

TRUY VẤN TRONG CSDL MySQL (Chương 1-B)

Su Kim Anh

TRUY VẤN TRONG CSDL MYSQL - SELECT

1. Truy vấn đơn giản
2. Sắp xếp
3. Truy vấn sử dụng as, limit và Distinct
4. Lọc theo mẫu tin
5. Truy vấn trên nhiều bảng
6. Truy vấn có nhóm và hàm tính toán trên nhóm
7. Lọc theo nhóm

TRUY VẤN ĐƠN GIẢN

```
SELECT [DISTINCT] list_col / * /  
FROM table_name  
[WHERE col_name operator value ]  
[ORDER BY col_names ASC|DESC ]
```

TRUY VẤN CÓ SỬ DỤNG AS, LIMIT VÀ DISTINCT

AS

- Sử dụng AS để đặt lại tên (alias) cho các cột
- Cú pháp

```
SELECT Tên_cột AS Định_danh_cột, ...  
FROM Tên_bảng  
[ORDER BY Tên_cột_sắp_xếp [DESC, ...]]
```

TRUY VẤN CÓ SỬ DỤNG AS, LIMIT VÀ DISTINCT

□ Giới hạn mẫu tin LIMIT

- Câu lệnh SELECT ... FROM kết hợp với mệnh đề LIMIT n,m giúp lấy ra m mẫu tin trong bảng tính từ vị trí n, theo một tiêu chuẩn sắp xếp nào đó
- Cú pháp:

```
SELECT Danh_sách_các_cột  
FROM Tên_bảng  
[ORDER BY Tên_cột_sắp_xếp [DESC, ...]]  
LIMIT n,m
```

TRUY VẤN CÓ SỬ DỤNG AS, LIMIT VÀ DISTINCT

□ DISTINCT

- Trong 1 bảng, 1 cột có thể chứa nhiều giá trị trùng lặp; và đôi khi ta chỉ muốn lấy ra danh sách những giá trị duy nhất không trùng lặp => Sử dụng từ khóa DISTINCT
- Cú pháp:

```
SELECT DISTINCT Tên_cột_1, Tên_cột_2,...  
FROM Tên_bảng  
[ORDER BY Tên_cột_sắp_xếp [DESC, ...]]
```

Lọc theo mẫu tin

□ WHERE

- Câu lệnh SELECT ... FROM kết hợp với mệnh đề WHERE giúp lọc các dòng dữ liệu bên trong bảng, dữ liệu này phải thỏa điều kiện đưa ra trong mệnh đề WHERE
- Cú pháp:

```
SELECT Danh_sách_các_cột  
FROM Tên_bảng  
WHERE Điều_kiện_lọc  
[ORDER BY Tên_cột_sắp_xếp [DESC, ...]]
```
- Các phép toán thường gặp trong điều kiện lọc
 - So sánh: >, >=, <, <=, =, !=, <>
 - Logic: and, or, not, in, not in, between, like, not like

Truy vấn trên nhiều bảng

□ Loại 1

- Có thể sử dụng mệnh đề WHERE để liên kết dữ liệu của nhiều bảng trong truy vấn
- Cú pháp:

```
SELECT Danh_sách_các_cột  
FROM Tên_bảng_1, Tên_bảng_2, ...  
WHERE Tên_bảng_1.tên_cột =  
Tên_bảng_2.tên_cột  
[ORDER BY Tên_cột_sắp_xếp [DESC, ...]]
```

Truy vấn trên nhiều bảng

□Loại 2

- Khi muốn liên kết các bảng có quan hệ với nhau để lấy ra dữ liệu chung → kết hợp **SELECT ... FROM** với mệnh đề **JOIN**.
- Khi sử dụng JOIN để nối các bảng cần phải lưu ý những bảng này phải có các cột liên hệ với nhau và thứ tự quan hệ được chỉ định giữa các bảng sẽ làm ảnh hưởng tới kết quả truy vấn.

Truy vấn trên nhiều bảng

□ Inner join

- Khi kết nối các bảng dùng **INNER JOIN** → chỉ định việc so sánh giá trị trong các cột của các bảng là tương đương – dữ liệu đều có ở cả hai bảng.
- Kết quả sau khi thực hiện câu lệnh truy vấn kết hợp INNER JOIN là các mẫu tin thỏa điều kiện quan hệ ở cả hai bảng
- Cú pháp:
SELECT Danh_sách_các_cột
FROM Tên_bảng
INNER JOIN Tên_bảng_liên_kết **ON** Điều_kiện_liên_kết
[WHERE Điều_kiện_lọc
[ORDER BY Danh_sách_các_cột_sắp_xếp **[DESC]]**

Truy vấn trên nhiều bảng

□Left join, right join

- Khi kết nối các bảng dùng **LEFT|RIGHT JOIN** → chỉ định việc so sánh giá trị trong các cột của các bảng được ưu tiên cho mỗi quan hệ bên nhánh trái | nhánh phải. Việc thay đổi thứ tự ưu tiên này sẽ làm ảnh hưởng tới kết quả truy vấn.
- Cú pháp:
SELECT Danh_sách_các_cột
FROM Tên_bảng
LEFT|RIGHT JOIN Tên_bảng_liên_kết **ON** Điều_kiện_liên_kết
[WHERE Điều_kiện_lọc
[ORDER BY Danh_sách_các_cột_sắp_xếp **[DESC]]**

Truy vấn trên nhiều bảng

□ Mệnh đề liên kết dữ liệu nhiều bảng

- Cú pháp:
SELECT Danh_sách_các_cột
FROM Tên_bảng_1
INNER|LEFT|RIGHT JOIN Tên_bảng_2
ON Điều_kiện_liên_kết_bang_1_2
INNER|LEFT|RIGHT JOIN Tên_bảng_3
ON Điều_kiện_liên_kết...
...
[WHERE Điều_kiện_lọc
[ORDER BY Danh_sách_các_cột_sắp_xếp **[DESC]]**

Truy vấn có nhóm và hàm tính toán trên nhóm

□ Group by

- Câu lệnh SELECT ... FROM kết hợp với mệnh đề GROUP BY giúp nhóm dữ liệu của các dòng dữ liệu bên trong bảng và sử dụng thêm các hàm thống kê đi kèm để tính toán dữ liệu có tính chất thống kê

- Cú pháp:

```
SELECT Danh_sách_các_cột, Hàm_thống_kê [as tên]
FROM Tên_bảng
[WHERE Điều_kiện_lọc]
GROUP BY Danh_sách_các_cột_nhóm_dữ_liệu
[ORDER BY Tên_cột_sắp_xếp [DESC, ...]]
```

Truy vấn có nhóm và hàm tính toán trên nhóm

□ Group by

- Các hàm thống kê:

- AVG: hàm trả về giá trị trung bình theo nhóm trong câu lệnh truy vấn trên bảng
- COUNT: hàm trả về số lượng mẫu tin theo nhóm trong câu truy vấn trên bảng
- MIN: hàm trả về giá trị nhỏ nhất theo nhóm
- MAX: hàm trả về giá trị lớn nhất theo nhóm
- SUM: hàm trả về tổng các giá trị theo nhóm

Lọc theo nhóm

□ HAVING

- Câu lệnh SELECT ... FROM kết hợp với mệnh đề HAVING giúp lọc lại dữ liệu sau khi đã gom nhóm dữ liệu bằng mệnh đề GROUP BY

- Cú pháp:

```
SELECT Danh_sách_các_cột, Hàm_thống_kê [as tên]
FROM Tên_bảng
[WHERE Điều_kiện_lọc]
GROUP BY Danh_sách_các_cột_nhóm_dữ_liệu
HAVING Điều_kiện_lọc_sau_khi_nhóm
[ORDER BY Tên_cột_sắp_xếp [DESC, ...]]
```

Truy vấn con

□ Khái niệm

- Truy vấn con là một câu lệnh truy vấn SELECT được lồng vào các câu lệnh truy vấn khác nhằm thực hiện các truy vấn tính toán phức tạp.
- Lưu ý: khi dùng truy vấn con cần tuân theo các quy tắc
 - Truy vấn con phải được đặt trong cặp ngoặc đơn ()
 - Truy vấn con chỉ có thể tham chiếu đến 1 cột hoặc 1 biểu thức.
- Kết quả trả về của truy vấn con có thể là một giá trị hoặc một danh sách các giá trị.

Truy vấn con

❑ Truy vấn con trả về danh sách các giá trị

- Là truy vấn con mà kết quả trả về là tập hợp các giá trị.
- Toán tử IN hoặc NOT IN thường được dùng trong trường hợp này vì nó so sánh một phần tử có thuộc (hay không thuộc) tập hợp các giá trị hay không.

Truy vấn con

❑ Làm việc với toán tử so sánh

- Các toán tử so sánh thường được sử dụng trong truy vấn con có thể là: >, >=, <, <=, =, <>.
- **Lưu ý:** Thông thường các toán tử so sánh được sử dụng khi dùng truy vấn con trả về một giá trị

Truy vấn con

❑ Làm việc với toán tử truy vấn con

- Các toán tử truy vấn con thường hay sử dụng là: ANY, SOME, ALL, IN, NOT IN, EXISTS, NOT EXISTS.
- Lưu ý: thông thường các toán tử truy vấn con được sử dụng khi dùng truy vấn con trả về tập hợp các giá trị.
- Quy tắc:
IN ⇔ =ANY
NOT IN ⇔ <>ALL

Truy vấn con

❑ Làm việc với toán tử truy vấn con

- Sử dụng từ khóa EXISTS hoặc NOT EXISTS để kiểm tra tính tồn tại (hay không tồn tại) của dữ liệu.
- Sau EXISTS hoặc NOT EXISTS là câu lệnh truy vấn con mà kết quả trả về là một tập hợp trống hoặc có phần tử

CẬP NHẬT DỮ LIỆU

```
UPDATE table_name SET column1=value,  
column2=value2,...  
[WHERE some_column=some_value ]
```

CẬP NHẬT DỮ LIỆU

❑ Lấy dữ liệu từ các bảng khác

- Kết hợp giữa UPDATE và SELECT để lấy dữ liệu từ bảng khác cập nhật vào bảng
- Cú pháp:

```
UPDATE Tên_bảng  
SET Tên_cột = (SELECT ... FROM ... WHERE ...)  
WHERE Điều_kiện_cập_nhật
```

Xóa mẫu tin

```
DELETE FROM table_name  
[WHERE some_column=some_value ]
```

Xóa mẫu tin

❑ Lệnh xóa dữ liệu có điều kiện được lấy từ các bảng khác

- Khi việc xóa phức tạp hơn vì có liên quan tới các quy tắc ràng buộc dữ liệu → kết hợp câu lệnh DELETE với SELECT
- Cú pháp:

```
DELETE FROM Tên_bảng  
WHERE Tên_cột toán_tử (SELECT ... FROM ...  
WHERE)
```

Thảo luận

