

ỦY BAN NHÂN DÂN TỈNH BR – VT  
**TRƯỜNG CAO ĐẲNG NGHỀ**



## GIÁO TRÌNH

**MÔ ĐUN PHP & MySQL**

**NGHỀ: LẬP TRÌNH MÁY TÍNH**

**TRÌNH ĐỘ: CAO ĐẲNG**

*Ban hành kèm theo Quyết định số: 01/QĐ-CDN, ngày 04 tháng 01 năm 2016  
của Hiệu trưởng trường Cao đẳng nghề tỉnh Bà Rịa – Vũng Tàu*



***Bà Rịa – Vũng Tàu, năm 2016***



## **TUYÊN BỐ BẢN QUYỀN**

Tài liệu này thuộc loại sách giáo trình nên các nguồn thông tin có thể được phép dùng nguyên bản hoặc trích dùng cho các mục đích về đào tạo và tham khảo.

Mọi mục đích khác mang tính lệch lạc hoặc sử dụng với mục đích kinh doanh thiếu lành mạnh sẽ bị nghiêm cấm.

# LỜI GIỚI THIỆU

Giáo trình “PHP & MySQL” được biên soạn dành cho sinh viên cao đẳng nghề Lập trình máy tính với mục tiêu cung cấp cho người học các kiến thức và kỹ năng cần thiết để có thể hoàn thành ứng dụng web quảng cáo sản phẩm, thương mại điện tử, ... bằng ngôn ngữ PHP sử dụng cơ sở dữ liệu MySQL.

Tác giả đã nghiên cứu một số tài liệu, công nghệ hiện đại kết hợp với kinh nghiệm làm việc thực tế để viết nên giáo trình này. Nội dung được tác giả trình bày cô đọng, dễ hiểu kèm theo các bước hướng dẫn thực hành chi tiết giúp cho người học dễ tiếp thu các kiến thức cần thiết và hình thành được kỹ năng nghề.

Trong quá trình biên soạn, chắc chắn giáo trình còn nhiều thiếu sót. Tác giả rất mong nhận được ý kiến đóng góp của quý thầy/cô và các em sinh viên để tiếp tục hoàn thiện hơn.

Xin chân thành cảm ơn quý đồng nghiệp, bạn bè đã có những ý kiến đóng góp trong quá trình xây dựng giáo trình này.

Bà Rịa – Vũng Tàu, ngày ..... tháng ..... năm 2016

Tham gia biên soạn

1. Phan Hữu Phước – Chủ biên

# MỤC LỤC

## MÔ ĐƠN: PHP & MySQL

### Vị trí, tính chất, ý nghĩa và vai trò của mô đơn:

Sau khi học xong Thiết kế web, Cấu trúc dữ liệu & Giải thuật, Lập trình hướng đối tượng, Cơ sở dữ liệu.

Cung cấp cho người học kiến thức và kỹ năng để có thể thiết kế, xây dựng và phát triển ứng dụng web theo hướng mã nguồn mở bằng ngôn ngữ lập trình PHP với Hệ quản trị cơ sở dữ liệu MySQL

### Mục tiêu của mô đơn:

Biết kiến thức tổng quan về PHP và MySQL.

Cài đặt Hệ quản trị cơ sở dữ liệu MySQL.

Cài đặt PHP và cấu hình IIS trên hệ điều hành Windows.

Sử dụng phpmyadmin để quản trị MySQL

Xây dựng và phát triển ứng dụng web bằng ngôn ngữ PHP với cơ sở dữ liệu MySQL.

Thực hiện các bài tập ứng dụng về web đảm bảo đúng trình tự. An toàn cho người và thiết bị.

### Nội dung của mô đơn:

T T	Tên các bài trong mô đơn	Thời gian	Hình thức giảng dạy
1	Tổng quan về PHP và MySQL	5	Tích hợp
2	Quản trị MySQL trên MySQL Command	7	Tích hợp
3	Quản trị MySQL bằng phpmyadmin	5	Tích hợp
	Kiểm tra bài 1 → 3	1	Thực hành
4	Tổng quan về PHP	4	Tích hợp
5	Đọc dữ liệu từ địa chỉ và form	3	Tích hợp
6	Thao tác với các kiểu dữ liệu trong PHP	6	Tích hợp
7	Cấu trúc điều khiển	5	Tích hợp
	Kiểm tra bài 4 → 7	3	Thực hành
8	Xây dựng và sử dụng hàm	4	Tích hợp
9	Sử dụng include và require	2	Tích hợp
10	Tạo các lớp đối tượng	5	Tích hợp
	Kiểm tra bài 8 → 10	2	Thực hành

<b>11</b>	Truy xuất MySQL trong PHP	20	Tích hợp
	Kiểm tra bài 8 → 11	3	Thực hành
<b>12</b>	Các đối tượng quản lý trong PHP	5	Tích hợp
<b>13</b>	Sử dụng jQuery	13	Tích hợp
	Kiểm tra bài 8 → 13	4	Thực hành
	<b>Cộng</b>	<b>105</b>	





# BÀI 1.

## TỔNG QUAN VỀ PHP VÀ MYSQL

### **Giới thiệu:**

Các ngôn ngữ lập trình ứng dụng web hiện nay gồm: ASP.NET, PHP, JSP. Trong đó, phổ biến nhất là ASP.NET và PHP (facebook được viết bằng PHP). Mỗi ngôn ngữ lập trình thường phù hợp với một CSDL thông dụng. CSDL thông dụng trong ngôn ngữ PHP là MySQL.

### **Mục tiêu:**

Biết nguồn gốc của PHP, MySQL

Lợi ích khi dùng PHP & MySQL để xây dựng và phát triển web

Biết địa chỉ để tải MySQL và PHP

Cài đặt và cấu hình MySQL

Cài đặt và cấu hình PHP

Chọn và cài đặt đúng version phù hợp với hệ điều hành

### **Giới thiệu MySQL**

MySQL là cơ sở dữ liệu nguồn mở phổ biến nhất Thế giới cho phép phân phối các ứng dụng cơ sở dữ liệu dựa trên nền tảng web với chi phí thấp và hiệu suất cao. MySQL do tập đoàn ORACLE cung cấp.

### **Giới thiệu PHP**

PHP là ngôn ngữ kịch bản thông dụng đặc trưng phù hợp cho việc phát triển ứng dụng web.

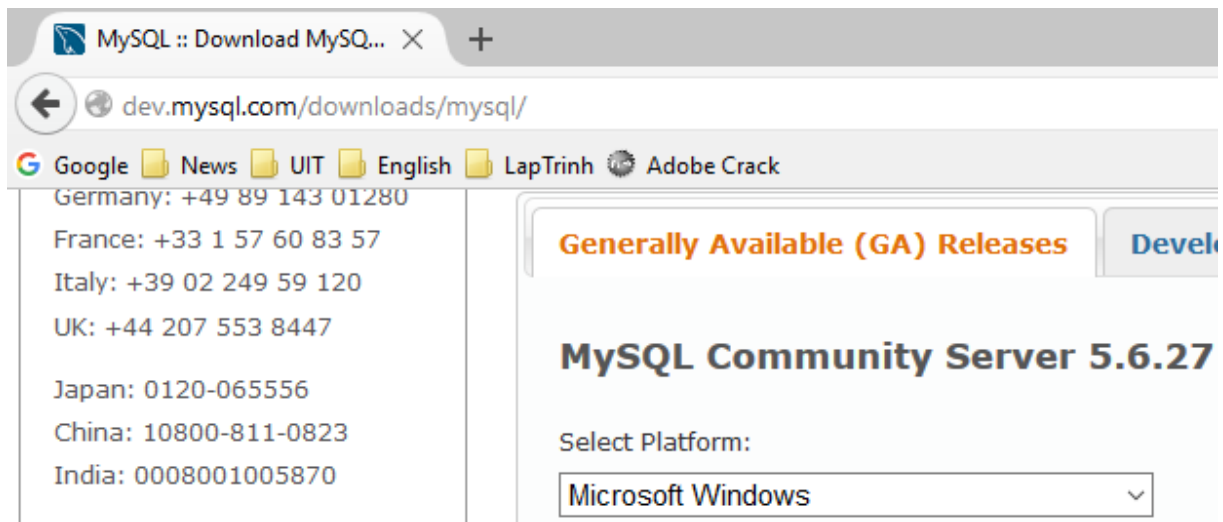
Nhanh chóng, linh hoạt và thực tế, PHP phát huy sức mạnh trên mọi ứng dụng từ blog cho đến những website phổ biến nhất trên Thế giới.

### **Cài đặt MySQL**

#### 3.1. Tải MySQL

Bước 1: truy cập vào trang web <http://dev.mysql.com/downloads/mysql/>

Bước 2: chọn hệ điều hành



Hình 1.1. Chọn version MySQL phù hợp với hệ điều hành

Bước 3: click chuột lên **Download** tại dòng **Installer MSI** để tải tập tin thực thi cài đặt.



### 3.2. Cài đặt và cấu hình MySQL

Click đôi chuột lên tập tin vừa tải về để cài đặt MySQL.

### Cài đặt và cấu hình PHP

#### 4.1. Tải PHP

Bước 1: truy cập trang web <http://php.net/downloads.php>

Bước 2: click chuột **Windows downloads** trong nhóm **Current Stable PHP**

• [Windows downloads](#)

Bước 3: chọn tải tập tin zip Thread Safe phù hợp với hệ điều hành (32 bit hay 64 bit).

#### 4.2. Cài đặt và cấu hình PHP

Giải nén tập tin zip vừa tải và click đôi chuột lên tập tin thực thi cài đặt.

## BÀI 2.

# QUẢN TRỊ MYSQL TRÊN MYSQL COMMAND

### Giới thiệu:

MySQL là ứng dụng nguồn mở nên được nhiều nhà phát triển tạo ra các công cụ để quản lý. Sau khi cài đặt MySQL, chúng ta có được một công cụ mặc định để quản lý (MySQL Command) bằng dòng lệnh.

### Mục tiêu:

Biết cấu trúc các lệnh sql trong MySQL

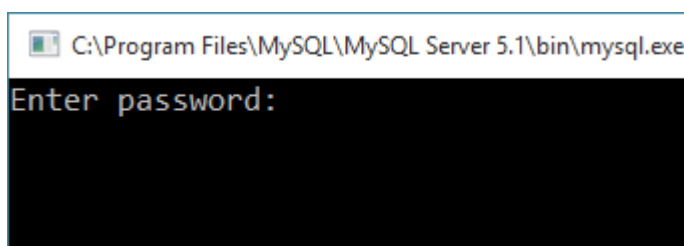
Tạo được CSDL, cấu trúc table trong MySQL trên MySQL Command

Thêm, xóa, sửa dữ liệu trên MySQL Command

Cẩn thận khi thao tác trên các table có quan hệ 1-n

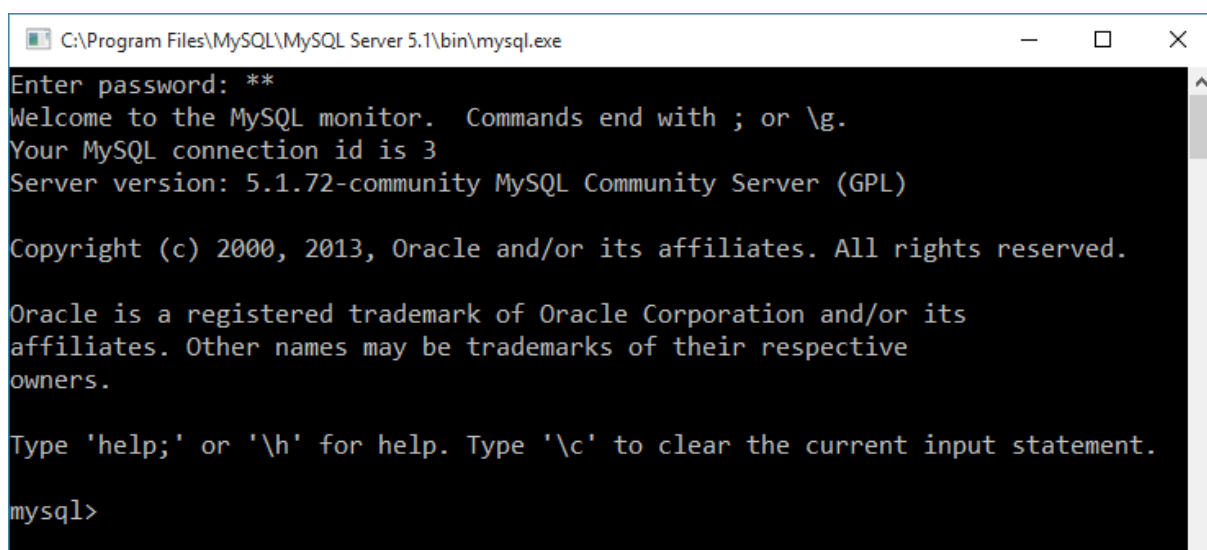
### 1. Đăng nhập và chọn CSDL

Bước 1: Mở MySQL Command Line Client.



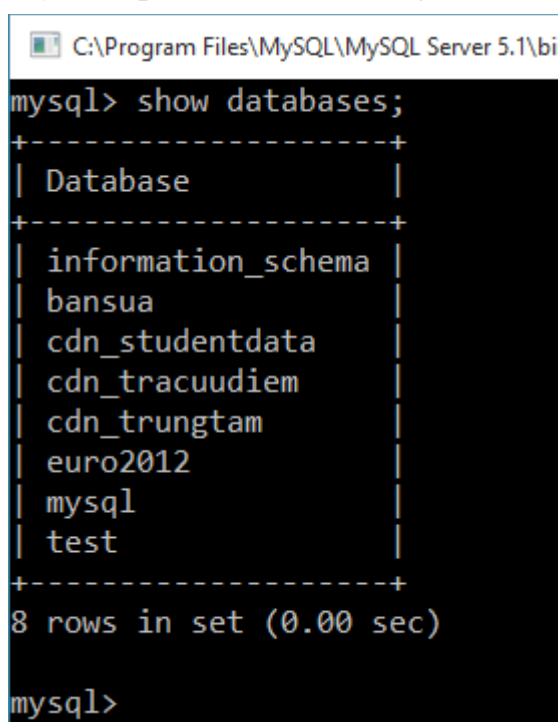
Hình 2.1. Nhập mật khẩu đăng nhập tài khoản sa của MySQL

Bước 2: nhập mật khẩu (đã nhập trong quá trình cài đặt MySQL).



Hình 2.2. Giao diện console quản lý MySQL

Bước 3: hiển thị danh sách CSDL đã có bằng lệnh **SHOW DATABASES;**  
Chú ý: mỗi lệnh của MySQL phải kết thúc bằng dấu chấm phẩy “;”.

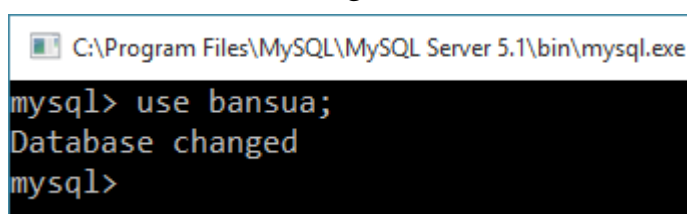


```
C:\Program Files\MySQL\MySQL Server 5.1\bin
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| bansua |
| cdn_studentdata |
| cdn_tracuudiem |
| cdn_trungtam |
| euro2012 |
| mysql |
| test |
+-----+
8 rows in set (0.00 sec)

mysql>
```

Hình 2.3. Xem danh sách CSDL trong MySQL

Bước 4: chọn CSDL cần làm việc bằng lệnh **USE <Tên CSDL>;**



```
C:\Program Files\MySQL\MySQL Server 5.1\bin\mysql.exe
mysql> use bansua;
Database changed
mysql>
```

Hình 2.4. Chọn CSDL làm việc

### Tạo CSDL mới

Tạo CSDL: dùng lệnh **CREATE DATABASE <Tên CSDL>;**

Ví dụ: CREATE DATABASE ThuongMaiDienTu;

Hủy CSDL: dùng lệnh **DROP <Tên CSDL>;**

Ví dụ: DROP ThuongMaiDienTu;

### Tạo và hủy table

Hiển thị danh sách table trong CSDL: **SHOW tables;**

Tạo table

```
CREATE TABLE <TênTable> (
    <TênCột> <KDL> [NOT NULL],
```

```

[<TênCột_i> <KDLi> [NOT NULL],]
PRIMARY KEY <PK_name> (<TênCột_j>[, <TênCột_k>]
[, FOREIGN KEY <FK_name> (<TênCột_x>[, <TênCột_y>])
REFERENCES <TableCha> (<TênCột_xc>[, <TênCột_yc>])
[ON DELETE ref_option]
[ON UPDATE ref_option]
]
);

```

Các giá trị của **ref\_option**: RESTRICT | CASCADE | SET NULL | NO ACTION

Ví dụ 1: Tạo 2 table Lop và HSSV. Mỗi lớp có nhiều hssv, mỗi hssv thuộc 1 lớp.

```

CREATE TABLE Lop(
    lopID INT NOTNULL AUTO_INCREMENT,
    tenlop VARCHAR(50)
    PRIMARY KEY PK_lop (lopID)
);
CREATE TABLE HSSV(
    hssvID VARCHAR(5),
    lopID INT,
    hoten VARCHAR(50),
    PRIMARY KEY PK_hssv (hssvID),
    FOREIGN KEY PK_hssv_lop (lopID)
        REFERENCES Lop (lopID)
        ON DELETE RESTRICT
        ON UPDATE RESTRICT
);

```

Hủy table: **DROP TABLE <Tên table>**

### **Đọc, thêm, xóa, cập nhật dữ liệu**

Sử dụng cấu trúc câu lệnh SQL trong MĐ Cơ sở dữ liệu.

## **CÂU HỎI, BÀI TẬP**

Tạo cơ sở dữ liệu CuaHangThoiTrang gồm các table theo yêu cầu sau:

### 2.1. PhanLoai

PhanLoaiID: int, AUTO\_INCREMENT

TenPhanLoai: varchar(100)

BiDanh: varchar(100)

## 2.2. Nhomsp

NhomspID: int, AUTO\_INCREMENT

PhanLoaiID: int

TenNhomsp: varchar(100),

BiDanh: varchar(100)

## 2.3. SanPham

SanPhamID: int, AUTO\_INCREMENT

NhomspID: int

TenSanPham: varchar(100)

MoTa: TEXT

DonGia: int

GiaKM: int

Hinh: varchar(255)

NgayCapNhat: date

NgungBan: bit

BiDanh: varchar(100)

2.4. Tạo quan hệ: Mỗi phân loại có nhiều nhóm sản phẩm, mỗi nhóm sản phẩm có nhiều sản phẩm.

## BÀI 3.

# QUẢN TRỊ MYSQL TRÊN PHPMYADMIN

### Giới thiệu:

Một trong những công cụ để quản lý MySQL được sử dụng phổ biến nhất hiện nay là ứng dụng web phpmyadmin. Với phpmyadmin, chúng ta có thể dễ dàng quản lý MySQL bằng các thao tác click chuột trên giao diện web mà không cần phải nhớ hết các cấu trúc lệnh sql.

### Mục tiêu:

Biết địa chỉ tải phpmyadmin

Tải và cấu hình phpmyadmin

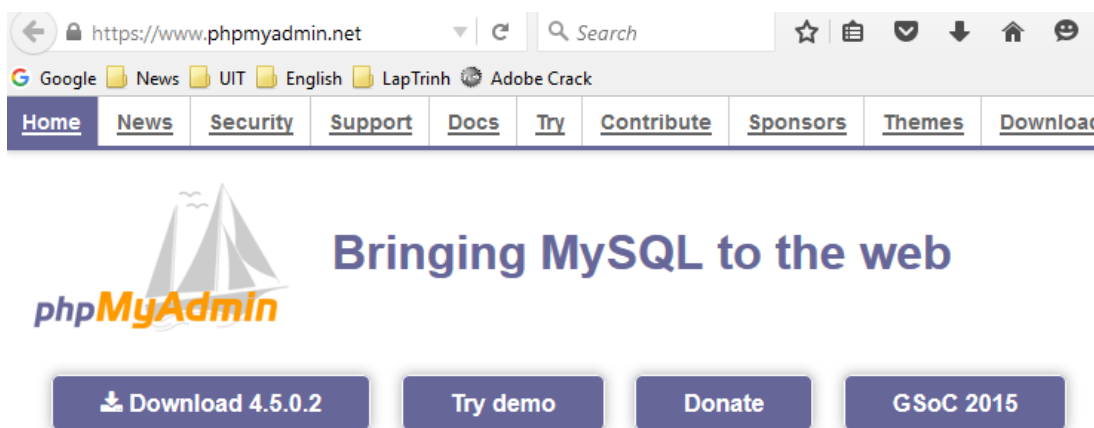
Quản trị được MySQL trên phpmyadmin

Tải đúng version phpmyadmin phù hợp với hệ điều hành, MySQL và PHP

### 1. Tải và cấu hình phpmyadmin

Bước 1: truy cập trang web <http://phpmyadmin.net>

Bước 2: click chuột lên Download để tải tập tin zip.

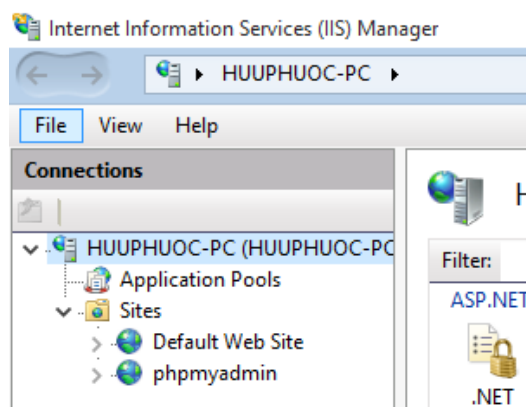


Hình 3.1. Trang web phpmyadmin.net

Bước 3: giải nén tập tin đã tải.

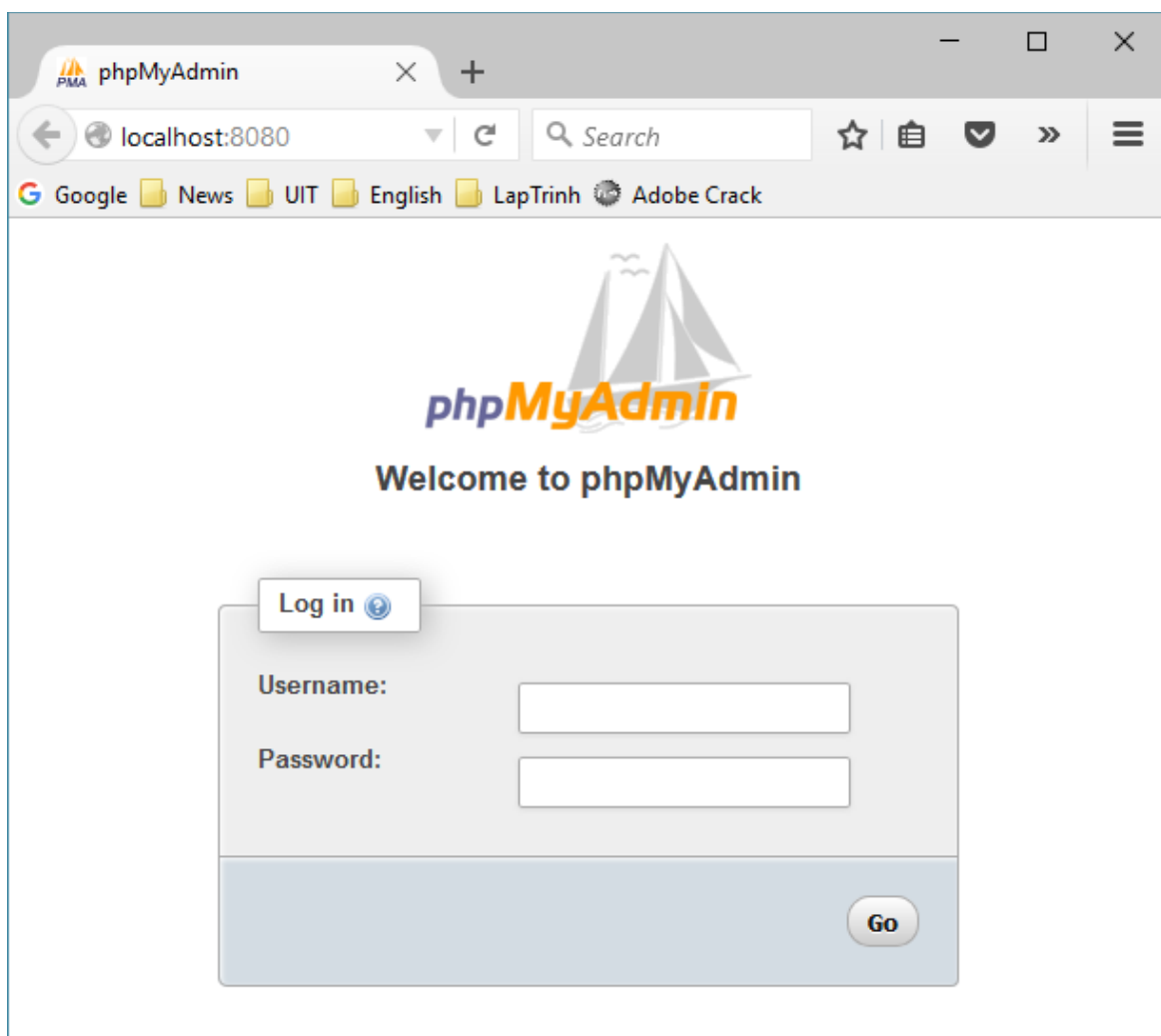
Bước 4: add website phpmyadmin vào IIS.





Hình 3.2. Cấu hình phpmyadmin trong IIS

Bước 5: truy cập website phpmyadmin theo địa chỉ đã cấu hình trên IIS.



Hình 3.3. Đăng nhập vào phpmyadmin

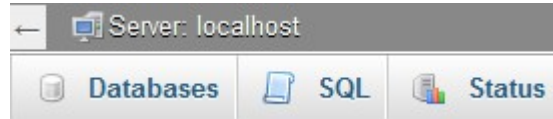
Bước 6: đăng nhập bằng username và password hợp lệ.

### Tạo CSDL

Bước 1: Trong giao diện phpmyadmin, click chuột lên biểu tượng Home ở khung trái.

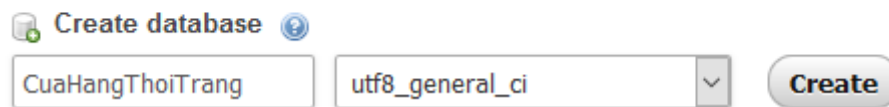


Bước 2: chọn Databases ở khung phải



Bước 3: Nhập tên CSDL cần tạo, chọn utf8\_general\_ci trong danh sách Collation rồi chọn Create

## Databases



## Hủy CSDL

Bước 1: chọn biểu biểu Home

Bước 2: chọn Databases

Bước 3: Đánh dấu chọn CSDL cần hủy rồi click chọn biểu tượng Drop

Database	Action
<input type="checkbox"/> bansua	Check Privileges
<input type="checkbox"/> cdn_studentdata	Check Privileges
<input type="checkbox"/> cdn_tracuudiem	Check Privileges
<input type="checkbox"/> cdn_trungtam	Check Privileges
<input checked="" type="checkbox"/> cuahangthoitrang	Check Privileges
<input type="checkbox"/> euro2012	Check Privileges
<input type="checkbox"/> information_schema	Check Privileges
<input type="checkbox"/> mysql	Check Privileges
<input type="checkbox"/> test	Check Privileges
<b>Total: 9</b>	

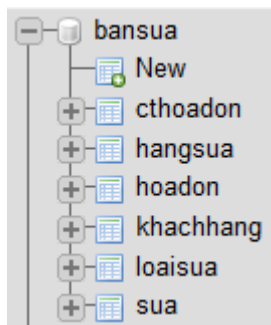
↑  Check All With selected: Drop

Hình 3.4. Đánh dấu chọn CSDL

## Tạo, hiệu chỉnh cấu trúc và hủy table

### 4.1. Tạo table

Bước 1: click chuột lên biểu tượng New trong CSDL



Hình 3.5. Chọn New để tạo CSDL mới

Bước 2: nhập tên table cần tạo, khai báo danh sách cột và kiểu dữ liệu

Table name:  Add  column(s)

Name	Type	Length/Values	Default
<input type="text"/>	INT	<input type="text"/>	None
<input type="text"/>	INT	<input type="text"/>	None

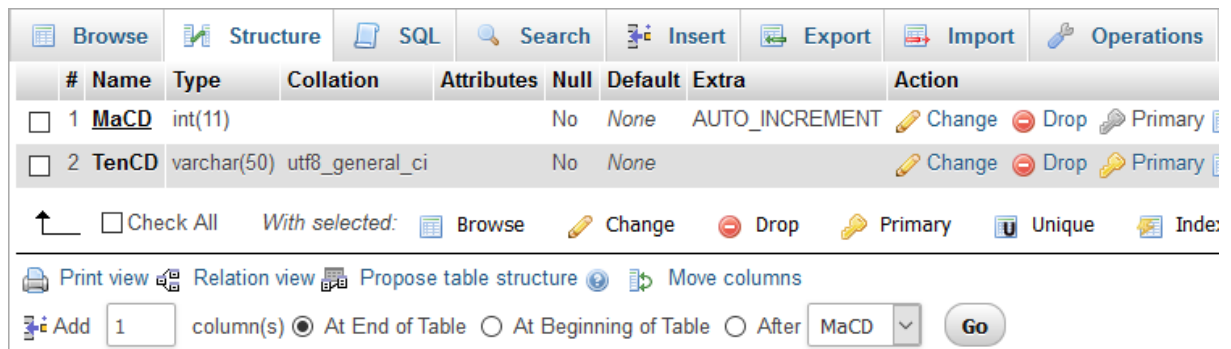
Hình 3.6. Tạo cấu trúc cho table

Bước 3: click chuột lên nút

### 4.2. Hiệu chỉnh cấu trúc table

Bước 1: ở khung trái, chọn table cần hiệu chỉnh

Bước 2: chọn Structure ở khung phải



Hình 3.7. Chọn Structure để hiệu chỉnh cấu trúc table

Mỗi dòng là một cột của table.

Muốn xóa cột này thì click chuột lên biểu tượng Drop trên dòng tương ứng.

Muốn thay đổi thông tin của cột thì click chuột lên Change

Muốn thêm cột thì nhập số cột cần thêm vào ô bên dưới rồi chọn Go

### 4.3. Hủy table

Bước 1: chọn CSDL trong khung trái

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> cthoadon		~0	InnoDB	utf8_general_ci	16 KiB	-
<input type="checkbox"/> hangsua		~7	InnoDB	utf8_general_ci	16 KiB	-
<input type="checkbox"/> hoaddon		~0	InnoDB	utf8_general_ci	16 KiB	-
<input type="checkbox"/> khachhang		~7	InnoDB	utf8_general_ci	16 KiB	-
<input type="checkbox"/> loaisua		~3	InnoDB	utf8_general_ci	16 KiB	-
<input type="checkbox"/> sua		~5	InnoDB	utf8_general_ci	16 KiB	-
6 tables	Sum	22	InnoDB	utf8_general_ci	96 KiB	0 B

Hình 3.8. Đánh dấu chọn table cần hủy

Bước 2: Trong hình trên, mỗi dòng là một table bên trong CSDL. Muốn hủy table nào thì chọn click chuột lên biểu tượng Drop trên dòng tương ứng.

Biểu tượng Empty sẽ xóa tất cả dữ liệu trong table nhưng không hủy table.

### Đọc, thêm, xóa, sửa dữ liệu

Bước 1: chọn table trong khung trái

Bước 2: chọn Browse trong khung phải để xem dữ liệu

Showing rows 10 - 14 (16 total, Query took 0.0008 sec)


```
SELECT *  
FROM `dmdoi`  
LIMIT 10, 5
```

Sort by key: None

	iddoi	idbang	tendoi	anh
<input type="checkbox"/>	NED	B	Netherlands	images/anhdoibong/NED.png
<input type="checkbox"/>	POL	A	Poland	images/anhdoibong/POL.png
<input type="checkbox"/>	POR	B	Portugal	images/anhdoibong/POR.png
<input type="checkbox"/>	RUS	A	Russia	images/anhdoibong/RUS.png
<input type="checkbox"/>	SWE	D	Sweden	images/anhdoibong/SWE.png

With selected: Change Delete Export

Hình 3.9. Chọn Browse để thêm, xóa, sửa dữ liệu trong table

Click chuột lên biểu tượng  Edit để hiện chỉnh dòng dữ liệu tương ứng

Click chuột lên biểu tượng  Delete để xóa dòng dữ liệu



Bước 3: muốn thêm dữ liệu thì click chuột lên biểu tượng

Bước 4: nhập thông tin cho dòng dữ liệu cần thêm rồi click chuột lên nút Go.

Column	Type	Function	Null	Value
iddoi	varchar(10)	<input type="text"/>		<input type="text"/>
idbang	varchar(5)	<input type="text"/>		<input type="text"/>
tendoi	varchar(50)	<input type="text"/>		<input type="text"/>
anh	varchar(100)	<input type="text"/>		<input type="text"/>

Hình 3.10. Giao diện thêm dòng dữ liệu mới

## Sao lưu và phục hồi dữ liệu

### 6.1. Sao lưu dữ liệu

Bước 1: chọn CSDL ở khung trái.

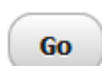


Bước 2: chọn

#### Export Method:

- Quick - display only the minimal options  
 Custom - display all possible options

#### Format:



Hình 3.11. Chọn định dạng export dữ liệu

Bước 3: click chuột lên nút Go.

Bước 4: chọn ổ đĩa, thư mục chứa tập tin sao lưu.

## 6.2. Phục hồi dữ liệu

Bước 1: chọn CSDL



Bước 2: chọn

Bước 3: click chuột lên Browse, chọn đến tập tin đã sao lưu trước đó

### Importing into the database "euro2012"

#### File to Import:

---

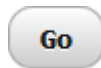
File may be compressed (gzip, zip) or uncompressed.

A compressed file's name must end in `.[format].[compression]`. Example: `.sql.zip`

Browse your computer:  No file selected. (Max: 2,048KiB)

Character set of the file:

*Hình 3.12. Chọn dữ liệu import*



Bước 4: click chuột lên nút bên dưới

## CÂU HỎI, BÀI TẬP

3.1. Tạo CSDL CuaHangThoiTrang bằng phpmyadmin.

3.2. Sao lưu dữ liệu CuaHangThoiTrang

3.3. Xóa CSDL CuaHangThoiTrang rồi phục hồi lại.

## BÀI 4.

# TỔNG QUAN VỀ PHP

### Giới thiệu:

Các trang web PHP là sự kết hợp giữa các HTML element và đoạn mã thực thi PHP. Chúng ta sẽ cùng tìm hiểu việc nhúng đoạn mã PHP trong các HTML element như thế nào.

### Mục tiêu:

Biết cấu trúc của đoạn mã PHP

Biết sử dụng biến, toán tử, các lệnh xuất trong PHP

Cẩn thận kiểu dữ liệu phù hợp khi tính toán

Cẩn thận phân biệt chữ in, thường, cú pháp của PHP

### 1. Cấu trúc đoạn mã PHP

Đoạn mã PHP được đặt trong các cặp thẻ sau:

Thẻ mở	Thẻ đóng
<code>&lt;?php</code>	<code>?&gt;</code>
<code>&lt;script language="php"&gt;</code>	<code>&lt;/script&gt;</code>

Ví dụ:

```
<?php
echo "PHP chào bạn!";
?>
```

Chú ý: mỗi lệnh của PHP kết thúc bằng dấu chấm phẩy ";".

### Sử dụng các lệnh xuất

```
echo string $arg1 [. string $... ]
```

Ví dụ:

```
<?php
$holot = "Nguyễn Văn";
$ten = "Thành";
echo "Họ tên: Nguyễn Văn Thành";
echo "Họ: $holot";
echo "Tên: $ten";
echo "Họ tên: $holot $ten";
echo "Họ tên: ". $holot. " ". $ten;
```

?>

Muốn xuống hàng thì dùng ký tự “\n”.

Ví dụ:

```
<?php
$holot = "Nguyễn Văn";
$ten = "Thành";
echo "Họ tên:\n$holot $ten";
?>
```

```
print ( string $arg )
```

Ví dụ:

```
<?php
print("Họ tên: $holot $ten");
?>
```

`sprintf ( string $format [, mixed $args [, mixed $... ] ] )`: trả về một chuỗi được định dạng.

`$format`: chuỗi định dạng.

`$args`: các tham số truyền vào chuỗi định dạng.

Các định dạng đại diện trong `$format`

`%b`: tham số là số nguyên, hiển thị dưới dạng nhị phân

`%c`: tham số là số nguyên, hiển thị dưới dạng ký tự ASCII

`%d`: tham số là số nguyên, hiển thị dưới dạng số nguyên có dấu.

`%f`: tham số là số thực, hiển thị dưới dạng số thực chấm động

`%o`: tham số là số nguyên, hiển thị dưới dạng hệ bát phân.

`%s`: tham số là chuỗi, hiển thị dưới dạng chuỗi.

`%u`: tham số là số nguyên, hiển thị dưới dạng số nguyên không dấu.

`%x, %X`: tham số là số nguyên, hiển thị dưới dạng thập lục phân.

Ví dụ:

```
<?php
$hoten = "Nguyễn Văn Thành";
$tuoi = 20;
$s = sprintf("Họ tên: %s, tuổi: %d", $hoten, $tuoi);
echo $s;
```



?>

`printf ( string $format [, mixed $args [, mixed $... ] ] )`: kết hợp giữa `echo` và `sprintf`

Ví dụ:

```
<?php
$hoten = "Nguyễn Văn Thành"; $tuoi=20;
printf("Họ tên: %s, tuổi: %d",$hoten,$tuoi);
?>
```

## **Khai báo và sử dụng biến**

Khai báo biến: `$tên_biến;`

Quy tắc đặt tên biến:

Bắt đầu bằng ký tự \$, theo sau là 1 ký tự hoặc dấu \_, tiếp đó là ký tự, hoặc số hoặc dấu \_.

Nêu khởi tạo giá trị ban đầu ngay khi khai báo biến.

Không trùng tên biến với tên hàm.

Không được bắt đầu bằng ký số.

Lưu ý: phân biệt chữ HOA – chữ thường.

Gán giá trị cho biến: `$tên_biến = <Giá trị>;`

Khai báo biến toàn cục: dùng từ khóa **global** đặt trước tên biến.

Sử dụng hằng

Hằng là một giá trị không thay đổi trong quá trình thực thi ứng dụng

Ví dụ: PI

Quy tắc đặt tên hằng: giống tên biến, thường dùng ký tự IN HOA

Khai báo hằng: `define ("TÊN_HẰNG", <Giá trị>;`

Ví dụ: `define ("PI", 3.14);`

Các phương thức kiểm tra giá trị của biến

`isset($tên_biến)`: kiểm tra biến có tồn tại hay không.

`empty($tên_biến)`: kiểm tra biến có rỗng (null, không chứa giá trị) hay không.

is\_numeric(\$tên\_biến): kiểm tra biến có phải là kiểu số hay không

is\_double(\$tên\_biến): kiểm tra biến có double hay không

gettype(\$tên\_biến): xác định kiểu dữ liệu của biến

### Sử dụng các toán tử

Toán tử	Ý nghĩa
$\$a = \$b$	Gán giá trị của $\$b$ cho $\$a$
$\$c = \$a + \$b$	Tính tổng $\$a$ và $\$b$ rồi gán cho $\$c$
$\$a++$	Tăng $\$a$ lên 1 đơn vị
$\$a += \$b$	Tương đương với $\$a = \$a + \$b$
$\$c = \$a - \$b$	Tính hiệu của $\$a$ trừ $\$b$ rồi gán cho $\$c$
$\$a--$	Giảm $\$a$ 1 đơn vị
$\$a -= \$b$	Tương đương với $\$a = \$a - \$b$
$\$c = \$a * \$b$	Tính tích của $\$a$ và $\$b$ rồi gán cho $\$c$
$\$a *= \$b$	Tương đương với $\$a = \$a * \$b$
$\$c = \$a / \$b$	Tính thương $\$a$ chia $\$b$ rồi gán cho $\$c$
$\$a /= \$b$	Tương đương $\$a = \$a / \$b$
$\$a == \$b$	TRUE nếu $\$a$ bằng $\$b$
$\$a === \$b$	TRUE nếu $\$a$ bằng $\$b$ và cùng kiểu dữ liệu
$\$a != \$b$ hoặc $\$a <> \$b$	TRUE nếu $\$a$ khác $\$b$
$\$a !== \$b$	TRUE nếu $\$a$ khác $\$b$ hoặc khác kiểu dữ liệu
$\$a < \$b$	TRUE nếu $\$a$ nhỏ hơn $\$b$
$\$a > \$b$	TRUE nếu $\$a$ lớn hơn $\$b$
$\$a <= \$b$	TRUE nếu $\$a$ nhỏ hơn hoặc bằng $\$b$
$\$a >= \$b$	TRUE nếu $\$a$ lớn hơn hoặc bằng $\$b$
$\$a <=> \$b$	Nếu $\$a$ nhỏ hơn $\$b$ thì trả về -1 Nếu $\$a$ bằng $\$b$ thì trả về 0 Nếu $\$a$ lớn hơn $\$b$ thì trả về 1

## BÀI 5.

# ĐỌC DỮ LIỆU TỪ ĐỊA CHỈ VÀ FORM

### **Giới thiệu:**

Bài này giúp HSSV biết kiến thức và kỹ năng truyền và truy xuất dữ liệu giữa các trang.

### **Mục tiêu:**

Biết phân biệt GET và POST

Biết nhận và xử lý dữ liệu từ địa chỉ và FORM

Thực hiện được các biện pháp an toàn cho máy tính

### **1. \$\_GET**

Là mảng kết hợp các cặp Key-Value được truyền thông qua tham số kèm theo URL.

Ví dụ: <http://localhost:8081/?option=chude&id=1>

\$\_GET["option"] có giá trị là "chude"

\$\_GET["id"] có giá trị là 1

### **\$\_POST**

Là mảng kết hợp các cặp Key-Value được truyền khi người dùng submit từ một form.

Thông tin được gửi từ form với phương thức POST không giới hạn lượng thông tin gửi đi và sẽ không được hiển thị trên URL nên người dùng không thể thấy được.

Cú pháp: \$\_POST["tên\_điều\_khiển"]

Ví dụ: Trên trang trang1.php có form như sau:

```
<form method="POST" action="trang2.php">  
    Nhập họ tên: <input type="text" name="hoten" />  
    <input type="submit" value="GỬI" />  
</form>
```

Trên trang2.php, muốn hiển thị dữ liệu người dùng đã nhập từ trang1.php

```
<?php  
    $s = $_POST["hoten"];
```

```
    echo "Họ tên đã nhập: ", $s;  
?>
```

## **\$\_REQUEST**

Là mảng kết hợp bao gồm \$\_GET và \$\_POST

Tức là, có thể dùng \$\_REQUEST thay cho \$\_GET và \$\_POST

## **CÂU HỎI, BÀI TẬP**

5.1. Tạo trang hiển thị table có m dòng và n cột với m và n là các tham số trên URL. Mỗi ô trong table có nội dung bất kỳ.

5.2. Tạo trang chứa form cho phép nhập thông tin cho phân loại thời trang, hiển thị lại những dữ liệu đã nhập khi người dùng submit.

# BÀI 6.

## THAO TÁC VỚI CÁC KIỂU DỮ LIỆU TRONG PHP

### Giới thiệu:

Bài này cung cấp cho HSSV kiến thức và kỹ năng tính toán, sử dụng hàm với từng kiểu dữ liệu trong PHP.

### Mục tiêu:

Biết sử dụng các kiểu dữ liệu trong PHP

Biết sử dụng các phép toán, hàm liên quan với các kiểu dữ liệu

Tạo được trang web nhận, xử lý và hiển thị kết quả

Cẩn thận truyền tham số có kiểu dữ liệu phù hợp cho hàm

### 1. Luận lý

Kiểu bool, chỉ có một trong hai giá trị là TRUE hoặc FALSE

### Số nguyên

Giá trị có thể là số trong hệ thập phân, thập lục phân và bát phân.

Ví dụ:

`$x = 1234; //hệ thập phân`

`$y = -123; //số âm hệ thập phân`

`$z = 0123; //hệ bát phân (bắt đầu bằng số 0)`

`$t = 0x1A; //hệ thập lục phân`

### Số thực

Kiểu float hoặc double

### Hàm toán học

Hàm	Công dụng
<code>abs(\$x)</code>	Trả về giá trị tuyệt đối của \$x
<code>ceil(\$x)</code>	Trả về số nguyên nhỏ nhất lớn hơn \$x
<code>floor(\$x)</code>	Trả về số nguyên lớn nhất nhỏ hơn \$x
<code>pi()</code>	Giá trị của PI
<code>min</code>	Tìm giá trị nhỏ nhất
<code>max</code>	Tìm giá trị lớn nhất

round(\$x, \$n)	Trả về số thực từ \$x được làm tròn còn \$n số thập phân
pow(\$x, \$y)	Tính \$x lũy thừa \$y
rand(\$x, \$y)	Trả về số nguyên ngẫu nhiên trong đoạn từ \$x đến \$y

## Chuỗi, ký tự

Mỗi ký tự chiếm 1 byte.

Mỗi chuỗi có thể chứa một hay nhiều ký tự.

Chuỗi không có giới hạn về kích thước.

## Hàm trên chuỗi

Hàm	Công dụng
trim(\$s)	Bỏ các khoảng trắng ở 2 đầu chuỗi s
strlen(\$s)	Trả về chiều dài của \$s
strcmp(\$s1, \$s2)	So sánh hai chuỗi \$s1 và \$s2
strpos(\$s_main, \$s_sub)	Trả về vị trí đầu tiên tìm thấy \$s_sub trong \$s_main. Nếu không tìm thấy thì trả về FALSE
str_replace(\$s1, \$s2, \$s_main)	Tìm trong \$s_main, thay \$s1 bằng \$s2
substr(\$s, \$pos, [\$len])	Từ vị trí \$pos trong \$s, lấy ra \$len ký tự
implode(";", array(\$s1, \$s2))	Kết hợp \$s1 và \$s2 thành một chuỗi cách nhau bằng dấu chấm phẩy ";"
explode(";", \$s)	Tách \$s thành mảng các chuỗi con dựa vào dấu chấm phẩy ";"

## Mảng

### 6.1. Khai báo

```
$tên_mảng = array();
```

Ví dụ:

```
$a = array();
```

Khai báo và khởi tạo

```
$tên_mảng = array([khóa=>]giá trị, ...);
```

Ví dụ

```
$a = array(1, 2, 3, 9, 7);
```

```
$b = array("k1"=>2, "k2"=>5);
```

## 6.2. Truy xuất

Cú pháp: `$tên_mảng[<khóa>]`

Ví dụ:

```
$x = $a[1]; // $x sẽ có giá trị là 2
$y = $b["k2"]; // $y sẽ có giá trị là 5
```

Đến số phần tử trong mảng: `$n = count($tên_mảng)`

Duyệt mảng

Cách 1: duyệt mảng có khóa tự động

```
for ($i=0; $i<$n; $i++)
{
    //Xử lý trên mỗi phần tử $tên_mảng[$i]
}
```

Ví dụ:

```
$mang = array(1, 2, 3, 5, 6, 4);
$n = count($mang);
for ($i=0; $i<$n; $i++)
    echo "\t" . $mang[$i];
```

Cách 2: duyệt mảng có khóa do người dùng tạo, chỉ quan tâm đến giá trị

```
foreach ($tên_mảng as $giá_trị)
{
    //Xử lý trên $giá_trị
}
```

Ví dụ:

```
$mang = array(1=> "Một", 2=> "Hai", 3=> "Ba", 4=> "Bốn");
foreach($mang as $x)
    echo "\t $x";
```

Cách 3: duyệt mảng có khóa do người tạo, lấy cả khóa và giá trị

```
foreach ($tên_mảng as $khóa=>$giá_trị)
{
    //Xử lý trên $khóa và $giá_trị
}
```

Ví dụ:

```
$mang = array("mot"=>1, "hai"=>2, "ba"=>3)
foreach ($mang as $k=>$v)
    echo "Khóa [$k] => Giá trị: $v";
```

### 6.3. Hàm trên mảng

Hàm	Công dụng
array_search	Trả về vị trí có giá trị cần tìm trong mảng Ví dụ: <code>\$k = array_search(&lt;Giá trị cần tìm&gt;, \$tên_mảng);</code>
array_merge	Ghép 2 mảng vào 1 mảng <code>\$a = array_merge(\$tên_mảng_1, \$tên_mảng_2);</code>
array_count_values	Đếm số lần xuất hiện của từng giá trị trong mảng Ví dụ: <code>\$mang = array(1, "hello", 1, "world", "hello", 2, "chào", 1)</code> <code>\$mang_slxh = array_count_values(\$mang);</code> <code>// \$mang_slxh là Array([1]=&gt;3, "hello"=&gt;2, "world"=&gt;3, 2=&gt;1, "chào"=&gt;1)</code>
array_unique	Tạo mảng chứa mỗi giá trị xuất hiện 1 lần duy nhất Ví dụ: <code>\$mang1 = array(1, 3, 1, 2, 5, 1, 3, 4);</code> <code>\$mang2 = array_unique(\$mang1);</code> <code>// \$mang2 là Array(1, 3, 2, 5, 4)</code>
array_diff	Tạo mảng chứa giá trị chỉ thuộc mảng 1 mà không thuộc mảng 2 Ví dụ: <code>\$mang1 = array("a"=&gt;"xanh", "đỏ", "tím", "vàng");</code> <code>\$mang2 = array("b"=&gt;"xanh", "vàng", "đỏ");</code> <code>\$mang_con_1 = array_diff(\$mang1, \$mang2);</code> <code>// \$mang_con_1 là array([1]=&gt;"tím")</code>

### CÂU HỎI, BÀI TẬP

6.1. Tạo trang web cho phép nhập họ tên, ngày sinh của một người. Sau khi submit thì xuất ra họ, tên và tuổi của người đó.

6.2. Tạo trang web nhập vào một số nguyên dương. Sau khi submit thì xuất ra n đường kẻ ngang (2 đường gần nhau phải có màu khác nhau).

### GỢI Ý, ĐÁP ÁN

6.1. Dùng hàm explode để tách họ, tên và ngày, tháng, năm sinh. Tra cứu hàm lấy năm hiện tại trên php.net để tính tuổi.

6.2. Dùng vòng lặp foreach. Dùng thẻ `<hr />` kết hợp với thuộc tính class (hoặc style) để tạo đường kẻ ngang có màu sắc khác nhau.



## BÀI 7.

# CẤU TRÚC ĐIỀU KHIỂN

### Giới thiệu:

Bài này cung cấp cho HSSV sử dụng được các cấu trúc điều khiển trong PHP.

### Mục tiêu:

Biết công dụng của các cấu trúc điều khiển

Sử dụng các cấu trúc điều khiển để giải các bài toán trong PHP

Thực hiện được các biện pháp an toàn cho máy tính

Cẩn thận tránh vòng lặp vô hạn

### 1. if, if ... else

#### 1.1. Cấu trúc if

Công dụng: kiểm tra nếu thỏa điều kiện thì thực hiện.

Cú pháp:

```
if (<Điều kiện>
{
    //Các lệnh được thực hiện khi điều kiện đúng
}
```

Ví dụ:

```
$a=3; $b=5;
if($b>$a)
    echo $b . ">" . $a;
```

#### 1.2. Cấu trúc if ... else

Công dụng: kiểm tra nếu điều kiện đúng thì thực hiện CV1, nếu điều sai thì thực hiện CV2.

Cú pháp:

```
if (<Điều kiện>
{
    //Các công việc 1
}
else
{
```

```

        //Các công việc 2
    }
Ví dụ:
$a=3; $b=5;
if($b>$a)
    echo $b . ">" . $a;
else
    echo $a . ">" . $b;

```

### **switch**

Công dụng: thực hiện công việc theo mỗi trường hợp.

Cú pháp:

```

switch(<Biểu thức>)
{
    case <gt1>:
        //thực hiện các công việc khi <Biểu thức> có giá
        trị là <gt1>
        break;
    case <gt2>:
        //thực hiện các công việc khi <Biểu thức> có giá
        trị là <gt1>
        break;
    [
    case <gt_i>:
        //thực hiện công việc khi <Biểu thức> có giá trị
        <gt_i>
        break;
    ]
    [
    default:
        //thực hiện các công việc khi <Biểu thức> có giá trị
        khác
    ]
}
Ví dụ:
$t=2;
switch ($t)
{
    case 1: echo "Một"; break;

```

```
    case 2: echo "Hai"; break;
    case 3: echo "Ba"; break;
    default: echo "Không hợp lệ";
}
```

## **for, foreach**

### 3.1. Vòng lặp for

Cú pháp:

```
for (bt1; bt2; bt3)
    statement
```

bt1: biểu thức khởi tạo biến đếm

bt2: biểu thức điều kiện

bt3: biểu thức thay đổi giá trị biến đếm

Ví dụ:

```
<?php
/* example 1 */

for ($i = 1; $i <= 10; $i++) {
    echo $i;
}

/* example 2 */

for ($i = 1; i<=20 ; $i++) {
    if ($i > 10) {
        echo $i;
    }
}
```

### 3.2. Vòng lặp foreach

Cú pháp:

```
foreach (array_expression as $value)
    statement
foreach (array_expression as $key => $value)
    statement
```

Ví dụ:

```
<?php
$arr = array(1, 2, 3, 4);
foreach ($arr as &$value) {
    $value = $value * 2;
}
// $arr is now array(2, 4, 6, 8)
unset($value); // break the reference with the last element
?>
```

```
$a = array(
    "one" => 1,
    "two" => 2,
    "three" => 3,
    "seventeen" => 17
);

foreach ($a as $k => $v) {
    echo "\$a[$k] => $v.\n";
}
```

## **while, do ... while**

Cú pháp:

```
while (expr)
    statement
```

Ví dụ:

```

<?php
/* example 1 */

$i = 1;
while ($i <= 10) {
    echo $i++; /* the printed value would be
                $i before the increment
                (post-increment) */
}

/* example 2 */

$i = 1;
while ($i <= 10):
    echo $i;
    $i++;
endwhile;
?>

```

### Các lệnh break, continue

Công dụng: thoát khỏi vòng lặp

Ví dụ:

```

<?php
$arr = array('one', 'two', 'three', 'four', 'stop', 'five');
while (list(, $val) = each($arr)) {
    if ($val == 'stop') {
        break; /* You could also write 'break 1;' here. */
    }
    echo "$val<br />\n";
}

```

## CÂU HỎI, BÀI TẬP

- 7.1. Tạo trang web cho phép nhập vào các hệ số a, b, c của phương trình bậc 2. Sau khi submit thì hiển thị kết quả nghiệm của phương trình.
- 7.2. Tạo trang web nhập vào năm dương lịch. Sau khi submit thì hiển thị CAN-CHI của năm.
- 7.3. Tạo trang web hiển thị lịch (dương lịch) của 1 năm.



## BÀI 8.

# XÂY DỰNG VÀ SỬ DỤNG HÀM

### **Giới thiệu:**

Bài này cung cấp cho HSSV kiến thức và kỹ năng xây dựng và sử dụng hàm tự định nghĩa.

### **Mục tiêu:**

Biết cú pháp định nghĩa hàm

Xây dựng và sử dụng hàm

Tạo trang web xử lý và hiển thị dữ liệu có sử dụng hàm

Tránh xây dựng các hàm mà PHP đã có sẵn

### **1. Định nghĩa hàm**

Cú pháp:

```
function <TênHàm> ($ts1, $ts2, ...)  
{  
    //các lệnh  
    [return <giá trị>;]  
}
```

Ví dụ 1:

```
function Xuat ($n)  
{  
    for ($i=0; $i<$n; $i++)  
        echo $i;  
}
```

Ví dụ 2:

```
function LuyThua ($a, $b)  
{  
    $kq = 1;  
    for ($i=1; $i<=$b; $i++)  
        $kq = $kq * $a;  
    return $kq;  
}
```

### **Tham số của hàm**

Tham trị: như các ví dụ ở mục 1

Tham biến: thêm dấu & trước tham số

Ví dụ:

```
function add_some_extra(&$string)
{
    $string .= 'and something extra.';
}
$str = 'This is a string, ';
add_some_extra($str);
echo $str;    // outputs 'This is a string, and something extra.'
```

Tham số có giá trị mặc định

```
function makecoffee($type = "cappuccino")
{
    return "Making a cup of $type.\n";
}
echo makecoffee();
echo makecoffee(null);
echo makecoffee("espresso");
```

## CÂU HỎI, BÀI TẬP

8.1. Làm lại các bài tập của Bài 7 (sử dụng hàm).



## **BÀI 9.**

### **SỬ DỤNG INCLUDE VÀ REQUIRE**

#### **Giới thiệu:**

Bài này cung cấp cho HSSV kiến thức và kỹ năng dùng các lệnh include và require để chia nhỏ chức năng vào từng trang và sử dụng cho các trang khác.

#### **Mục tiêu:**

Biết công dụng của các lệnh include, require

Sử dụng các lệnh include, include\_once, require, require\_once

Tạo trang web xử lý có sử dụng các tập tin thư viện

Xác định cẩn thận tập tin cần sử dụng 1 lần hay nhiều lần

#### **1. include, include\_once**

Công dụng chung: đặt các lệnh từ một tin thư viện vào vị trí hiện tại của lệnh include (include\_once).

Khác nhau: include\_once sẽ kiểm tra nếu chưa sử dụng tập tin thư viện thì mới đặt các lệnh vào.

Ví dụ 1:

```

vars.php
<?php

$color = 'green';
$fruit = 'apple';

?>

test.php
<?php

echo "A $color $fruit"; // A

include 'vars.php';

echo "A $color $fruit"; // A green apple

?>

```

Ví dụ 2:

```

config.php
<?php
return array("test">1);

-----
//first
$config = include_once("config.php");
var_dump($config);

$config = include_once("config.php");
var_dump($config);

-----
output will be
array(
    "test"=>1,
)

nothing

```

**require, require\_once**

require: tương tự include

require\_once: tương tự include\_once

## **CÂU HỎI, BÀI TẬP**

9.1. Làm lại bài tập của Bài 8, sử dụng hàm và require, include

# BÀI 10.

## TẠO CÁC LỚP ĐỐI TƯỢNG

### Giới thiệu:

Bài này cung cấp cho HSSV kiến thức và kỹ năng lập trình hướng đối tượng trong PHP.

### Mục tiêu:

Biết cú pháp định nghĩa lớp đối tượng

Xây dựng và sử dụng lớp đối tượng.

Tạo trang web sử dụng các lớp đối tượng xử lý

Xác định cẩn thận các lớp đối tượng cần xây dựng phù hợp với CSDL

### 1. Tổng quan

#### Ý nghĩa

Đối tượng (Object) là những thực thể tồn tại trong thế giới thực.



Hình 10.1: Mối tương quan giữa đối tượng thực tế với đối tượng trong lập trình

#### Ví dụ

Con người

Sinh viên Nguyễn Văn A

Nhân viên Huỳnh Thị B

Đồ vật

Phòng học 1PM1

Máy tính PCCH1

## Tiếp cận hướng đối tượng

Là kỹ thuật biểu diễn tự nhiên đối tượng trong thực tế với đối tượng trong chương trình.

## Lớp đối tượng

Là một khái niệm trong lập trình hướng đối tượng, mô tả những thực thể có chung tính chất và hành vi. Class định nghĩa những thuộc tính và hành vi được dùng cho những đối tượng của lớp đó.

Kết quả của sự trừu tượng hóa (Abstraction) các đối tượng là:

Cùng loại

Cùng các thông tin mô tả về đối tượng

## Xây dựng lớp đối tượng (class)

### 2.1. Tạo lớp đối tượng

```
Class <TênLớp>
```

```
{  
//Khai báo các thuộc tính  
//Gán và lấy giá trị các thuộc tính  
//Định nghĩa các phương thức  
}
```

### 2.2. Khai báo thuộc tính

Thuộc tính: thành phần lưu trữ các tính chất, đặc điểm của đối tượng.

Cú pháp: [<khóa truy cập>] <Tên thuộc tính>

<khóa truy cập>: private, protected, public

Nếu không có khóa truy cập thì sẽ hiểu là public.

Chú ý: có thể thiết lập những giá trị mặc định ban đầu cho tất cả các thuộc tính được tạo ra từ lớp đó.

### Ví dụ:

```
class NHANVIEN{  
    private $holot;  
    private $ten;  
    private $heso;  
}
```

### 2.3. Xây dựng phương thức

Phương thức là chức năng mà đối tượng có thể thực hiện, có thể có hoặc không có giá trị trả về.

Cú pháp:

```
[<khóa truy cập>] function <Tên phương thức> (các tham số)
{
    //các lệnh xử lý
}
```

Phương thức khởi tạo: phương thức có tên là `__construct` hoặc trùng với tên lớp.

Ví dụ:

```
public function NHANVIEN(){
    $heso=0;
}
```

hoặc

```
public function __construct(){
    $heso=0;
}
```

Phương thức hủy: phương thức có tên là `__destruct`

Ví dụ:

```
public function __destruct(){
    //Xử lý giải phóng vùng nhớ
}
```

Các phương thức xử lý khác

```
public function GetHolot(){
    return $holot;
}

public function SetHolot($s){
    $holot=$s;
}

public function HoTen(){
    return $holot . " " . $ten;
}
```

Ghi chú:

Trong các phương thức của lớp, có thể truy cập các thuộc tính hay phương thức thông qua con trỏ \$this.

Con trỏ \$this được dùng để chỉ đối tượng hiện tại đang thực thi phương thức.

Ví dụ:

```
public function SetHolot($holot){
    $this->holot=$holot;
}
```

#### 2.4. Từ khóa static

Sử dụng từ khóa static khi khai báo thuộc tính (hoặc phương thức) để cho biết đó là thuộc tính (hoặc phương thức) lớp.

Ví dụ thuộc tính static:

```
private $holot;
private $ten;
private $heso;
private static $luongcb;
```

Ví dụ phương thức static:

```
public static function GetLuongCB(){
    return $luongcb;
}
public static function SetLuongCB($luongcb){
    NHANVIEN::$luongcb = $luongcb;
}
```

### Sử dụng lớp đối tượng

#### 3.1. Khởi tạo đối tượng

Cú pháp: \$<tên biến> = new <Tên lớp>();

Ví dụ: \$nv = new NHANVIEN();

#### 3.2. Gán giá trị cho các thuộc tính

Cú pháp: \$<tên biến>-><tên thuộc tính> = <giá trị>;

Ví dụ:

\$nv->holot = "Nguyễn Văn";

\$nv->ten = "An";

#### 3.3. Gọi sử dụng các phương thức

Phương thức không có giá trị trả về

Cú pháp:

```
$<tên biến>-><tên phương thức>(các giá trị truyền vào nếu có);
```

Ví dụ: `$nv->SetHoLot("Trần Thanh");`

Phương thức có giá trị trả về

Cú pháp:

```
$<tên biến> = $<tên biến đối tượng>-><Tên phương thức>(các giá trị truyền vào nếu có);
```

Ví dụ:

```
$hoten = $nv->HoTen();
```

```
echo $hoten;
```

Truy xuất thuộc tính/phương thức static

Cú pháp:

```
<Tên lớp>::<Tên thuộc tính/Phương thức>
```

Ví dụ:

```
NHANVIEN::SetLuongCB(1150000);
```

```
$luongcb = NHANVIEN::GetLuongCB();
```

```
echo $luongcb;
```

## **CÂU HỎI, BÀI TẬP**

10.1. Xây dựng các lớp đối tượng tương ứng với table trong CSDL. Mỗi lớp đối tượng đặt trong một tập tin .php.



# BÀI 11.

## TRUY XUẤT MYSQL TRONG PHP

### Giới thiệu:

Bài này cung cấp cho HSSV kiến thức và kỹ năng lập trình PHP để truy xuất dữ liệu từ MySQL và hiển thị trên trang web.

### Mục tiêu:

Biết các bước đọc, ghi dữ liệu MySQL bằng PHP

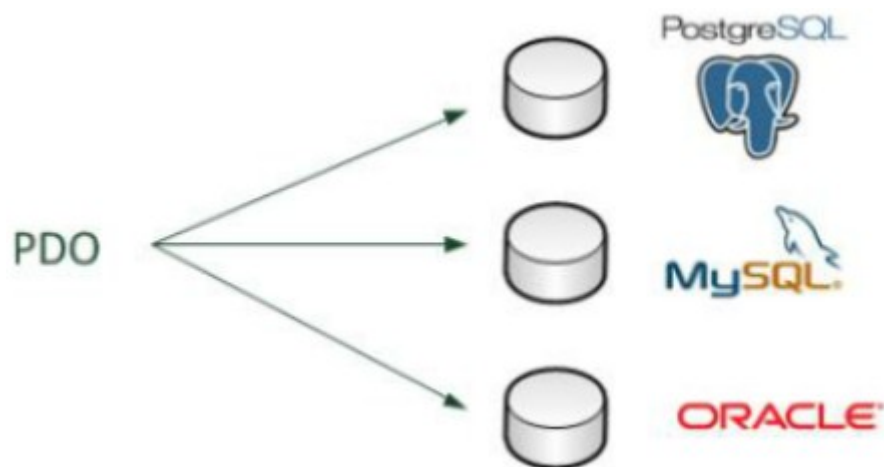
Tạo được lớp đối tượng xử lý dữ liệu MySQL

Tạo trang web đơn giản hiển thị dữ liệu MySQL

Thực hiện được các biện pháp an toàn cho máy tính

### 1. Giới thiệu PDO

PDO – PHP Data Objects: là một lớp truy cập dữ liệu cung cấp phương thức chuẩn để truy cập vào nhiều loại CSDL khác nhau.



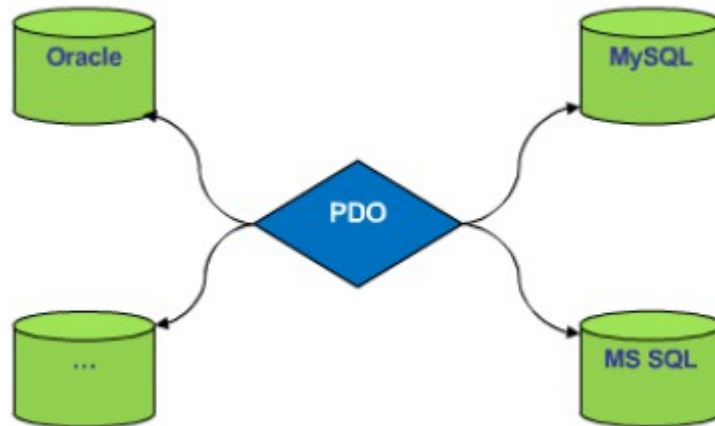
Hình 11.1: Các CSDL mà PDO có thể truy cập

PDO cung cấp một lớp truy cập dữ liệu trừu tượng. Nghĩa là, dù chúng ta đang sử dụng CSDL nào, đều có thể sử dụng phương thức như nhau để truy xuất CSDL.

PDO không cung cấp một CSDL trừu tượng. Vì vậy, chúng ta cần phải sử dụng một lớp trừu tượng nếu chúng ta cần CSDL đó.

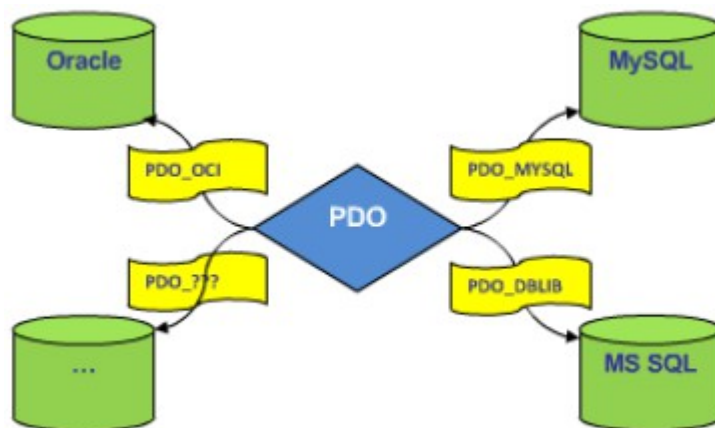
## Lý do chọn PDO

Hướng đối tượng: rất phù hợp để tích hợp vào mô hình MVC (rất nhiều framework lớn hiện nay sử dụng PDO).



Hình 11.2: Mô hình làm việc giữa PDO và các loại CSDL

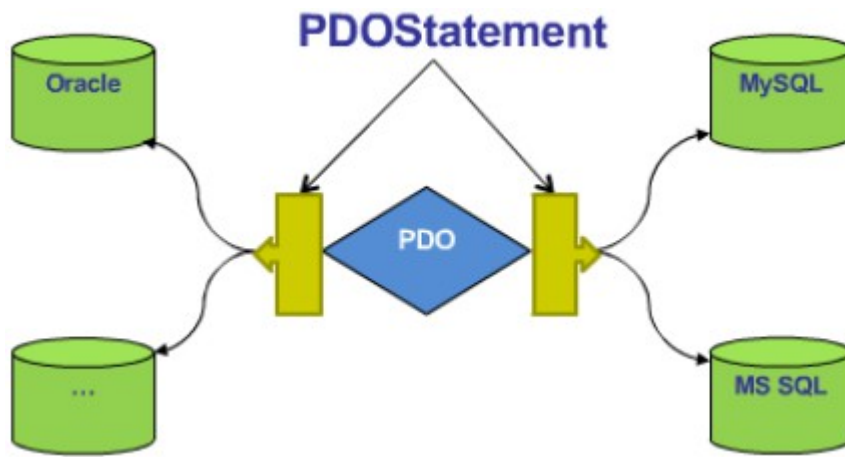
Nhanh và đơn giản: việc thao tác với từng loại CSDL sẽ do từng driver tương ứng đảm nhiệm.



Hình 11.3: Mô hình driver tương ứng với các CSDL của PDO

Linh động: Để chuyển đổi sang CSDL khác, chúng ta chỉ cần thay đổi tên driver tương ứng.

An toàn: với giải pháp PDOStatement, chúng ta không còn lo sợ SQL injection tấn công.



Hình 11.4: Mô hình giải pháp PDOStatement

## Tạo kết nối – Hủy kết nối

### 3.1 Tạo kết nối

Cú pháp:

```
$dbh = new PDO(<driver>, <username>, <password>, <driver_options>);
```

Tham số:

<driver>: tên driver dùng để làm việc với CSDL

<username>: tên đăng nhập vào CSDL

<password>: mật khẩu đăng nhập vào CSDL

<driver\_options>: tham số tùy chọn, dùng để bổ sung thêm các trạng thái kết nối của PDO.



Hình 11.5: Mô hình dùng PDO kết nối đến CSDL từ PHP

Server hỗ trợ khá nhiều driver:

PDO\_DBLIB (Microsoft SQL Server)

PDO\_IBM (IBM DB2)

PDO\_MYSQL (MySQL)

PDO\_OCI (Oracle Call Interface)

PDO\_PGSQL (Postgre SQL)

Để xem server hỗ trợ những loại driver nào:

```
<?php
printf_r(PDO:getAvailableDrivers());
?>
```

Ví dụ: kết nối đến MySQL

```
$dbh = new PDO('mysql:host=localhost; dbname=test',
'root', 'sa');
```

### 3.2. Hủy kết nối

```
$dbh = null;
```

### 3.3. Xử lý lỗi kết nối

```
try
{
    $dbh = new PDO('mysql:host=localhost; dbname=test',
    'root', 'sa');
    //Các lệnh đọc, ghi dữ liệu
    $dbh = NULL;
}
catch (PDOException $e)
{
    printf 'Error: ' . $e->getMessage() . '<br />';
    die();
}
```

## Thực thi lệnh SQL

### 4.1. Các lệnh Insert, Update, Delete

Cú pháp

```
$dbh -> exec(<sqlstatement>);
```

<sqlstatement>: tham số bắt buộc, là câu lệnh được gửi đi.

Kết quả trả về: số dòng bị ảnh hưởng. Nếu không có dòng nào bị tác động thì kết quả là 0. Nếu không thực thi được thì kết quả là *false*.

### 4.2. Câu lệnh Select

Cú pháp

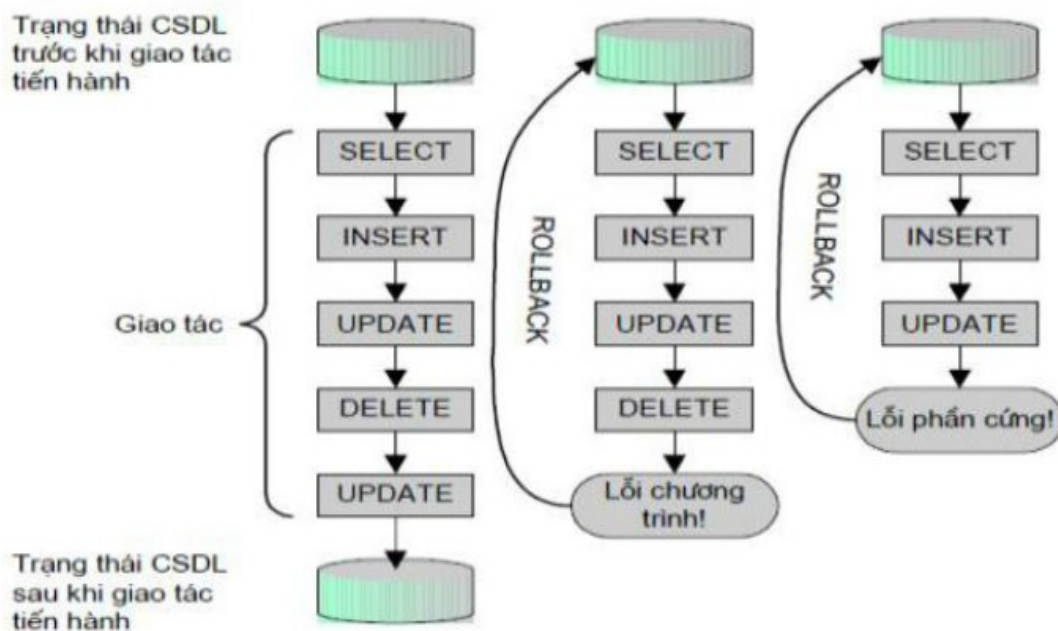
```
$dbh -> query(<sqlstatement>);
```

Tham số: <sqlstatement>: tham số bắt buộc, là câu lệnh được gửi đi.

## Transaction

Một giao tác (transaction) là một chuỗi hoặc nhiều câu lệnh SQL được kết hợp lại với nhau thành một khối công việc. Việc kết hợp các câu lệnh lại với nhau trong một giao tác nhằm đảm bảo tính toàn vẹn dữ liệu và khả năng phục hồi dữ liệu.

Trong một giao tác, các câu lệnh có thể độc lập với nhau nhưng tất cả các câu lệnh trong một giao tác đòi hỏi phải thực thi trọn vẹn hoặc không một câu lệnh nào được thực thi.



Hình 11.6: Mô hình Transaction

Transaction được định nghĩa dựa trên các câu lệnh xử lý sau đây:

**BEGIN TRANSACTION:** bắt đầu một transaction

**COMMIT:** đánh dấu kết thúc transaction. Câu lệnh này báo hiệu sự kết thúc thành công của một transaction. Sau câu lệnh này, một transaction mới sẽ được bắt đầu.

**ROLLBACK:** hủy transaction và đưa CSDL về trạng thái như trước khi transaction bắt đầu. Một transaction mới sẽ bắt đầu sau khi câu lệnh ROLLBACK được thực thi.

## PDO Statement

### 6.1. Prepared statement

Sử dụng prepared statement sẽ giúp tránh các SQL injection.

Cú pháp: `$statement = $dbh->prepare(<query>);`

Tham số:

<query>: tham số bắt buộc, là câu lệnh được gửi đi.

Ví dụ:

```
$statement = $dbh->prepare('SELECT * FROM sua');
```

## 6.2. Thực thi

Cú pháp: `$statement->execute();`

## 6.3. Duyệt dữ liệu

### Duyệt lần lượt từng mẫu tin

Cú pháp: `PDOStatement->fetch([int $fetch_style]);`

Dữ liệu được lấy thông qua phương thức `fetch()`. Trước khi gọi `fetch()`, cần phải cho PDO biết các dữ liệu muốn lấy thông qua tham số tùy chọn `$fetch_style`.



Hình 11.7: Mô hình tương quan giữa đối tượng và dãy mẫu tin thông qua PDO

Giá trị của `$fetch_style`:

`PDO::FETCH_NUM`: kết quả trả về là một mảng (có chỉ số) chứa giá trị của một dòng dữ liệu với mỗi phần tử là nội dung của một cột → truy cập bằng cách gọi từng phần tử của mảng `$row[0]`, `$row[1]`, `$row[2]`, ...

`PDO::FETCH_ASSOC`: kết quả trả về là một mảng (có chỉ số chuỗi) chứa giá trị của một dòng dữ liệu với mỗi phần tử là nội dung của một cột → truy cập bằng cách gọi từng phần tử của mảng: `$row["tên cột 1"]`, `$row["tên cột 2"]`, ...

`PDO::FETCH_BOTH`: kết quả trả về là một mảng (có chỉ số chuỗi và số) chứa giá trị của một dòng dữ liệu với mỗi phần tử là nội dung của một cột → truy cập bằng cách gọi từng phần tử của mảng theo `FETCH_NUM` hoặc `FETCH_ASSOC`.

`PDO::FETCH_OBJ`: kết quả trả về là một mẫu tin trong bộ các mẫu tin như là một đối tượng → truy cập bằng cách gọi từng thuộc tính của đối

tượng: \$<tên đối tượng>-><tên cột 1>, \$<tên đối tượng>-><tên cột 2>,  
...

Ví dụ:

```
$sql= 'SELECT * FROM sua';  
$statement = $dbh->prepare($sql);  
$ds = $statement->execute();  
foreach ($ds as $t){  
echo $t[0];  
}
```

### **Duyệt danh sách các mẫu tin**

```
PDOStatement->fetchAll([int $fetch_style, [$column_index]])
```

Tham số:

\$fetch\_style:

PDO::FETCH\_BOTH: giá trị mặc định. Kết quả trả về là mảng hai chiều, mỗi phần tử là mảng 1 chiều (có chỉ số chuỗi và số) chứa giá trị của một dòng dữ liệu với mỗi phần tử là nội dung của một cột.

PDO::FETCH\_COLUMN: kết quả trả về là mảng 1 chiều chứa tất cả các giá trị của một cột được chỉ định.

\$column\_index: kết quả trả về là một cột được chỉ định khi \$fetch\_style có giá trị là PDO::FETCH\_COLUMN.

## 6.4. Truyền tham số trong câu truy vấn

Cú pháp:

```
PDOStatement->bindParam(mixed $parameter $mixed &$variable  
[, int $data_type, [,int $length [, $mixed  
$driver_options]]]);
```

Tham số:

\$parameter: tên tham số

\$variable: giá trị tham số

\$data\_type: tùy chọn kiểu dữ liệu của tham số, có cú pháp  
PDO::PARAMS\_\*

\$length: tùy chọn, chiều dài của kiểu dữ liệu

\$driver\_options: tùy chọn driver

## 6.5. Đếm số lượng mẫu tin

Cú pháp: `PDOStatement->rowCount ( ) ;`

Trả về số lượng mẫu tin bị tác động bởi câu lệnh insert, update, delete

Ví dụ: đếm số lượng mẫu tin bị xóa

## **CÂU HỎI, BÀI TẬP**

11.1. Tạo các trang xem, xóa, sửa dữ liệu trên các table trong CSDL.

## **GỢI Ý, TRẢ LỜI**

11.1. Kết hợp PDO với các lớp đối tượng được xây dựng ở bài trước.



## BÀI 12.

# CÁC ĐỐI TƯỢNG QUẢN LÝ TRONG PHP

### **Giới thiệu:**

Bài này cung cấp cho HSSV kiến thức và kỹ năng sử dụng các đối tượng quản lý của PHP để truy xuất thông tin người truy cập.

### **Mục tiêu:**

Biết công dụng của các đối tượng `$_SESSION`, `$_COOKIE`.

Tạo trang đăng nhập, kiểm tra đăng nhập

Tạo chức năng giỏ hàng, đặt hàng trực tuyến

Cẩn thận không lưu mật khẩu trong `$_SESSION` hoặc `$_COOKIE`

### **1. Session**

#### 1.1. Khái niệm

Cho phép lưu trữ thông tin người dùng trên trình duyệt (tên người dùng, giỏ hàng, ...) trong suốt quá trình truy cập ứng dụng web.

Thông tin của session chỉ tạm thời và thông tin này sẽ bị xóa sau khi người truy cập rời khỏi ứng dụng web.

Nếu cần, phải lưu trữ thông tin trong CSDL.

#### 1.2. Cách thức hoạt động

Làm việc bằng cách tạo ra một địa chỉ duy nhất (UID) cho mỗi người truy cập.

UID có giá trị là một dãy số ngẫu nhiên.

UID có thể được lưu trong COOKIE hoặc truyền lên URL.

Ngoài UID, có thể khai báo, khởi tạo và sử dụng một biến session khác, tất cả các session này có giá trị cho mỗi người sử dụng khi họ truy cập đến ứng dụng web.

#### 1.3. Khởi động session

Trước khi lưu trữ thông tin người dùng vào session, cần khởi động session.

Chú ý: hàm khởi động session phải đặt trước thẻ <html>

Cú pháp: `session_start();`

#### 1.4. Đăng ký session

Sử dụng biến `$_SESSION` nhận và lưu trữ giá trị của biến session.

Cú pháp: `$_SESSION['tên biến session'] = <giá trị>;`

#### 1.5 Sử dụng session

Khi muốn sử dụng các biến session hoặc giá trị lưu trong biến session đã đăng ký → dùng biến `$_SESSION`.

Cú pháp: `$<biến> = $_SESSION['tên biến'];`

#### 1.6. Hủy toàn bộ các biến session

Khi không cần dùng đến các biến session nữa thì có thể hủy bỏ toàn bộ các biến session đã đăng ký bằng hàm `session_destroy()`.

Cú pháp: `session_destroy();`

#### 1.7. Hủy một biến session

Khi không cần dùng đến biến session nào thì có thể dùng hàm `unset()` để hủy biến session đó.

Cú pháp: `unset($_SESSION['tên biến']);`

## **COOKIE**

### 2.1. Khái niệm

Sử dụng để xác định thông tin người dùng.

Là một tập tin nhỏ được server lưu trữ xuống từng máy tính của người dùng.

Mỗi khi máy tính này yêu cầu một trang tới trình duyệt, nó cũng sẽ gửi theo cookie.

Với PHP, ta có thể tạo ra và sử dụng giá trị của biến cookie.

### 2.2. Khai báo cookie

Chú ý: khai báo cookie ở phía trên thẻ <html>.

Cú pháp: `setcookie (name, value, expire[, path, domain]);`

name: tên biến cookie

value: giá trị

expire: thời gian giới hạn cho cookie – đơn vị tính là giây. Nếu thời gian này không được thiết lập trong hàm setcookie(), biến cookie này sẽ còn hiệu lực cho đến khi người dùng xóa tập tin cookie.

path: đường dẫn

domain: tên miền của website

### 2.3. Sử dụng cookie

Dùng biến `$_COOKIE` để đọc giá trị biến cookie.

Cú pháp: `$<biến> = $_COOKIE['tên biến cookie'];`

### 2.4. Hủy cookie

Khi muốn hủy một biến cookie thì cần kiểm tra lại thời gian giới hạn dành cho biến cookie này.

Sử dụng hàm `setcookie()` để hủy bằng cách đặt giá trị của biến cookie bằng “” và gởi gian = `-<thời gian giới hạn>`.

Cú pháp: `setcookie(name, "", time() - <thời gian giới hạn>)`

## CÂU HỎI, BÀI TẬP

12.1. Tạo thư mục administrator trong website, tạo các trang đăng nhập, quản lý nội dung CSDL. Đảm bảo phải đăng nhập thành công mới được quản lý nội dung.

Cho phép người dùng đăng nhập và tạo giỏ hàng.

## GỢI Ý, ĐÁP ÁN

Sử dụng `$_SESSION` để lưu thông tin đăng nhập.

Sử dụng `$_COOKIE` hoặc `$_SESSION` lưu tên đăng nhập và quản lý giỏ hàng của khách truy cập.

## BÀI 13.

# SỬ DỤNG JQUERY

### Giới thiệu:

Bài này cung cấp cho HSSV kiến thức và kỹ năng sử dụng thư viện jQuery để tạo hiệu ứng, tương tác người dùng trên trình duyệt.

### Mục tiêu:

Biết khái niệm, hoạt động và lợi ích của jQuery

Khai báo thư viện jQuery

Sử dụng thư viện jQuery để trang trí, tạo hiệu ứng cho trang web

Tránh sử dụng nhiều hiệu ứng làm trình duyệt xử lý chậm

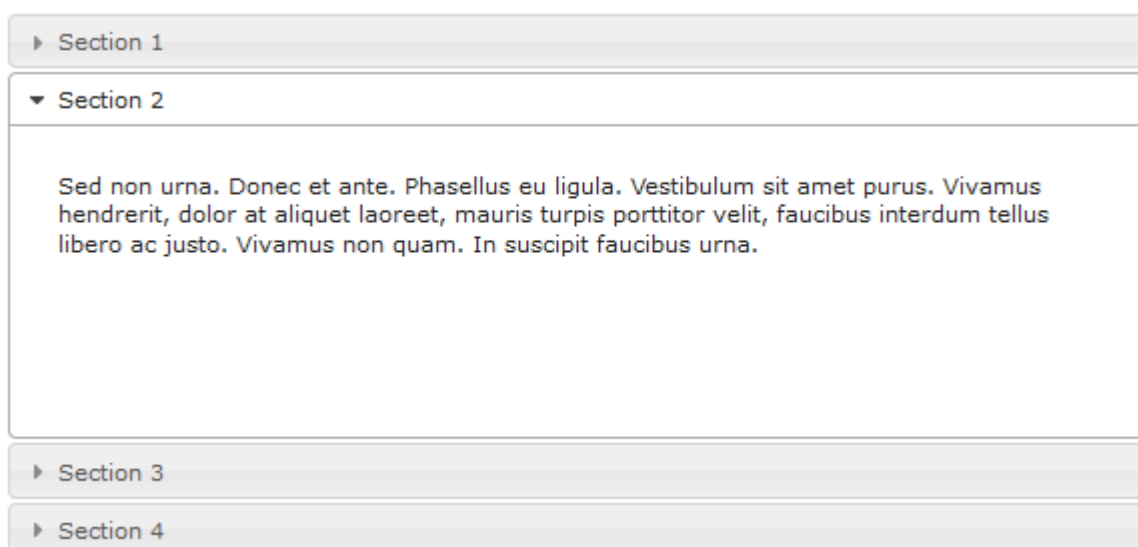
Cẩn thận sử dụng version mới nhất của jQuery

Đảm bảo an toàn cho tài khoản facebook

### 1. Cài đặt jQuery plugin

Tải thư viện jquery và jquery-ui tại [jquery.com](http://jquery.com) và lưu vào thư mục Scripts của website

### Đóng/mở nội dung trong không gian giới hạn



```
<head>
  <!-- -->
  <script>
    $(function() {
```

```

        $( "#accordion" ).accordion();
    });
</script>
</head>
<body>
<div id="accordion">
    <h3>Section 1</h3>
    <div><p>Nội dung Section 1</p></div>
    <h3>Section 2</h3>
    <div><p>Nội dung Section 2</p></div>
</div>
</body>

```

### Tạo hộp nhập tự động gợi ý nội dung

```

<head>
    <!-- -->
    <script>
        $(function() {
            var availableTags = [
                "ActionScript",
                "AppleScript",
                "Asp",
                "BASIC",
                "C",
                "C++",
                "Java",
                "JavaScript"
            ];
            $( "#tags" ).autocomplete({
                source: availableTags
            });
        });
    </script>
</head>
<body>
    <label for="tags">Tags: </label>
    <input id="tags">
</body>

```

## Tạo khả năng kéo/thả

```
<head>
  <!-- -->
  <style>
    #draggable { width: 150px; height: 150px; padding:
0.5em; }
  </style>
  <script>
    $(function() {
      $( "#draggable" ).draggable();
    });
  </script>
</head>
<body>
  <div id="draggable" class="ui-widget-content">
    <p>Drag me around</p>
  </div>
</body>
```

## Tạo khả năng kéo/thả để chọn

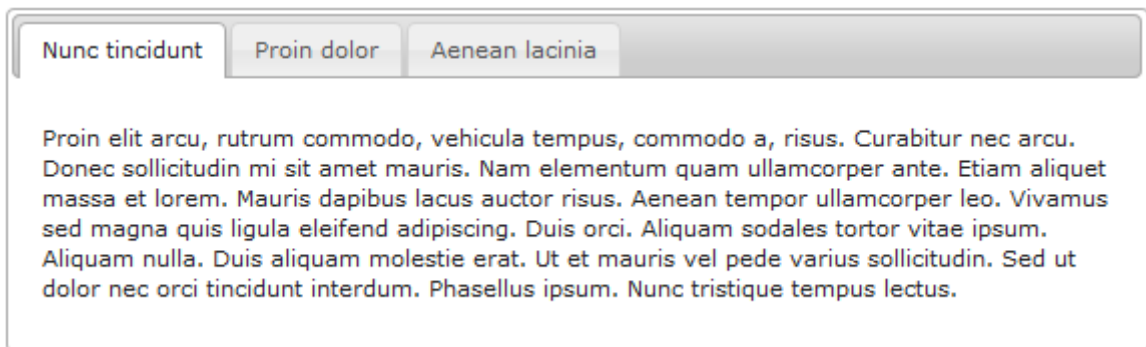
```
<head>
  <!-- -->
  <style>
    #draggable { width: 100px; height: 100px; padding:
0.5em; float: left; margin: 10px 10px 10px 0; }
    #droppable { width: 150px; height: 150px; padding:
0.5em; float: left; margin: 10px; }
  </style>
  <script>
    $(function() {
      $( "#draggable" ).draggable();
      $( "#droppable" ).droppable({
        drop: function( event, ui ) {
          $( this )
            .addClass( "ui-state-highlight" )
            .find( "p" )
              .html( "Dropped!" );
        }
      });
    });
```

```

    });
  </script>
</head>
<body>
  <div id="draggable" class="ui-widget-content">
    <p>Drag me to my target</p>
  </div>
  <div id="droppable" class="ui-widget-header">
    <p>Drop here</p>
  </div>
</div>
</body>

```

## Tạo giao diện Tabs



```

<head>
<!-- -->
  <script>
    $(function() {
      $( "#tabs" ).tabs();
    });
  </script>
</head>
<body>
<div id="tabs">
  <ul>
    <li><a href="#tabs-1">Nunc tincidunt</a></li>
    <li><a href="#tabs-2">Proin dolor</a></li>
    <li><a href="#tabs-3">Aenean lacinia</a></li>
  </ul>
  <div id="tabs-1">
    <p>Nội dung tab1</p>
  </div>

```

```

<div id="tabs-2">
  <p>Nội dung tab2</p>
</div>
<div id="tabs-3">
  <p>Nội dung tab3</p>
</div>
</div>
</body>

```

## Tạo Model Popup

```

<head>
<!-- -->
  <script>
    $(function() {
      $( "#dialog" ).dialog();
    });
  </script>
</head>
<body>
<div id="dialog" title="Basic dialog">
  <p>Nội dung hộp thoại Popup</p>
</div>
</body>

```

## Tạo Datepicker



```

<head>
<!-- -->
  <script>
    $(function() {
      $( "#datepicker" ).datepicker();
    });
  </script>

```



```
</head>
<body>
<p>Date: <input type="text" id="datepicker"></p>
</body>
```

## Tạo nút Fanpage

Đăng nhập Facebook

Tạo Fan Page

Vào <https://developers.facebook.com/docs/plugins/page-plugin>

Khai báo các tùy chọn cho Fan Page

Get Code

Click chuột lên

Sao chép và đặt đoạn code sau ngay sau thẻ mở <body>

```
<div id="fb-root"></div>
<script>(function(d, s, id) {
  var js, fjs = d.getElementsByTagName(s)[0];
  if (d.getElementById(id)) return;
  js = d.createElement(s); js.id = id;
  js.src = "//connect.facebook.net/en_US/sdk.js#xfbml=1&version=v2.4";
  fjs.parentNode.insertBefore(js, fjs);
})(document, 'script', 'facebook-jssdk');
```

Đặt đoạn code sau tại vị trí muốn hiển thị thông tin Fanpage

```
<div class="fb-page" data-href="https://www.facebook.com/facebook" data-small-
header="false" data-adapt-container-width="true" data-hide-cover="false" data-show-
facepile="true" data-show-posts="true"><div class="fb-xfbml-parse-ignore">
<blockquote cite="https://www.facebook.com/facebook"><a
href="https://www.facebook.com/facebook">Facebook</a></blockquote></div></div>
```

## Tạo chức năng Comment

Tạo Application trên Facebook tại <http://developers.facebook.com/setup/>

Create application

Ghi lại App ID và App Secret sau khi click

Thực hiện các bước tiếp theo để nhận được đoạn code XFBML

Đặt đoạn code XFBML vào vị trí cần hiển thị nội dung Comment facebook

## TÀI LIỆU THAM KHẢO

<https://facebook.com>

<http://jquery.com>

<http://mysql.com>

<http://php.net>