

# Import và phương thức Recovery trong MySQL

Có hai cách đơn giản trong MySQL để tải dữ liệu vào trong MySQL Database từ một file đã được sao lưu trước đó.

## Import với LOAD DATA

MySQL cung cấp một lệnh LOAD DATA mà hoạt động như một Bulk Data Loader. Ví dụ về lệnh sau sẽ đọc một file là dump.txt từ thư mục hiện tại của bạn và tải nó vào trong bảng sinhvienk60 trong cơ sở dữ liệu hiện tại:

```
mysql> LOAD DATA LOCAL INFILE 'dump.txt' INTO TABLE sinhvienk60;
```

- Nếu từ khóa LOCAL không có mặt, MySQL tìm kiếm datafile trên Server host bởi nhìn vào pathname tuyệt đối xác định vị trí của file đó, bắt đầu từ Root của hệ thống file. MySQL đọc file từ vị trí đã cho.
- Theo mặc định, LOAD DATA giả sử rằng datafile chứa các dòng mà được kết thúc bởi linefeed (newline) và các giá trị dữ liệu đó bên trong một line được phân biệt bởi các tab.
- Để xác định một định dạng file một cách tường minh, sử dụng một mệnh đề FIELDS để miêu tả đặc trưng của trường bên trong một line, và một mệnh đề LINES để xác định dãy line-ending. Lệnh LOAD DATA sau xác định rằng datafile chứa các giá trị được phân biệt nhau bởi dấu phẩy và line kết thúc bởi ký tự carriage-return và newline.

```
mysql> LOAD DATA LOCAL INFILE 'dump.txt' INTO TABLE sinhvienk60 -> FIELDS  
TERMINATED BY ':' -> LINES TERMINATED BY '\r\n';
```

- LOAD DATA giả sử các cột trong datafile có cùng thứ tự như các cột trong bảng. Nếu điều đó không đúng, bạn có thể xác định một danh sách để chỉ dẫn cột nào trong bản nên được tải vào. Giả sử bảng của bạn có các cột a, b, và c, nhưng các cột liên tiếp trong datafile là b, c và a. Bạn có thể tải file này như sau:

```
mysql> LOAD DATA LOCAL INFILE 'dump.txt' -> INTO TABLE sinhvienk60 (b, c,  
a);
```

## Import với mysqlimport

MySQL cũng cung cấp một chương trình tiện ích tên là **mysqlimport** mà hoạt động như một Wrapper xung quanh LOAD DATA để mà bạn có thể tải input file một cách trực tiếp từ command line.

Để tải một file từ dump.txt vào trong sinhvienk60, sử dụng lệnh sau tại UNIX prompt.

```
$ mysqlimport -u root -p --local ten_database dump.txt password *****
```

Nếu bạn sử dụng mysqlimport, tùy chọn command-line cung cấp các Format specifier. Các lệnh mysqlimport mà tương ứng với hai lệnh LOAD DATA trước đó sẽ trông giống như:

```
$ mysqlimport -u root -p --local --fields-terminated-by=":" \ --lines-terminated-by="\r\n" ten_database dump.txt password *****
```

Thứ tự trong đó bạn xác định các tùy chọn không là vấn đề với mysqlimport, ngoại trừ rằng chúng nên đặt ở trước database name.

Lệnh **mysqlimport** sử dụng tùy chọn --column để xác định thứ tự cột.

```
$ mysqlimport -u root -p --local --columns=b,c,a \ ten_database dump.txt password *****
```

## Xử lý trích dẫn và các ký tự đặc biệt

Mệnh đề FIELDS có thể xác định các tùy chọn định dạng khác bên cạnh TERMINATED BY. Theo mặc định, LOAD DATA giả sử rằng các giá trị không được trích dẫn và thông dịch dấu gạch chéo ngược (\) như là một ký tự thoát cho các ký tự đặc biệt. Để chỉ dẫn giá trị trích dẫn các ký tự một cách tường minh, sử dụng ENCLOSED BY. MySQL sẽ lược ký tự đó từ phần cuối dữ liệu trong khi xử lý input. Để thay đổi ký tự thoát mặc định, sử dụng ESCAPED BY.

Khi bạn xác định ENCLOSED BY để chỉ rằng các ký tự trích dẫn nên bị lược bỏ từ các giá trị dữ liệu, nó là có thể để bao ký tự trích dẫn bên trong giá trị dữ liệu bằng việc sử dụng nó hai lần (tăng gấp đôi) hoặc bằng việc đặt trước nó với ký tự thoát. Ví dụ, nếu ký tự trích dẫn và ký tự thoát là " và \, thì giá trị đầu vào là "a""b\"c" sẽ được thông dịch là a"b"c.

Với mysqlimport, các tùy chọn command-line tương ứng để xác định giá trị trích dẫn và giá trị thoát là --fields-enclosed-by và --fields-escaped-by.

