...  
FROM 表名  
WHERE 字段名 IS [NOT] NULL;
```



# 空值查詢——例子

查詢性別為空值的學生：

- `SELECT * FROM student WHERE gender IS NULL;`

**參考上述例子寫出以下SQL語句：查詢student表中性別不為空值的學生的編號、姓名和性別(answer#6)，並把查詢結果截圖保存(截圖6)**





# DISTINCT關鍵字查詢——例子

- `SELECT gender FROM student;`
- `SELECT DISTINCT gender FROM student;`

以上2個SQL語句的查詢結果有何區別(answer#7,無需截圖)

只有gender和name都相同時才被認為是重複:

- `SELECT DISTINCT gender, name FROM student;`

往student插入右側記錄後再次執行:

grade	name	gender
45	Amy	m

- `SELECT DISTINCT gender, name FROM student;`

以上的SQL語句結果是甚麼？和上一條有何異同？為甚麼？(answer#8,無需截圖)



# 帶LIKE關鍵字的查詢

SELECT \* | 字段1,字段2,...

FROM 表名

WHERE 字段名 [NOT] LIKE '匹配字符串' ;



# 帶LIKE關鍵字的查詢——例子

%匹配0個或多個字符

查詢以a開頭的名字的學生:

- `SELECT * FROM student WHERE name LIKE 'A%';`

查詢以D開頭d結尾的名字的學生:

- `SELECT * FROM student WHERE name LIKE 'D%d';`

grade	name	gender
40	Amy	m
100	Bob	m
90	Cathy	m
88	David	f
66	Emily	f
86	Flora	m
92	Grace	m
90	Hugh	空值
45	Amy	m

參考上述例子寫出以下SQL語句：查詢student表中姓名含有a的學生的名字(需要去重複)(answer#9)， 並把查詢結果截圖保存(截圖9)



# 帶LIKE關鍵字的查詢——例子

## \_匹配1個字符

查詢結構為 A?y 的名字的學生：

- `SELECT * FROM student WHERE name LIKE 'A_y';`

查詢結構為 F???a 的名字的學生：

- `SELECT * FROM student WHERE name LIKE 'F___a';`

往student表中插入右側2條記錄：

grade	name	gender
40	I%y	m
100	J%y	m

參考上述例子寫出以下SQL語句：查詢student表中姓名含有%的學生的記錄(answer#10)，  
並把查詢結果截圖保存(截圖10) [ 提示：使用轉義字符\ ]



# 帶AND關鍵字的查詢

SELECT \* | 字段1,字段2,...

FROM 表名

WHERE 條件表達式1 AND 條件表達式 2 [... AND 條件表達式 n]



# 帶AND關鍵字的查詢——例子

查詢id少於5的女學生:

- `SELECT * FROM student WHERE id<5 and gender='f';`

查詢名字含o的女學生:

- `SELECT * FROM student WHERE name LIKE '%o%' and gender='f';`

**參考上述例子寫出以下SQL語句：查詢student表中id小於7的男學生且成績高於60分的記錄(需要去重複)(answer#11)， 並把查詢結果截圖保存(截圖11)**



# 帶OR關鍵字的查詢

SELECT \* | 字段1,字段2,...

FROM 表名

WHERE 條件表達式1 OR 條件表達式2 [ ... OR 條件表達式 n ]



# 帶OR關鍵字的查詢——例子

查詢id少於5或性別為女的學生：

- `SELECT * FROM student WHERE id<5 or gender='f';`

查詢名字含o或性別為女的學生：

- `SELECT * FROM student WHERE name LIKE '%o%' or gender='f';`

grade	name	gender
40	Amy	m
100	Bob	m
90	Cathy	m
88	David	f
66	Emily	f
86	Flora	m
92	Grace	m
90	Hugh	空值
45	Amy	m
40	I%y	m
100	J%y	m

**參考上述例子寫出以下SQL語句：查詢student表中姓名以A開頭或以y結尾的學生的姓名和編號  
(answer#12)， 並把查詢結果截圖保存(截圖12)**





# 帶AND和OR關鍵字的查詢

參考之前的例子寫出以下SQL語句：查詢student表中(成績高於60分的男同學)或(成績少於90分的女同學)的記錄(answer#13)， 並把查詢結果截圖保存(截圖13)

參考之前的例子寫出以下SQL語句：查詢student表中(成績 $\geq 90$ 分或成績 $\leq 70$ )的女同學的記錄(answer#14)， 並把查詢結果截圖保存(截圖14)



# 高級查詢



# 聚合函數

實際開發中，需要對數據進行統計，如求最大值、最小值、平均值等。MYSQL提供了一些聚合函數來實現這些功能。

函數名	作用
COUNT()	返回行數
SUM()	返回總和
AVG()	返回平均值
MAX()	返回最大值
MIN()	返回最小值



# 聚合函數——例子

- 返回student表中成績的行數：SELECT COUNT(grade) FROM student;
- 返回student表中成績的總和：SELECT SUM(grade) FROM student;
- 返回student表中成績的平均值：SELECT AVG(grade) FROM student;
- 返回student表中成績的最大值：SELECT MAX(grade) FROM student;
- 返回student表中成績的最小值：SELECT MIN(grade) FROM student;

利用聚合函數寫出以下SQL語句：查詢student表中女學生的平均成績(answer#15)，並把查詢結果截圖保存(截圖15)

利用聚合函數寫出以下SQL語句：查詢student表中id>4的最高分成績(answer#16)，並把查詢結果截圖保存(截圖16)



# 對查詢結果排序

SELECT \* | 字段1, 字段2, ...

FROM 表名

ORDER BY 字段名1 [ASC|DESC], 字段名2 [ASC|DESC], ...



# 對查詢結果排序——例子

以下查詢為：從student表中選擇所有列，並

- 按成績進行排序：

- SELECT \* FROM student ORDER BY grade;

- 按成績進行升序排序：

- SELECT \* FROM student ORDER BY grade ASC;

- 按成績進行降序排序：

- SELECT \* FROM student ORDER BY grade DESC;

- 先按性別的值進行升序排序，如有相同性別的記錄則再按成績進行降序排序：

- SELECT \* FROM student ORDER BY gender ASC, grade DESC;



# 分組查詢——例子

單獨用GROUP BY，查詢的是每個分組中的一條記錄，如：

SELECT \* FROM student GROUP BY gender; **失敗？**

一般GROUP BY會與聚合函數一齊用

- SELECT COUNT(\*),gender FROM student GROUP BY gender;



# 分組查詢——例子

## GROUP BY與HAVING一起用：

查詢根據性別分組，計算每個性別的成績總和，並且只顯示總和小於300的結果：

- `SELECT sum(grade), gender from student GROUP BY gender HAVING SUM(grade)<300;`

查詢根據性別分組，計算每個性別的成績總和：

- `SELECT sum(grade), gender from student GROUP BY gender;`

查詢男學生並計算他們的成績總和：

- `SELECT sum(grade), gender from student WHERE gender='m';`





# 使用LIMIT限制查詢結果的數量

SELECT \* | 字段1,字段2,...

FROM 表名

LIMIT [OFFSET,] 記錄數



# 使用LIMIT限制查詢結果的數量——例子

- `SELECT * FROM student LIMIT 4;`
- `SELECT * FROM student LIMIT 2, 4;`

以上2句SQL執行的結果有何不同，為甚麼？(answer#17,無需截圖)



# 函數——例子

MYSQL有很多函數，具體可以看「MYSQL數據庫入門, 清華大學出版社」第117-118頁，  
或 <https://www.runoob.com/mysql/mysql-functions.html>

例子：

函數名	作用	例子
ABS(X)	返回絕對值	
SQRT(X)	返回X的非負2次方根	
CONCAT(S1,S2,...)	返回一個或多個字符串連接產生的新字符串	SELECT CONCAT(id,'_',name,'_',gender) FROM student;
IF(expr, v1, v2)	如果expr為真，返回v1，否則v2	SELECT id, IF(gender='m', 'boy', 'girl') FROM student;

# 為表和字段取別名



# 為表和字段取別名——例子

為student表取別名s並在該表中查詢所有性別為女的學生：

- `SELECT * FROM student AS s WHERE s.gender='f';`

執行下列查詢：

- `SELECT name AS stu_name, gender AS stu_gender FROM student;`
- `SELECT name, gender FROM student;`

**以上2句SQL執行的結果有何異同， 為甚麼？(answer#18,無需截圖)**



# 課後作業



# 課後作業

Student_XXX				
屬性名	SID	SNAME	SEX	AGE
數據類型	varchar(5)	varchar(10)	varchar(2)	int
約束	主碼	非空		
數據	S1	Amy	F	18
	S2	Bob	M	19
	S3	Cathy	F	21
	S4	David	M	24

Course_XXX			
屬性名	CID	CNAME	CREDIT
數據類型	varchar(5)	varchar(20)	int
約束	主碼	非空	
數據	C1	Python	4
	C2	Java	2
	C3	Php	2
	C4	Database	4

SC_XXX			
屬性名	SID	CID	SCORE
數據類型	varchar(5)	varchar(5)	int
約束	無約束	無約束	無約束
數據	S1	C1	90
	S1	C2	95
	S1	C3	93
	S2	C2	70
	S2	C4	65
	S3	C2	80
	S3	C3	88
	S3	C4	100

○建立一個名為**db\_w7\_xxx**的數據庫，然後按要求在該數據庫中建立左側三個表並插入數據。

○其中：

○學生表Student\_XXX：

SID學號、SNAME姓名、SEX性別、AGE年齡

○課程表Course\_XXX：

CID課程號、CNAME課程名、CREDIT學分

○選課表SC\_XXX：

SID學號、CID課程號、SCORE成績

# 課後作業

○ 在HW07模版.docx中，寫出以下查詢並附上輸出結果截圖：

1. 查詢Student\_XXX表中，年齡不小於20的學生的學號、姓名和年齡。
2. 查詢Student\_XXX表中，性別不是女的記錄。
3. 查詢SC\_XXX中，成績在[80,90]區間範圍的記錄。
4. 查詢選了課程號為C2的學生的學號。
5. 查詢選了Python或Java的學生的姓名，結果需要去重覆。
6. 查詢選了課程名中包含「a」的課的學生的學號，結果需要去重覆。
7. 查詢選了課程名中包含「a」的課的學生的姓名及該門課的課程名。
8. 查詢選了課程名中僅包含2個「a」的課的學生的姓名及該門課的課程名。
9. 查詢選了課的學生的學號及其平均分。
10. 查詢選了課的學生的學號、姓名及其最高分。
11. 查詢女學生的選課記錄中，成績最高分的那條選課記錄的分數。





# 課後作業

把HW07模版.docx改名為HW07\_你的名字\_你的學號.docx

## 注意：

- 若不使用HW07模版.docx答題，本次作業將不獲得任何分數。
- 文中的XXX為你名字每個字的拼音首字母，若寫錯XXX，本次作業將不獲得任何分數。

