

# Xử lý bản sao trong MySQL

Các bảng hoặc tập hợp kết quả đôi khi chứa bản sao các bản ghi. Đôi khi, nó được cho phép nhưng đôi khi nó là bắt buộc để ngăn chặn các bản sao này. Đôi khi, nó là bắt buộc để nhận diện các bản sao và xóa chúng khỏi bảng. Chương này miêu tả cách ngăn cản các bản sao dữ liệu xuất hiện trong một bảng và cách để gỡ bỏ chúng.

## Ngăn cản bản sao xuất hiện trong một bảng

Bạn có thể sử dụng một **PRIMARY KEY** hoặc **UNIQUE** Index trên một bảng cho các trường thích hợp để ngăn cản sự xuất hiện của các bản sao. Ví dụ: bảng *nhanvienIT* sau không chứa chỉ mục dạng này, vì thế nó cho phép xuất hiện các bản sao cho **ho** và **ten**:

```
CREATE TABLE nhanvienIT ( ho CHAR(20), ten CHAR(20), diachi CHAR(30) );
```

Để ngăn cản nhiều bản ghi với cùng giá trị họ và tên với các giá trị họ và tên đã được tạo trong bảng này, thêm một PRIMARY KEY vào định nghĩa của nó. Khi bạn làm điều này, cũng cần thiết phải khai báo các cột được lập chỉ mục là NOT NULL, bởi vì một PRIMARY KEY không cho phép các giá trị NOT NULL.

```
CREATE TABLE nhanvienIT ( ho CHAR(20) NOT NULL, ten CHAR(20) NOT NULL, diachi CHAR(30), PRIMARY KEY (ho, ten) );
```

Sự có mặt của một chỉ mục UNIQUE trong một bảng thường gây ra một lỗi nếu bạn chèn một bản ghi vào trong một bảng là bản sao của một bản ghi đang tồn tại trong cột hoặc các cột định nghĩa chỉ mục đó.

Sử dụng **INSERT IGNORE** thay cho **INSERT**. Nếu một bản ghi không sao một bản ghi đang tồn tại, MySQL chèn nó như bình thường. Nếu bản ghi là một bản sao, từ khóa IGNORE nói cho MySQL để loại bỏ nó một cách lặng lẽ mà không cần tạo ra một lỗi.

Ví dụ sau không tạo lỗi và cùng lúc đó nó sẽ không chèn bản sao các bản ghi.

```
mysql> INSERT IGNORE INTO nhanvienIT (ho, ten) -> VALUES( 'Jay', 'Thomas');  
Query OK, 1 row affected (0.00 sec) mysql> INSERT IGNORE INTO nhanvienIT (ho,  
ten) -> VALUES( 'Jay', 'Thomas'); Query OK, 0 rows affected (0.00 sec)
```

Sử dụng **REPLACE** thay cho **INSERT**. Nếu bản ghi là mới, nó được chèn với lệnh **INSERT**. Nếu nó là bản sao, bản ghi mới sẽ thay thế cho bản ghi cũ.

```
mysql> REPLACE INTO nhanvienIT (ho, ten) -> VALUES( 'Ajay', 'Kumar'); Query OK, 1 row affected (0.00 sec)
mysql> REPLACE INTO nhanvienIT (ho, ten) -> VALUES( 'Ajay', 'Kumar'); Query OK, 2 rows affected (0.00 sec)
```

Bạn nên chọn INSERT IGNORE và REPLACE theo cách xử lý bản sao bạn muốn. INSERT IGNORE giữ lại tập hợp bản sao bản ghi đầu tiên và loại bỏ phần còn lại. REPLACE giữ tập hợp bản sao cuối và xóa các phần trước.

Cách khác để áp dụng tính duy nhất là thêm một chỉ mục UNIQUE thay vì một PRIMARY KEY cho một bảng.

```
CREATE TABLE nhanvienIT ( ho CHAR(20) NOT NULL, ten CHAR(20) NOT NULL,
diachi CHAR(30) UNIQUE (ho, ten) );
```

## Đếm và xác định các bản sao trong MySQL

Ví dụ sau là truy vấn để đếm bản sao với họ và tên trong một bảng.

```
SELECT COUNT(*) as sobansao, ho, ten FROM nhanvienIT GROUP BY ho, ten
HAVING sobansao > 1;
```

Truy vấn này sẽ trả về một danh sách tất cả bản sao trong bảng nhanvienIT. Nói chung, để xác định tập hợp các giá trị có bị sao hay không, bạn thực hiện:

- Xác định xem cột nào chứa các giá trị mà có thể bị sao.
- Liệt kê tất cả các cột đó, cùng với COUNT(\*).
- Liệt kê các cột trong mệnh đề GROUP BY.
- Thêm một mệnh đề HAVING để loại bỏ các giá trị duy nhất bằng việc yêu cầu giá trị đếm nhóm lớn hơn 1.

## Loại bỏ bản sao từ một kết quả truy vấn

Bạn có thể sử dụng **DISTINCT** cùng với lệnh SELECT để tìm các bản ghi duy nhất có trong một bảng.

```
SELECT DISTINCT ho, ten FROM nhanvienIT ORDER BY ten;
```

Ngoài DISTINCT, bạn thêm một mệnh đề GROUP BY để xác định các cột bạn đang chọn.

```
SELECT ho, ten FROM nhanvienIT GROUP BY (ho, ten);
```

## Xóa bản sao bởi sử dụng bảng thay thế

Nếu bạn có các bản sao trong một bảng và bạn muốn xóa tất cả bản sao từ bảng đó, thì dưới đây là thủ tục:

```
CREATE TABLE nhanvienIT1 SELECT ho, ten, diachi FROM
nhanvienIT;
GROUP BY (ho, ten); DROP TABLE nhanvienIT;
ALTER TABLE nhanvienIT1 RENAME TO nhanvienIT;
```

Một cách dễ dàng để xóa các bản sao từ một bảng là thêm một INDEX hoặc PRIMARY KEY cho bảng đó. Ngay cả khi bảng này đã có sẵn, bạn có thể sử dụng kỹ thuật này để xóa các bản sao.

```
ALTER IGNORE TABLE nhanvienIT ADD PRIMARY KEY (ho, ten);
```