

CÁC THAO TÁC KẾT NỐI CƠ SỞ DỮ LIỆU MYSQL

Các bước xây dựng chương trình có kết nối tới CSDL My SQL:

Thông thường, trong một ứng dụng có giao tiếp với CSDL, ta phải làm theo bốn trình tự sau:

Bước 1: Thiết lập kết nối tới CSDL.

Bước 2: Lựa chọn CSDL.

Bước 3: Tiến hành các truy vấn SQL, xử lý các kết quả trả về nếu có

Bước 4: Đóng kết nối tới CSDL.

Chú ý: Trước khi sử dụng PHP để lập trình với MySQL, hãy sử dụng chương trình quản lý phpMyAdmin để tạo trước một CSDL, vài bảng cũng như người dùng... để tiện thực hành. Bài hướng dẫn sử dụng phpMyAdmin sẽ được trình bày sau.

I. Thiết lập kết nối tới MySQL.

Để kết nối tới MySQL, ta sử dụng hàm `mysql_connect()`

Cú pháp:

```
mysql_connect(host,tên_truy_cập,mật_khẩu);
```

Trong đó:

host là chuỗi chứa tên (hoặc địa chỉ IP) của máy chủ cài đặt MySQL.

tên_truy_cập là chuỗi chứa tên truy cập hợp lệ của CSDL cần kết nối

mật_khẩu là chuỗi chứa mật khẩu tương ứng với tên truy cập

Ví dụ:

```
<?
mysql_connect ("localhost", "root","root");
?>
```

Khi kết nối tới MySQL thành công, hàm sẽ trả về giá trị là một số nguyên định danh của liên kết, ngược lại, hàm trả về giá trị false.

Ta có thể dùng hàm if để kiểm tra xem có kết nối được tới MySQL hay không:

```
<?php
$link = mysql_connect("localhost", "mysql_user", "mysql_password");
if (!$link)
{
    echo "Không thể kết nối được tới CSDL MySQL";
}
?>
```

II. Lựa chọn CSDL

Để lựa chọn một CSDL nào đó mà người sử dụng có tên là `tên_truy_cập` có quyền sử dụng, ta dùng hàm `mysql_select_db`:

```
mysql_select_db (tên_CSDL);
```

Hàm này thường được dùng sau khi thiết lập kết nối bằng hàm `mysql_connect`. Hàm này trả về `true` nếu thành công, `false` nếu thất bại.

Code:

```
<?php
$link = mysql_connect("localhost", "mysql_user", "mysql_password");
if (!$link)
{
    echo "Không thể kết nối được tới MySQL";
}
if (!mysql_select_db ("nhatnghecms"))
{
    echo "Không thể lựa chọn được CSDL Forums";
}
?>
```

III. Đóng kết nối tới CSDL:

Để đóng kết nối tới CSDL, ta dùng hàm `mysql_close`.

Hàm này có nhiệm vụ đóng kết nối tới CSDL có mã định danh được tạo ra bởi hàm

```
mysql_connect()
```

Cú pháp:

```
mysql_close(mã định danh kết nối).
```

Ví dụ:

```
<?php
$link = mysql_connect("localhost", "mysql_user", "mysql_password");
if (!$link)
{
    echo "Không thể kết nối được tới MySQL";
}
mysql_close($link);
?>
```

IV. Thực hiện các truy vấn thêm, xóa, sửa bản ghi thông qua câu lệnh SQL

Trong bài viết này, sử dụng MySQL chạy trên nền máy chủ localhost, người sử dụng có username là `mysql_user`, mật khẩu là `mysql_password`.

CSDL có tên là `nhatnghecms`, trong đó có một bảng là `nncms_menus` để lưu thông tin các menu của trang web.

Bảng `nncms_menus` có các trường sau:

Field	Type
<u>idMenu</u>	int(255)
idCapCha	int(255)
TieuDe	varchar(255)
LienKet	varchar(500)
ThuTu	int(10)
AnHien	tinyint(1)

Để thực thi một câu lệnh SQL bất kỳ trong PHP tác động lên MySQL, ta dùng hàm

```
mysql_query (chuỗi_câu_lệnh_SQL)
```

1. Thêm một bản ghi vào bảng.

Ví dụ chèn thêm một menu liên hệ vào bản và link tới file `contact.php` :

```
<?
$sql = "INSERT INTO `nncms_menus` (`idMenu`, `idCapCha`, `TieuDe`, `LienKet`,
`ThuTu`, `AnHien`) VALUES (NULL, '0', 'Liên Hệ', 'contact.php', '1', '1') ";

mysql_query ($sql);
?>
```

2. Xóa một bản ghi khỏi bảng.

Trong trường hợp này, ta sử dụng câu lệnh SQL `DELETE FROM`:

VD: Xóa khỏi bảng `nncms_menus` với menu có `idMenu = 30`

```
<?
$sql = "DELETE FROM `nncms_menus` WHERE `idMenu` = 30 ";
mysql_query ($sql);
?>
```

3. Sửa thông tin của bản ghi trong bảng.

Trong trường hợp này, tôi sử dụng cú pháp `UPDATE`:

VD: Cập nhật cấp cha cho menu Liên hệ có `idMenu = 30` :

```
<?
$sql = "UPDATE `nncms_menus` SET `idCapCha` = '1' WHERE `idMenu` = 30 ";
mysql_query ($sql);
?>
```

4. Sử dụng SQL để truy vấn dữ liệu trong PHP

Dùng hàm `mysql_query()` để tiến hành các truy vấn sửa, thêm và xóa. Bài này tập trung vào việc ứng dụng PHP để lấy thông tin từ CSDL.

Cú pháp hàm `mysql_query` như sau:

```
mysql_query(câu lệnh sql).
```

VD: Để lấy tất cả các bản ghi trong bảng "`nncms_menus`", ta sẽ gọi hàm `mysql_query()` như sau:

```
<?
mysql_query ("Select * from nncms_menus")
?>
```

Vấn đề là khi thực hiện truy vấn này, dữ liệu sẽ được trả về dưới dạng một bảng. Ta sẽ tiến hành lấy từng dòng của bảng đó ra dưới dạng một mảng (sử dụng hàm `mysql_fetch_array()`), tiếp đó, việc xử lý các phần tử của mảng đó như thế nào là tùy ý các bạn.

Ví dụ:

```
<?php
$result = mysql_query("SELECT * FROM nncms_menu");
while ($row = mysql_fetch_array($result, MYSQL_ASSOC))
{
    printf("ID: %s Tiêu Đề: %s", $row["TieuDe"]);
}

mysql_free_result($result);
?>
```

Đoạn lệnh trên lặp duyệt qua bảng kết quả `result`, đối với mỗi dòng của bảng, hàm `mysql_fetch_array()` sẽ đảm nhiệm việc tách dòng đó rồi đưa vào một mảng có tên là `$row`. Mỗi phần tử của mảng này sẽ là giá trị của cột có tên tương ứng.

Như vậy, trước khi xác định dữ liệu cần lấy là gì, chúng ta phải thiết kế câu truy vấn SQL sao cho hợp lý và đảm bảo rằng truy vấn đó chỉ lấy ra những thông tin gì cần sử dụng. Nếu lấy ra nhiều quá sẽ gây quá tải hệ thống dẫn đến treo máy.

Sau khi đã có được câu truy vấn SQL, ta chỉ việc kết hợp với hàm `mysql_query()` để trả kết quả về dưới dạng một biến nào đó.

Phần tiếp theo, chúng ta sẽ dùng một hàm có tên là `mysql_fetch_array()` để lấy từng dòng dữ liệu ra để xử lý.

Hàm **mysql_fetch_array** có hai tham số:

- Tham số đầu tiên xác định biến nắm giữ kết quả trả về bởi hàm **mysql_query()**
- Tham số thứ hai xác định kiểu định danh trường:
 - o + **MYSQL_ASSOC**: Đưa ra một mảng với chỉ số của mảng là tên trường
 - o + **MYSQL_NUM**: Đưa ra một mảng với chỉ số của mảng là số thứ tự của trường được liệt kê trong câu truy vấn SQL
 - o + **MYSQL_BOTH**: Đưa ra một mảng với chỉ số kết hợp giữa hai loại trên

Thông thường, ta dùng **MYSQL_ASSOC** để sử dụng chỉ số bằng tên các trường.

Khi truy vấn CSDL MySQL, ta có thể sử dụng thêm một số hàm phụ trợ:

mysql_num_rows(biến kết quả): trả về số các dòng lấy được từ biến kết quả.

VD:

```
<?php
$link = mysql_connect("localhost", "mysql_user", "mysql_password");
mysql_select_db("database", $link);
$result = mysql_query("SELECT * FROM table1", $link);
$num_rows = mysql_num_rows($result);
echo "Có $num_rows bản ghi được tìm thấy\n";
?>
```

Tương tự ta có:

mysql_num_fields (biến kết quả): Trả về số trường (cột) trong tập kết quả

mysql_affected_rows (): Trả về số bản ghi bị ảnh hưởng bởi các câu lệnh SQL INSERT, UPDATE, DELETE.

Ví dụ:

```
<?php
mysql_pconnect("localhost", "mysql_user", "mysql_password") or
die ("Could not connect");
mysql_query("DELETE FROM mytable WHERE id < 10");
printf("Số bản ghi bị xóa: %d\n", mysql_affected_rows());
?>
```

mysql_data_seek (nguồn kết quả, dòng cần truy cập): Hàm này chuyển con trỏ dòng của nguồn kết quả đến một dòng nào đó. Dòng đầu tiên trong bảng kết quả được đánh số là 0, dòng cuối cùng chính là **mysql_num_rows()-1**. Hàm trả về TRUE nếu thành công, ngược lại là FALSE

Chú ý rằng hàm này chỉ sử dụng cùng với hàm **mysql_query**

Ví dụ:

```
<?php
$link = mysql_pconnect("localhost", "mysql_user", "mysql_password")
or die("Could not connect");

mysql_select_db("samp_db")
or exit("Could not select database");

$query = "SELECT last_name, first_name FROM friends";
$result = mysql_query($query)
or die("Query failed");

/* fetch rows in reverse order */
for ($i = mysql_num_rows($result) - 1; $i >= 0; $i--) {
    if (!mysql_data_seek($result, $i)) {
        echo "Cannot seek to row $i\n";
        continue;
    }

    if!($row = mysql_fetch_object($result))
        continue;

    echo "$row->last_name $row->first_name<br />\n";
}

mysql_free_result($result);
?>
```

