



TRƯỜNG TRUNG CẤP NGHỀ KỸ THUẬT CÔNG NGHỆ HÙNG VƯƠNG  
**KHOA: Công Nghệ Thông Tin**

**MÔN : Lập trình Web PHP & MySQL**

**HIỆU TRƯỞNG TRƯỞNG KHOA GIÁO VIÊN BIÊN SỌAN**

**TÀI LIỆU LƯU HÀNH NỘI BỘ**

**Tháng 06 năm 2012**

*Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL*

**MỤC LỤC**

**CHƯƠNG 1**

- 1.1 Giới thiệu PHP.....
- 1.2. Cài đặt và cấu hình Wamp.....
- 1.3 Cài đặt và cấu hình Xampp.....

1.4	Cài đặt và cấu hình PHP Designer IDE	
1.5	Cài đặt và cấu hình NetBeans IDE	
<b>CHƯƠNG 2</b>		
2.1	PHP Script	
2.2	Ghi chú trong PHP	
2.3	In nội dung bằng PHP	
2.4	Kiểu dữ liệu và biến	
2.5	Khai báo và sử dụng hằng	
2.6	Dữ liệu mảng	
2.7	Chuyển đổi kiểu dữ liệu	
2.8	Phép toán và phát biểu có điều kiện trong php	
2.9	Xử lý chuỗi	
2.10	Làm việc với mảng dữ liệu	
2.11	Kiểu DateTime	
<b>CHƯƠNG 3</b>		
3.1	Giới thiệu toán tử	
	3.2 Phát biểu có điều kiện If	
	3.3 Switch	
	3.4 While	
	3.5 For	
	3.6 Do-While	
<b>CHƯƠNG 4</b>		
4.1	Biến form submit với phương thức GET	
	4.2 Form	
<b>CHƯƠNG 5</b>		
5.1	Xử lý dữ liệu nhập trên form và thông báo lỗi với javascript	
5.2	Regular Expression	
<b>CHƯƠNG 6</b>		
6.1	Session	
	6.2 Cookie	
<b>CHƯƠNG 7</b>		
7.1	Các hàm xử lý lỗi	
7.2	Làm việc mảng dữ liệu	
	7.3 Kiểu DateTime	
<b>CHƯƠNG 8</b>		
8.1	Khai báo hàm trong PHP	
	8.2 Thống nhất kích thước của mọi trang PHP	
<b>CHƯƠNG 9</b>		
9.1	Hướng đối tượng trong PHP	
<b>CHƯƠNG 10</b>		
10.1	Giới thiệu CSDL MySQL	
10.2	Cài đặt MySQL	
10.3	Tạo CSDL người dung	
	10.4 Kiểu dữ liệu trong MySQL	

10.5	Câu	lệnh
SQL.....		10.6 Các hàm
thông dụng trong MySQL.....		10.7 Các hàm
về toán học .....		10.8 Phát
biểu SQL dạng Select .....		10.9
Phát biểu SQL dạng Join .....		
10.10 Tạo kịch bản SQL- SQL Script.....		

## **CHƯƠNG 11**

11.1	Stored	procedure	trong	MySQL
.....				11.2 Con trỏ trong stored procedure
.....				11.3
Trigger.....				

## **CHƯƠNG 12**

12.1 Kết nối CSDL
-------------------

.....	<b>CHƯƠNG 13</b>
-------	------------------

13.1 Liệt kê dữ liệu
----------------------

.....	<b>CHƯƠNG 14</b>
-------	------------------

<a href="#">14.1 Mô hình</a>
------------------------------

<a href="#">MVC.....</a>	<a href="#">14.2</a>
--------------------------	----------------------

<a href="#">Friendly URL.....</a>	
-----------------------------------	--

## **Tài liệu tham khảo**

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 3

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

## LỜI NÓI ĐẦU

### **1. Lời giới thiệu tài liệu giảng dạy:**

- Giáo trình "**Lập trình web PHP & MySQL**" là tài liệu dùng trong giảng dạy và học tập của học viên đang theo học hệ trung cấp nghề Thiết kế web trường **TCN KTCN Hùng Vương** tại địa chỉ 161 - 165 Nguyễn Chí Thanh, P12, Q5, TpHCM.

- Ngày nay các ứng dụng website rất phổ biến và áp dụng rộng rãi trong rất nhiều lĩnh vực. Hầu hết các tổ chức hay doanh nghiệp đều xây dựng website riêng nhằm mục tiêu giới thiệu thương hiệu các hoạt động và hình thức kinh doanh trên website, một số doanh nghiệp xây dựng loại hình thương mại điện tử trên website. Trong lĩnh vực thiết kế website thì có nhiều ngôn ngữ hỗ trợ như Java, DotNet, PHP,... Trong đó PHP và hệ quản trị cơ sở dữ liệu MySQL được nhiều lập trình viên lựa chọn vì tính đơn giản và phổ biến cao. Rất nhiều hệ thống mã nguồn mở nổi tiếng và thông dụng được xây dựng trên nền tảng PHP và MySQL như : Joomla, Moodle, NukeViet, Magento, OpenCart, Wordpress.....Điều đó chứng tỏ PHP đang trở thành 1 ngôn ngữ lập trình web thông dụng và được sử dụng nhiều trên thế giới hiện nay.

- Giáo trình được chia làm 13 chương, đi từ cơ bản đến chuyên sâu cung cấp các kiến thức nền tảng để xây dựng lập trình hoàn chỉnh 1 website sử dụng ngôn ngữ lập trình nguồn mở PHP và cơ sở dữ liệu MySQL.

## **2. Mục tiêu môn học**

- Hiểu rõ các kỹ thuật thiết kế giao diện và lập trình website hoàn chỉnh - Cung cấp các kiến thức nền tảng về các kỹ thuật phát triển web hiện đại - Hình thành và phát triển tư duy lập trình và tư duy phát triển ứng dụng web - Xây dựng được hoàn chỉnh một ứng dụng web tuân thủ đầy đủ các bước trong quy trình phát triển phần mềm với 3 dạng web phổ biến:

- + Website chuyên về trình bày thông tin – tin tức
- + Website chuyên về bán hàng – thương mại điện tử
- + Website chuyên về quản trị nội dung CMS

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 4

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

## **CHƯƠNG 1**

### **1.1. Giới thiệu PHP**

- PHP viết tắt của chữ Personal Home Page ra đời năm 1994 do phát minh của Rasmus Lerdorf, và nó tiếp tục được phát triển bởi nhiều cá nhân và tập thể khác, do đó PHP được xem như một sản phẩm của mã nguồn mở.
- PHP là kịch bản trình chủ (*server script*) chạy trên phía server (*server side*) như cách server script khác (*asp, jsp, cold fusion*).
- PHP là kịch bản cho phép chúng ta xây dựng ứng dụng web trên mạng internet hay intranet tương tác với mọi cơ sở dữ liệu như mySQL, PostgreSQL, Oracle, SQL Server và Access.
- Lưu ý rằng, từ phiên bản 4.0 trở về sau mới hỗ trợ session, ngoài ra PHP cũng như Perl là kịch bản xử lý chuỗi rất mạnh chính vì vậy bạn có thể sử dụng PHP trong những có yêu cầu về xử lý chuỗi.

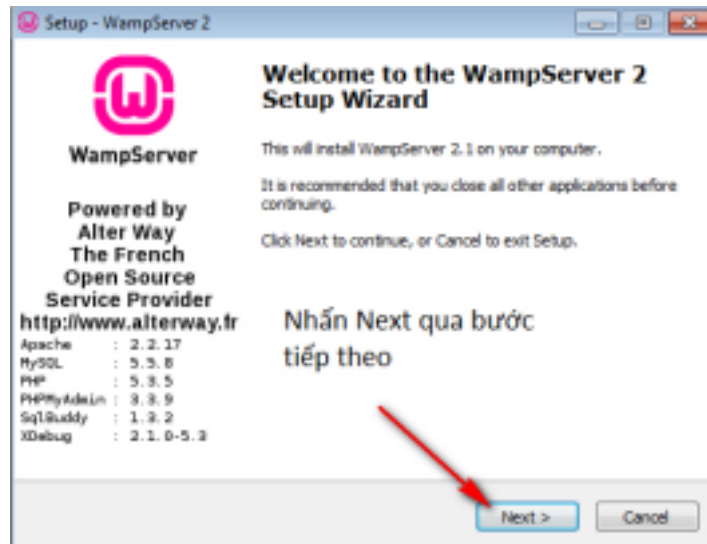
## 1.2. Cài đặt và tìm hiểu Wamp

- Cài đặt: Các bước cài đặt wampserver như sau:

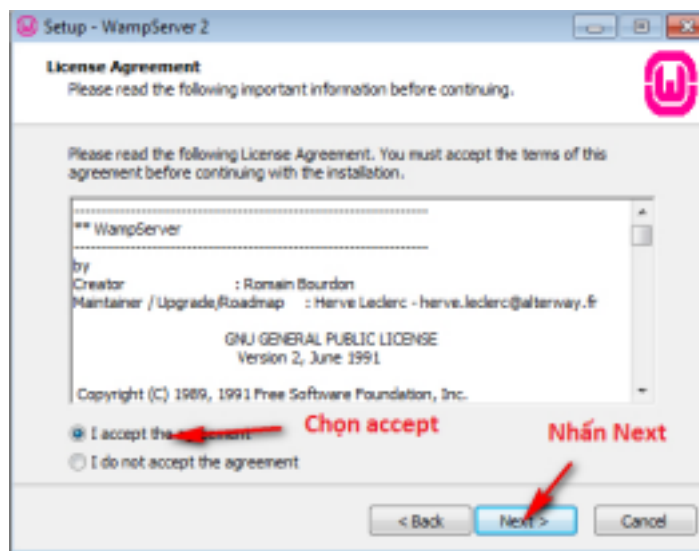
Bước 1: download chương trình wampserver 2.0 theo địa chỉ sau :

<http://www.wampserver.com/en/download.php>

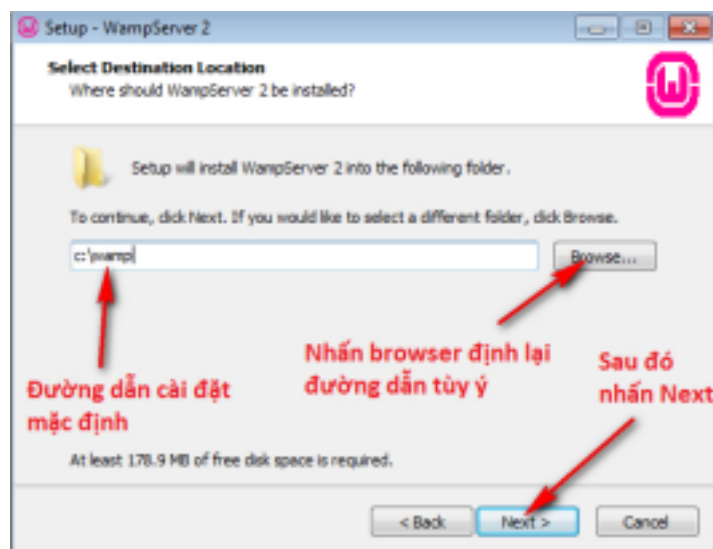
Bước 2: làm theo như hình chỉ dẫn



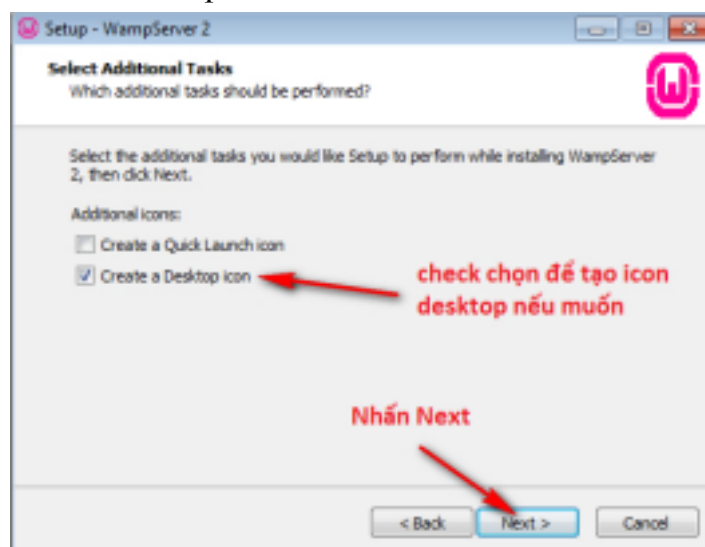
- Nhấn *Next* để tiếp tục.



- Click chọn mục ***I accept the agreement*** để đồng ý việc cài đặt
- Nhấn ***Next*** để tiếp tục.

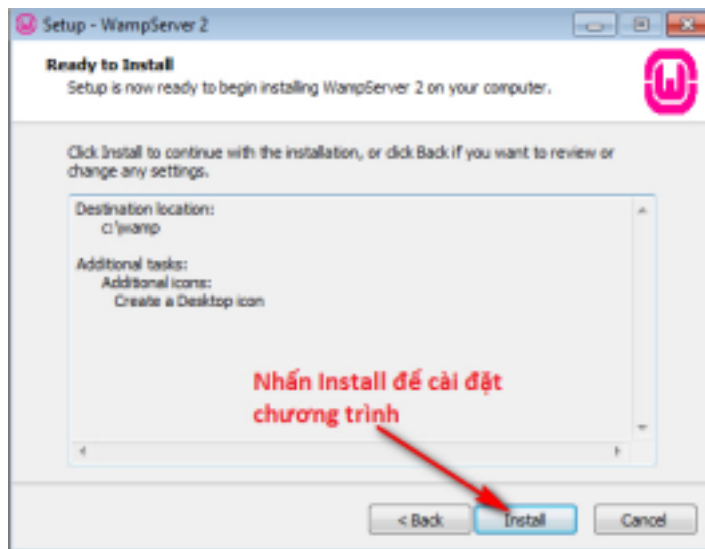


- Đường dẫn cài đặt mặc định sẽ là ***c:\wamp***.
- Nếu muốn cài đặt tùy ý thì chọn ***Browse..*** để chọn nơi cài đặt khác.
- Sau đó nhấn ***Next*** để tiếp tục.

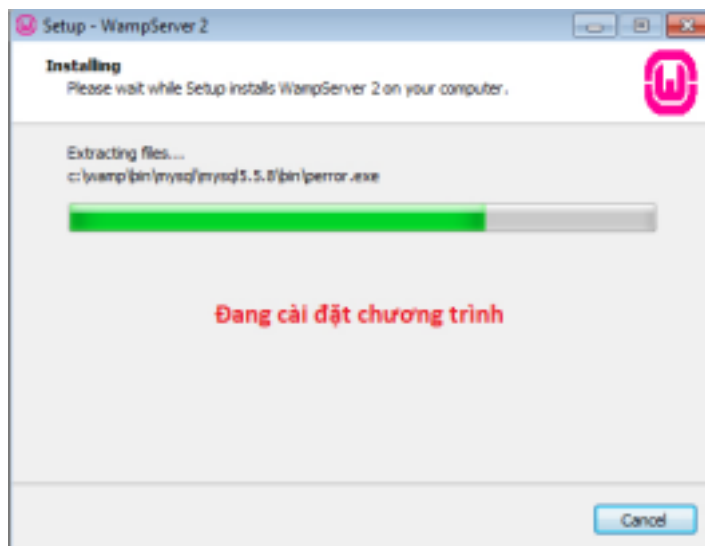


- Check vào ***Create a Desktop icon*** để tạo icon ngoài màn hình Desktop

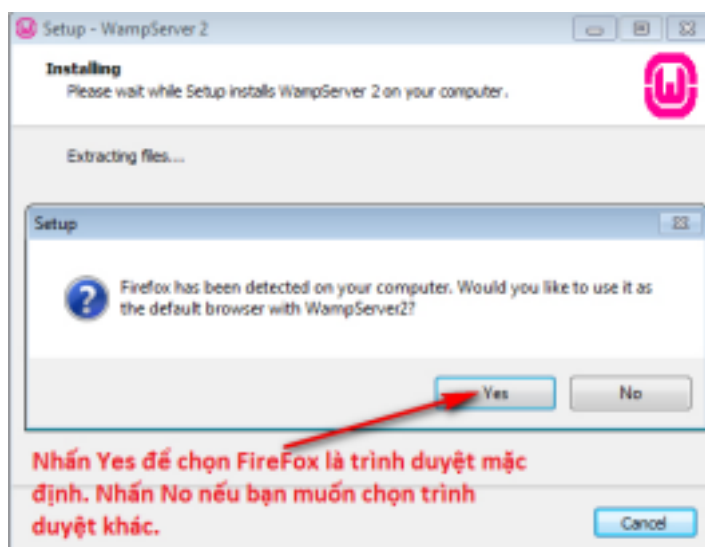
tiếp tục.



- Nhấn **Install** để bắt đầu quá trình cài đặt.



- Đang cài đặt **WampServer**



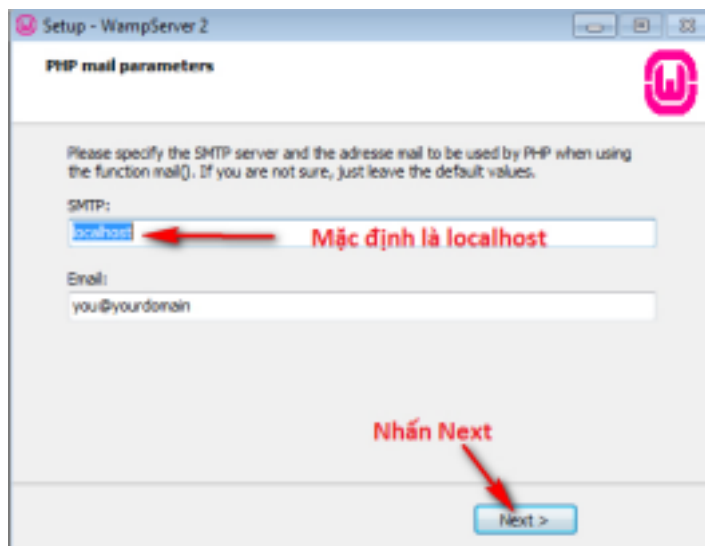
BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 7

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

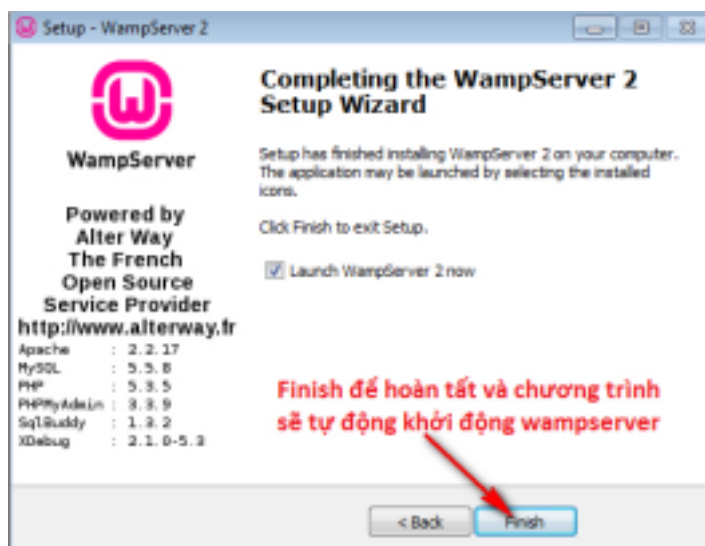
- Sau khi cài đặt xong. Chương trình sẽ hiện thị thông báo yêu cầu chọn trình duyệt để sử dụng. Ở đây mặc định sẽ là **firefox**, nếu chọn mặc định thì nhấn **Yes**, nếu



không thì nhấn **No** và chọn đến trình duyệt tùy ý.



- SMTP mặc định là **localhost**
- Email mặc định là **you@yourdomain** (có thể thay đổi tùy ý)
- Nhấn **Next** để tiếp tục.



- Nhấn **Finish** để hoàn thành cài đặt.

**Biểu tượng sau khi  
wampserver khởi động**



- Biểu tượng của WampServer sau khi khởi động chương trình.



- **Localhost** là server chạy trực tiếp chính máy chúng ta.
- **PhpMyAdmin** là cơ sở dữ liệu.



- Giao diện khi chạy **localhost**.



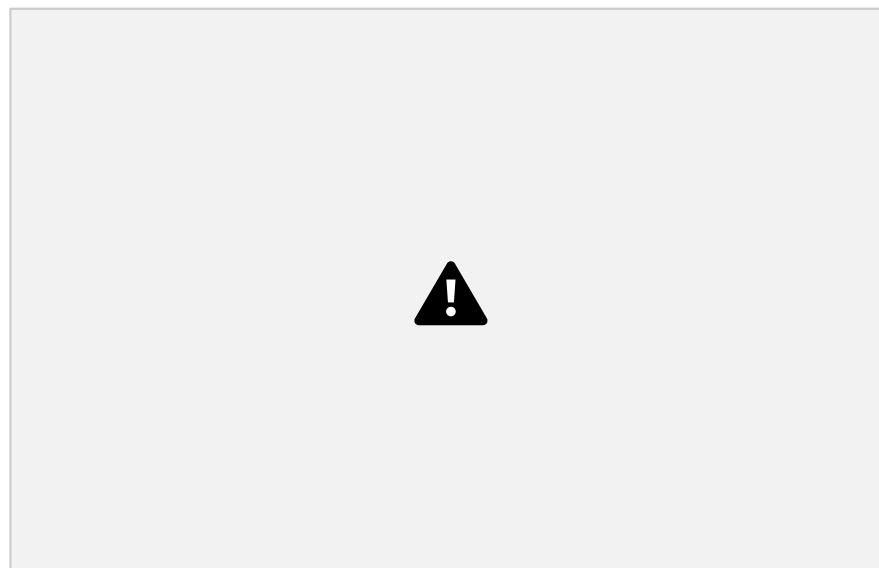
- Giao diện cơ sở dữ liệu **phpMyAdmin**
- Bên trái là danh sách tên cơ sở dữ liệu có trong **phpMyAdmin**

- Bên phải là thông tin về MySQL, WebServer và version phpMyAdmin và các chức năng của phpMyAdmin

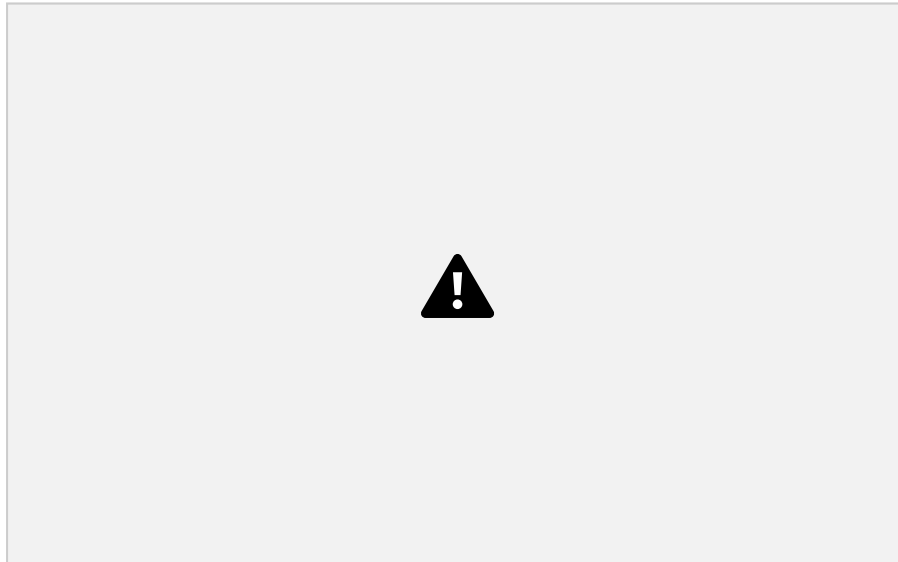
## Tìm hiểu:



- Bên trái là danh sách tên cơ sở dữ liệu có trong phpMyAdmin
- Bên phải là thông tin về MySQL, WebServer và version phpMyAdmin và các chức năng của phpMyAdmin
- Phần trên là menu chức năng như import, export hay tạo mới database



- Thẻ SQL dùng để viết query truy vấn cơ sở dữ liệu.



- Các thẻ export và import như đã nói ở phần trên là dùng để export và import database.

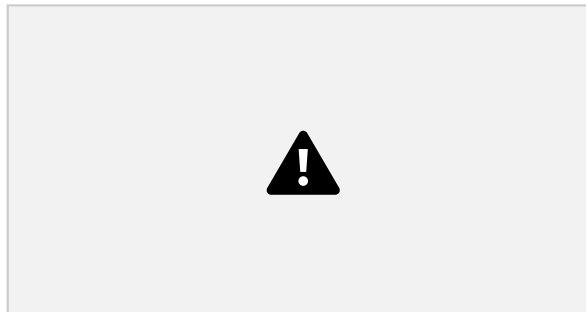
### **1.3. Cài đặt và tìm hiểu Xampp**

#### **Cài đặt:**

Bước 1: download chương trình xampp về theo địa chỉ sau:

<http://www.apachefriends.org/en/xampp-windows.html#641>

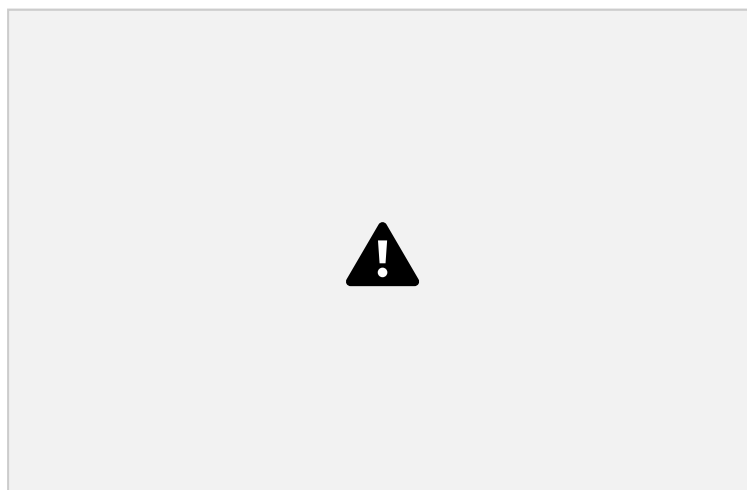
Bước 2: thực hiện theo hình chỉ dẫn sau:



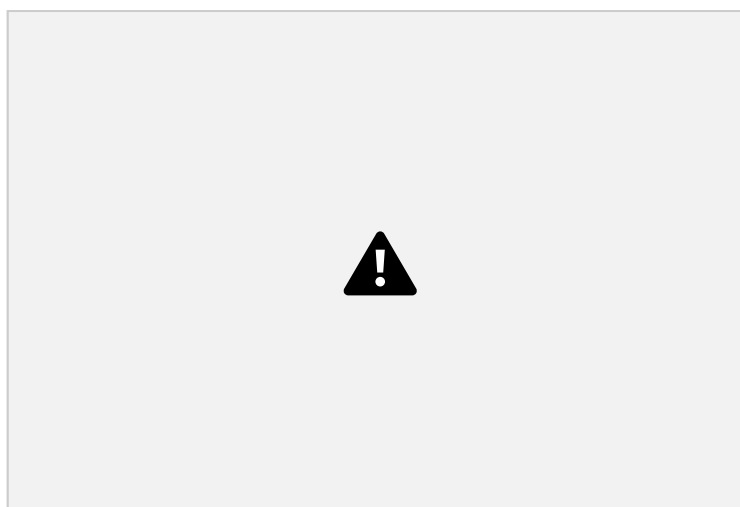
- Chạy file setup vừa download về.
- Nhấn OK để bắt đầu chương trình cài đặt.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 11

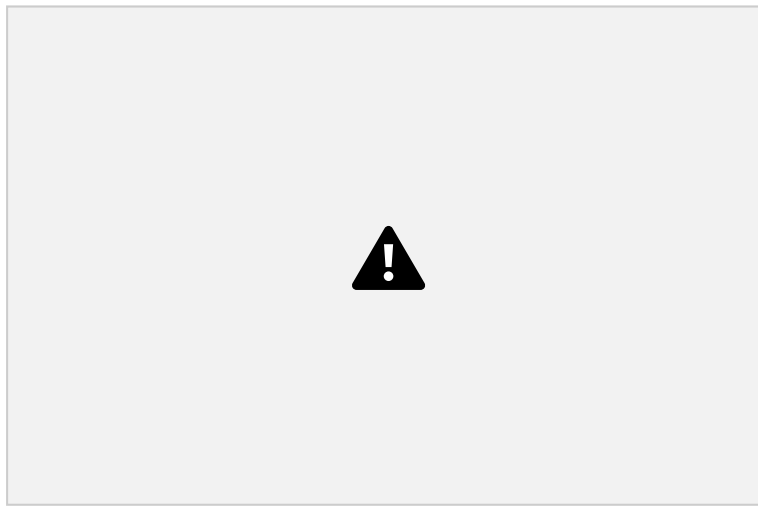
***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



- Nhấn ***Next*** để tiếp tục.



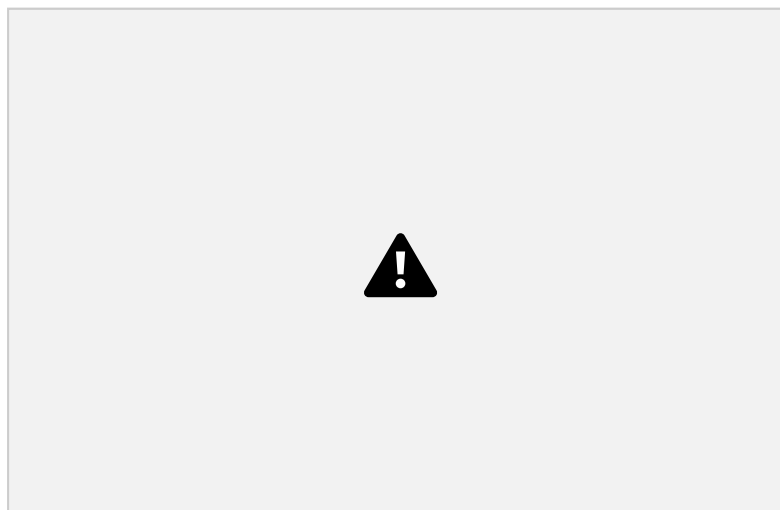
- Mặc định chương trình sẽ cài đặt mặc định là ổ C
- Nếu muốn cài đặt tùy ý thì chọn ***Browse*** để chọn đường dẫn khác tùy ý.



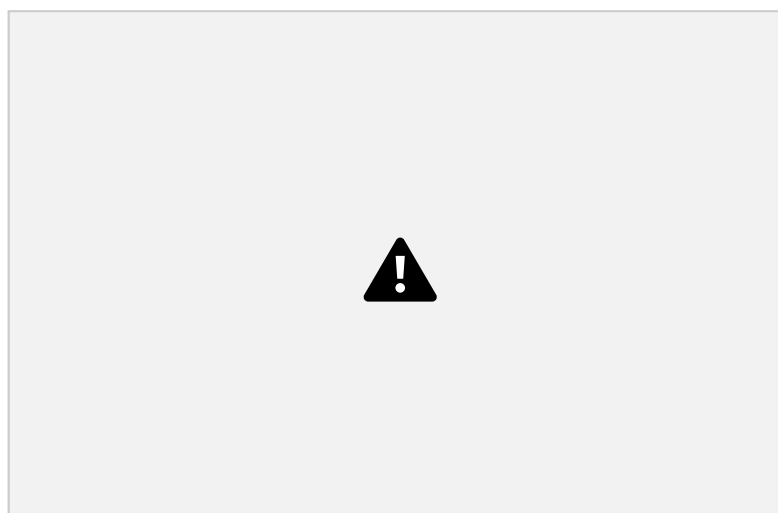
-Nhấn ***Install*** để tiến hành quá trình cài đặt ***xampp***

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 12

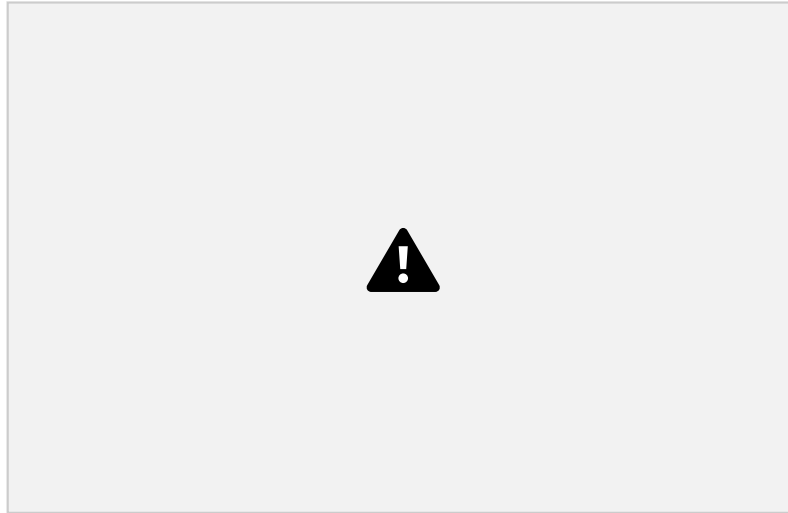
***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



- Đang trong quá trình cài đặt ***xampp***



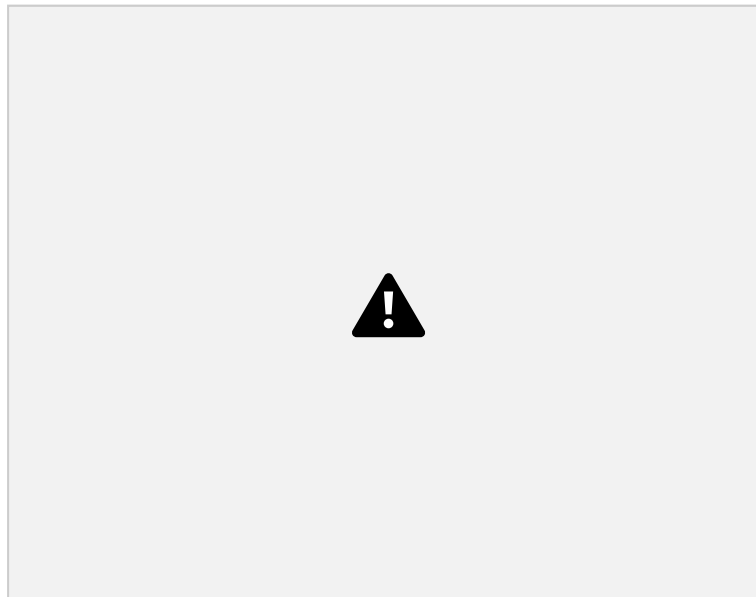
- Nhấn ***Finish*** để hoàn thành cài đặt ***xampp***



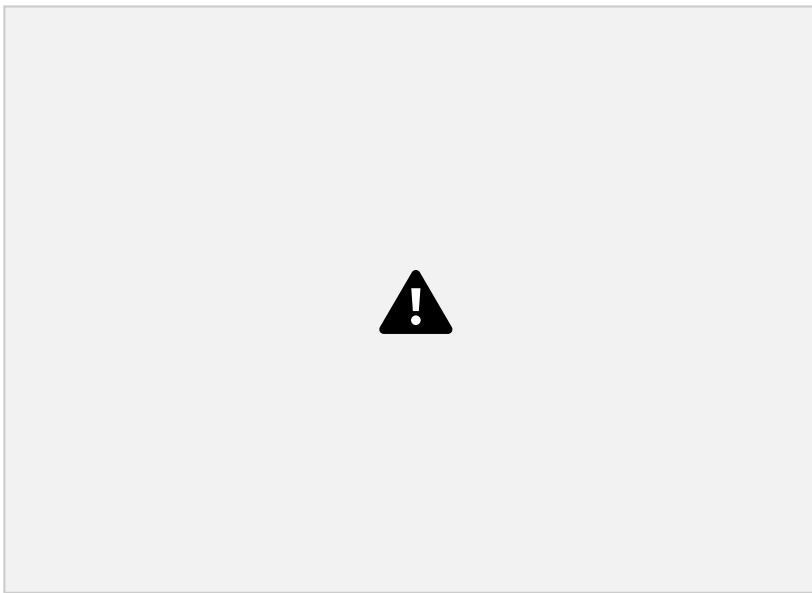
- Chương trình hiện thị thông báo có khởi động *xampp* sau khi cài đặt không. Nhấn **Yes** để khởi động chương trình.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 13

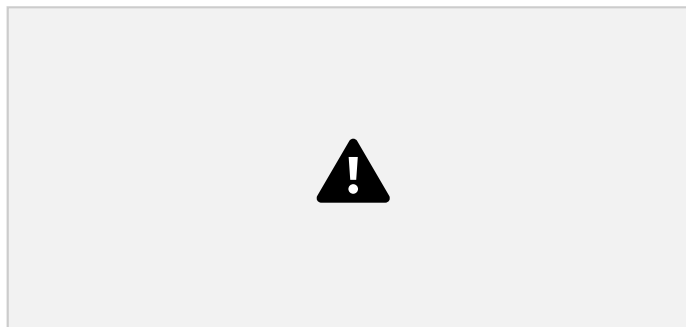
***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



- Hiện tại server và CSDL MySQL chưa được khởi động. - Nhấn **Start** để khởi động server localhost và cơ sở dữ liệu MySQL



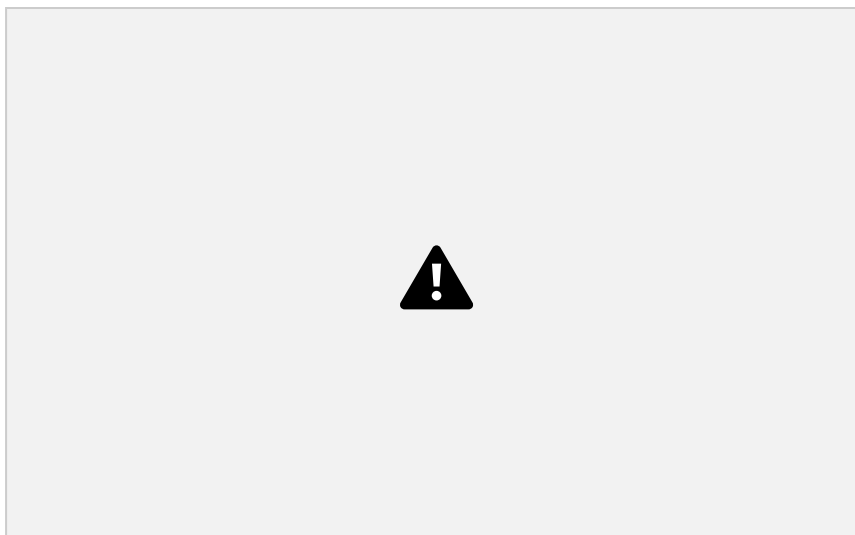
- Server và CSDL MySQL đã được khởi động.



- Để chạy localhost nhấn ***Admin*** như hình vẽ.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 14

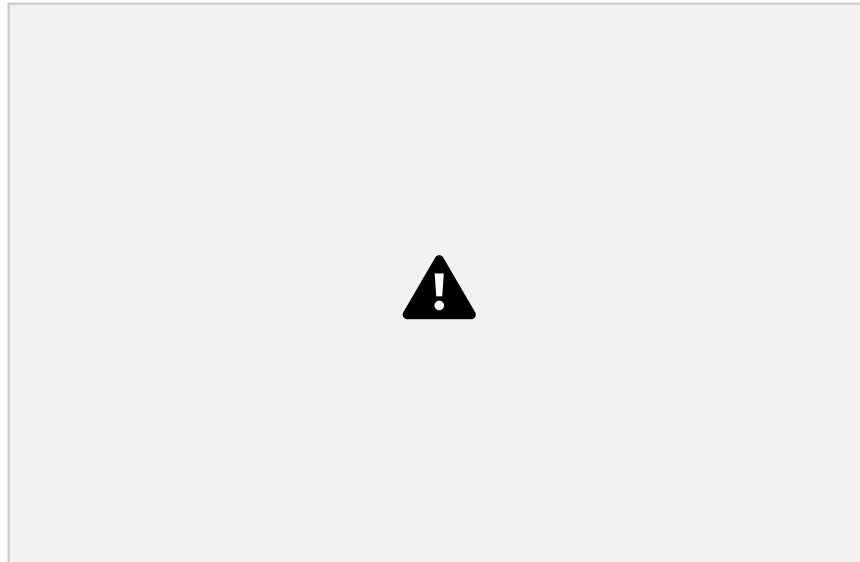
***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



- Nếu muốn vào hệ thống CSDL MySQL thì chọn ***phpMyAdmin*** bên trái như hình



vẽ.



**Tìm hiểu:**

- Bởi vì hệ quản trị csdl của *Xampp* cũng tương tự giống của *WampServer* nên chỉ cần xem lại hệ csdl wampserver đã viết ở trên.

**1.4. Cài đặt và cấu hình PHP Designer IDE**

Bước 1: download phần mềm PHP Designer theo địa chỉ sau:

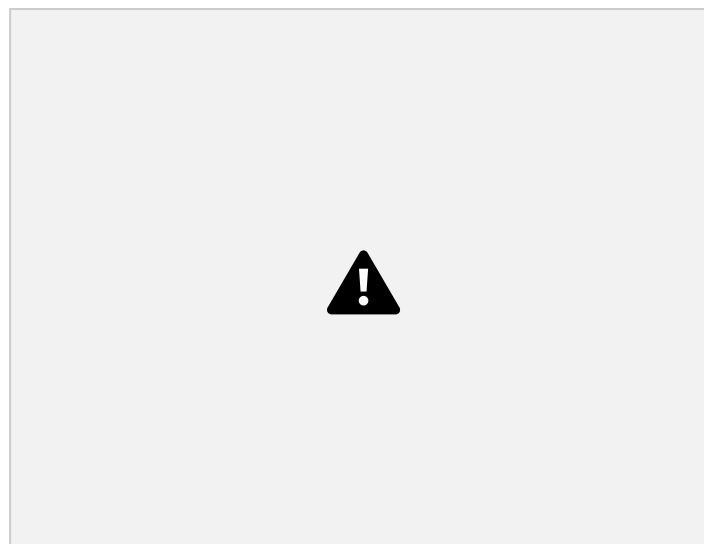
[http://www.mpsoftwareweb.com/phpdesigner\\_7\\_2\\_5\\_setup.exe](http://www.mpsoftwareweb.com/phpdesigner_7_2_5_setup.exe)

Còn đây là link mediafire <http://www.mediafire.com/?aasic74i214eq6t>

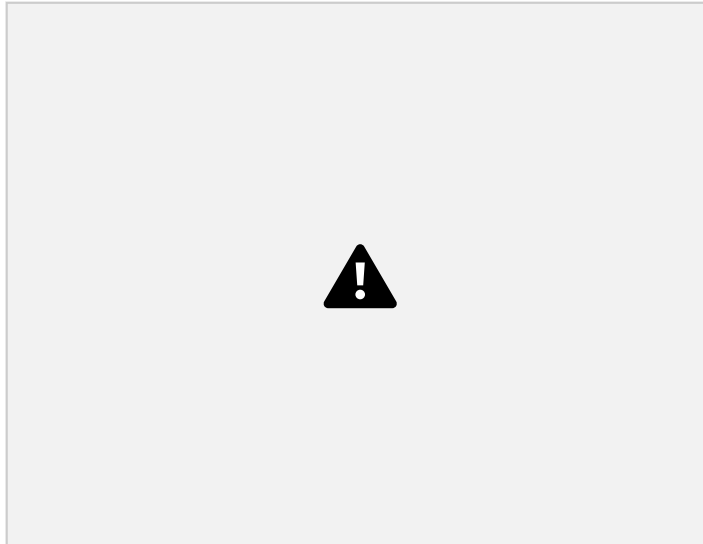
Bước 2: làm theo hình chỉ dẫn sau:

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 15

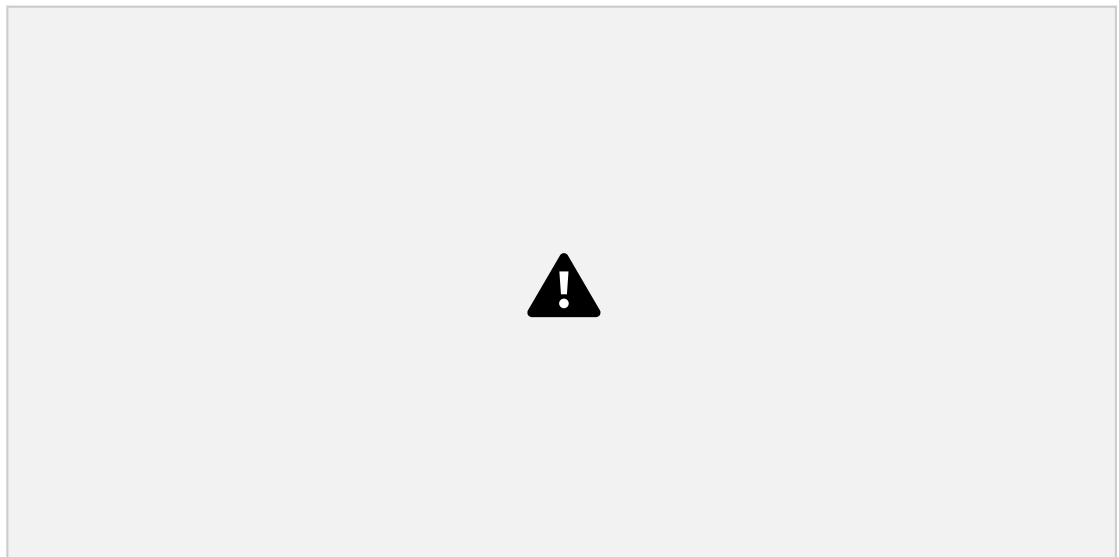
***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



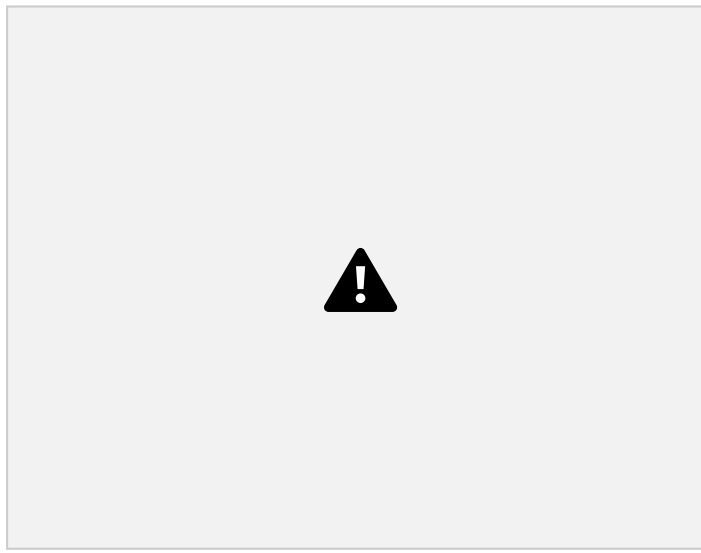
- Nhấn **Next** để tiếp tục.



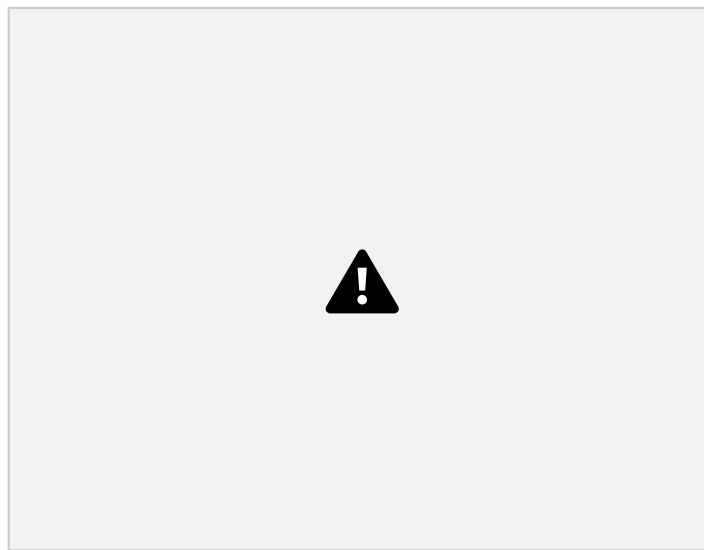
- Click chọn **I accept the agreement** và Nhấn **Next** để tiếp tục cài đặt.



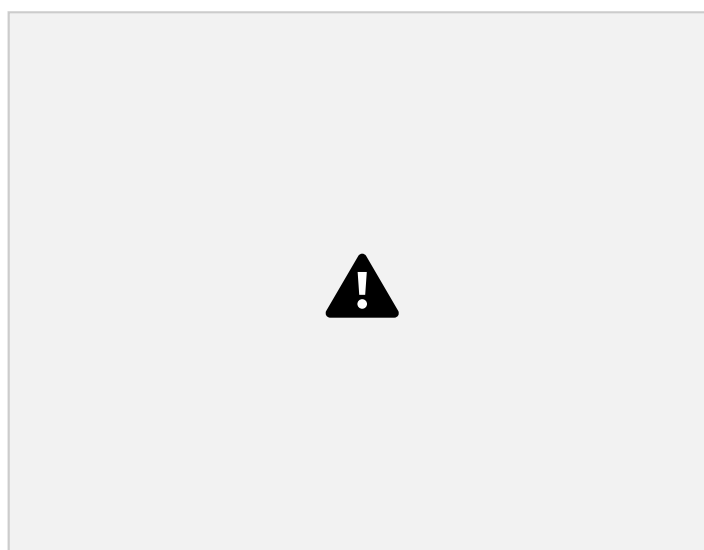
- Chương trình sẽ cài vào đường dẫn mặc định.
- Nhấn **Browse** để chọn lại đường dẫn tùy ý. Sau đó nhấn **Next** để tiếp tục cài đặt chương trình.



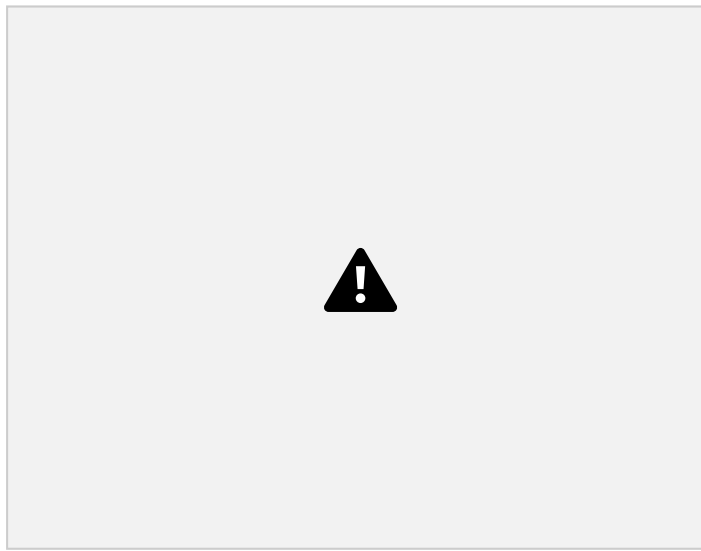
- Bước này nhấn **Next** để tiếp tục.



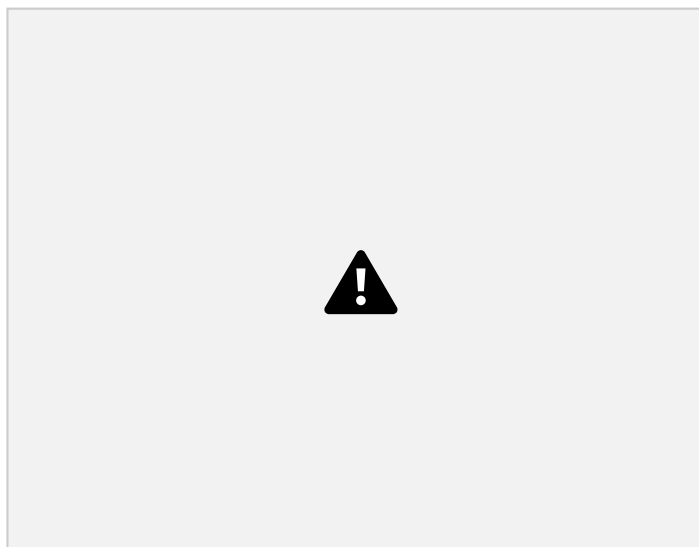
- Vẫn nhấn **Next** để tiếp tục việc cài đặt.



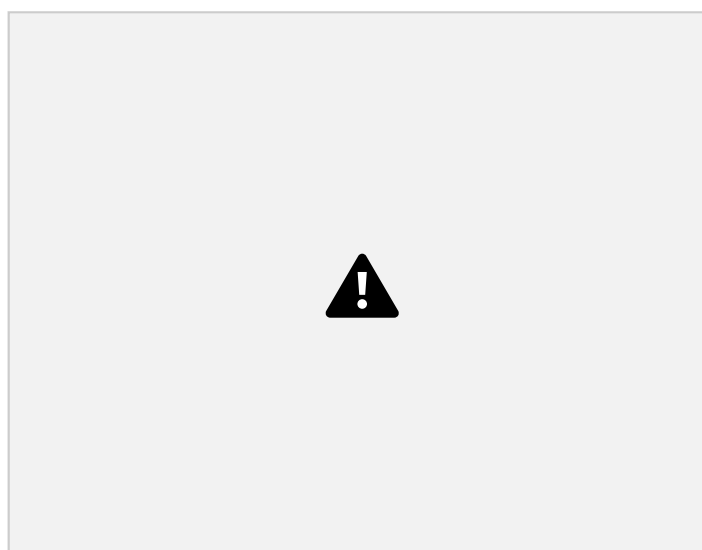
- Click check vào **Create a desktop icon** để tạo icon chương trình ngoài desktop.  
- Sau đó nhấn **Next** để tiếp tục quá trình cài đặt.



- Nhấn ***Install*** để bắt đầu quá trình cài đặt chương trình vào ổ cứng.

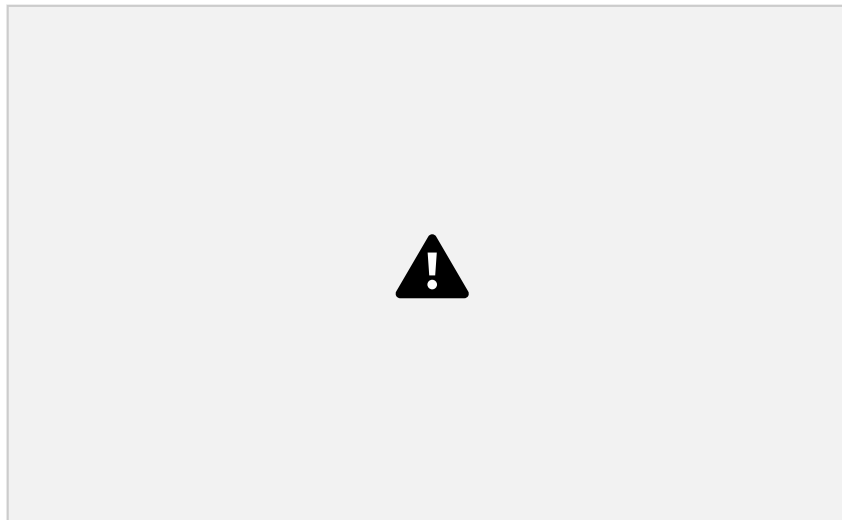


- Chương trình đang được cài đặt.



- Nhấn ***Finish*** để hoàn tất quá trình cài đặt.

## Tìm hiểu:



- Khởi động chương trình *phpDesigner*. Giao diện chương trình sẽ như hình vẽ trên.
- Bên trái là danh sách các ngôn ngữ lập trình mà chương trình hỗ trợ. Chúng ta chọn 1 ngôn ngữ lập trình rồi double click vào ngôn ngữ đó để tạo mới.
- Chính giữa là phần hiển thị danh sách các file đã mở. Chúng ta có thể mở file bằng cách nhấn vào biểu tượng *Open*
- Phía trên là menu chính cung cấp đầy đủ các chức năng của chương trình.

## 1.5. Cài đặt và cấu hình NetBeans IDE

### Nội dung:

Cài đặt:

- Đầu tiên để cài đặt được NetBeans, máy tính cần phải có JDK. Download JDK theo địa chỉ sau: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

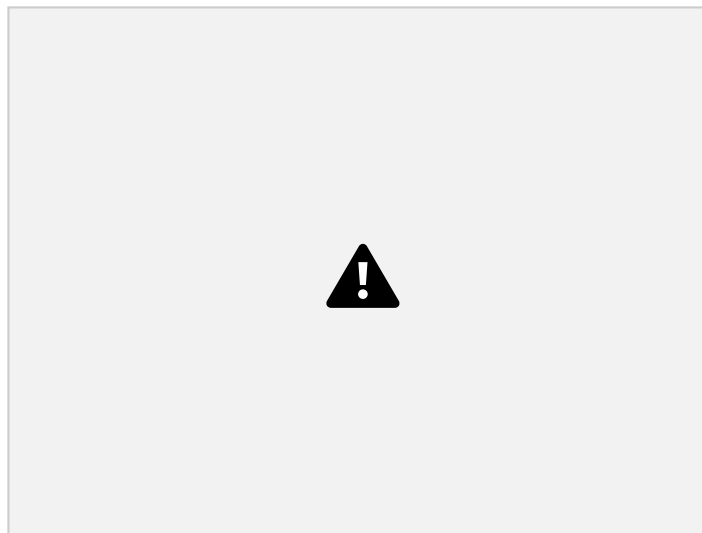
Bước 1: download phần mềm NetBeansIDE 7.0 full theo địa chỉ sau:

<http://netbeans.org/downloads/start.html?platform=windows&lang=en&option=all>  
phiên bản này bao gồm tất cả ngôn ngữ lập trình như java, php, c++, hoặc download bản chỉ gồm PHP

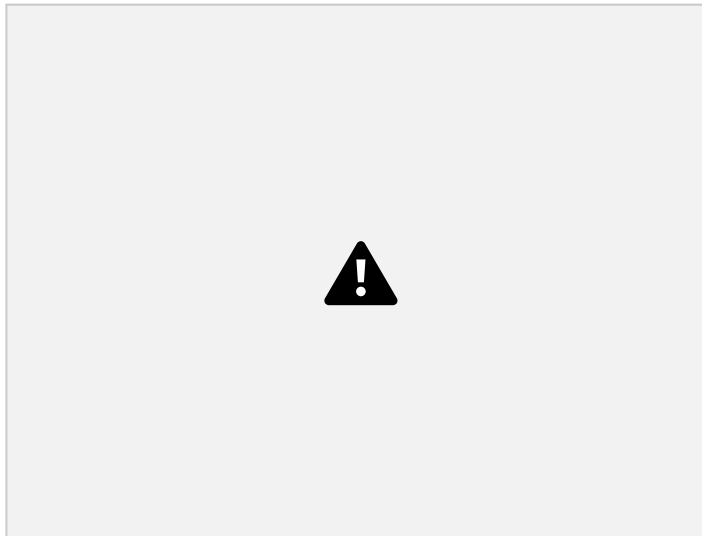
<http://netbeans.org/downloads/start.html?platform=windows&lang=en&option=php>

Bước 2: làm theo hình chỉ dẫn sau:

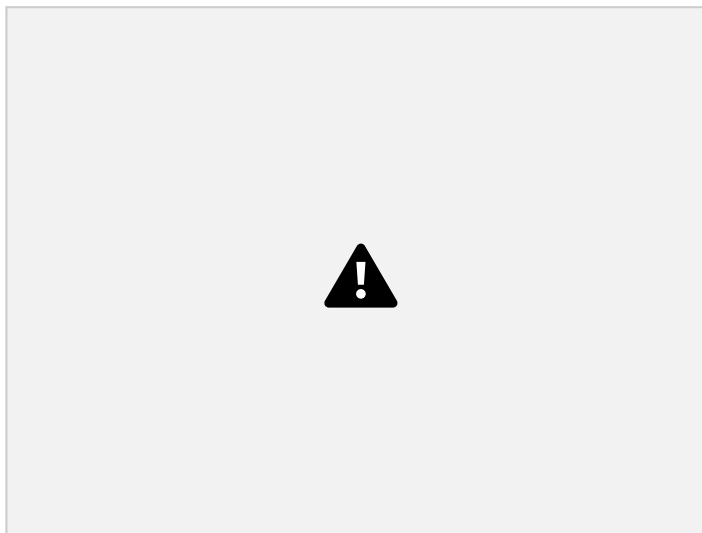
Hướng dẫn cài đặt JDK



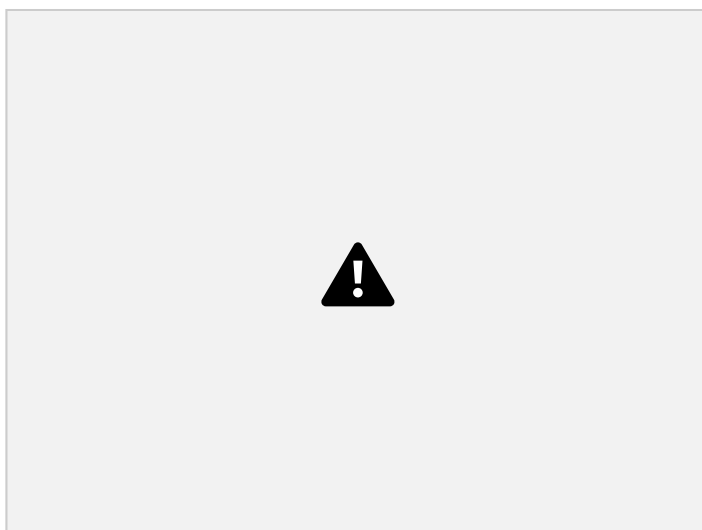
- Nhấn ***Next*** để tiếp tục quá trình.



- Nhấn ***Change*** để thay đổi đường dẫn tùy ý.
- Nhấn ***Next*** để bắt đầu cài đặt.

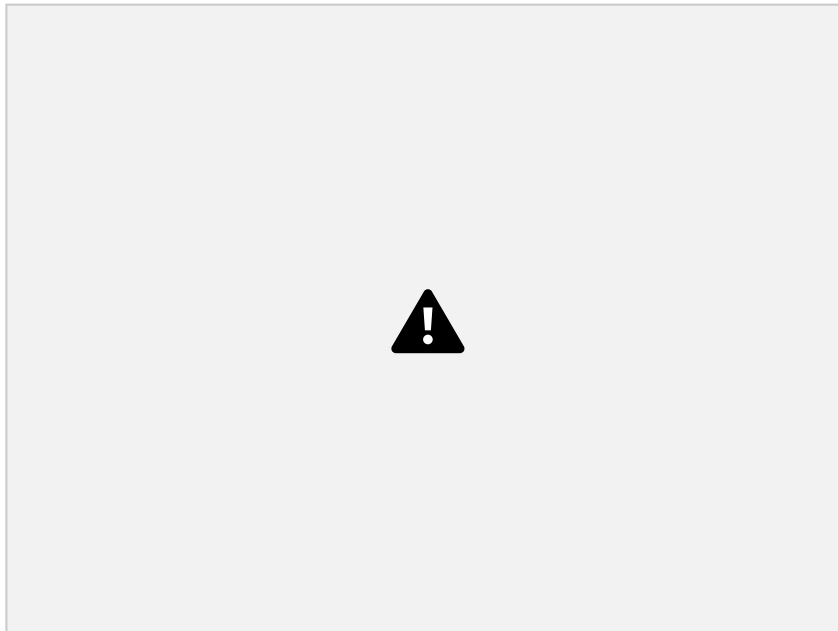


- Sau khi cài đặt JDK xong, chạy file cài đặt netbean 7.0

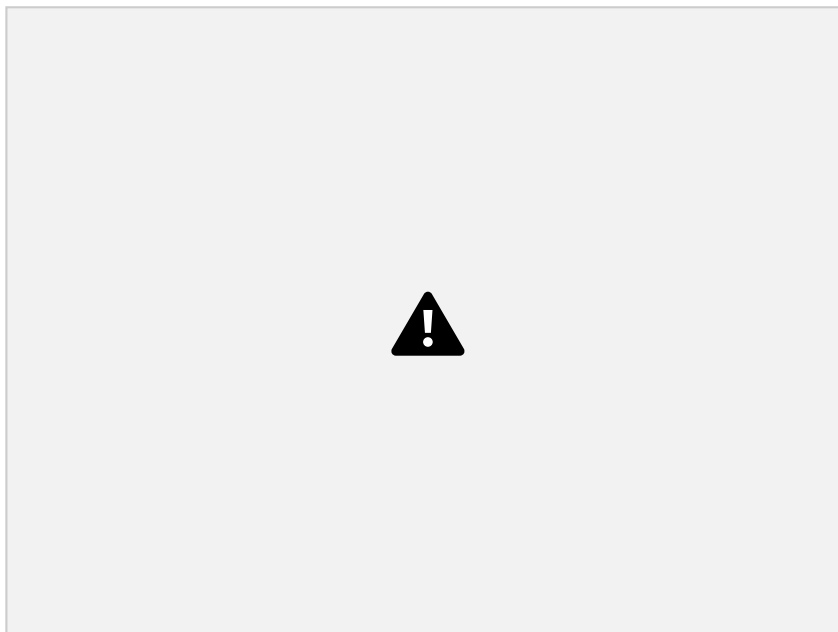


- Nhấn ***Customize*** để chọn ngôn ngữ mà bạn muốn cài đặt để lập trình.

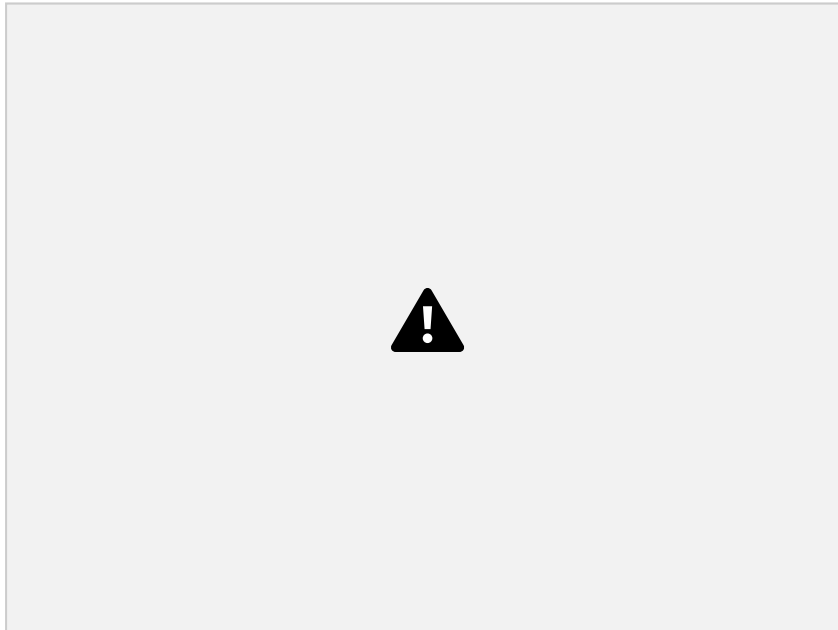
- Sau đó nhấn **Next** để tiếp tục.



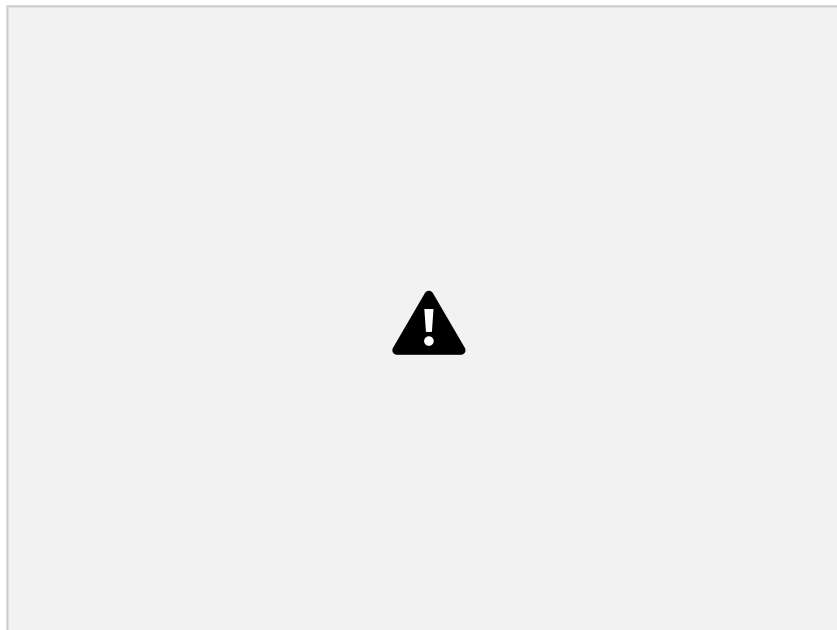
- Check vào ***I accept the terms in the license agreement.***
- Nhấn **Next** để tiếp tục cài đặt.



- Check vào ***I accept the terms in the license agreement. Install JUnit.***
- Nhấn **Next** để tiếp tục cài đặt.



- **NetBeans IDE** tự động tìm kiếm JDK đã cài đặt trước. Nếu bạn chưa cài JDK thì không thể cài đặt NetBeans IDE.
- Nhấn **Browse** phía trên nếu muốn thay đổi đường dẫn cài đặt chương trình. Sau đó nhấn **Next** để tiếp tục quá trình.



- Nhấn Next để tiếp tục.



BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 22

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



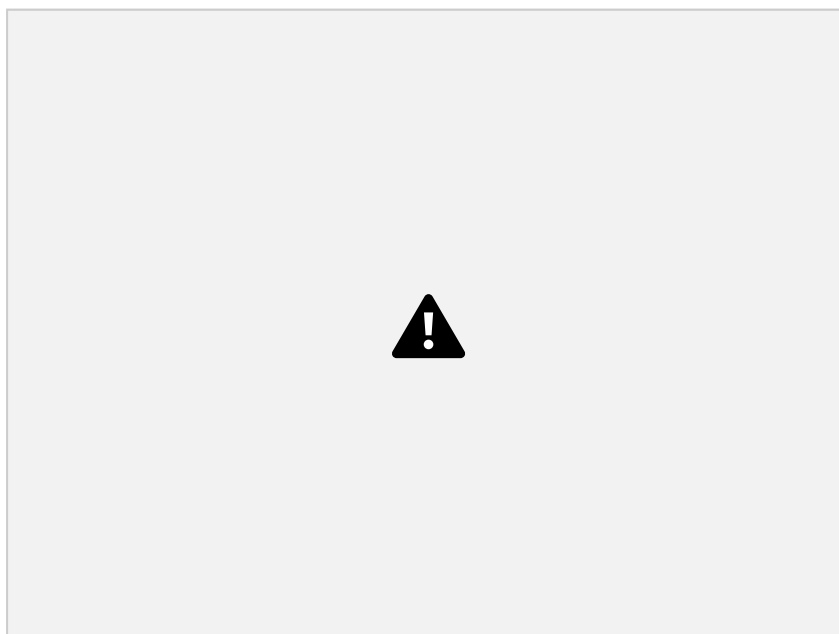
- Nhấn ***Install*** để cài đặt chương trình.



- Chương trình đang được cài đặt.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 23

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



- Nhấn ***Finish*** để hoàn tất quá trình cài đặt NetBeans IDE.
- Sau đó khởi động ***NetBeans IDE***.
- Giao diện ***NetBeans IDE*** sẽ như hình bên dưới.



- Hướng dẫn cách tạo ***Project PHP***

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 24

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



- Để tạo mới ***Project PHP***, chúng ta vào menu ***File*** chọn ***New Project***. Sau đó chương trình sẽ xuất hiện như hình sau:



- Nếu bạn lập trình PHP thì chọn ngôn ngữ PHP bên trái. Sau đó chọn ***PHP Application*** bên phải cửa sổ. Cuối cùng là nhấn ***Next*** để tiếp tục.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 25

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



- Nhấn ***Finish*** để tạo mới một project lập trình ngôn ngữ PHP.

### **Câu hỏi ôn tập:**

Câu 1: Làm sao để tạo mới database trong *Wamp/ Xampp* ?

Câu 2: Làm sao viết câu truy vấn trong *Wamp/Xampp* ?

Câu 3: Làm sao backup, import, export database trong *Wamp/Xampp*?

Câu 4: So sánh ưu & khuyết điểm của *Wampp* và *Xampp*?

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 26

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL CHƯƠNG 2***

## **2.1. PHP script**

### **2.1.1 Script**

- Code PHP có thể được chèn lẫn lộn và bất cứ ở đâu trong body HTML, chính vì vậy, để phân biệt code PHP bạn sử dụng cặp dấu `<?=Giá trị/Biểu thức/Biến?>` .

Chúng ta có thể khai báo như sau :

`<br>`

1-Giá trị biến MemberNo : `<?=$memberNo?>`

2-Giá trị biến k : `<?=$k?>`

3-Giá trị cụ thể : `<?=8000?>`

*Ví dụ :*

```
<HTML>
<HEAD>
<TITLE>Welcome to PHP World</TITLE>
</HEAD>
<BODY>
Hello : <?="PHP World"?>
</BODY>
</HTML>
```

### 2.1.2 Scriptlet

- Khác với cách khai báo Script, bạn chỉ được khai báo 1 biến. Scriptlet dùng trong trường hợp khi khai báo nhiều biến. Điều này có nghĩa là sử dụng cặp dấu trên như <?php Khai báo ?> với các khai báo PHP với cú pháp của C như sau:

```
<?php
    $tenSinhVien=$studentName;
    $column=$filed;
    $totalSalary = 8000;
    $studentNo="";
    ?>
-Khai báo trên là Scriptlet
<?= $paging ?>
-Khai báo này là Script
```

*Lưu ý: Khi kết thúc mỗi câu trong cript PHP phải dùng dấu ;  
Ví dụ :*

```
<HTML>
<HEAD>
<TITLE> Welcome to PHP World </TITLE>
</HEAD>
<BODY>
<?php
    $tenSinhVien=$studentName;
    $column=$filed;
    $totalSalary = 8000;
```

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 27

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

```
$studentNo="";
    $mySchool="TCN KTCN Hùng Vương";
    ?>
Giá trị của mySchool : <?=$mySchool?>
</BODY>
</HTML>
```

Lưu ý rằng, nếu bạn muốn sử dụng script hay scriptlet như ASP thì bạn khai báo trong tập tin php.ini như sau:

```
asp_tags = On ;  
Allow ASP-style <% %> tags. mặc định là Off
```

Khi đó , thay vì bạn khai báo cú pháp PHP là :

```
<?php  
    $tenSinhVien=$studentName;  
    $column=$filed;  
    $totalSalary = 8000;  
    $studentNo="";  
?>
```

Thì bạn có thể khai báo như sau:

```
<%  
    $tenSinhVien=$studentName;  
    $column=$filed;  
    $totalSalary = 8000;  
    $studentNo="";  
%>
```

## 2.2. Ghi chú trong PHP

- Ghi chú trong code PHP cũng tương tự như trong ngôn ngữ lập trình C. Để ghi chú 1 dòng trong PHP thì sử dụng một cặp dấu // hoặc #. Chẳng hạn chúng ta sẽ khai báo như sau :

```
<?php  
//Ghi chu mot dong trong code PHP  
    #$tenSinhVien=$studentName;  
    //$column=$filed;  
    #$totalSalary = 8000;  
    //$studentNo="";  
?>
```

- Ngoài ra, Nếu có nhiều dòng mà chúng ta cần ghi chú .Chúng ta nên sử dụng cặp dấu /\* Nội dung ghi chú \*/.

Ví dụ :

```
<?php
/*Ghi chu nhieu dong trong code PHP
    $tenSinhVien=$studentName;
    $column=$filed;
    $totalSalary = 8000;
    $studentNo="";*/
?>
```

### 2.3. In nội dung bằng PHP

- Khác với các kịch bản như ASP, JSP, Perl, đối với code PHP để in ra giá trị của biến, biểu thức, hàm hoặc giá trị cụ thể thì bạn có thể sử dụng script như sau:

```
Giá trị của mySchool : <?=$mySchool?>
```

Ngoài ra **echo** cũng được sử dụng khi in ra giá trị của biến, biểu thức, hàm, giá trị cụ thể trong PHP, chúng được khai báo như sau :

```
<?php
    $mySchool="TCN KTCN Hùng Vương";
    echo $ mySchool;
?>
```

Ví dụ :

```
<HTML>
<HEAD>
<TITLE> Welcome to PHP World </TITLE>
</HEAD>
<BODY>
<?php
$tenSV="Nguyen Van A";
$tuoi="19";
$soDT = "0975587553";
/*Dùng echo để xuất ra thông tin */
echo "Thông Tin Sinh Viên";
echo $tenSV;
        echo $ tuoi;
        echo $ soDT;

?>
</BODY>
</HTML>
```

### 2.4: Khái niệm cú pháp trong PHP

- Cú pháp PHP được xây dựng trên nền tảng trong ngôn ngữ C, vì vậy các bạn làm quen với ngôn ngữ C thì có lợi thế trong việc tìm hiểu và lập trình PHP sau này. Để lập trình tốt bằng ngôn ngữ PHP chúng ta cần chú ý những đặc điểm sau: - Cuối câu lệnh có dấu ;



- Biến trong PHP có tiền tố là \$
- Mỗi phương thức đều bắt đầu { và đóng bằng dấu }
- Khi khai báo biến thì không có kiểu dữ liệu

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 29

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

- Nên có giá trị khởi đầu cho biến khai báo
- Phải có chi chú (comment) cho mỗi thuộc tính mới
- Sử dụng dấu // hoặc # để giải thích cho mỗi câu ghi chú
- Sử dụng /\* và \*/ cho mỗi đoạn ghi chú
- Khai báo biến có phân biệt chữ hoa hay thường.

## 2.5: Khai báo biến trong PHP

- Khi khai báo biến trong C, chúng ta cần phải biết tuân thủ những quy định như : đầu tiên là kiểu dữ liệu , tiếp theo sau kiểu dữ liệu là tên biến và phải có giá trị khởi tạo. Nhưng trong PHP thì không cần khai báo phức tạp như vậy, chỉ cần sử dụng tiền tố \$ trước tên biến. Chúng ta có thể khai báo biến như sau :

```
$count = 0;
$queryStr = "SELECT * FROM tblSinhVien where status = 0";
$location = array("first", "last", "company");
$check = true;
```

Ví dụ :

```
<HTML>
  <HEAD>
<TITLE>Welcome to PHP World</TITLE>
</HEAD>
  <BODY>
<h4>Khai bao bien</h4>
<?php
$soSV = 10;
$test = true;

        $sql = "SELECT * FROM tblStudent";
$location = array("first", "last", "company");
        $myarrs[] = array(3);
        $myarrs[0]="SV001";
        $myarrs[1]="SV002";
        $myarrs[2]="SV003";
        echo $myarr[2];echo "<br>";
        echo $myarrs[1];

        ?>
  </BODY>
</HTML>
```

## 2.6: Kiểu dữ liệu

Danh sách các kiểu dữ liệu cơ bản và thường dùng :

- **Boolean** True hay false

- **Integer** giá trị lớn nhất xấp xỉ 2 tỷ
- **Float** ~1.8e308 gồm 14 số lẻ
- **String** Lưu chuỗi ký tự chiều dài vô hạn
- **Object** Kiểu đối tượng
- **Array** Mảng với nhiều kiểu dữ liệu

### 2.6.1: Thay đổi kiểu dữ liệu

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 30

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

- Thay đổi kiểu dữ liệu (ép kiểu). Để thay đổi từ kiểu dữ liệu này sang kiểu dữ liệu khác, tương tự như trong ngôn ngữ lập trình C hay Java, bạn có thể sử dụng cách ép kiểu. Chẳng hạn, khai báo ép kiểu như sau:

```
<HTML>
<HEAD>
<TITLE>Welcome to PHP World</TITLE>
</HEAD>
<BODY>
<h4>Thay doi kieu du lieu(Ep Kiep)</h4>
$i="T50T";
    echo $i."<br>";
    echo ($i + 20)."<br>";
    $i="200A";
    $j=(float)$i;
    echo $j."<br>";
    $j+=10;
    echo $j."<br>";
    $q=25;$p=4;
    echo "q/p: ".(float)$q/$p;
</BODY>
</HTML>
```

Lưu ý rằng PHP tự động nhận biết giá trị chuỗi đằng sau số, vì vậy các giá trị chuỗi sẽ không được chuyển sang kiểu số như trường hợp trên.

Ngoài phương pháp ép kiểu, hàm **settype** cũng có thể được sử dụng để chuyển đổi từ kiểu dữ liệu này sang một kiểu dữ liệu khác. Ví dụ chúng ta khai báo như sau :

```

<HTML>
<HEAD>
<TITLE>Welcome to PHP World</TITLE>
</HEAD>
<BODY>
<h4>Thay doi kieu du lieu(Settype)</h4>
    <?php
$var="99-THD";
    $check=true;
    echo $var."<br>";
    echo $check."<br>";
    settype($var,"integer");
    echo $var."<br>";
    settype($check,"string");
    echo $check;

?>
</BODY>
</HTML>

```

### 2.6.2: Kiểm tra kiểu dữ liệu của biến

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 31

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

- Để kiểm tra kiểu dữ liệu của biến, chúng ta sử dụng những hàm sau (kết quả trả về là true = 1):
- **is\_int** Kiểm tra biến có phải kiểu **Integer** không.
- **is\_array** Kiểm tra biến có phải kiểu **Mảng** không.
- **is\_bool** Kiểm tra biến có phải kiểu **Boolean** không.
- **is\_double** Kiểm tra biến có phải kiểu **Double** không.
- Ngoài ra còn có những hàm khác như : **is\_callable**, **is\_float**, **is\_integer**, **is\_long**, **is\_null**, **is\_numeric**, **is\_object**, **is\_real**, **is\_string**

Ví dụ :

```

<HTML>
    <HEAD>
        <TITLE>.:Welcome to PHP</TITLE>
    </HEAD>
    <BODY>
        <h4>Kiem tra kieu du lieu cua bien</h4>
        <?php
$record=5;
            $myarr = array("first", "last", "company");
            echo is_bool($record);
            echo "<br>";

            echo is_array($myarr);
            ?>
        </BODY>
</HTML>

```

### 2.6.3: Thay đổi kiểu dữ liệu biến

- Khi khai báo biến với một giá trị khởi tạo cho biến đó. Sau đó bạn muốn sử dụng giá trị của biến đó thành tên biến và có giá trị chính là giá trị của biến ban đầu mà ta khai báo, thì sử dụng cặp dấu **\$\$**. Chúng ta xem xét ví dụ sau :

```
<HTML>
  <HEAD>
    <TITLE>::Welcome to PHP</TITLE>
  </HEAD>
  <BODY>
    <h4>Thay doi kieu du lieu bien</h4>
    <?php
$obj="math";
        echo $obj;
        echo "<br>";
        $$obj=10;
        echo $math;
    ?>
  </BODY>
</HTML>
```

Ở ví dụ trên, biến **\$obj** có giá trị là *math*, sau đó chúng ta muốn đổi *math* thì tên biến thì sử dụng **\$\$obj**.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 32

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

### 2.6.4: Kiểu Array

#### Mảng 1 chiều

- Kiểu Array là một mảng các giá trị do người dùng định nghĩa. Thứ tự index trong một Array luôn bắt đầu từ 0 và tăng dần lên 1, cú pháp của kiểu Array được khai báo như sau :

```
//Mang gom cac kieu chuoi
$myarrs=array("SV001", "SV002", "SV003");
```

Hay chúng ta có thể khai báo như sau :

```
$myarrs[]=array(3);
$myarrs[0]=" SV001";
$myarrs[1]=" SV002";
$myarrs[2]=" SV003";
```

Ở ví dụ dưới đây, chúng ta sẽ minh họa cho việc sử dụng kiểu dữ liệu Array.

```

<HTML>
  <HEAD>
    <TITLE>::Welcome to PHP</TITLE>
  </HEAD>
  <BODY>
    <h4>Mang 1 chieu</h4>
    <?php
$room[]=array(4);
$room[0]="Room001";
        $room[1]=" Room002";
        $room[2]=" Room003";
        $room[3]=" Room004";
        echo $room[0];
        echo $room[1];
        echo $room[2];
        echo $room[3];
        echo "<br>";
        $languge=array("English", "VietNam", "Japan");
        echo $languge[2];

    ?>
  </BODY>
</HTML>

```

### **Mảng 2 chiều**

- Về cơ bản mảng 2 chiều trong php tương tự với trong C nhưng có một điểm khác đó là trong PHP không cần phải khai báo kiểu dữ liệu của mảng . Cú pháp khai báo mảng 2 chiều được minh họa trong ví dụ sau đây :

```

<HTML>
  <HEAD>
    <TITLE>Welcome to PHP</TITLE>
  </HEAD>

```

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 33

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

```

  <BODY>
    <h4>Mang 2 chieu</h4>
    <?php
        $student[][]=array(2,3);
        $student[0][0]="SV001";
        $student[0][1]="SV002";
        $student[0][2]="SV003";
        $student[1][0]="SV004";
        $student[1][1]="SV005";
        $student[1][2]="SV006";
        echo $student[0][2];

    ?>
  </BODY>

```

</HTML>

Ghi chú : ở ví dụ trên chúng ta khai báo một mảng 2 chiều có tên là **student** gồm 2 dòng và 3 cột, tương ứng sẽ có 6 phần tử trong mảng 2 chiều đó .

### 2.6.5 Kiểu đối tượng

- Trong PHP để khai báo một đối tượng, chúng ta sử dụng khái niệm Class như trong ngôn ngữ lập trình C hay Java. Ngoài ra các phương thức trong PHP còn được biết đến như một hàm(Function). Nếu một hàm trong Class bất kì có cùng tên với tên Class thì hàm đó cũng được gọi là hàm khởi tạo(Constructor). Khi ta khai báo 1 Class và Constructor. Thì khi khởi tạo một đối tượng, tự động Constructor được gọi. Sau đó gọi hàm trong Class đó (*nội dung chi tiết liên quan đến hướng đối tượng được trình bày trong chương 9*)

```
<HTML>
  <HEAD>
    <TITLE>Welcome to PHP World</TITLE>
  </HEAD>
  <BODY>
    <h4>Kieu doi tuong</h4>
    <?php
      class clsA
      {
function clsA() {
    echo "Day la ham khoi tao(Constructor)"; echo "<br />\n";
}
function B() {
    echo "Learning PHP very well";
    echo "<br />\n";
}
      }
      //Khoi tao mot doi tuong thuoc clsA
      $b = new clsA();
      echo "<br>";
      //Goi phuong thuc B().
      $b->B();
    }
  </BODY>
</HTML>
```

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 34

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

```
    ?>
  </BODY>
</HTML>
```

### 2.6.6 Tầm vực của biến

- Tầm vực của biến phụ thuộc vào nơi khai báo biến, nếu biến khai báo bên ngoài của một hàm hoặc lớp thì sẽ có tầm vực trong cả trang PHP. Ngược lại trong trường hợp biến khai báo trong hàm hoặc lớp thì chỉ có hiệu lực trong hàm hoặc lớp đó.

Ví dụ: chúng ta có biến **\$var** khai báo bên ngoài hàm nhưng khi vào trong hàm thì biến **\$var** được khai báo lại, thì biến **\$var** trong hàm có tầm vực nhỏ hơn khác với biến **\$var** ngoài hàm có tầm vực lớn hơn. Tương tự như vậy, khi biến **\$k** khai báo trong hàm thì chỉ có tầm vực bên trong hàm cho dù chúng được khai báo lại bên ngoài như ví dụ sau đây :

```
<HTML>
  <HEAD>
    <TITLE>::Welcome to PHP</TITLE>
  </HEAD>
  <BODY>
    <h4>Tam vuc cua bien</h4>
    <?php
      //bien toan cuc
      $var = 100;
      $k=1000;
      function Test() {
        //bien cuc bo trong ham
        $k=10;
        $var=10;
        echo "var=$var";echo"<br>";
        echo "k=$k";echo"<br>";
        /* reference to local scope variable */
      }
      Test();
      echo "var=$var"; echo"<br>";
      echo "k=$k"; echo"<br>";
    ?>
  </BODY>
</HTML>
```

-Ngoài ra từ khóa **global** được sử dụng khi chúng ta muốn biến toàn cục được dùng bên trong hàm.

Ví dụ :

```
<HTML>
  <HEAD>
    <TITLE>Welcome to PHP World</TITLE>
  </HEAD>
  <BODY>
```

```

        <h4>Su dung tu khoa Global</h4>
        <?php
            //bien toan cuc
$var = 100;

            function check() {
global $var;
$k=10;

                $var+=10;
echo "<br>var=$var";

                echo "<br>k=$k";
            }

            check();
            echo "<br>var=$var";
            $k=1000;
            echo "<br>k=$k";

        ?>
    </BODY>
</HTML>

```

*Ghi chú: khai báo biến \$var bên ngoài hàm check(), sau đó bên trong hàm chúng ta sử dụng từ khoá global cho biến \$var, khi đó biến \$var sẽ được sử dụng và giá trị đó có hiệu lực sau khi ra khỏi hàm.*

## 2.7: Hằng trong PHP

### 2.7.1: Khai báo và sử dụng hằng

- Hằng là một giá trị không bao giờ bị thay đổi kể từ khi chúng được khai báo, bạn có thể sử dụng từ khóa ***define*** để khai báo hằng, xem ví dụ sau :

```

<HTML>
    <HEAD>
        <TITLE>Welcome to PHP World</TITLE>
    </HEAD>
    <BODY>
        <h4>Hang trong PHP</h4>
        <?php
            define("day",24);
            function Test() {
echo "<br>day=".day;
echo "<br>day=".constant("day");
            }
            Test();
            echo "<br>day=".day;
            echo "<br>day=".constant("day");

        ?>
    </BODY>
</HTML>

```

### 2.7.2: Kiểm tra hằng



- Khi sử dụng hằng, mà chúng ta không biết hằng đó có tồn tại hay chưa thì bạn sử dụng hàm *defined*. Ví dụ :

```
<HTML>
```

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 36

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

```
<HEAD>
    <TITLE>Welcome to PHP World</TITLE>
</HEAD>
<BODY>
    <h4>Kiem tra hang</h4>
    <?php
        define("week",7);
        //define("sec",60);
        function Test()
        {
            if(defined("week")) {
                echo "<br>week=".week;
            }
            else {
                echo "<br>week is not defined";
            }
            if(defined("sec")) {
                echo "<br>sec=".sec;
            }
            else {
                echo "<br>sec not defined";
            }
        }
        Test();
    ?>
</BODY>
</HTML>
```

### **Câu hỏi ôn tập:**

Câu 1: In ra màn hình danh sách thông tin của sinh viên của các lớp. Biết có 2 lớp là a và b, trong mỗi lớp có 5 sinh viên, mỗi sinh viên có 1 mã sinh viên. (thông tin sinh viên ở đây là chỉ in mã sinh viên).

Câu 2: Viết ứng dụng máy tính cá nhân có chức năng thực hiện các phép tính (cộng, trừ, nhân, chia) đơn giản 2 chữ số. với các yêu cầu cơ bản sau:

- Tạo Form nhập dữ liệu
- Trong Form có 2 trường textbox để nhập các số thứ nhất và số thứ hai
- Trong Form có hộp selectbox để lựa chọn các phép toán (+, -, x, /)
- Trong Form có nút kết quả để khi bấm kết quả sẽ cho in ra kết quả ngay tại trang tính toán
- Nếu các số cần thực hiện phép tính chưa được nhập thì khi bấm kết quả sẽ thông

báo “Các trường nhập số không được để trống”.

- Nếu nội dung nhập vào không phải là số thì sẽ thông báo “Các trường nhập vào phải là số”.

Để hoàn thành một ứng dụng đạt yêu cầu, cần có một số gợi ý như sau:

- Để kiểm tra dữ liệu nhập vào có phải là một chuỗi rỗng hay không, chúng ta sử dụng `if(bien_kiem_tra == “”)`

- Để kiểm tra dữ liệu nhập vào có phải là một số hay không sử dụng hàm `isnumeric(bien_kiem_tra)`. Kết quả trả về TRUE nếu là số và FALSE nếu không phải số.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 37

### **Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL CHƯƠNG 3**

#### **3.1: Giới thiệu toán tử**

- **Toán tử số học:** là phép toán cộng trừ nhân chia trong số học, ngoài ra còn có thêm phép chia lấy dư (%), phép tăng thêm 1, giảm đi 1.

<b>Toán tử</b>	<b>Miêu tả</b>	<b>Ví dụ</b>	<b>Kết quả</b>
+	Cộng	$x = 2$ $x+2$	4
-	Trừ	$x = 2$ $5-x$	3
*	Nhân	$x = 4$ $x*5$	20
/	Chia	$15/5$ $5/2$	3 2.5
%	Chia lấy dư	$5\%2$ $10\%8$ $10\%2$	1 2 0
++	Cộng thêm 1	$x = 5$ $x++$	$x = 6$
-	Giảm đi 1	$x = 5$ $x--$	$x = 4$

- **Toán tử gán:** là phép gán trong PHP, sử dụng kí tự “=” kết hợp các phép toán khác để tính toán và gán giá trị của toán hạng bên phải cho toán hạng bên trái.

<b>Toán tử</b>	<b>Ví dụ</b>	<b>Giải thích</b>
=	$x = y$	$x = y$
+=	$x += y$	$x = x + y$
-=	$x -= y$	$x = x - y$
*=	$x *= y$	$x = x * y$

/=	x /= y	x = x / y
.=	x .= y	x = x . y
%=	x %= y	x = x % y

- **Toán tử so sánh:** là phép so sánh giữa 2 toán hạng với nhau, có thể bằng, khác, lớn hơn, nhỏ hơn, ....

Toán tử	Miêu tả	Ví dụ
=	So sánh bằng	5 = 8 trả về false
!=	Khác	5 != 8 trả về true
<>	Khác	5 <> 8 trả về true
>	Lớn hơn	5 > 8 trả về false
<	Nhỏ hơn	5 < 8 trả về true
>=	Lớn hơn hoặc bằng	5 >= 8 trả về false
<=	Nhỏ hơn hoặc bằng	5 <= 8 trả về true

- **Toán tử logic:** là phép toán kiểm tra logic (true/false) của 1 điều kiện cho trước.

Toán tử	Miêu tả	Ví dụ
&&, AND	Và	x = 6 y = 3 (x < 10 && y > 1) Trả về true

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 38

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

, OR	Hoặc	x = 6 y = 3 (x == 5    y == 5) Trả về false
!	Phủ định	x = 6 y = 3 ! (x == y) Trả về true

Ví dụ:



### 3.2: Phát biểu có điều kiện

- **Phát biểu if (nếu):** là phát biểu kiểm tra điều kiện nào đó. **Nếu** thỏa mãn và đúng với điều kiện **thì** sẽ 1 đoạn lệnh nào đó. **Nếu** không thỏa mãn điều kiện **thì** sẽ không thực thi đoạn lệnh bên trong.

#### Syntax (cách viết):

if ( điều kiện )

{

// TODO

}

Ví dụ:



BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 39

*Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL*



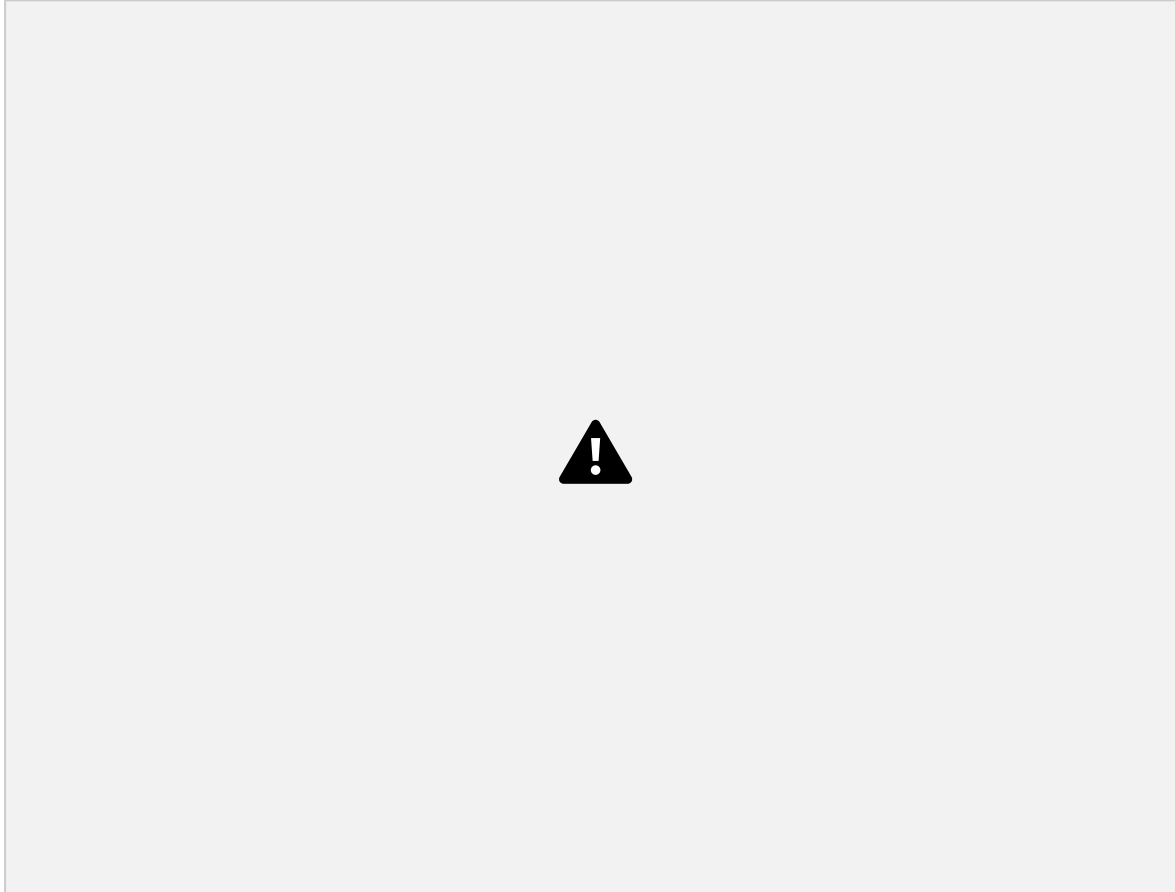
- **Phát biểu if - else:** cũng giống như phát biểu If. Phát biểu này cho phép ta có thể

kiểm tra nhiều điều kiện khác nhau trong trường các phát biểu If phía trên không thỏa mãn điều kiện.

**Syntax (cách viết):**

```
if( điều kiện thứ 1 ) {  
    // TODO  
} else if( điều kiện thứ 2 ) {  
    // TODO  
} else {  
    // TODO  
}
```

Ví dụ:



BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 40

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

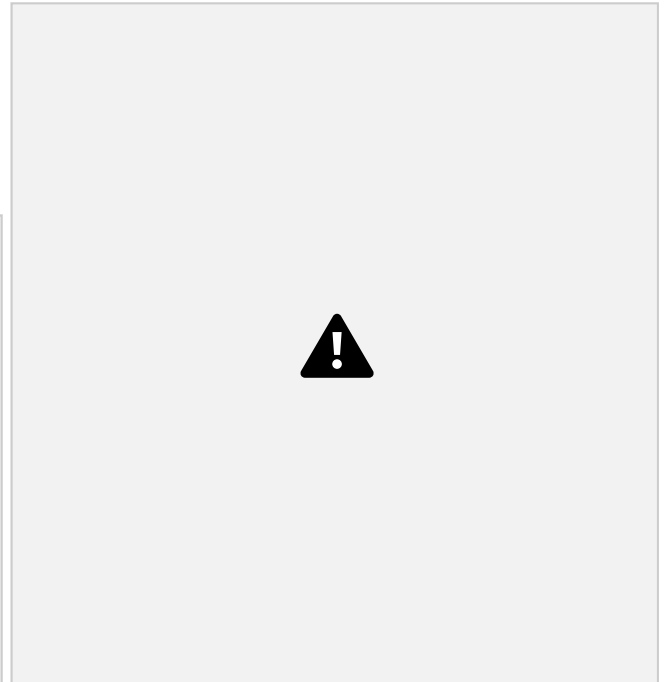
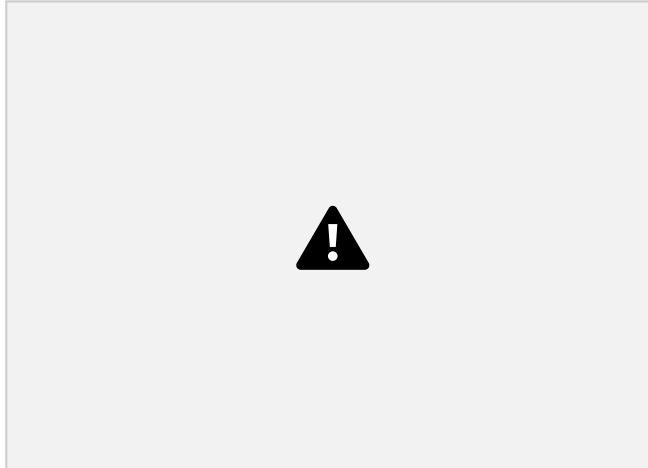
**3.3 Switch:** cũng như giống giống với phát biểu If. Trong nhiều trường hợp, bạn có thể so sánh chung 1 biến với nhiều giá trị value khác nhau và thực thi cái phần code khác nhau phụ thuộc trên cái giá trị mà được chọn.

**Syntax (cách viết):**

```
switch( biến ) {  
    case giá trị 1: {  
        //code thực thi  
        break;  
    }  
    case giá trị 2: {  
        //code thực thi  
        break;  
    }  
    default :{
```

//code thực thi nếu không thỏa các giá trị trên

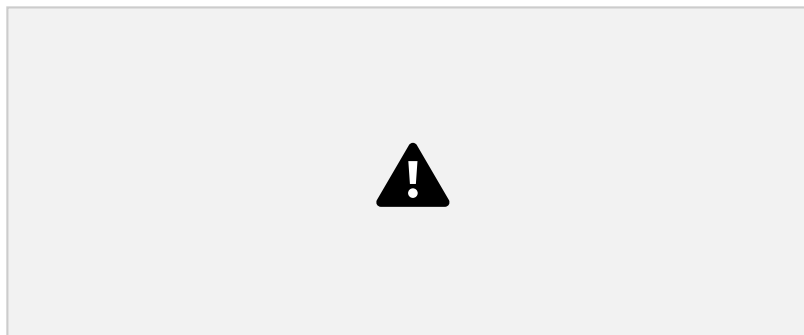
```
}  
}
```



**3.4 While:** cũng như mệnh đề if thì bắt buộc thỏa điều kiện trong while thì mới vào được vòng while để thực thi các lệnh.

**Syntax :**

```
While( Điều kiện ) {  
    //đoạn code sẽ được thực thi.  
}
```



BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 41

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

**3.5 For:** cách hoạt động của vòng lặp for cũng giống với while nhưng chỉ khác ở cách viết syntax.

**Syntax:**

```
For( điều kiện 1; điều kiện 2; điều kiện 3)  
{  
    //đoạn code sẽ được thực thi.  
}
```

VD:



**3.6Do – While:** giống với while nhưng **do while** thực thi đoạn code 1 lần rồi sau đó mới xét tới điều kiện trong while().

**Syntax:**

```
do {  
    //đoạn code thực thi  
}while( điều kiện );
```

**\*Chú ý:** sau while có dấu chấm “;”

VD:



**Câu hỏi ôn tập:**

Câu 1: In ra màn hình danh sách các số nguyên tố từ 1 đến 30.

Câu 2: Tạo 1 form đăng ký thành viên với các thông tin sau:

- Tên Đăng Nhập
- Họ Tên
- Email
- Điện Thoại
- Năm Sinh (combobox cho chọn từ năm 1950-2000)

Dữ liệu được ràng buộc bằng javascript, sau khi nhập đầy đủ thì click nút đăng ký thì hệ thống thông báo đăng ký thành công.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 42

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

## **CHƯƠNG 4**

### **4.1 Biến Form:**

- Biến Form trong PHP được hiểu như là một loại biến. Thay vì, nếu muốn sử dụng một biến nào đó thì chúng ta phải khởi tạo còn biến Form chính là thuộc tính **name** của bất kỳ một thẻ nhập nào trong Form hoặc tham số trên querystring.

#### 4.1.1 Form submit với phương thức POST:

- Khi từ một trang người dùng **submit** đến 1 trang khác (*phương thức action của thẻ form*), nếu khai báo thuộc tính name của bất kì một thẻ nào trong form là student thì biến form đó được định nghĩa là **\$student**.

Ví dụ:

```
...
<form action=example.php method=post>
  <tr>
    <td>Name :</td>
    <td><input type=text name=studentName></td>
  </tr>
  <tr>
    <td>Gender :</td>
    <td>
      <input type=radio value=M name=gender> Male
      <input type=radio value=F name=gender> Female
    </td>
  </tr>
  <tr>
    <td>&nbsp;</td>
    <td><input type=submit value=Submit></td>
  </tr>
</form>
...
```

- Khi người dùng nhập nội dung vào textbox của Student Name và chọn Giới Tính một trong hai radio Male hoặc Female, sau đó nếu chúng ta nhấn button Submit thì trang **resulExample.php** sẽ được gọi, trong trang này các giá trị nhập từ trang Example.php sẽ được lấy ra bằng cách sử dụng biến form.

Ví dụ :

```
<HTML>
  <HEAD>
    <TITLE>Welcome to PHP World</TITLE>
  </HEAD>
  <BODY>
    <h4>Bien Form</h4>
    <table>
      <tr>
        <td>Student Name :</td>
        <td><?=$studentName?></td>
      </tr>
    </table>
```

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 43

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

```
      <td>Gender :</td>
      <td><?=$gender?> </td>
    </tr>
```



```

</table>
</BODY>
</HTML>

```

- Ở ví dụ trên, chúng ta dễ dàng nhận thấy 2 biến \$studentName và \$gender là tên của hai thẻ input trong form của trang Example.php. Trong 2 Ví dụ phần này chúng ta sử dụng phương thức POST để truyền dữ liệu giữa 2 trang Example.php và resultExample.php (**method = post**).

#### 4.1.2 Form submit với phương thức GET:

- Khác với phương thức POST không truyền biến form lên QueryString, phương thức GET có thể lấy giá trị của các tham số trên chuỗi QueryString bằng biến form.

```

...
<form action=Example.php method=get>
  <tr>
    <td>Khoa-Trung Tam :</td>
    <td>
      <select name=khoa>
        <option value=CNTH>Cong Nghe Thong Tin</option> | <option value=CDT>Co
        Dien Tu</option> | <option value=DL>Dien Lanh</option> | </select>
      </td>
    </tr>
    <tr>
      <td>Country :</td>
      <td>
        <select name=country multiple>
          <option value=USA>United State</option> | <option value=VN>Viet
          Nam</option>
          <option value=CAN>Canada</option> | <option value=ITI >Italia</option>
        </select>
      </td>
    </tr>
    <tr>
      <td>&nbsp;</td>
      <td><input type=submit value=Submit></td>
    </tr>
  </form>
  ...

```

#### Câu hỏi ôn tập:

Câu 1: Viết trang Information.php và tạo Form thông tin cho phép người dùng nhập

vào thông tin cá nhân của mình với các thông tin như sau:

- họ tên : text
- giới tính : radio
- ngày sinh : text
- địa chỉ : textarea
- email : text

Và button Submit để chuyển sang trang PrintInfo.php với phương thức tùy ý. PrintInfo.php phải hiển thị đầy đủ thông tin mà người dùng đã nhập vào.

Câu 2: Viết trang đăng nhập với các thông tin email, mật khẩu. Chương trình tự động validate dữ liệu sử dụng javascript, nếu nhập thông tin đúng email là: [admin@hungvuongtech.edu.vn](mailto:admin@hungvuongtech.edu.vn) với mật khẩu là hungvuong thì thông báo đăng nhập thành công, các trường hợp khác thông báo thất bại.

## CHƯƠNG 5

### 5.1 Xử lý dữ liệu nhập trên form và thông báo lỗi với javascript.

B1: Xây dựng 1 form bất kỳ. Ở đây chúng ta sẽ xây dựng form đăng nhập để kiểm tra dữ liệu nhập vào có phù hợp hay không?



Giải thích: trong đoạn code trên cho phép người dùng đăng nhập vào hệ thống website khi nhập đúng tên đăng nhập và mật khẩu. Ở đây gọi hàm onsubmit với điều kiện trả về là phải thỏa mãn đoạn code trong hàm check() mà chúng ta sẽ viết tiếp theo trong phần sau.

Form đăng nhập sẽ như hình bên dưới.



Phần javascript chúng ta khai báo sai khi đóng thẻ title. Xem hình bên dưới.



*Giải thích:*

- + Từ khóa var là khai báo 1 biến mới.
- + Để lấy đối tượng HTML trong javascript, chúng ta dùng cú pháp:  
`document.getElementById("tenDangNhap");`
- + Cú pháp này nhận vào id của đối tượng HTML mà chúng ta đã tạo id cho đối tượng HTML trong đoạn code tạo form đăng nhập ở phía trên.  
(Chú ý: Chúng ta nên khai báo thuộc tính name và id cho đối tượng HTML giống nhau để tránh trường hợp nhầm lẫn)
- + Sau đó để lấy được giá trị nhập vào từ đối tượng đó (tùy thuộc vào đối tượng mà thuộc tính lấy sẽ khác nhau). Ví dụ ở đây là đối tượng input với type là text nên để lấy được giá trị text thì chúng ta chỉ cần `.value` như đoạn code javascript phía trên (VD: `tenDangNhap.value`)
- +Cú pháp return false hoặc true như sau:
  - Nếu return false tức điều kiện không thỏa mãn nên action form đăng nhập sẽ không chuyển qua trang home.php như code ví dụ.
  - Nếu return true đã thỏa mãn điều kiện nên lúc đó khi nhấn submit thì sẽ chuyển qua trang home.php.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 47

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



Thông báo lỗi “*Tên đăng nhập hoặc mật khẩu không được để trống.*” sẽ xuất tên đăng nhập và mật khẩu đều để trống.



Còn đây là trường hợp lỗi đăng nhập thất bại khi người dùng cung cấp thông tin tên đăng nhập hoặc mật khẩu không chính xác.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 48

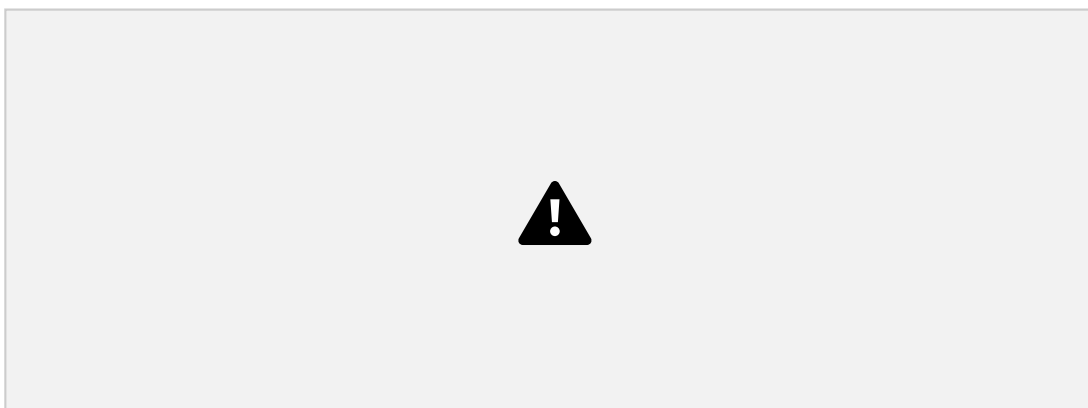
***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



Khi người dùng cung cấp thông tin chính xác (*tên đăng nhập là admin, mật khẩu là*

123) lúc này hệ thống kiểm tra thấy chính xác điều kiện thì xuất thông báo như hình trên.

Sau đó sẽ chuyển trang qua trang home.php. Vì trang home.php chúng ta chưa viết nên hệ thống sẽ báo lỗi là không tìm thấy trang home.php



## 5.2 Regular Expression

+Sơ bộ về Regular expression (*thường gọi tắt là RegEx*).

+RegEx là gì? RegEx là một biểu thức thường dùng để mô tả một mẫu câu chuỗi (*còn gọi là string pattern*).

+Mẫu chuỗi là gì? Mẫu chuỗi là các chuỗi văn bản tuân thủ theo 1 quy luật sắp xếp mẫu nào đó. Ví dụ dễ hiểu nhất là email, email được đặt theo quy định là gồm 2 thành phần: phần username và phần domain, 2 phần này được quy định là cách nhau bởi dấu @. Phần domain thì có quy luật riêng theo cách đặt tên của domain. +RegEx được dùng để làm gì và có tác dụng gì? RegEx được dùng để thiết lập để tạo ra 1 quy luật nào đó. RegEx có tác dụng là yêu cầu chương trình tìm kiếm theo quy luật vừa được thiết lập phía trên. (*Cụ thể là ví dụ về email đã được nhắc đến phía trên*).

+Cú pháp RegEx trong javascript: có 2 cách để tạo regular expression trong javascript:

- **Cách 1:** var reExample = /pattern/thamso;
- **Cách 2:** cách này tạo regular expression theo cấu trúc constructor RegExp() : var reExample = new RegExp("pattern","thamso");

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 49

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

thamso : tham số này xác định cách thức tìm kiếm mẫu chuỗi và chứa các giá trị sau đây:

○ **g** : (“global”) lặp lại quá trình tìm kiếm cho đến khi kết thúc chuỗi. ○ **i** : so sánh không phân biệt hoa thường.

○ **m** : so sánh ở chế độ đa dòng.

+Các ký tự dùng trong mẫu trong pattern:

• **\** : loại bỏ các ký tự đặc biệt.

• **{n}** : lặp 1 ký tự hoặc 1 biểu thức con trước đó n lần

• **{n,m}**: lặp 1 ký tự hoặc 1 biểu thức con trước đó từ n đến m lần • **{n,}**: lặp 1 ký tự hoặc 1 biểu thức con trước đó lớn hơn hoặc bằng n lần • **(pattern)**: tìm và bắt lấy /pattern/

• **(?:pattern)**: tìm nhưng không bắt lấy /pattern/

• **|** : tìm thay phiên

• **[characters]**: tìm trong chuỗi chứa bất kỳ các ký tự characters • **[^characters]**: tìm trong chuỗi không chứa bất kỳ các ký tự trong character. • **[a-z]**: tìm 1 ký tự

bất kỳ nằm trong tập hợp a đến z.

- \* : lặp 1 ký tự hoặc 1 biểu thức con trước đó lớn hơn bằng 0 lần
- + : lặp 1 ký tự hoặc 1 biểu thức con trước đó lớn hơn bằng 1 lần
- ? : lặp 1 ký tự hoặc 1 biểu thức con trước đó 0 hoặc 1 lần
- ^ : trả về chuỗi kết quả trong trường hợp chuỗi này nằm ở vị trí đầu của chuỗi gốc
- \$ : trả về chuỗi kết quả trong trường hợp chuỗi này nằm ở vị trí cuối của chuỗi gốc.
- . : tìm bất cứ ký tự nào trừ ký tự xuống dòng
- \b : trả về chuỗi kết quả trong trường hợp chuỗi này nằm ở vị trí đầu của 1 từ trong chuỗi gốc. Nếu cần so sánh ở vị trí cuối từ hãy đặt \b ở vị trí cuối từ.
- \B : trả về chuỗi kết quả trong trường hợp chuỗi này không nằm ở vị trí đầu của 1 từ trong chuỗi gốc. (Tùy thuộc vào vị trí đặt \B ở đầu hay cuối của 1 từ.)
- ?= : trả về chuỗi kết quả nếu theo sau đó là chuỗi nào đó được chỉ định trước
- ?! : trả về chuỗi kết quả nếu theo sau đó không phải là chuỗi nào đó được chỉ định trước
- \w : tìm 1 ký tự dạng từ a-Z, 0-9 và dấu gạch dưới.
- \W : ngược lại với \w
- \d : tìm 1 ký tự thuộc tập ký tự từ 0-9
- \D : ngược lại với \d
- \s : tìm ký tự cách (khoảng trắng)
- \S : ngược lại với \s

#### **Gộp nhóm các biểu thức:**

- () : tìm kiếm 1 nhóm các ký tự bên trong cặp dấu ngoặc và lưu vào chuỗi kết quả.
- (?:) : tìm kiếm 1 nhóm các ký tự bên trong cặp dấu ngoặc nhưng không lưu vào chuỗi kết quả.
- | : phép toán hoặc dùng để kết hợp các mệnh đề với nhau vào chung 1 biểu thức.

+Các phương thức có hỗ trợ RegEx trong javascript: ở đây ví dụ chúng ta đã có 1 chuỗi và đặt là var str = "chao buoi sang ban An";

- **str.search(/pattern/)**: phương thức này trả về vị trí đầu tiên tìm được của chuỗi này trong chuỗi pattern.

Vd: document.write(str.search(/bui/));

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 50

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

⇒ Kết quả trả về sẽ là: 5

- **str.replace(/pattern/,"chuỗi thay thế")**: phương thức này tìm và thay thế pattern tìm được trong chuỗi str và thay thế thành chuỗi thay thế.

Vd: document.write(str.replace(/sang/,"toi"));

⇒ Kết quả trả về sẽ là: chao buoi toi ban An

- **str.split(/pattern/)**: phương thức này sẽ tìm pattern đó chuỗi và sẽ dùng pattern đó để cắt chuỗi thành mảng.

Vd: document.write(str.split(/s/g));

⇒ Kết quả trả về sẽ là: chao,bui,sang,ban,An

- **str.match(/pattern/)**: trả về 1 mảng các thông tin nếu tìm thấy và false nếu không tìm thấy /pattern/ trong chuỗi str.

Vd: var str = "Watch out for the rock!";

document.write(str.match(/r?or?/g));

⇒ Kết quả là: o,or,ro

- **/pattern/.test(str)**: trả về true nếu pattern tìm thấy trong str và ngược lại trả về false.



Vd: `var ketqua = /example/.test("My Example");`

⇒ Kết quả trả về là: false vì javascript phân biệt hoa thường.

○ **/pattern/.exec(str):** trả về array không nếu /pattern/ tìm thấy trong chuỗi str và ngược lại.

Vd: `document.write(/s(amp)le/i.exec("Sample text"));`

⇒ Kết quả trả về là: Sample,amp

### Câu hỏi ôn tập:

- Viết trang Register.php với 2 thông tin như sau:

- Tên người dùng : text
- Mật khẩu : password
- Button Tạo mới

Khi click button sẽ chuyển đến trang xử lý Process.php.

Chú ý: kiểm tra như sau trước khi chuyển đến Process.php

- Các fiel không được để rỗng.
- Mật khẩu nhập vào phải ít nhất 5 kí tự và phải bao gồm chữ cái và số.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 51

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

## CHƯƠNG 6

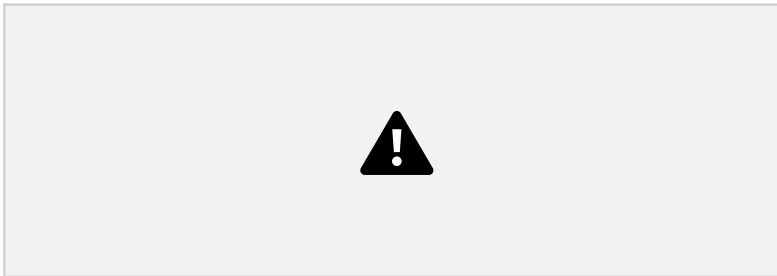
### 6.1 Đối tượng Session

**Tổng quan về session:** Một cách khác quản lý người sử dụng là session. Session được hiểu là khoảng thời gian người sử dụng giao tiếp với 1 ứng dụng. Một session được bắt đầu khi người sử dụng truy cập vào ứng dụng lần đầu tiên, và kết thúc khi người sử dụng thoát khỏi ứng dụng. Mỗi session sẽ có được cấp một định danh (ID) khác nhau và nội dung được lưu trong thư mục thiết lập trong file php.ini (tham số session.save\_path).

**Khởi tạo session:** việc khởi tạo session thì bắt buộc trang nào chúng ta đều phải start session lên thì mới có thể sử dụng session. Để khởi tạo 1 session, ta sử dụng cú pháp: `session_start();`. Đoạn code này phải nằm trên các kịch bản HTML hoặc những lệnh echo, printf.

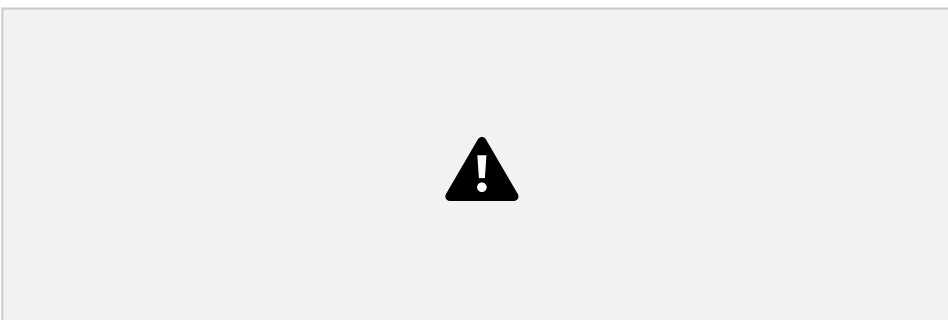
**Thiết lập giá trị cho session:** ngoài việc cho phép bắt đầu thực thi session, ta phải đăng ký 1 giá trị cho session. Ta sử dụng cú pháp: session\_register("Tên session")

VD: Chúng ta sẽ bắt đầu khởi tạo session và đăng ký 1 session có tên là userName

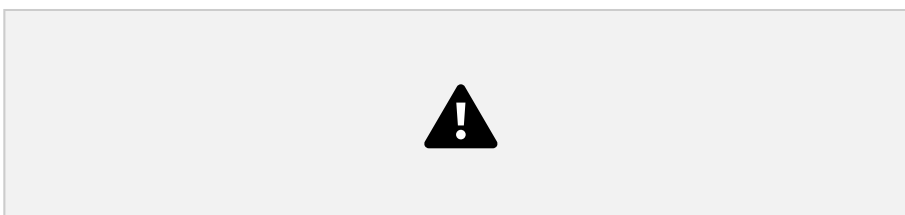


Gán giá trị vào session và lấy giá trị từ session:

+Việc gán giá trị vào session được thực hiện như ví dụ sau:



+Việc lấy giá trị từ session như sau:



+**Hủy session**: việc hủy session có 2 cách hủy session:

Hủy 1 session, cú pháp là: unset(\$\_SESSION["tên session"]);

Hủy toàn bộ session, cú pháp là: session\_destroy();

## 6.2 Cookie

+**Tổng quan về cookie:**

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 52

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

- Cookie là 1 đoạn dữ liệu được ghi vào đĩa cứng hoặc bộ nhớ của máy người sử dụng. Nó được trình duyệt gửi ngược lên lại server mỗi khi browser tải 1 trang web từ server.

- Những thông tin được lưu trữ trong cookie hoàn toàn phụ thuộc vào website trên server. Mỗi website có thể lưu trữ những thông tin khác nhau trong cookie, ví dụ thời điểm lần cuối ta ghé thăm website, đánh dấu ta đã login hay chưa, v.v... Cookie được tạo ra bởi website và gửi tới browser, do vậy 2 website khác nhau (cho dù cùng host trên 1 server) sẽ có 2 cookie khác nhau gửi tới browser. Ngoài ra, mỗi browser quản lý và lưu trữ cookie theo cách riêng của mình, cho nên 2 browser cùng truy cập vào 1 website sẽ nhận được 2 cookie khác nhau.

+**Thiết lập cookie:**

Để thiết lập cookie ta sử dụng cú pháp:

**Setcookie("tên cookie","giá trị", thời gian sống);**

*Tên cookie là tên mà chúng ta đặt cho phiên làm việc.*

*Giá trị là thông số của tên cookie.*

Ví dụ:

**Setcookie("userName","admin", time() +3600)**

Như ví dụ trên ta thấy với tên là username và giá trị là admin, có thời gian sống là 1 giờ tính từ thời điểm thiết lập.

Chú ý: Kịch bản cookie phải đặt trên mọi giá trị trả về bao gồm thẻ HTML và lệnh echo.

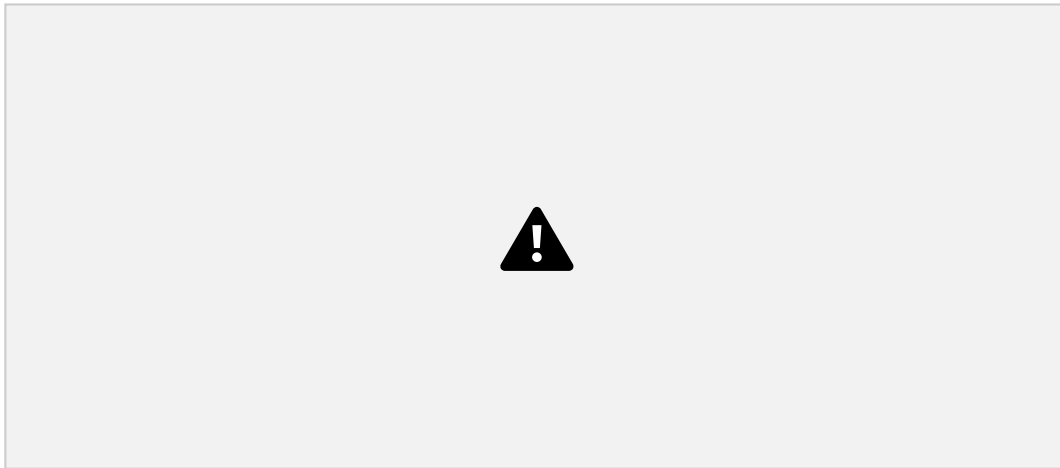
+Sử dụng cookie:

Để sử dụng lại cookie vừa thiết lập, chúng ta sử dụng .

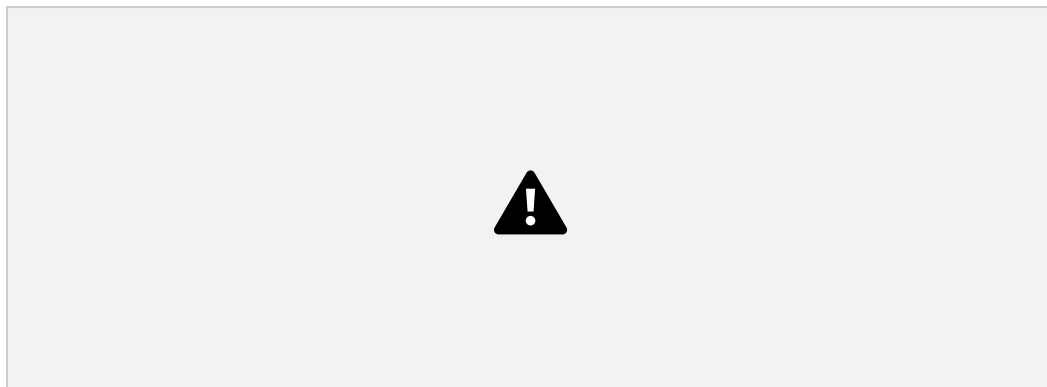
Cú pháp: **\$\_COOKIE["tên cookies"];**

Tên cookie là tên mà chúng ta thiết lập phía trên.

Ví dụ: page1.php



page2.php sẽ có code như sau:



Kết quả in ra trình duyệt là:



+**Hủy cookie:** Để hủy 1 cookie đã được tạo ta có thể dùng 1 trong 2 cách sau:

- Cú pháp: `setcookie("Tên cookie")`

Gọi hàm `setcookie` với chỉ duy nhất tên cookie mà thôi

- Dùng thời gian hết hạn cookie là thời điểm trong quá khứ.

Ví dụ: `setcookie("userName", "Nguyen Van C", time()-3600);`

### Câu hỏi ôn tập:

Câu 1: Viết trang `login.php` và lưu trữ thông tin vào biến session `user`, sau đó in ra thông tin đăng nhập thông qua session.

Câu 2: Xây dựng trang web theo mẫu sau:



*Click vào nút thêm vào giỏ thì sử dụng session để lưu thông tin tên sản phẩm và giá sản phẩm. Thiết kế thêm 1 menu xem giỏ hàng, lúc đó hiển thị tất cả các sản phẩm được chọn để người dùng cập nhật và thanh toán.*

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 54

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

## CHƯƠNG 7

### 7.1 Các hàm xử lý chuỗi :

Định dạng chuỗi:

- Chuyển chuỗi từ chữ thường thành hoa: hàm `strupper`. Vd: `strupper("chuong")`  
=> kết quả: `CHUONG`

- Chuyển chuỗi từ chữ hoa thành thường: hàm `strtolower`. Vd: `strtolower("CHUONG")`  
=> kết quả: `chuong`

- Viết hoa kí tự đầu tiên: hàm `ucfirst`. Vd: `ucfirst("chuong")` => kết quả: `Chuong` -

Viết hoa kí tự đầu của mỗi từ: hàm ucwords. Vd: ucwords(“quoc chuong”) => kết quả: Quoc Chuong

Kết hợp hay tách chuỗi:

- Hàm explode: hàm dùng để tách chuỗi. Hàm này trả về 1 mảng.

Vd:



Kết quả:



- Hàm implode thì ngược lại với hàm explode. Hàm này ghép nối 2 phần tử của mảng lại và nối giữa 2 phần tử này là 1 kí tự hoặc 1 chuỗi nào đó.

Vd: như nối 2 phần của email trên vd của explode đã tách ra bằng @



BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 55

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

Hàm substr: hàm cắt chuỗi theo điều kiện. Hàm này như sau:

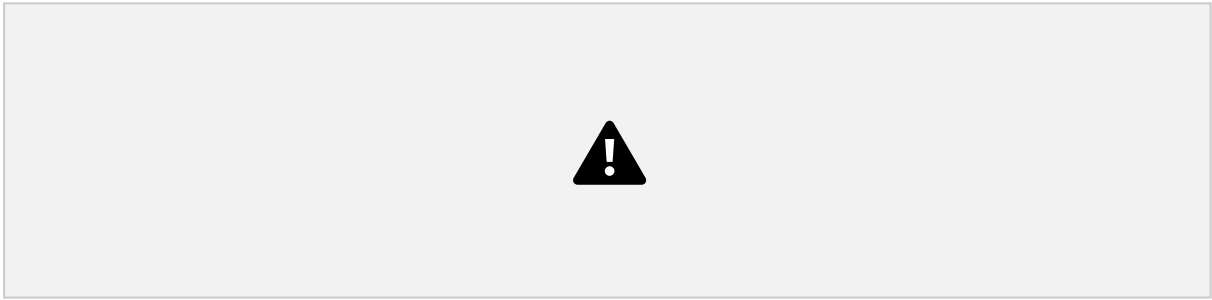
*substr(tham số 1, tham số 2);*

Chú thích: tham số 1 là chuỗi cần cắt, tham số 2 là bắt đầu cắt từ đâu. Chuỗi sẽ được lấy từ tham số 2 trở về sau chuỗi đó.

*substr(tham số 1, tham số 2, tham số 3)*

Chú thích: tham số 1 là chuỗi cần cắt, tham số 2 là bắt đầu cắt. Tham số 3 là số lượng kí tự cần lấy.

Ví dụ:

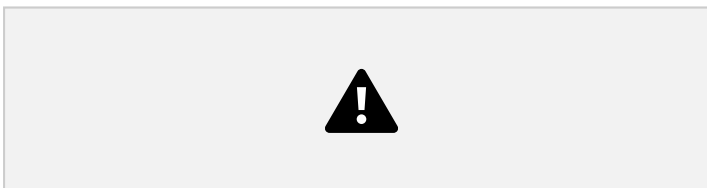


Và kết quả là:



Hàm strlen: hàm này sẽ trả về chiều dài của chuỗi.

Ví dụ:



Kết quả trả về là 12.

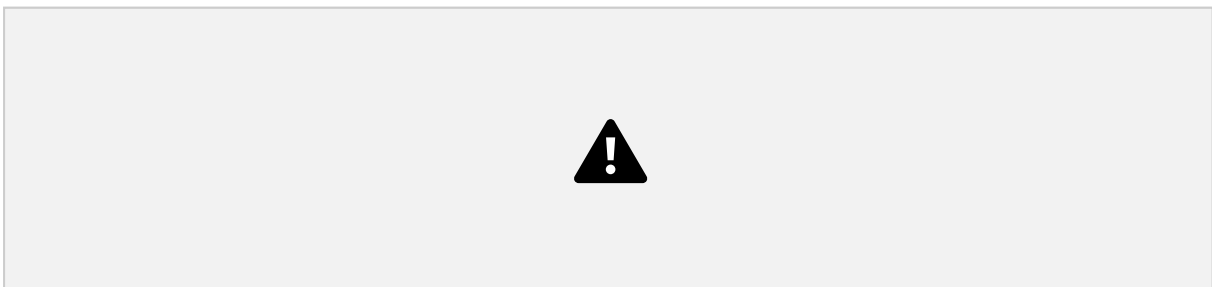
Hàm strcmp(), strcasecmp(), strnatcmp(): để so sánh chuỗi ta có thể dùng hàm strcmp(). Nếu hàm này trả về 0 khi 2 chuỗi này bằng nhau. Nếu chuỗi 1 lớn hơn chuỗi 2 thì hàm sẽ trả về giá trị lớn hơn 0, ngược lại thì sẽ nhỏ hơn 0.

Syntax của strcmp(): strcmp(chuỗi 1, chuỗi 2);

Chú ý: hàm so sánh chuỗi strcmp() này có phân biệt chữ hoa và chữ thường. Trong trường hợp ngược lại thì sử dụng hàm strcasecmp(), strnatcmp().

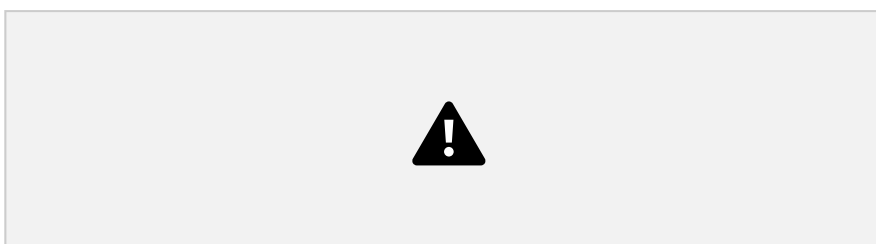
Hàm thay thế chuỗi: str\_replace(), substr\_replace().

Ví dụ:



BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 56

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



## **7.2 Làm việc với mảng dữ liệu**

Mảng 1 chiều: Cách khai báo mảng 1 chiều như sau:

Syntax:

Khai báo 1 mảng rỗng: `$arr = array();`

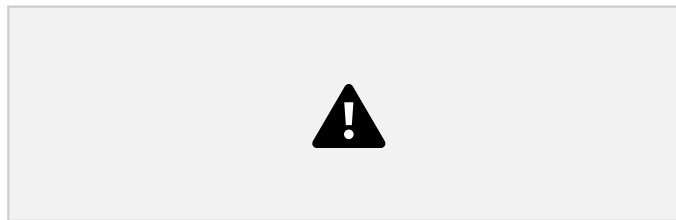
Khai báo 1 mảng có sẵn phần tử:

`$arr = array(phần tử 1, phần tử 2, phần tử 3,...);`

Ví dụ:



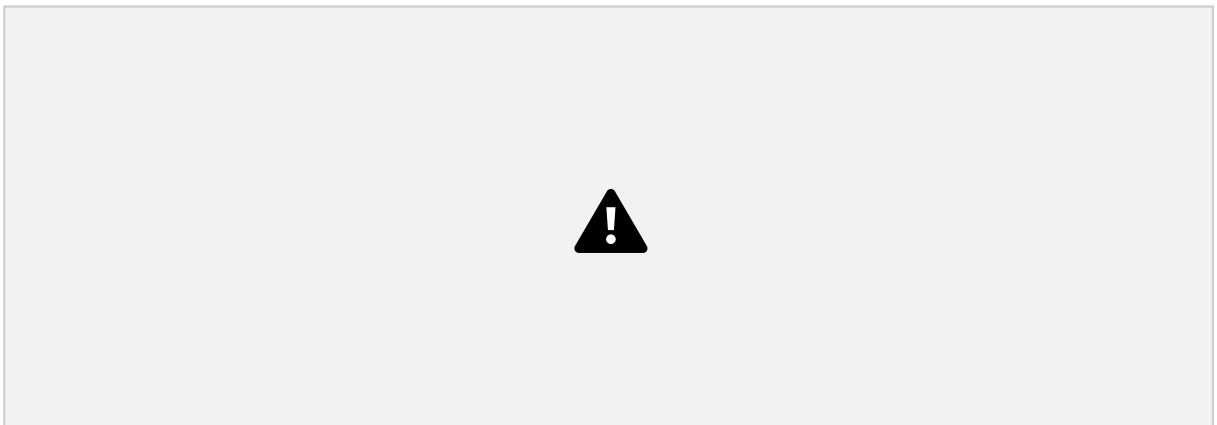
Kết quả in ra là:



Mảng 2 chiều: thực chất ra cũng là mảng 1 chiều nhưng được gọi là 2 chiều là vì trong mảng 1 chiều có chứa thêm 1 mảng 1 chiều.

`$arr = array(array(),array(),...);`

Ví dụ:



### 7.3 Kiểu datetime

- Để làm việc với kiểu dữ liệu Date và Time, ta sử dụng hàm của PHP có sẵn. Chẳng hạn, muốn trình bày chuỗi ngày tháng, ta dùng hàm date với các tham số như ví dụ sau:



Kết

quả:



Các tham số được quy định trong bảng sau:

<u>Kí tự</u>	<u>Ý nghĩa</u>
<b>a</b>	am/pm
<b>A</b>	AM/PM
<b>d</b>	Day (01-31) trong tháng với 2 số, nếu ngày 1-9 sẽ có kèm thêm số 0
<b>D</b>	Day (Mon-Sun) trong tuần với 3 kí tự
<b>F</b>	Tháng (January-December) trong năm với tên tháng đầy đủ dạng text
<b>g</b>	Hour (1-12) trong ngày, 1 hoặc 2 số (không kèm thêm số 0 nếu giờ từ 1-9)
<b>G</b>	Hour (0-23) trong ngày, 1 hoặc 2 số (không kèm thêm số 0 nếu giờ từ 1-9)
<b>h</b>	Hour (01-12) trong ngày, 2 số (kèm 0 nếu giờ từ 01-09)
<b>H</b>	Hour (00-23) trong ngày, 2 số (kèm 00 nếu giờ từ 00-09)
<b>i</b>	Minutes (01-59) đã trôi qua (kèm 00 nếu phút từ 00-59)
<b>j</b>	Day (1-31) 1 hoặc 2 số (không kèm 0 nếu ngày từ 1-9)



<b>l</b>	Day (Monday-Sunday) trong tuần dạng text
<b>L</b>	Năm nhuận trả về 1, ngược lại trả về 0
<b>m</b>	Month (01-12) trong năm 2 số (kèm 00 nếu tháng từ 01-09)
<b>M</b>	Month (Jan-Dec) trong năm 3 ký tự
<b>n</b>	Month (1-12) 1 hoặc 2 số (không kèm 0 nếu tháng từ 1-9)

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 58

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

<b>s</b>	Second (01-59) đã trôi qua (kèm 00 nếu giây từ 00-59)
<b>S</b>	Thêm 2 ký tự st, nd, rd hay th theo sau ngày dạng 2 ký tự số
<b>t</b>	Trả về tổng số ngày trong tháng từ 28-31
<b>T</b>	Ký tự Timezone của server với 3 ký tự chẳng hạn như EST
<b>U</b>	Tổng số second từ 1 January 1970 tới hôm nay ứng với UNIX Time Stamp
<b>w</b>	Day (0-6) của tuần, 0 ứng với Sunday và 6 ứng với Saturday
<b>y</b>	Năm định dạng 2 con số (11)
<b>Y</b>	Năm định dạng 4 con số (2011)
<b>z</b>	Ngày trong năm, 1 hoặc 2 con số (0-365)
<b>X</b>	Timezone hiện tại tính bằng giây từ -43200 đến 43200

**Câu hỏi ôn tập:**

-Viết trang Register.php với các thông tin như sau:

- Tên đăng nhập
- Mật khẩu
- Xác nhận mật khẩu
- Giới tính
- Ngày sinh
- Địa chỉ
- Button Register

Yêu cầu:

- Các fiel không được rỗng.
- Mật khẩu và xác nhận mật khẩu phải giống nhau
- Ngày sinh theo chuẩn dd-mm-yyyy
- Sau đó lưu tất cả thông tin vào mảng (1 hoặc 2 chiều) và lưu vào session register.
- Cuối cùng in ra thông tin thông qua trong session.

## CHƯƠNG 8

### 8.1 Khai báo HÀM trong PHP:

#### Cú pháp:

```
function TenFunction(Các tham số nếu có)
{
    //các mã lệnh thực thi
}
```

**\*Lưu ý:** Tên hàm phải bắt đầu bằng chữ cái hoặc dấu gạch dưới.

Ví dụ: viết hàm tính tổng 2 số.

```
function fnTinhTong2So($a,$b)
{
    return $a+$b;
}
```

Kiểm tra hàm

```
echo "Tổng 2 số = ".fnTinhTong2So(2,5);
```

⇒ Kết quả ra là: Tổng 2 số = 7

### 8.2 Thống nhất kích thước của mọi trang PHP

+ Khi xây dựng ứng dụng web chuyên nghiệp, điều đầu tiên bạn nên quan tâm là sự thống nhất kích thước của các phần trên trang web. Điều này có nghĩa khi người dùng thay đổi trang web khi duyệt các phần như top,bottom,left,right có kích thước như nhau.

+Tùy theo bố cục của mỗi người mà trang web được chia theo bao nhiêu đó phần. Ở

đây chúng ta sẽ lấy ví dụ trang web gồm 5 phần :

- Top
- Bottom
- Left
- Right
- Body

Phần **Top**: thường trình bày các thuộc tính như quảng cáo (banner), logo (biểu tượng công ty), menu của ứng dụng web đó và 1 số thông tin khác. Phần **Left**: là trình bày thông tin của các menu phụ hay còn gọi là menu của các menu chính, bên cạnh các menu con này trang web thường có các liên kết về liên hệ, quảng cáo, mailing list (đăng ký email), gửi đến bạn bè(send to friend),... Phần **Right**: thường là phần giới thiệu về các thông tin đặc biệt và quảng cáo, chẳng hạn đối với ứng dụng bán sách, phần right thường là danh sách các nhóm sách bán chạy, sắp phát hành,....

Phần **Bottom**: là phần thường trình bày thông tin liên lạc với công ty, người quản trị website và bản quyền. Ngoài ra, phần bottom đôi khi là danh sách các menu con khác.

Phần **Body**: thể hiện nội dung chính của website

Nói chung là tùy thuộc ý tưởng thiết kế bố cục và thiết kế website của mỗi người thì trang website sẽ có sự khác nhau. Tùy theo bố cục của mỗi người mà website có thể có các phần nêu trên.

Vd: đây là bố cục của 1 website gồm 5 phần như nói trên.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 60

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**



Hình ảnh trên chỉ mang tính chất minh họa cho bố cục của website gồm 5 phần: Top, Bottom, Left, Right, Body

Còn đây là đoạn code thể hiện bố cục 5 phần của 1 website như hình trên

```

<!DOCTYPE html PUBLIC "-//W3C//Dtd XHTML 1.0
transitional//EN"
"http://www.w3.org/tr/xhtml1/Dtd/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"
/> <title>Thông nhất kích thước trang PHP</title>
</head>

<body bottommargin="0" leftmargin="0" topmargin="0" rightmargin="0">
<table cellpadding="2" cellspacing="2" width="778" border="1"
align="center"> <tr height="100">
<td align="center" colspan="3">TOP</td>
</tr>
<tr height="280">
<td valign="top" width="20%">LEFT</td>
<td valign="top" width="60%">BODY</td>
<td valign="top" width="20%">RIGHT</td>
</tr>
<tr height="50">
<td colspan="3" align="center">BOTTOM</td>
</tr>
</table>
</body>
</html>

```

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 61

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

**\*Chú ý:** Sử dụng 2 hàm sau để nhúng trang:

- Hàm **include**("Tên trang muốn nhúng vào");
- Hàm **require**("Tên trang muốn nhúng vào");

**Câu hỏi ôn tập:**

Câu 1: Thiết kế layout website sử dụng các tập tin dùng chung theo mẫu sau:



BM02–QT7.3/2

Ngày hiệu lực: 15/9/2009 Trang 62

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

## **Chương 9: Hướng đối tượng trong PHP**

### **9.1 Hướng đối tượng trong PHP**

**+Định nghĩa:** Lập trình hướng đối tượng (gọi tắt là OOP viết tắt của từ Object Oriented Programming) là kỹ thuật lập trình hỗ trợ công nghệ hướng đối tượng. OOP được xem là giúp tăng năng suất, đơn giản hóa độ phức tạp khi bảo trì cũng như là việc mở rộng phần mềm bằng cách cho phép lập trình viên tập trung vào các đối tượng phần mềm ở bậc cao hơn.

Những đối tượng trong một ngôn ngữ OOP là các kết hợp giữa mã và dữ liệu mà chúng được nhìn nhận như là 1 đơn vị duy nhất. Mỗi đối tượng có 1 tên riêng biệt và tất cả các tham chiếu đến đối tượng đó được tiến hành qua tên của nó. Như vậy, mỗi đối tượng có khả năng nhận vào các thông báo, xử lý dữ liệu bên trong của nó, gửi ra hay trả lời đến các đối tượng khác hay đến môi trường.

**+Các tính chất cơ bản của lập trình hướng đối tượng:**

Một đối tượng bao gồm 2 thành phần:

1. Các thuộc tính (hay còn gọi là các biến)
2. Các phương thức (hay còn gọi là các hàm)

Thuộc tính là tập hợp các dữ liệu đặc trưng của 1 đối tượng. Vd: xe máy có đặc trưng như:

1. Xe máy thuộc hãng xe nào
2. Model đời xe máy
3. Màu xe
4. Tốc độ của 1 chiếc xe máy
5. ....

=>Đó là các thuộc tính của 1 chiếc xe máy.

Phương thức là các phương tiện hành động của 1 đối tượng được dùng để sử dụng 1 đối tượng. Vd như trên nhưng phương thức của xe máy là chạy và dừng,....

Lập trình hướng đối tượng là một phương pháp lập trình có các tính chất chính sau:

**Tính trừu tượng (abstraction):** Đây là khả năng của chương trình bỏ qua hay không chú ý đến một số khía cạnh của thông tin mà nó đang trực tiếp làm việc lên, nghĩa là nó có khả năng tập trung vào những cốt lõi cần thiết. Mỗi đối tượng phục vụ như là một "động tử" có thể hoàn tất các công việc một cách nội bộ, báo cáo, thay đổi trạng thái của nó và liên lạc với các đối tượng khác mà không cần cho biết làm cách nào đối tượng tiến hành được các thao tác. Tính chất này thường được gọi là sự trừu tượng của dữ liệu.

Tính trừu tượng còn thể hiện qua việc một đối tượng ban đầu có thể có một số đặc điểm chung cho nhiều đối tượng khác như là sự mở rộng của nó nhưng bản thân đối tượng ban đầu này có thể không có các biện pháp thi hành. Tính trừu tượng này thường được xác định trong khái niệm gọi là lớp trừu tượng hay hay lớp cơ sở trừu tượng.

**Tính đóng gói (encapsulation) và che dấu thông tin (information hiding):** Tính chất này không cho phép người sử dụng các đối tượng thay đổi trạng thái nội tại của một đối tượng. Chỉ có các phương thức nội tại của đối tượng cho phép thay đổi trạng thái của nó. Việc cho phép môi trường bên ngoài tác động lên các dữ liệu nội tại của một đối tượng theo cách nào là hoàn toàn tùy thuộc vào người viết mã. Đây là tính chất đảm bảo sự toàn vẹn của đối tượng.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 63

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

**Tính đa hình (polymorphism):** Thể hiện thông qua việc gửi các thông điệp (message). Việc gửi các thông điệp này có thể so sánh như việc gọi các hàm bên trong của một đối tượng. Các phương thức dùng trả lời cho một thông điệp sẽ tùy theo đối tượng mà thông điệp đó được gửi tới sẽ có phản ứng khác nhau. Người lập trình có thể định nghĩa một đặc tính (chẳng hạn thông qua tên của các phương thức) cho một loạt các đối tượng gần nhau nhưng khi thi hành thì dùng cùng một tên gọi mà sự thi hành của mỗi đối tượng sẽ tự động xảy ra tương ứng theo đặc tính của từng đối tượng mà không bị nhầm lẫn.

Thí dụ khi định nghĩa hai đối tượng "hình\_vuong" và "hình\_tron" thì có một phương thức chung là "chu\_vì". Khi gọi phương thức này thì nếu đối tượng là "hình\_vuong" nó sẽ tính theo công thức khác với khi đối tượng là "hình\_tron".

**Tính kế thừa (inheritance):** Đặc tính này cho phép một đối tượng có thể có sẵn các đặc tính mà đối tượng khác đã có thông qua kế thừa. Điều này cho phép các đối tượng chia sẻ hay mở rộng các đặc tính sẵn có mà không phải tiến hành định nghĩa lại. Tuy nhiên, không phải ngôn ngữ định hướng đối tượng nào cũng có tính chất này.

## **Lớp (class)**

Một lớp được hiểu là một kiểu dữ liệu bao gồm các thuộc tính và các phương thức được định nghĩa từ trước. Đây là sự trừu tượng hóa của đối tượng. Một đối tượng sẽ được xác lập khi nó được thực thể hóa từ một lớp. Khác với kiểu dữ liệu thông thường, một lớp là một đơn vị (trừu tượng) bao gồm sự kết hợp giữa các phương thức và các thuộc tính. Để có một đối tượng (mà có thể được xem như là một biến) hoạt động được thì việc thực thể hóa sẽ có thể bao gồm việc cài đặt các giá trị ban đầu của các thuộc tính cũng như việc đăng ký bộ nhớ, mà công việc này thường được giao cho các phương thức gọi là "máy kết cấu" (constructor) hay hàm dựng. Ngược lại khi một đối tượng thuộc về một lớp không còn sử dụng nữa thì cũng có thể có một phương thức để xử lý gọi là "máy hủy diệt" (destructor) hay hàm hủy.

Như vậy, để có được các đối tượng thì người lập trình OOP cần phải thiết kế lớp của các đối tượng đó bằng cách xây dựng các thuộc tính và các phương thức có các đặc tính riêng.

Mỗi một phương thức hay một thuộc tính đầy đủ của một lớp còn được gọi tên là một thành viên (member) của lớp đó.

## **Lớp con (subclass)**

Lớp con là một lớp thông thường nhưng có thêm tính chất kế thừa một phần hay toàn bộ các đặc tính của một lớp khác. Lớp mà chia sẻ sự kế thừa gọi là lớp phụ mẫu (parent class).

## **Lớp trừu tượng hay lớp cơ sở trừu tượng (abstract class)**

Lớp trừu tượng là một lớp mà nó không thể thực thể hóa thành một đối tượng thực dụng được. Lớp này được thiết kế nhằm tạo ra một lớp có các đặc tính tổng quát nhưng bản thân lớp đó chưa có ý nghĩa (hay không đủ ý nghĩa) để có thể tiến hành viết mã cho việc thực thể hóa. (xem thí dụ)

Thí dụ: Lớp "hình\_phang" được định nghĩa không có dữ liệu nội tại và chỉ có các phương thức (hàm nội tại) "tính\_chu\_vi", "tính\_dien\_tich". Nhưng vì lớp hình\_phẳng này chưa xác

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 64

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

định được đầy đủ các đặc tính của nó (cụ thể các biến nội tại là tọa độ các đỉnh nếu là đa giác, là đường bán kính và tọa độ tâm nếu là hình tròn, ...) nên nó chỉ có thể được viết thành một lớp trừu tượng. Sau đó, người lập trình có thể tạo ra các lớp con chẳng hạn như là lớp "tam\_giac", lớp "hình\_tron", lớp "tu\_giac",.... Và trong các lớp con này người viết mã sẽ cung cấp các dữ liệu nội tại (như là biến nội tại r làm bán kính và hằng số nội tại Pi cho lớp "hình\_tron" và sau đó viết mã cụ thể cho các phương thức "tính\_chu\_vi" và "tính\_dien\_tich").

## **Phương Thức (method)**

Là hàm nội tại của một lớp (hay một đối tượng). Tùy theo đặc tính mà người lập trình gán cho, một phương pháp có thể chỉ được gọi bên trong các hàm khác của lớp đó, có thể cho phép các câu lệnh bên ngoài lớp gọi tới nó, hay chỉ cho phép các lớp có quan hệ đặc biệt như là quan hệ lớp con, và quan hệ bạn bè (friend) được phép gọi tới nó. Mỗi phương pháp đều có thể có kiểu trả về, chúng có thể trả các kiểu dữ liệu cổ điển hay trả về một kiểu là một lớp đã được định nghĩa từ trước. Một tên gọi khác của phương pháp của một lớp là hàm thành viên.

Người ta còn định nghĩa thêm vài loại phương pháp đặc biệt:

Hàm dựng (constructor) là hàm được dùng để cài đặt các giá trị ban đầu cho các biến nội tại và đôi khi còn dùng để khai báo về việc xử dụng bộ nhớ.

Hàm hủy (destructor) là hàm dùng vào việc làm sạch bộ nhớ và hủy bỏ tên của một đối tượng sau khi đã dùng xong, trong đó có thể bao gồm cả việc xóa các con trỏ nội tại và trả về các phần bộ nhớ mà đối tượng đã dùng.

Trong một số trường hợp thì hàm hủy hay hàm dựng có thể được tự động hóa bởi ngôn ngữ OOP như là trường hợp của Visual C++, C#.

Tiện ích (utility) là các hàm chỉ hoạt động bên trong của một lớp mà không cho phép môi trường bên ngoài gọi tới. Các hàm này có thể là những tính toán trung gian nội bộ của một đối tượng mà xét thấy không cần thiết phải cho thế giới bên ngoài của đối tượng biết là gì.

### **Thuộc tính (attribute)**

Thuộc tính của một lớp bao gồm các biến, các hằng, hay tham số nội tại của lớp đó. Ở đây, vai trò quan trọng nhất của các thuộc tính là các biến vì chúng sẽ có thể bị thay đổi trong suốt quá trình hoạt động của một đối tượng. Các thuộc tính có thể được xác định kiểu và kiểu của chúng có thể là các kiểu dữ liệu cổ điển hay đó là một lớp đã định nghĩa từ trước. Như đã ghi, khi một lớp đã được thực thể hoá thành đối tượng cụ thể thì tập hợp các giá trị của các biến nội tại làm thành trạng thái của đối tượng. Giống như trường hợp của phương pháp, tùy theo người viết mã, biến nội tại có thể chỉ được dùng bên trong các phương pháp của chính lớp đó, có thể cho phép các câu lệnh bên ngoài lớp, hay chỉ cho phép các lớp có quan hệ đặc biệt như là quan hệ lớp con, (và quan hệ bạn bè (friend) trong C++) được phép dùng tới nó (hay thay đổi giá trị của nó). Mỗi thuộc tính của một lớp còn được gọi là thành viên dữ liệu của lớp đó.

### **Thực thể (instance)**

Thực thể hóa (instantiate) là quá trình khai báo để có một tên (có thể được xem như là một biến) trở thành một đối tượng từ một lớp nào đó.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 65

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

Một lớp sau khi được tiến hành thực thể hóa để có một đối tượng cụ thể gọi là một thực thể. Hay nói ngược lại một thực thể là một đối tượng riêng lẻ của một lớp đã định trước. Như các biến thông thường, hai thực thể của cùng một lớp có thể có trạng thái nội tại khác nhau (xác định bởi các giá trị hiện có của các biến nội tại) và do đó hoàn toàn độc lập nhau nếu không có yêu cầu gì đặc biệt từ người lập trình.



## Công cộng (public)

Công cộng là một tính chất được dùng để gán cho các phương pháp, các biến nội tại, hay các lớp mà khi khai báo thì người lập trình đã cho phép các câu lệnh bên ngoài cũng như các đối tượng khác được phép dùng đến nó.

Thí dụ: Trong C++ khai báo public: `int my_var;` thì biến `my_var` có hai tính chất là tính công cộng và là một integer cả hai tính chất này hợp thành đặc tính của biến `my_var` khiến nó có thể được xử dụng hay thay đổi giá trị của nó (bởi các câu lệnh) ở mọi nơi bên ngoài lẫn bên trong của lớp.

## Riêng tư (private)

Riêng tư là sự thể hiện tính chất đóng mạnh nhất (của một đặc tính hay một lớp) <http://hocvui.net> . Khi dùng tính chất này gán cho một biến, một phương pháp thì biến hay phương pháp đó chỉ có thể được sử dụng bên trong của lớp mà chúng được định nghĩa. Mọi nỗ lực dùng đến chúng từ bên ngoài qua các câu lệnh hay từ các lớp con sẽ bị phủ nhận hay bị lỗi.

## Bảo tồn (protected)

Tùy theo ngôn ngữ, sẽ có vài điểm nhỏ khác nhau về cách hiểu tính chất này. Nhìn chung đây là tính chất mà khi dùng để áp dụng cho các phương pháp, các biến nội tại, hay các lớp thì chỉ có trong nội bộ của lớp đó hay các lớp con của nó (hay trong nội bộ một gói như trong Java) được phép gọi đến hay dùng đến các phương pháp, biến hay lớp đó.

So với tính chất riêng tư thì tính bảo tồn rộng rãi hơn về nghĩa chia sẻ dữ liệu hay chức năng. Nó cho phép một số trường hợp được dùng tới các đặc tính của một lớp (từ một lớp con chẳng hạn).

Lưu ý: Các tính chất công cộng, riêng tư và bảo tồn đôi khi còn được dùng để chỉ thị cho một lớp con cách thức kế thừa một lớp cha mẹ như trong C++.

## Đa kế thừa (multiple inheritance)

Đây là một tính chất cho phép một lớp con có khả năng kế thừa trực tiếp cùng lúc nhiều lớp khác.

Vài điểm cần lưu ý khi viết mã dùng tính chất đa kế thừa:

Khi muốn có một sự kế thừa từ nhiều lớp phụ mẫu thì các lớp này cần phải độc lập và đặc biệt tên của các dữ liệu hay hàm cho phép kế thừa phải có tên khác nhau để tránh lỗi "ambiguity". Bởi vì lúc đó phần mềm chuyển dịch sẽ không thể xác định được là lớp con sẽ thừa kế tên nào của các lớp phụ mẫu.

Không phải ngôn ngữ OOP loại phân lớp nào cũng hỗ trợ cho tính chất này.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 66

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

Ngoài các khái niệm trên, tùy theo ngôn ngữ, có thể sẽ có các chức năng OOP riêng biệt được cấp thêm vào.

Khai báo class và thể hiện của class trong PHP: 1 lớp bao gồm các thuộc tính và phương thức.

```
class tên_lớp  
{  
    //danh sách các biến, hằng, lớp, ...(thuộc tính)  
    //danh sách các hàm (phương thức)  
}
```

Trong đó, các lớp được khai báo thông qua từ khóa Class, các thuộc tính được khai báo dưới dạng các biến, còn các phương thức được xây dựng dưới dạng các hàm. Các thuộc tính và phương thức có thể thiết lập các tính chất đặc biệt như : private, public, ... Các tính chất này được đặt trước các khai báo thuộc tính và phương thức. Vd: tạo 1 lớp sinhvien có các thuộc tính public như masv, tensv,...

```
class sinhvien  
{  
    public $masv;  
    public $tensv;  
}
```

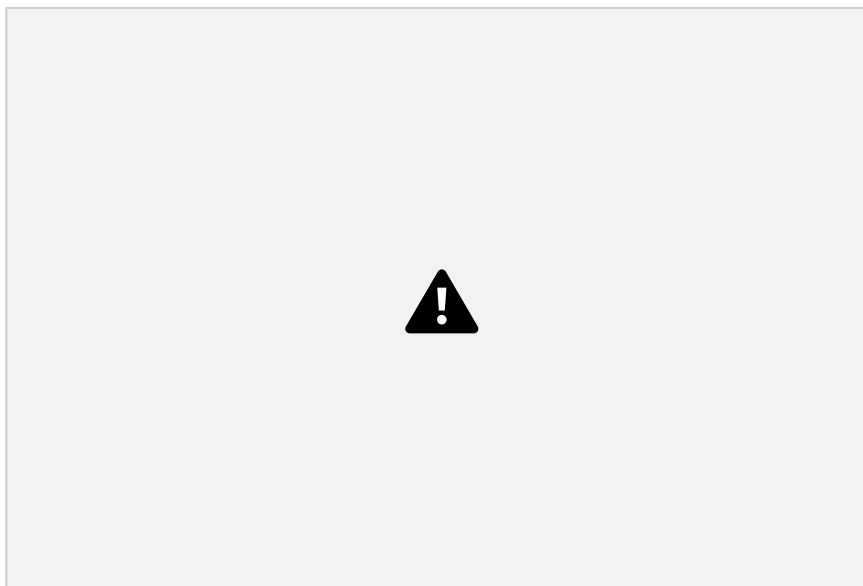
Để khai báo 1 thể hiện của 1 lớp, chúng ta dùng từ khóa new như cú pháp sau:

```
$ten_thuc_the = new Ten_lop();
```

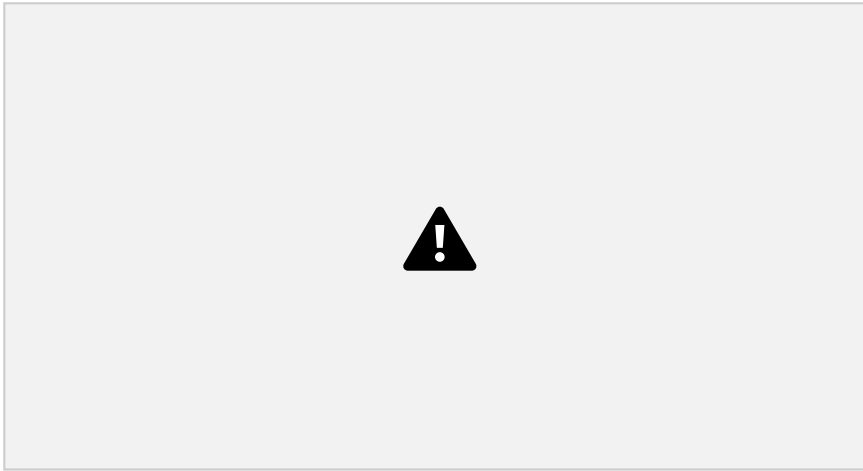
Để truy cập vào các thuộc tính hay phương thức của lớp đó, ta dùng toán tử -> với cú pháp như sau:

```
$ten_thuc_the->ten_thuoc_tinh;
```

Ví dụ sau sẽ tạo 1 lớp như ví dụ trên và sẽ thiết lập thêm các thuộc tính của lớp này:



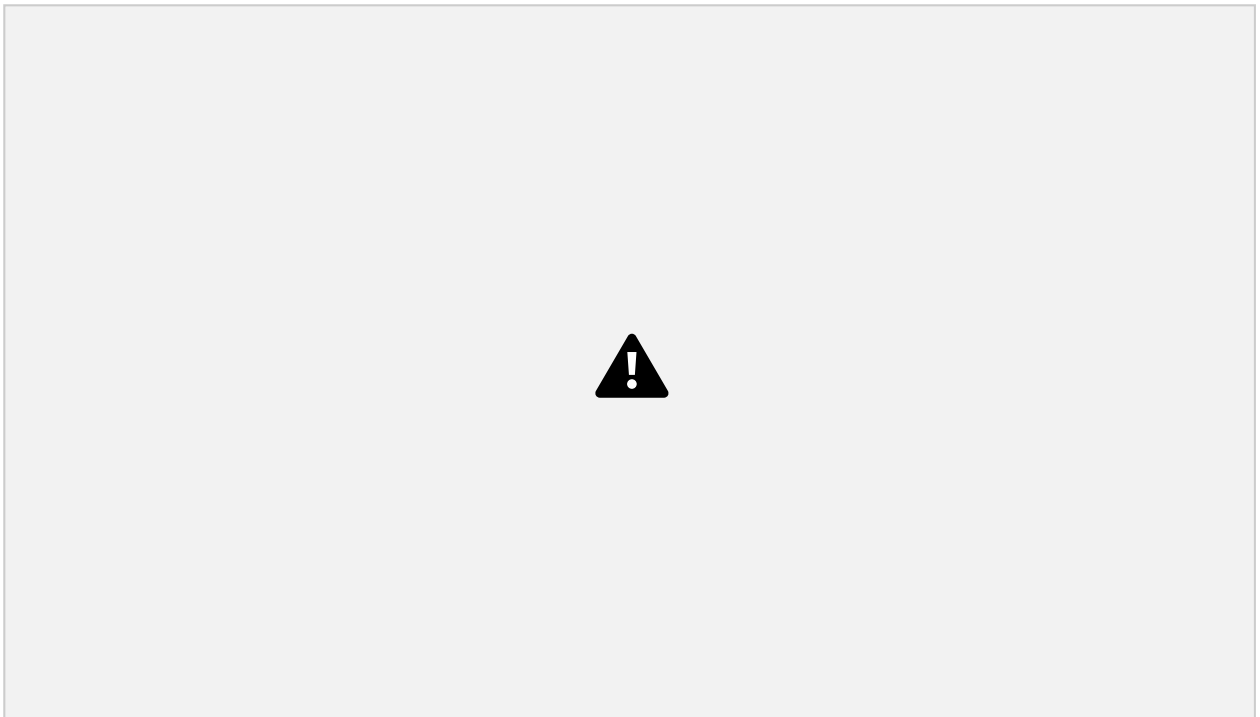
Kết quả:



**+Cú pháp tạo phương thức:**

```
[Kiểu truy xuất] function Ten_Phuong_Thuc(danh sách các tham số)
{
    //xử lý
}
```

**Vd:** tạo thêm phương thức thiết lập thông tin cho sinh viên có 2 tham số nhận vào là masv và tensv với tên phương thức là thiếtLapThongTin()



Hoặc có thể sử dụng construct được thiết lập của PHP. Cú pháp hàm khởi tạo như hình như:



Chúng ta có thể đưa tham số vào như hình hoặc tùy theo trường hợp mà sử dụng

```
function __construct(danh sách tham số nếu có)  
{  
    //mã thực thi  
}
```

Lập trình hướng đối tượng tập trung vào việc đóng gói các thuộc tính và phương thức của 1 đối tượng nào đó. Trong lập trình hướng đối tượng thì các thành viên của 1 lớp cần được xác định xem chúng có thể được truy xuất từ đâu. Có 3 cách truy xuất như sau: 1. Kiểu truy xuất public (công khai): nếu phương thức hay thuộc tính nào mang giá trị này thì nó được truy xuất tự do bên ngoài lớp và bên trong nó.

2. Kiểu truy xuất private (riêng tư): thì chỉ truy xuất được chính bên trong lớp đó 3. Kiểu truy xuất protected (bảo vệ): truy xuất được chính bên trong lớp đó và lớp thừa kế từ lớp đó.

Vd ta xây dựng lớp thành viên để quản lý thành viên, trong đó biến ID thành viên là không cần thiết xuất ra trình duyệt và đó là thông tin quan trọng trong cơ sở dữ liệu nên ta khai báo nó ở dạng private để tránh sự thay đổi giá trị không mong muốn do sự sơ xuất trong lập trình. Cách dùng các khoá bạn xem trong VD sau:

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 69

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



Lưu ý: phương thức, thuộc tính protected trong lớp cơ sở chỉ được thừa kế trong nội bộ của

lớp thừa kế, do đó chỉ các hàm trong lớp thừa kế mới có quyền gọi chúng, còn đối tượng của lớp thừa kế thì không thể gọi các \$ và hàm protected trong lớp cơ sở.

**+Thuộc tính hằng, thuộc tính tĩnh và phương thức tĩnh:** Trong PHP, thuộc tính hằng, phương thức tĩnh và thuộc tính tĩnh tồn tại ngay cả khi chưa có đối tượng nào được gọi. Thuộc tính hằng dùng để tạo ra một hằng số và giá trị của nó không thể thay đổi. Còn thuộc tính và phương thức tĩnh tồn tại ngay sau khi nó được định nghĩa (nghĩa là không cần gọi lớp vẫn có thể dùng dc).

Lưu ý: Phương thức tĩnh không được tác động và truy xuất đến các thuộc tính không tĩnh. Có thể truy xuất đến thuộc tính tĩnh. Có vẻ hơi trừu tượng, chúng ta hãy đi vào VD sau:

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 70

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***



Như vậy để sử dụng hằng, thuộc tính tĩnh và phương thức tĩnh ta dùng cú pháp sau: tên\_lớp::tên hằng/thuộc tính tĩnh/phương thức tĩnh. Nếu muốn truy cập thuộc tính tĩnh hoặc hằng trong một hàm của chính lớp đó thì thay vì dùng \$this-> ta có thể dùng self::tên hằng/thuộc tính tĩnh/phương thức tĩnh.

Khi nào thì dùng đến những thứ này? Trong một project web sẽ có lúc bạn phải dùng đến nó ví dụ như hàm kiểm tra sự chính xác của email, username nhập vào trước khi cho nó vào lớp...

**+Dẫn xuất, thừa kế, lớp cơ sở, chạy đề phương thức:**

Nếu lớp B thừa kế từ lớp A thì lớp B gọi là lớp dẫn xuất, lớp A là lớp cơ sở. Lớp dẫn xuất có thể sử dụng tất cả các phương thức và thuộc tính public ,protected của lớp cơ sở. Khi nào thì cần dùng thừa kế? Khi lớp B sử dụng một số phương thức và tính mà lớp A đã định nghĩa sẵn thì ta nên cho B thừa kế của A, chỉ khai báo trong B thêm những cái mới mà trong A không có.

Ví dụ: trên đây ta có class nguoi, giờ nếu ta khai báo class công nhân. Thì mỗi công nhân cũng phải có tên tuổi, mà hai yếu tố này ta đã khai báo, “xào nấu” sẵn trong class nguoi. Vậy thì ta hãy cho class cong\_nhan thừa kế class nguoi đi, khỏi khai báo thêm chi cho mệt!!!

Để cho class cong\_nhan thừa kế từ class nguoi ta theo cú pháp sau: class cong\_nhan extends nguoi { ... }

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 71

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***





Trong lớp dẫn xuất mà muốn truy xuất giá trị của thuộc tính tĩnh, hằng trong lớp cơ sở ta dùng cú pháp parent:: tên hằng/thuộc tính tĩnh/phương thức tĩnh.

Khi trong lớp dẫn xuất mà có phương thức trùng tên với phương thức trong lớp cơ sở thì sẽ ưu tiên chạy phương thức trong lớp dẫn xuất gọi là chạy đè phương thức. Ví dụ đơn giản như trong lớp cơ sở có hàm \*\*\* dùng để đặt tên là “chó” nhưng trong lớp dẫn xuất cũng có hàm \*\*\* và hàm này lại đặt tên là “mèo”. vậy khi chạy thì kết quả sẽ là “mèo”.

**+Truy cập phương thức trong lớp cơ sở:**

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 72

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***







Trong lớp dẫn xuất, nếu ta muốn truy cập các phương thức (kể cả phương thức tĩnh) trong lớp cơ sở thì theo cú pháp: `tên_lớp_csở::tên_hàm` hoặc `parent::tên_hàm`.

### **Câu hỏi ôn tập:**

- Tạo form register.php với thông tin:

- Tên người dùng
- Mật khẩu

Nhưng lần này sử dụng hướng đối tượng với đối tượng là người dùng bao gồm các thông tin trên.

Viết thêm các phương thức: kiểm tra người dùng đã tồn tại chưa (vì không có database nên tạm thời dùng mảng tạo sẵn thông tin của vài người dùng được kiểm tra) và các phương get và set thông tin người dùng vào trong 1 đối tượng.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 73

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL CHƯƠNG***

**10**

### **10.1 Giới thiệu CSDL MySQL**

MySQL là 1 phần mềm quản trị CSDL mã nguồn mở, miễn phí nằm trong nhóm LAMP(Linux- Apache – MySQL - PHP)

### **10.2 Cài đặt MySQL**

Phần cài đặt này đã được thực hiện khi chúng ta cài đặt wampserver.

### **10.3 Tạo CSDL & người dùng**

**Quản lý người dùng:**



BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 74

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

**+Tạo mới người dùng trong MySQL và Cấp quyền cho người dùng:**



**+Xóa quyền user**





## 10.4 Kiểu dữ liệu trong MySQL

### Numeric

+Các kiểu dữ liệu số nguyên:

Kiểu dữ liệu	Int	2 bytes
Tinyint	Bigint	3 bytes
Smallint	Kích thước	4 bytes
Mediumint	1 byte	8 bytes

Miền	-327
------	------

$-2^{31} \Rightarrow 2^{31} - 1$ hay
--------------------------------------

-127	-838 0..16
------	---------------

$-2^{63} \Rightarrow 2^{63} - 1$ hay
--------------------------------------

**+Kiểu dữ liệu true/false:**

Kiểu dữ liệu	Kích thước
--------------	------------

Miền giá trị
--------------

Bool/Boolean	1 byte
--------------	--------

Có hai giá trị là '0' và '1'
------------------------------

**+Kiểu dữ liệu dạng số thập phân: decimal và numeric**

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 76

***Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL***

Kiểu dữ liệu	Kích thước
--------------	------------

Miền giá trị
--------------

Decimal/Numeric	4 bytes
-----------------	---------

Có hai giá trị là '0' và '1'
------------------------------

**+Kiểu dữ liệu kiểu số thực:**

Kiểu dữ liệu	Kích thước
--------------	------------

Float	4 bytes
-------	---------

-3.402823466E+38 -1.175494351E -38 1.175494351E -38 3.402823466E +38
---

Double	8 bytes
--------	---------

-1.797693134862E+308 $\Rightarrow$ -2.225073858507367E+308; 0; 2.225073858507367E+308 $\Rightarrow$ 1.797693134862E+308
--

Miền
------

**+Kiểu dữ liệu dạng ngày giờ (DateTime)**

Kiểu dữ liệu	'1000-01-01' => '9999-12-31'	Miền giá trị
Date		Ngày với định dạng yyyy-mm dd
Datetime	'1000-01-01 00:00:00' => '9999- 12-31 23:59:59'	Ngày giờ với định dạng yyyy mm-dd hh:mm:ss
Time	'00:00:00' => '23:59:59'	Giờ với định dạng hh:mm:ss
Year[(2 4)]	4 ký số: '1901' => '2155' 2 ký số:	Năm với định dạng hoặc 4 ký số
Timestamp [(Kích cỡ định dạng)]	'1970' => '2069'	Timestamp trình bày dạng yyyy-mm-dd hh:mm:ss
Kích thước	'1970-01-01 00:00:01'	

### +Các Kiểu dữ liệu chuỗi

	Blob	Kích thước
Kiểu dữ liệu		
Char	Text	1 => 255 ký tự
Varchar	Mediumblob	1 => 255 ký tự
Tinyblob	Mediumtext	1 => 2 <sup>8</sup> -1 bytes (255 bytes)
Tinytext	Longblob	1 => 2 <sup>8</sup> -1 ký tự (255 ký tự)

1 => 2 <sup>16</sup> -1 bytes (65535 bytes)	Miền	Kiểu blob cỡ 65535
1 => 2 <sup>16</sup> -1 ký tự (65535 ký tự)	Chuỗi	Kiểu chuỗi dạng cỡ 65535 ký tự
1 => 2 <sup>24</sup> -1 bytes (16777215 bytes)	Chuỗi	Kiểu blob vừa cỡ ký tự
1 => 2 <sup>24</sup> -1 ký tự (16777215 ký tự)	Kiểu 255	Kiểu chuỗi dạng vừa 16777215 ký
1 => 2 <sup>32</sup> -1 bytes (4GB)	Kiểu cỡ 4GB	Kiểu blob lớn cỡ 4GB

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 77

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

Longtext 1 => 2 <sup>32</sup> -1 ký tự (4GB)	Kiểu chuỗi dạng văn bản lớn khoảng 4GB ký tự
---	---

## 10.5 Câu lệnh SQL

Nhóm phát biểu SQL

SQL dạng thay đổi và định nghĩa CSDL

**Phát biểu SQL dạng CREATE** : phát biểu này dùng để tạo mới 1 CSDL và bảng.

**Tạo CSDL - Create Database:** dùng để tạo mới cơ sở dữ liệu. Cú pháp như sau:

create database TenDatabaseMuốnTạo

Tạo bảng-Create table theo cú pháp sau:

**Create table**

```
(
    TenCot KieuDulieu,
    ...
)
```

**Chú ý:** Cuối cột cuối cùng không có dấu phẩy

Một số qui định khi thiết kế table: (có thể thêm nhiều giá trị theo sau nhưng phải theo thứ tự như sau)

Tên cột

Kiểu dữ liệu

Khóa chính hoặc khóa ngoại(Primary key/Foreign key)

Giá trị mặc định(default)

Số tự động (auto\_increment)

Null/Not Null

Thay đổi cấu trúc đối tượng với **ALTER**: là cú pháp thay đổi.

+**Đổi tên table**: **alter table** TenTable **rename** TenTableMoi

+**Thêm cột** vào table:

**Alter alter** TenTable **add column** TenCot KieuDuLieu

Nếu muốn thêm 1 cột mới vào cột đầu tiên thì ta sử dụng cú pháp:

**Alter table** TenTable **add column** TenCot KieuDuLieu **first**

Nếu muốn thêm 1 cột mới vào sau 1 cột nào đó thì ta sử dụng cú pháp: **Alter table** TenTable **add column** TenCot KieuDuLieu **after** TenCotKhac **Xóa cột** trong table: ta sử dụng cú pháp

**Alter table** TenTable **drop column** TenCot

Phát biểu SQL dạng **DROP**: là cú pháp dùng để **xóa** bảng hoặc database

+**Xóa database**: cú pháp

**Drop database** TenDatabase

+**Xóa bảng table**: cú pháp

**Drop table** TenTable

Phát biểu SQL thao tác dữ liệu:

**Khái niệm cơ bản về select:**

+ Thao tác dữ liệu là điều mà chúng ta sẽ làm thường xuyên. Để thao tác lấy dữ liệu thì chúng ta phải sử dụng cú pháp **select**, cú pháp này sẽ lấy dữ liệu theo yêu cầu mà chúng ta muốn lấy.

Vd: **select \*** => **select** toàn bộ dữ liệu, ta dùng **\***

**Select** TenCot => **select** cụ thể tên của cột mà ta muốn lấy.

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 78

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

**Phát biểu Select với mệnh đề From:** khi sử dụng phát biểu select thì phát biểu này thường đi cùng từ khóa **From**. Từ khóa **From** này chỉ ra nguồn bảng mà chúng ta muốn select dữ liệu.

vd: ta có bảng SinhVien gồm các cột: id,name,age và yêu cầu là select ra tên và tuổi của sinhvien từ bảng SinhVien. Cú pháp như sau:

**Select** name,age **from** SinhVien

**Phát biểu SQL dạng Select với mệnh đề Where:** phát biểu select đi chung với mệnh đề **where** để chỉ ra rõ điều kiện mà chúng ta muốn **select**. Ví dụ ta sử dụng lại bảng SinhVien như vd ở trên và đã có nhập sẵn dữ liệu. Trong dữ liệu đã nhập thì bảng SinhVien này chỉ có 2 sinh viên là có tuổi 18. Bây giờ chúng ta chỉ viết câu **select** để lấy sinh viên chỉ 18 tuổi. Cú pháp sẽ như sau:

**Select \*** **from** SinhVien **where** age = 18

*Chú ý: Nếu có từ khóa **where** đằng sau thì câu select sẽ lọc để thỏa điều kiện đi theo sau mệnh đề **where***

**Mệnh đề Order By:** là mệnh đề dùng để sắp xếp dữ liệu tăng dần hoặc giảm dần theo 1 cột nào đó mà ta quy định.

Vd1: Lấy toàn bộ danh sách sinh viên và sắp xếp tăng dần theo tuổi: **Select \*** **from** SinhVien **order by** age **asc**

*Chú ý: cú pháp **order by** TenCot **KieuSapXep**. Ở đây **asc** là tăng dần, ngược lại **desc** là giảm dần.*



Vd2: Lấy danh sách sinh viên theo điều kiện là tuổi phải lớn hơn 18 và sắp xếp giảm dần theo id.

**Select \* from SinhVien where age > 18 order by id desc**

**Mệnh đề Group By:** Mệnh đề **group by** được thêm vào SQL bởi vì các hàm tập hợp (như Sum) trả về 1 tập hợp các giá trị trong cột mỗi khi chúng được gọi. Và nếu không có **group by** ta không thể nào tính được tổng của các giá trị theo từng nhóm riêng lẻ trong cột. Cú pháp **group by** như sau:

**Select TenCot, SUM(TenCot) from TenBang Group By TenCot**

Vd: ta có bảng Sales như sau:

<b>Company</b>	<b>Amount</b>
W3Schools	5500
IBM	4500
W3Schools	7100

**Query 1: Select company,sum(amount) from sales**

Câu query trên sẽ trả về kết quả như hình bên dưới:

<b>Company</b>	<b>SUM(Amount)</b>
W3Schools	17100
IBM	17100
W3Schools	17100

**Query 2: select company,sum(amount) from sales group by company**

Câu query trên sẽ trả về kết quả như hình bên dưới:

<b>Company</b>	<b>SUM(Amount)</b>
W3Schools	12600
IBM	4500

BM02–QT7.3/2 Ngày hiệu lực: 15/9/2009 Trang 79

**Tài liệu giảng dạy Khoa: CNTT – Môn : Lập trình Web PHP & MySQL**

**Mệnh đề HAVING:** được thêm vào SQL vì mệnh đề where không áp dụng được đối với các hàm tập hợp (như sum). Nếu không có **HAVING** ta không thể nào kiểm tra được điều kiện đối với các hàm tập hợp.

Cú pháp **HAVING**:

**Select TenCot, Sum(TenCot)**  
**from TenBang**  
**group by TenCot**  
**Having Sum(TenCot) DieuKienGiaTri**

VD: sử dụng lại bảng sales ở ví dụ trên.

**Select company,sum(amount)**  
**From sales**  
**Group by company**  
**Having sum(amount) > 10000**

<b>Company</b>	<b>SUM(Amount)</b>
<b>W3Schools</b>	12600

## 10.6 Các hàm thông dụng trong MySQL

### Các hàm trong Group By

+**MySQL – Count** : hàm này trả về số lượng mẫu tin trong câu truy vấn trên bảng.

<b>Company</b>	<b>Amount</b>
<b>W3Schools</b>	5500
<b>IBM</b>	4500
<b>W3Schools</b>	7100

Vd: đếm số mẫu tin trên bảng Sales này

Select count(\*) from sales

+**MySQL – AVG** : hàm trả về giá trị bình quân của cột hay trường (column) trong câu truy vấn.

Vd : select avg(amount) from sales

+**MySQL – MIN**: hàm trả về giá trị nhỏ nhất của cột hay trường trong câu truy vấn.

Vd: select min(amount) from sales

+**MySQL – MAX**: hàm trả về giá trị lớn nhất của cột hay trường trong câu truy vấn .

Vd : select max(amount) from sales

+**MySQL – SUM** : hàm trả về tổng giá trị của trường (column), cột trong câu truy vấn.

### Các hàm xử lý chuỗi:

+**Hàm ASCII** : hàm trả về giá trị mã ASCII của ký tự bên trái của chuỗi. Vd: select ASCII('TOI') => Kết quả là 84

+**Hàm Char** : hàm này chuyển đổi kiểu mã ASCII từ số nguyên sang dạng chuỗi.

Vd: select char(35) => kết quả : #