

# LẬP TRÌNH WEB ĐỘNG VỚI **PHP / MySQL**

- ❖ GUESTBOOK
- ❖ CATALOG
- ❖ FORUM
- ❖ SHOPPING CART

## PHẦN 2

**Tống Phước Khải (tổng hợp & biên dịch)**

# **Phương pháp truy xuất CSDL**

## **MySQL**

- 1- Từ Command Prompt**
- 2- Bằng lệnh PHP**
- 3- Dùng phpMyadmin**

Để hiểu được chương này một cách rõ ràng, trước tiên bạn cần phải có một số kiến thức cơ bản về **Cơ Sở Dữ Liệu quan hệ**. Nếu bạn đã học qua một khoá căn bản về MS Access trong chương trình đào tạo chứng chỉ B chẳng hạn thì hẳn nhiên bạn có thể tiếp tục. Còn nếu như bạn chưa biết gì về nó thì tôi sẽ bàn đến nó trong phần Phụ Lục của giáo trình này hoặc bạn có thể tìm ngay một tài liệu tham khảo về CSDL, dễ nhất là tài liệu và MS Access ...

... Tôi chắc rằng bây giờ bạn đã có kiến thức về CSDL và hiểu biết Table là gì rồi! Có hàng khối công việc bạn sẽ phải làm việc đối với các Table và bạn sẽ được hướng dẫn cặn kẽ trong quyển sách này. Bạn sẽ phải vượt qua một số kiến thức về nó để mới có thể thành thạo trong thao tác với Table. Như bạn biết đấy: *Con đường đi đến thành công không có trải thảm sẵn đâu!*

Nếu bạn đã từng làm việc với *MS SQL Server* hay *Access* chúng đều có hỗ trợ việc tạo CSDL rất là dễ dàng với giao diện trực quan. Đối với MySQL bạn cũng có thể sử dụng công cụ trực quan đó là *phpMyadmin*.

Tuy nhiên, bạn phải học cách thao tác với CSDL bằng dòng lệnh, tôi chắc rằng điều này sẽ rất có ích cho bạn. Muốn chương trình của bạn trong lúc chạy thao tác tự động với CSDL thì bạn cần hàng tá lệnh PHP/SQL để thực hiện các yêu cầu của chương trình.

Trước khi chúng ta tạo các table trong CSDL của MySQL, có một vài thứ bạn cần phải hiểu rõ. Những khái niệm cơ bản mà tôi sắp giới thiệu sau đây rất quan trọng. Bạn hãy chắc rằng mình đã nắm kỹ về chúng trước khi thực hiện việc thiết kế dữ liệu.

## Null

Việc đầu tiên bạn phải làm trong việc thiết kế một table là quyết định xem một field có cho phép giá trị NULL hay không.

Trong CSDL quan hệ, giá trị NULL của một field đồng nghĩa với nó có thể chấp nhận không có dữ liệu trong đó. Nên nhớ rằng giá trị NULL khác với giá trị của một chuỗi không có ký tự trong đó hoặc số có giá trị 0.

Đôi khi trong chương trình, bạn sẽ thực hiện một số động tác so sánh xem một chuỗi nào đó có chứa giá trị hay không, nó có thể là một câu lệnh IF. Xét một ví dụ PHP như sau:

```
$var //this is a variable used in the test
if ($var == "")
{
echo "Var is an empty string";
} else {
echo $var;
}
```

Nếu bạn thực hiện việc so sánh xem giá trị số có phải 0 hay không thì cũng thực hiện tương tự.

Động tác so sánh trên sẽ không làm việc được đối với giá trị NULL. Bạn nên hiểu rằng NULL là không chứa bất kỳ giá trị gì trong đó, cho nên việc so sánh trị không mang ý nghĩa gì cả. Trong chương 3 bạn sẽ thấy rằng đối với giá trị NULL đòi hỏi lập trình viên phải rất cẩn nhắc khi viết lệnh liên kết table.

*Trong lệnh SELECT của SQL, có một số cách để bạn có thể kiểm tra nếu như một field chứa giá trị NULL. Trước hết bạn hãy sử dụng hàm Isnull(). Giả sử tìm một record trong table mà giá trị middle\_name là NULL, bạn có thể sử dụng query sau:*

**select \* from names where isnull(middle\_name);**

*Hoặc lấy các record mà middle\_name khác NULL:*

**select \* from names where !isnull(middle\_name);**

*Bạn cũng có thể sử dụng **is null** và **is not null**:*

**select \* from users where addr2 is null;**

**select \* from users where addr2 is not null;**

Để rõ hơn, bạn hãy xem chuyện gì xảy ra khi tôi cố gắng liên kết hai table sau:

### Khach\_hang

ten	ho_lot	ba_xa
Trung	Le Anh	1321
Khai	Tong Phuoc	Null

### Hon\_nhan

ba_xa	ten	ho_lot
1321	Diem	Nguyen Thuy

Nếu bạn muốn tìm tên các **khách hàng** và tên những **bà xã** của họ, bạn sẽ phải liên kết 2 table này thông qua field ba\_xa. (Xin bạn chờ lo lắng khi chưa hiểu về cú pháp, bạn sẽ học ngay ở phần tiếp theo thôi).

```
SELECT * FROM khach_hang, hon_nhan  
WHERE khach_hang.ba_xa = hon_nhan.ba_xa
```

Việc thực hiện này chỉ đúng đối với Trung, nhưng sẽ có vấn đề đối với Khai bởi vì anh ta hãy còn độc thân và ba\_xa của anh ta là NULL.

Trong chương 3 bạn sẽ khảo sát kỹ hơn về vấn đề này.

## Index

Người ta nói rằng ưu điểm vượt trội của Hệ quản trị CSDL quan hệ là nó thực hiện các việc tìm kiếm hay sắp xếp những khối lượng dữ liệu khổng lồ một cách rất nhanh chóng. Sở dĩ nó thực hiện được việc này là do nó có chứa một cơ cấu lưu trữ dữ liệu gọi là INDEX.

INDEX cho phép database server tạo được một field đặc trưng tìm kiếm với tốc độ khó ngờ. Các INDEX đặc biệt hỗ trợ một hoặc một nhóm các record trong một table chứa số lượng lớn các record. Chúng cũng hỗ trợ tốc độ cho các hàm liên kết hoặc tách nhóm dữ liệu như min(), max(), bạn sẽ tìm hiểu ở Chương 3.

Với các tính năng vượt trội này, tại sao người ta lại không tạo index trong tất cả các field của một table? Có một số điều trở ngại như sau: Thứ nhất, index có sẽ làm chậm một số tiến trình trong CSDL. Mỗi lần bảo trì các index Database Server phải mất khá nhiều thời gian. Có một vài trường hợp chính các index làm cho chúng chậm hẳn. Nếu như trên table của bạn tất cả các record đều giống y như nhau thì không có lý do gì để bạn tạo index. Các index dư thừa chỉ làm cho tốn thêm không gian đĩa của bạn mà thôi.

Đối với một table có gắn khoá chính (primary key) thì thông thường field có khoá này được dùng vào việc tìm kiếm cho nên index sẽ được gán tự động trên field này. Bạn sẽ gặp lại rất nhiều công việc tạo index ở các phần tiếp theo sau.

## Lệnh CREATE Database

Trước khi tạo được một Table thì điều tất yếu là bạn phải tạo được một Database cái đã. Việc này dễ dàng và nhanh chóng thôi. Lệnh CREATE được sử dụng như sau:

```
mysql> create database database_name;
```

*Nếu như bạn thắc mắc rằng sau khi tạo database nó sẽ nằm ở trong thư mục nào trên ổ đĩa của bạn thì bạn hãy tìm trong ..\mysql\data xem có không.*

Khi đặt tên cho database, hay đặt tên cho field và index gì đấy tránh trường hợp đặt những cái tên khó nhớ hoặc dễ bị lẫn lộn. Đối với một số hệ thống Unix chẳng hạn có sự phân biệt chữ HOA/thường thì CSDL chạy trên nó cũng ảnh hưởng theo.

Bạn hãy chọn một quy ước cho riêng mình trong việc đặt tên để khỏi nhầm lẫn về sau. Chẳng hạn tên của table và field đều đặt chữ thường chẳng hạn. Nên nhớ là không được sử dụng khoảng trắng.



Bây giờ bạn tìm hiểu cả hai cách tạo database: Cách thứ nhất tạo thủ công từ dấu nhắc dòng lệnh DOS, cách thứ hai sử dụng các lệnh trong PHP.

Cách thứ nhất tôi đã có trình bày ở chương giới thiệu và bạn đã tạo một database tên là guestbook. Cú pháp tạo như sau:

```
mysql> create database guestbook;
```

Cách thứ hai là sử dụng lệnh trong PHP, bạn có thể dùng hàm `mysql_create_db()` hoặc `mysql_query()`. Nhưng nên nhớ trước khi tạo bạn phải thực hiện được kết nối với database server.

```
$conn = mysql_connect("localhost", "username", "password")  
or die ("Could not connect to localhost");  
mysql_create_db("my_database") or  
die ("Could not create database");  
$string = "create database my_other_db";  
mysql_query($string) or  
die(mysql_error());
```

# Lệnh USE Database

Sau khi đã tạo được một database mới trong database server bạn sẽ bắt đầu chọn nó để sử dụng cho công việc của mình. Cách thực hiện như sau:

## 1. Command Prompt:

```
mysql> use database_name;
```

## 2. Trong PHP:

```
$conn = mysql_connect("localhost", "username", "password")  
or die ("Could not connect to localhost");  
mysql_select_db("test", $conn) or  
die ("Could not select database");
```