

Chương 6

Lập trình WEB ở Server side

Giảng viên: Nguyễn Hữu Hiếu
huuhieubk@gmail.com

TP.HCM

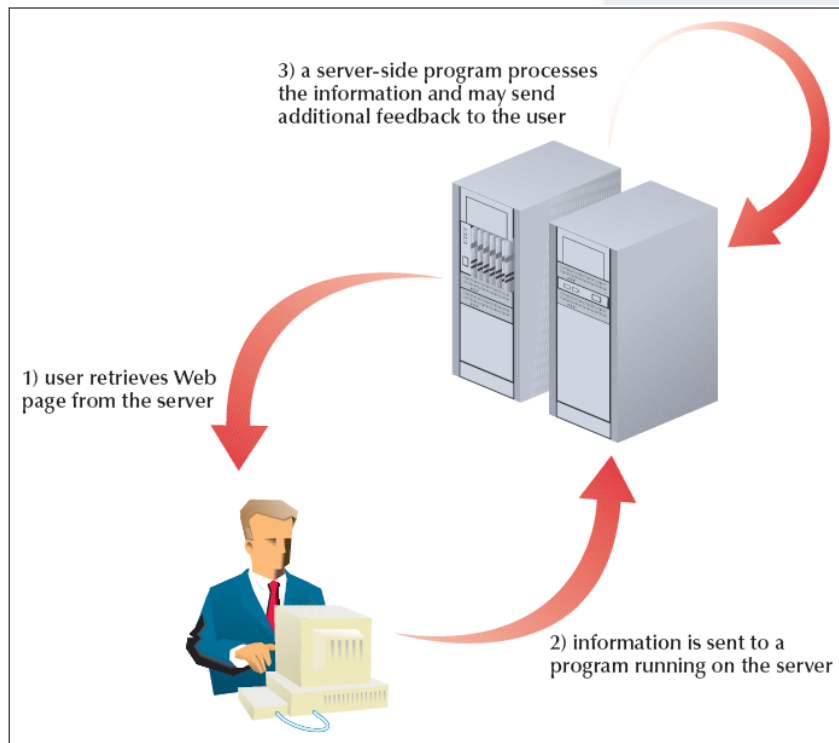
Nội dung

- Giới thiệu các ngôn ngữ lập trình web ở Server side
- Lập trình web với CGI, Servlet, JSP, ASP, PHP
- Giới thiệu cách truy xuất dữ liệu với cơ sở dữ liệu

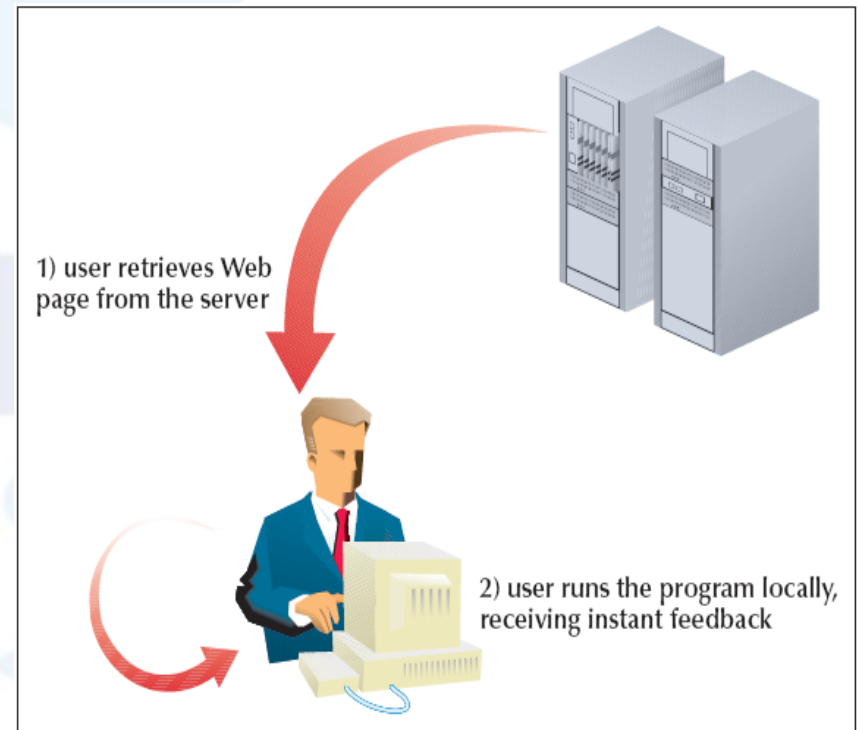


Lập Trình WEB ở Server vs. ở Client

Server-Side Programming



Client-Side Programming



Giới thiệu các ngôn ngữ phía Server



Một số ngôn ngữ dùng cho phía Server

- ASP
- Java via JavaServer Pages (*.jsp)
- Perl CGI (*.cgi, *.ipl, *.pl)
- PHP (*.php)
- ...



So sánh một số ngôn ngữ (1)

PHP	ASP.NET	ASP	JSP/Java
Tốc độ xử lý nhanh, hiệu quả cao	Tốc độ xử lý nhanh, hiệu quả cao	Tốc độ xử lý rất chậm	Hơi chậm
Ngôn ngữ free không tốn chi phí mua bản quyền	Chi phí giá thành cao (phải mua bản quyền)	Chi phí giá thành trung bình (do một phần phải mua bản quyền)	Chi phí giá thành cao
Thời gian code và triển khai nhanh, đơn giản	Thời gian code và triển khai hơi phức tạp, chậm hơn PHP	Thời gian code và triển khai trung bình	Thời gian code và triển khai hơi phức tạp, chậm hơn PHP

So sánh một số ngôn ngữ (2)

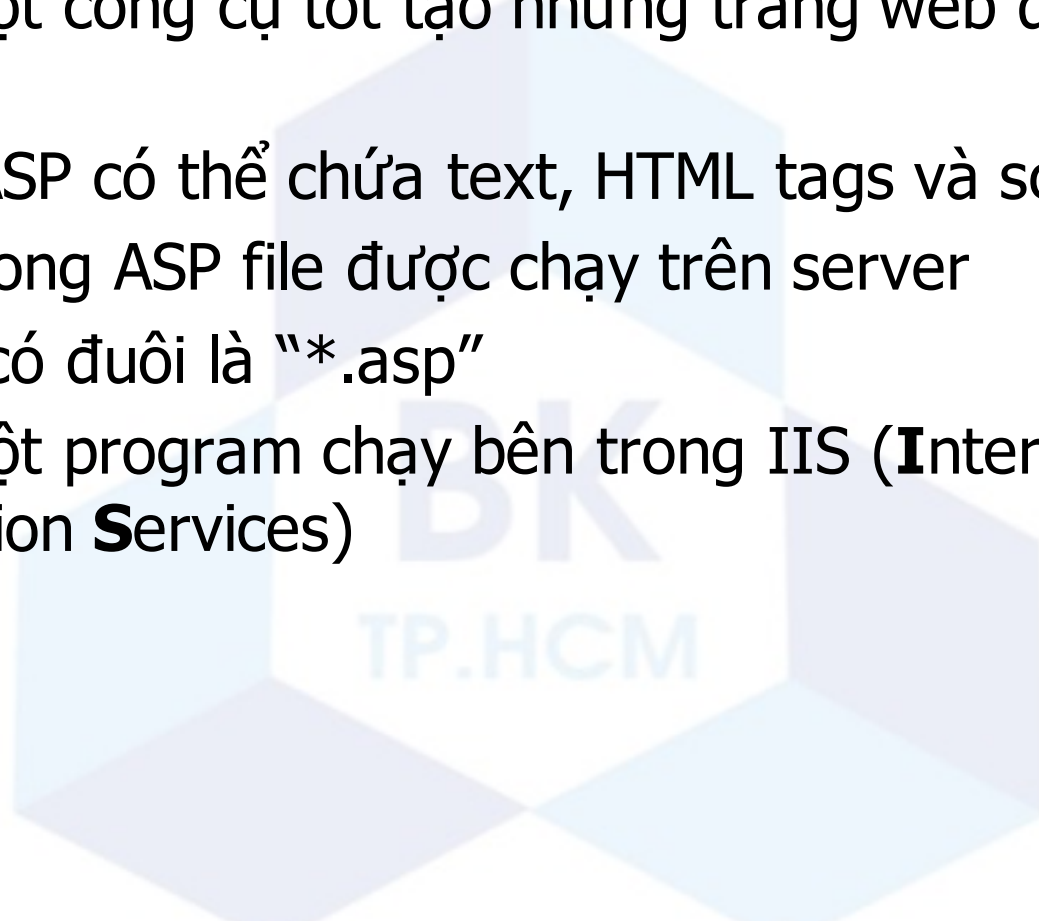
PHP	ASP.NET	ASP	JSP/Java
Số lượng nhà cung cấp hosting nhiều, dễ lựa chọn	Số lượng nhà cung cấp hosting không nhiều, vì vậy khó cho việc lựa chọn	Số lượng nhà cung cấp hosting không nhiều, vì vậy khó cho việc lựa chọn	Số lượng nhà cung cấp hosting ít, khó tìm
Số nhà cung cấp website nhiều, khả năng chọn lựa một trang web phù hợp tốt hơn	Số nhà cung cấp website nhiều, khả năng chọn lựa một trang web phù hợp tốt hơn	Số nhà cung cấp website ít	Số nhà cung cấp website ít, khả năng chọn lựa để xây dựng 1 trang web phù hợp rất khó

So sánh một số ngôn ngữ (3)

PHP	ASP.NET	ASP	JSP/Java
Khả năng mở rộng và phát triển dễ dàng và nhanh chóng	Mở rộng và phát triển website dễ dàng	Khả năng mở rộng và phát triển website khó khăn	Khả năng mở rộng và phát triển website khó khăn
Các công cụ và công nghệ hỗ trợ phong phú, đa dạng. Phát triển web trên nền web 2.0. Sử dụng công nghệ Ajax làm cho quá trình duyệt web nhanh chóng và thân thiện hơn đối với người dùng.	Các công cụ và công nghệ hỗ trợ phong phú, đa dạng. Phát triển web trên nền web 2.0. Sử dụng công nghệ Ajax làm cho quá trình duyệt web nhanh chóng và thân thiện hơn đối với người dùng.	Công cụ và công nghệ hỗ trợ ít, khó tìm.	Công cụ và công nghệ hỗ trợ ít, khó tìm.

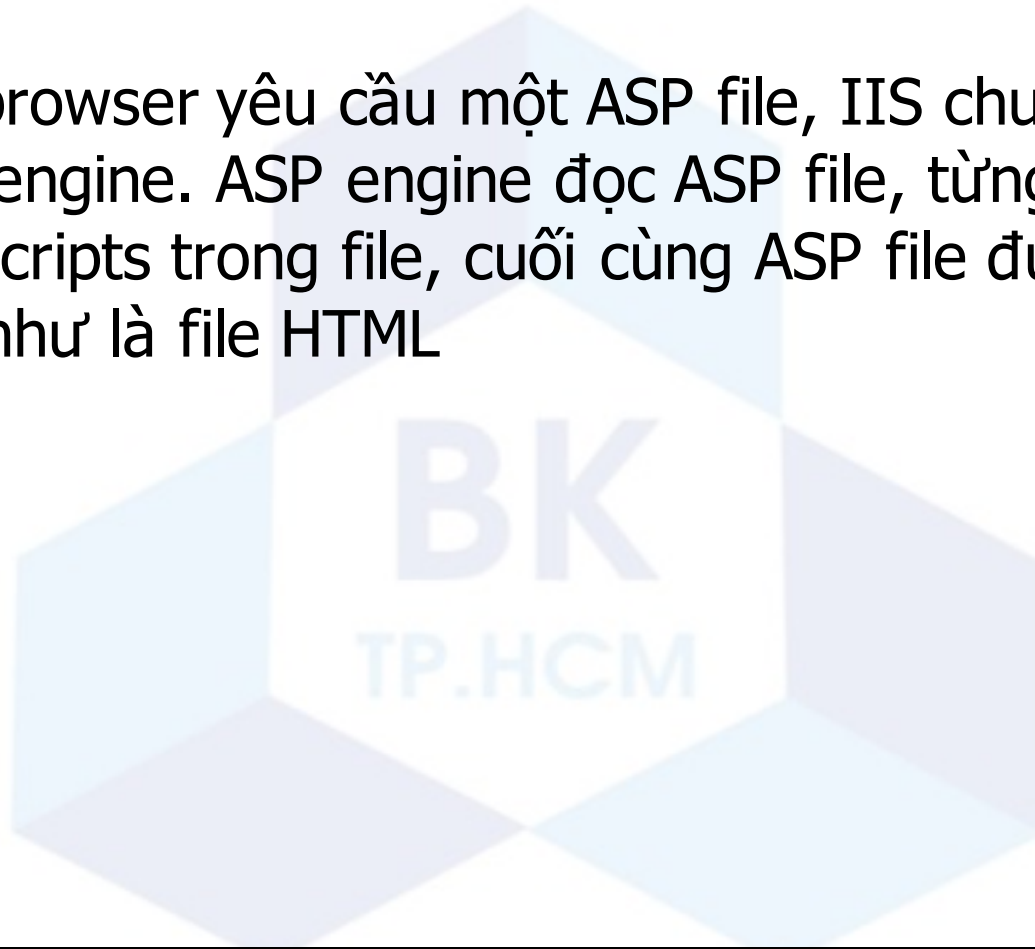
ASP (Active Server Pages)

- ASP là một công nghệ của Microsoft
- ASP là một công cụ tốt tạo những trang web động và tương tác
- Một file ASP có thể chứa text, HTML tags và scripts.
- Scripts trong ASP file được chạy trên server
- File ASP có đuôi là "*.asp"
- ASP là một program chạy bên trong IIS (**I**nternet **I**nformation **S**ervices)



ASP vs. HTML

- Khi một browser yêu cầu một HTML file, server trả về file HTML
- Khi một browser yêu cầu một ASP file, IIS chuyển yêu cầu đến ASP engine. ASP engine đọc ASP file, từng dòng một, và chạy scripts trong file, cuối cùng ASP file được gửi trả về browser như là file HTML



ASP có thể làm gì được cho chúng ta?

- Có thể cập nhật hoặc thêm nội dung của một trang web
- Trả lời các câu truy vấn của người dùng hoặc dữ liệu được gửi từ HTML forms
- Truy xuất dữ liệu hoặc CSDL và trả kết quả về cho browser
- Customize một trang web để nó có ích hơn cho người dùng
- Mặt mạnh của việc dùng ASP thay vì CGI và Perl là sự đơn giản và tốc độ
- Bảo mật – ASP code không thể view được từ browser
- Nếu lập trình khéo léo có thể tối ưu hóa tải trong mạng (network traffic)

Cú pháp cơ bản của ASP (1)

- Viết output cho browser
- Một ASP file thường chứa HTML tags, như là HTML file
- Tuy nhiên, một ASP file cũng có thể chứa những scripts, bên trong `<%` và `%>`
- Server scripts được chạy trên server, và có thể chứa những expressions, statements, procedures hoặc operators hợp lệ

```
<html>
<body>
  <%
    response.write("Hello World!")
  %>
</body>
</html>
```

ASP

```
<html>
```

```
<body>
```

```
<%
```

```
response.write("Hello World!")
```

```
%>
```

```
</body>
```

```
</html>
```



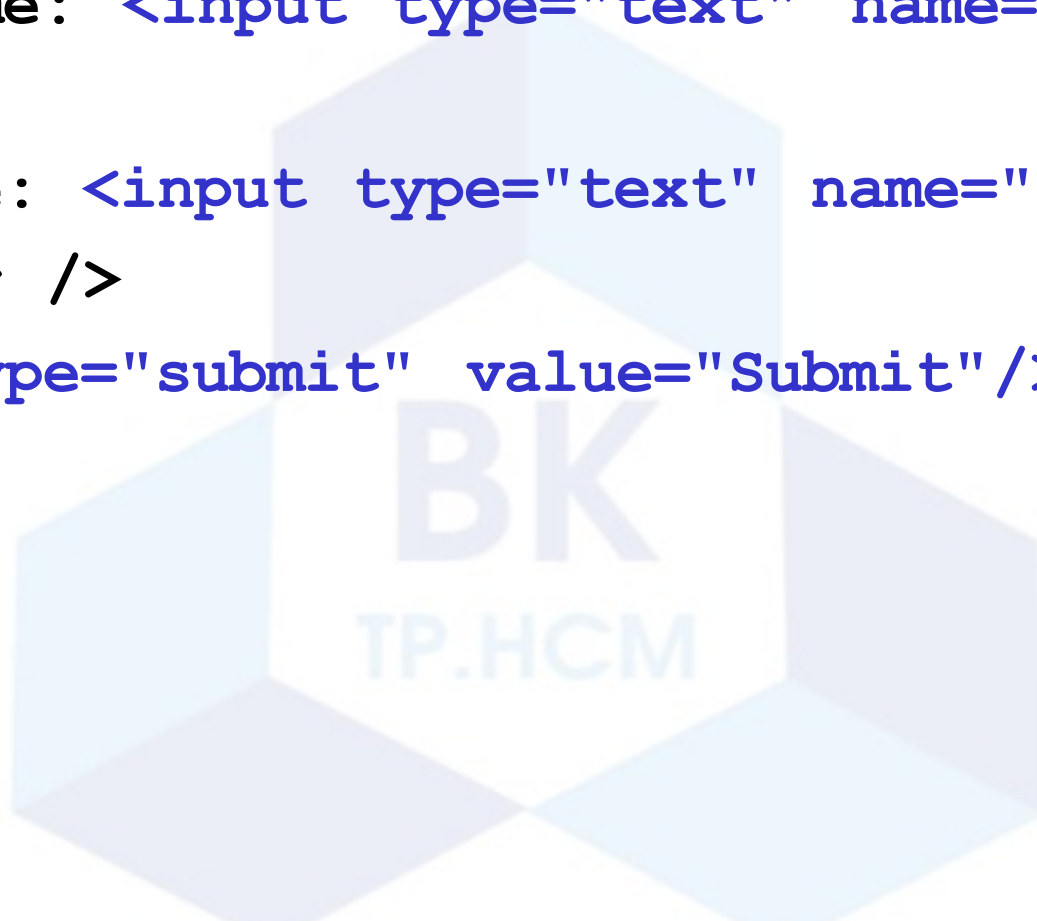
Javascript trong ASP

```
<%@ language="javascript"%>
<html>
<body>
  <%
    Response.Write("Hello World!")
  %>
</body>
</html>
```

BK
TP.HCM

Dùng form với ASP

```
<form method="get" action="simpleform.asp">  
First Name: <input type="text" name="fname"/>  
<br />  
Last Name: <input type="text" name="lname"/>  
<br /><br />  
<input type="submit" value="Submit"/>  
</form>
```



Cookies với ASP

- Dùng "Response.Cookies" để tạo cookie
- **Ghi chú**: "Response.Cookies" phải xuất hiện trước thẻ `<html>`

```
<%  
    Response.Cookies("firstname")="Alex"  
%>
```

- Cũng có thể gán thêm đặc tính cho cookie, chẳng hạn ngày tháng khi cookie quá hạn

```
<%  
    Response.Cookies("firstname")="Alex"  
    Response.Cookies("firstname").Expires=#May 10,2012#  
%>
```

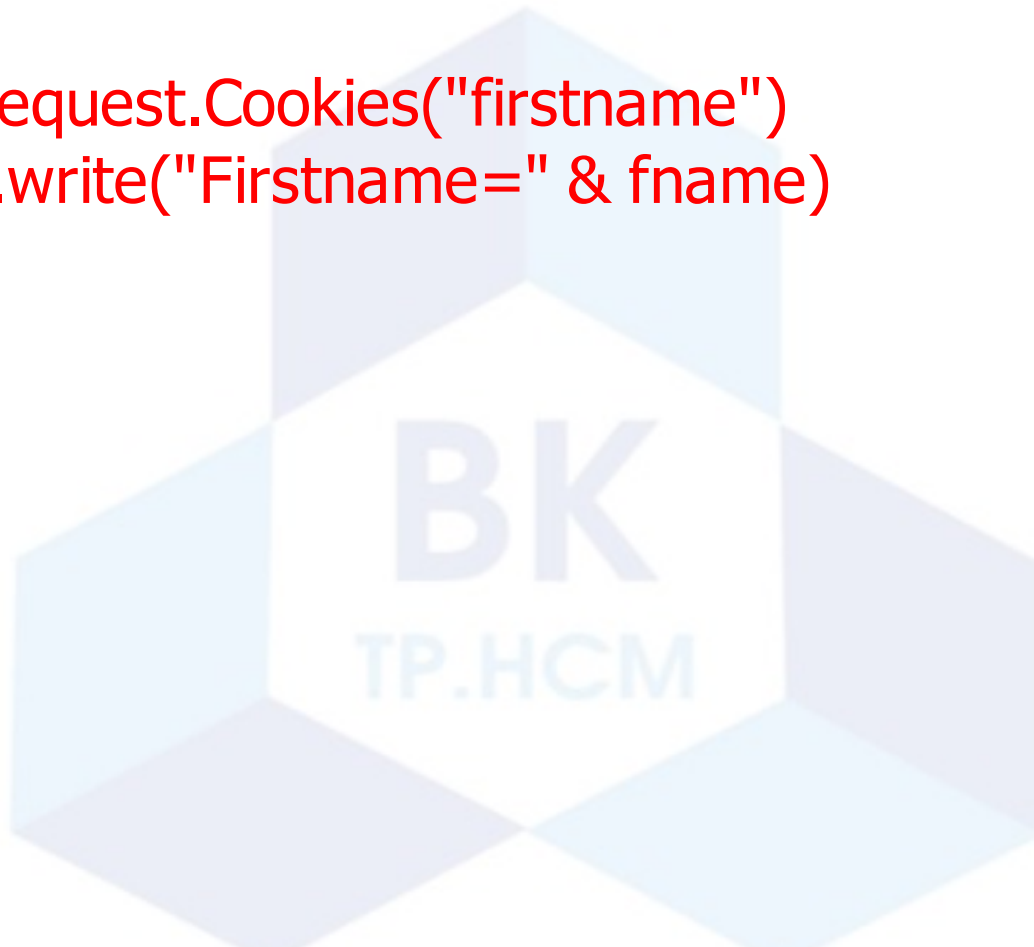

Nhận giá trị cookies với ASP

- Dùng "Request.Cookies" để nhận giá trị cookies

<%

```
fname=Request.Cookies("firstname")  
response.write("Firstname=" & fname)
```

%>



Dùng session với ASP

- Quản lý thông tin về phiên làm việc (session)
- Có thể lưu dữ liệu, đối tượng của một session
- Biến trong session:
- `Session("variable_name") = init_value;`
- Ví dụ:

<%

```
Session("username")="Donald Duck"
```

```
Session("age")=50
```

%>

Application với ASP

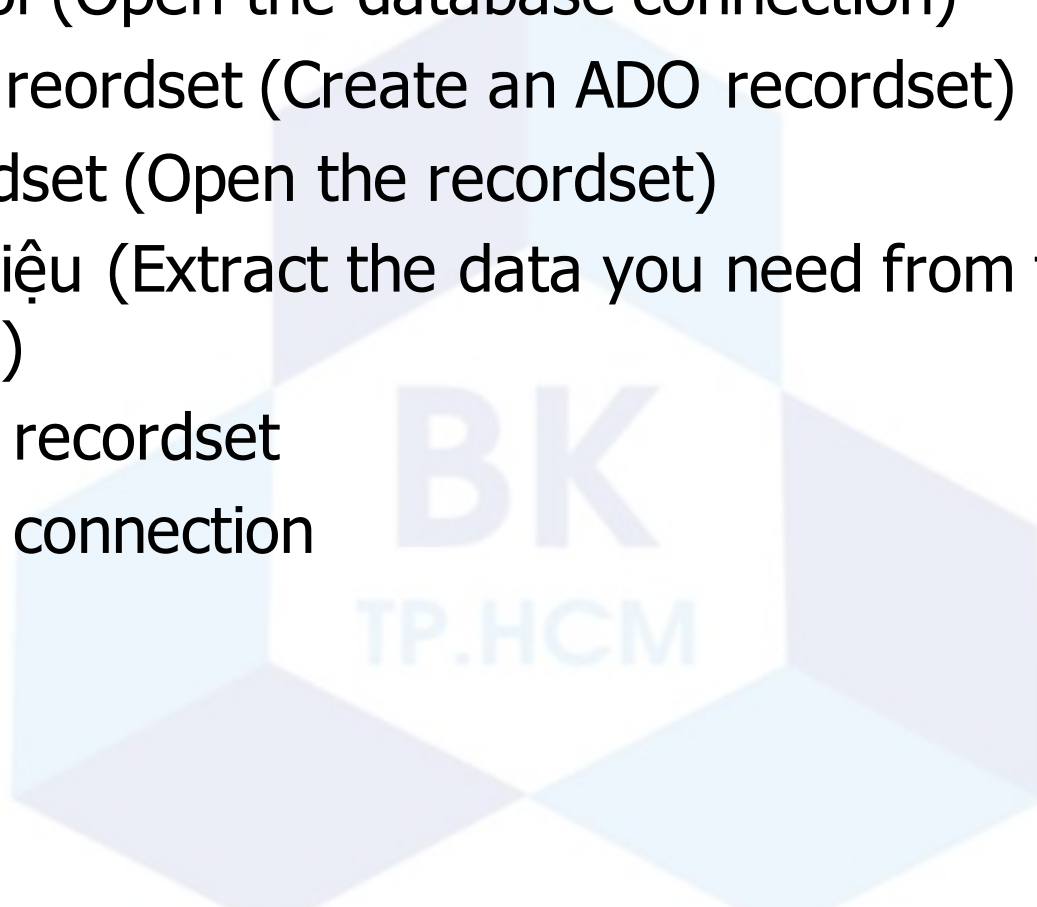
- Quản lý thông tin về ứng dụng
- Có thể dùng lưu trữ dữ liệu, đối tượng
- Chia sẻ giữa người dùng

`Application ("variable_name") = value;`

- Bạn có thể lock một ứng dụng với phương pháp "Lock". Khi một ứng dụng bị locked, người dùng khác không thể thay đổi biến Application. Bạn có thể unlock ứng dụng dùng phương pháp "Unlock"
- Thao tác cập nhật
 - Gọi Application.Lock();
 - Thực hiện các thao tác với các biến Application.
 - Gọi Application.Unlock();

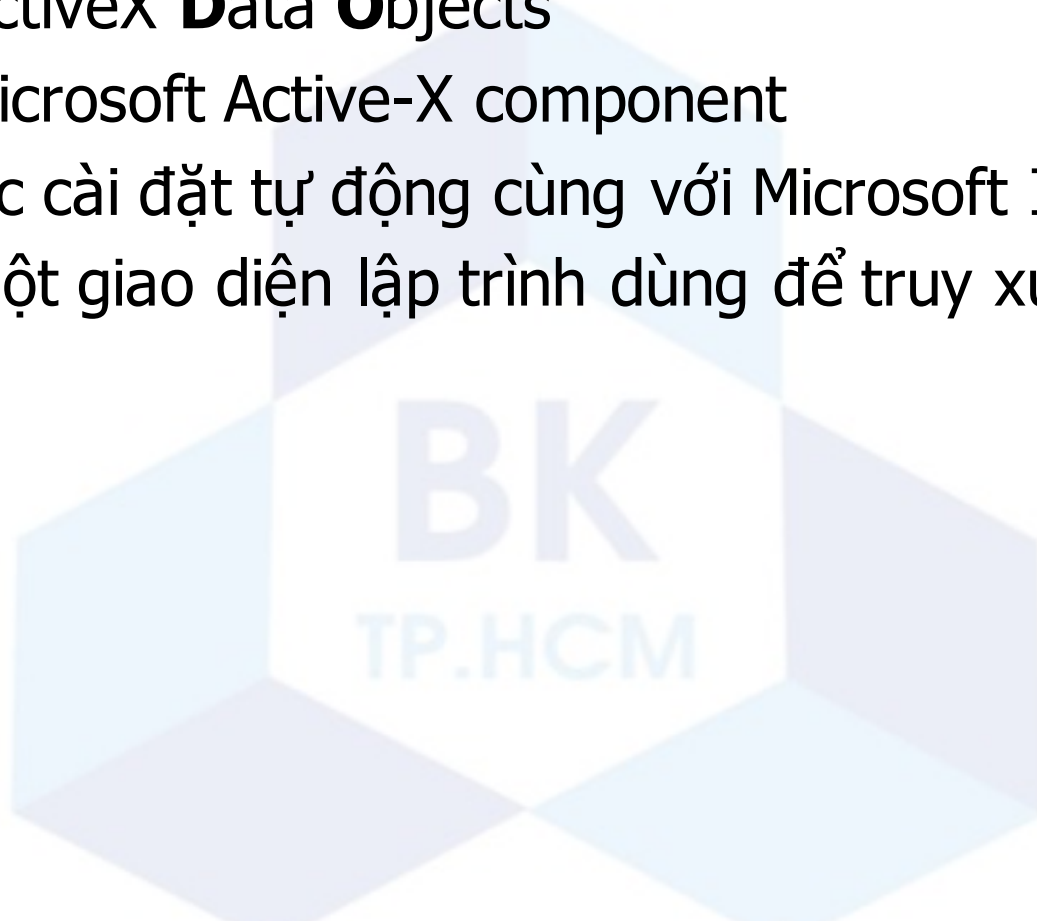
ASP với Cơ sở dữ liệu

- Tạo kết nối ADO (**A**ctiveX **D**ata **O**bjects) đến CSDL
- Mở kết nối (Open the database connection)
- Tạo ADO reordset (Create an ADO recordset)
- Mở recordset (Open the recordset)
- Trích dữ liệu (Extract the data you need from the recordset)
- Close the recordset
- Close the connection



ADO là gì?

- ADO là một công nghệ của Microsoft
- ADO = **A**ctiveX **D**ata **O**bjects
- ADO là Microsoft Active-X component
- ADO được cài đặt tự động cùng với Microsoft IIS
- ADO là một giao diện lập trình dùng để truy xuất CSDL



ASP với Cơ sở dữ liệu SQL

- Đọc dữ liệu bằng câu truy vấn SQL

```
strSQL = "Select * FROM table WHERE ...."
```

- Cập nhật thông tin trong record

```
strSQLUpdate = "UPDATE table SET field1=...,  
field2=... WHERE ..."
```

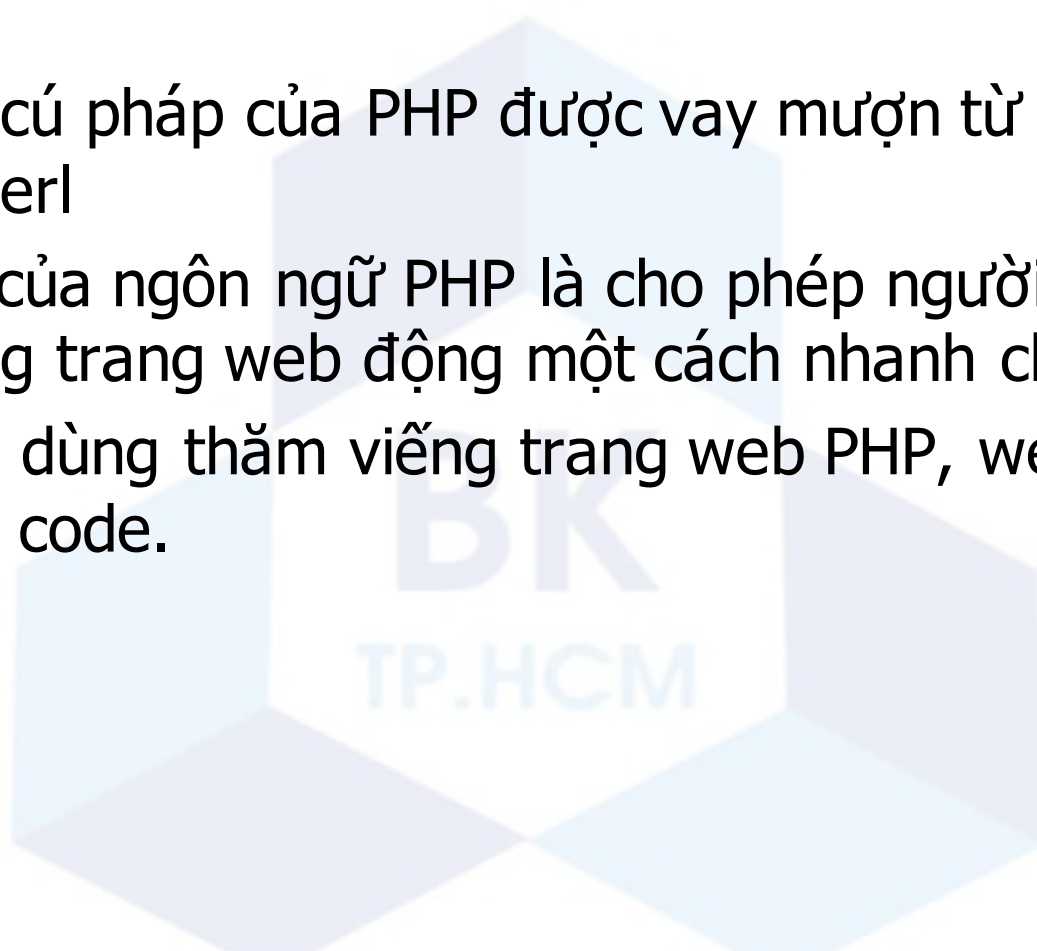
BK
TP.HCM

Ngôn ngữ lập trình PHP



PHP là gì ?

- PHP là ngôn ngữ thuộc dạng scripting nhúng vào bên trong HTML.
- Phần lớn cú pháp của PHP được vay mượn từ ngôn ngữ C, Java và Perl
- Mục tiêu của ngôn ngữ PHP là cho phép người thiết kế web viết những trang web động một cách nhanh chóng
- Khi người dùng thăm viếng trang web PHP, web server sẽ chạy PHP code.



PHP – Tiện ích

- PHP cho phép:
 - Rút ngắn thời gian tạo những website lớn
 - Cho kết quả dựa trên thông tin do người dùng cung cấp
 - Cho phép tạo những trang web thương mại online (shopping carts)



PHP – cú pháp

- Cú pháp và ngữ nghĩa của PHP tương tự như phần lớn các ngôn ngữ lập trình khác (C, Java, Perl), nhưng PHP code còn chứa thêm thẻ

```
<?php  
    (PHP code)  
?>
```

BK
TP.HCM

PHP – định dạng file PHP

- Bạn cần save lại file PHP dưới dạng có đuôi là *.php, thay vì là *.html
- Save file index.php thay vì là index.html



PHP – Ví dụ đơn giản

```
<html>
<head>
    <title>My First PHP Page</title>
</head>
<body>
    <?php
        echo "Hello World!";
    ?>
</body>
</html>
```

Dấu chấm phẩy bắt buộc

PHP – White space

- Whitespace bị bỏ đi (ignored) giữa những biểu thức PHP

```
<html>
<head>
<title>My First PHP Page</title>
</head>
<body>
<?php
    echo "Hello World!";

    echo "Hello World!";

?>
</body>
</html>
```

white space bị bỏ qua

PHP – biến

- Chúng ta có thể khai biến như trong ví dụ sau:

```
<?php
```

```
    $hello = "Hello World!";
```

```
    $a_number = 4;
```

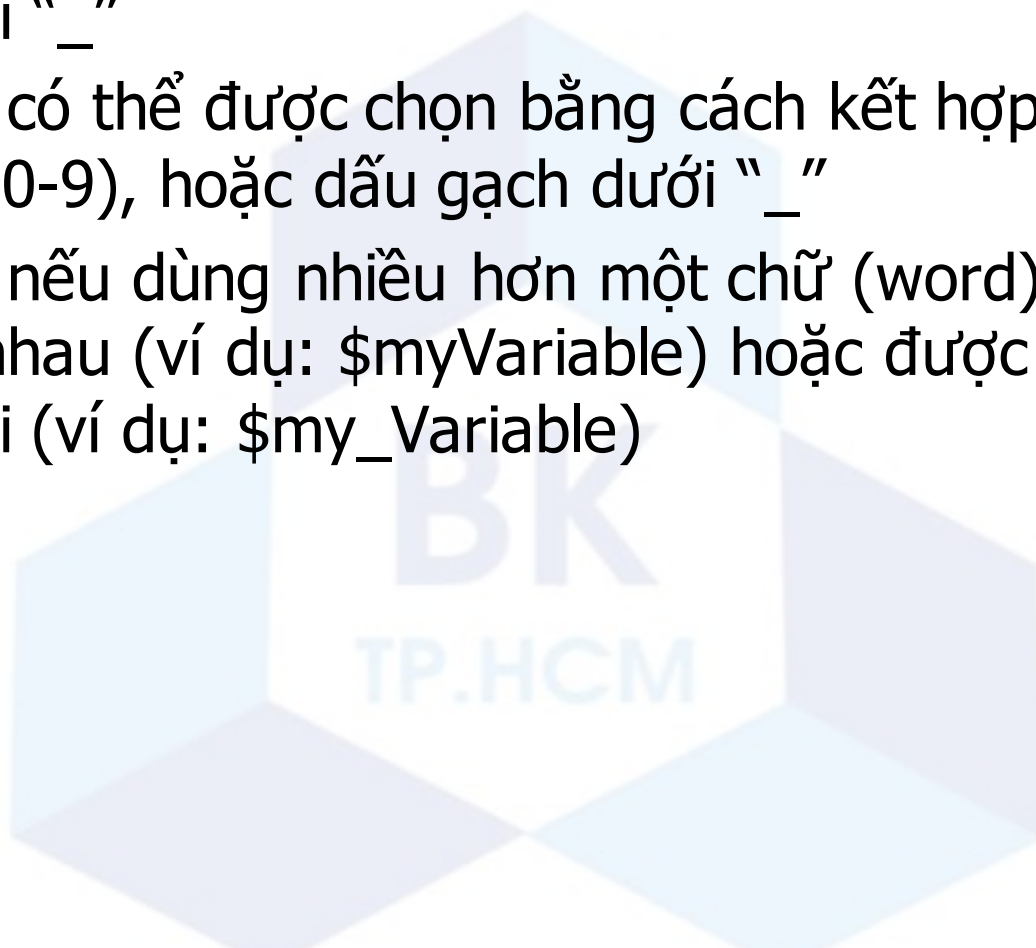
```
    $anotherNumber = 8;
```

```
?>
```

BK
TP.HCM

PHP – Quy định về biến

- Biến PHP bắt buộc phải bắt đầu bằng chữ cái hoặc dấu gạch dưới “_”
- Biến PHP có thể được chọn bằng cách kết hợp chữ cái (a-z, A-Z), số (0-9), hoặc dấu gạch dưới “_”
- Biến PHP nếu dùng nhiều hơn một chữ (word), có thể được viết liền nhau (ví dụ: `$myVariable`) hoặc được nối bằng dấu gạch dưới (ví dụ: `$my_Variable`)



PHP – viết ra chuỗi ký tự

- Dùng **echo** function

```
<?php
    $myString = "Hello!";
    echo $myString;
    echo "<h5>I love using PHP!</h5>";
?>
```



PHP – Careful When Echoing Quotes

- Không dùng quotes bên trong string (chuỗi ký tự)

```
<?php
// Không hợp lệ
echo "<h5 class='specialH5'>I love using PHP!</h5>";

// Hợp lệ
echo "<h5 class=\"specialH5\">I love using PHP!</h5>";

// Hợp lệ
echo "<h5 class='specialH5'>I love using PHP!</h5>";
?>
```

PHP – Tạo chuỗi ký tự

```
<?php  
$my_string = "Chào buổi sáng!";  
echo "Chào buổi sáng!";  
echo $my_string;  
?>
```



PHP – Phép toán

- Phân loại phép toán
 - Gán
 - Số học
 - So sánh
 - Chuỗi ký tự
 - Kết hợp gán và số học



PHP – Phép tính gán

- Phép tính gán được dùng để gán một giá trị vào một biến hoặc gán giá trị của một biến vào biến khác

- Ví dụ:

```
$my_var = 4;
```

```
$another_var = $my_var;
```



PHP – Phép tính số học

- `$addition = 2 + 4;`
- `$subtraction = 6 - 2;`
- `$multiplication = 5 * 3;`
- `$division = 15 / 3;`
- `$modulus = 5 % 2;`
- `echo "Perform addition: 2 + 4 = ".$addition."
;`
- `echo "Perform subtraction: 6 - 2 = ".$subtraction."
; echo "Perform multiplication: 5 * 3 = ".$multiplication."
;`
- `echo "Perform division: 15/3 = ".$division."
;`
- `echo "Perform modulus: 5 % 2 = " . $modulus . " .`
- Modulus is the remainder after the division operation has been performed. In this case it was 5 / 2, which has a remainder of 1."

PHP – Phép tính so sánh

==

!=

<

>

<=

>=



PHP – Phép tính với chuỗi ký tự

```
$a_string = "Hello";  
$another_string = " Billy";  
$new_string = $a_string . $another_string;  
echo $new_string . "!";
```



PHP – Kết hợp phép tính gán và số học

`+=`

`-=`

`*=`

`/=`

`%=`

`.=`



PHP – Comments

<!-- This is an HTML Comment -->

■ Single line comment

```
<?php
```

```
echo "Hello World!"; // This will print out Hello  
World!
```

```
echo "<br />Psst...You can't see my PHP comments!";
```

```
// echo "nothing";
```

```
// echo "My name is Humperdinkle!";
```

```
# echo "I don't do anything either";
```

```
?>
```

PHP – Multiple line comments

```
<?php
```

```
/* This Echo statement will print out my message  
to the the place in which I reside on.  
In other words, the World.
```

```
*/
```

```
echo "Hello World!";
```

```
/* echo "My name is Humperdinkle!";
```

```
echo "No way! My name is Uber PHP Programmer!";
```

```
*/
```

```
?>
```

PHP – Include file

■ Nội dung file menu.php

```
<html>
<body>
<a href="http://www.example.com/index.php">Home</a> - <a
  href="http://www.example.com/about.php">About Us</a> - <a
  href="http://www.example.com/links.php">Links</a> - <a
  href="http://www.example.com/contact.php">Contact Us</a>
<br />
```

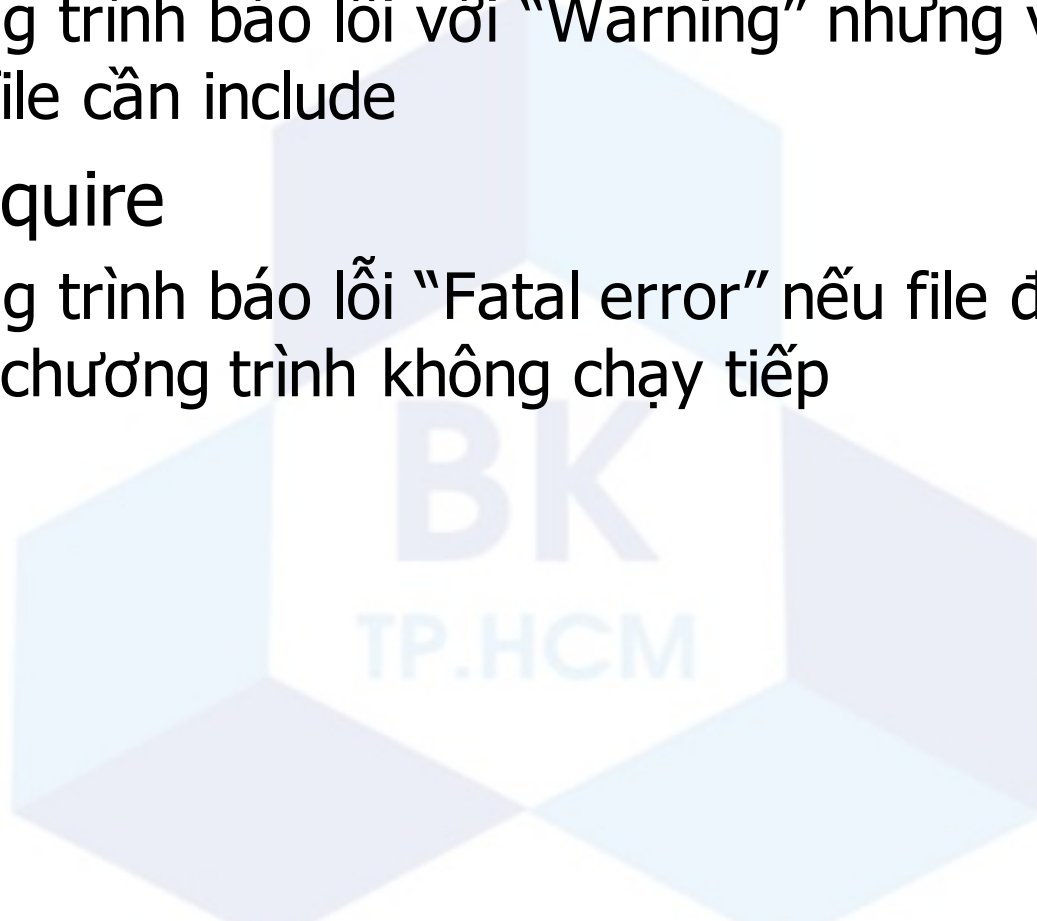
■ File chính

```
<?php include("menu.php"); ?>
```

```
<p>This is my home page that uses a common menu to save
  me time when I add new pages to my website!</p>
</body>
</html>
```

PHP – require vs. include

- Dùng include
 - Chương trình báo lỗi với “Warning” nhưng vẫn chạy nếu thiếu file cần include
- Dùng require
 - Chương trình báo lỗi “Fatal error” nếu file được require thiếu, chương trình không chạy tiếp



PHP – Hàm

- Một hàm là một tên chúng ta đặt cho một đoạn code có thể thực thi bất cứ khi nào chúng ta gọi
- Sử dụng hàm có thể tiết kiệm được rất nhiều thời gian và chương trình dễ đọc hơn
- Ví dụ: tạo một hàm

```
<?php
```

```
function myFunction()
```

```
{
```

```
}
```

```
?>
```

PHP – Ví dụ

```
<?php
function myGreeting($firstName)
{
    echo "Hello there ". $firstName . "!<br />";
}
```

```
myGreeting("East");
myGreeting("West");
myGreeting("South");
myGreeting("North");
?>
```

Display

```
Hello there East!
Hello there West!
Hello there South!
Hello there North!
```

BK
TP.HCM

PHP – Return value

- Khi hàm được dùng và đã kết thúc, nó sẽ có thể giữ lại một giá trị. Muốn dùng giá trị này, chúng ta có thể dùng một biến để nhận giá trị đó. Ví dụ:

```
$myVar = somefunction();
```

- Ví dụ:

```
<?php
```

```
function mySum($numX, $numY)
```

```
{
```

```
    $total = $numX + $numY;
```

```
    return $total;
```

```
}
```

```
$myNumber = 0;
```

```
echo "Before the function, myNumber = ". $myNumber . "<br />";
```

```
$myNumber = mySum(3, 4); // Store the result of mySum in
```

```
    $myNumber
```

```
echo "After the function, myNumber = " . $myNumber . "<br />";
```

```
?>
```

Truy xuất dữ liệu PHP với MySQL



Tạo kết nối đến CSDL

```
<?php
    $con =
    mysql_connect("localhost","peter","abc123");

    if (!$con)
    {
        die('Could not connect: ' .
        mysql_error());
    }
    // some code
?>
```

Tạo CSDL

```
<?php
    $con = mysql_connect("localhost","peter","abc123");
    if (!$con)
    {
        die('Could not connect: ' . mysql_error());
    }

    if (mysql_query("CREATE DATABASE my_db", $con))
    {
        echo "Database created";
    }
    else
    {
        echo "Error creating database: " . mysql_error();
    }

    mysql_close($con);

?>
```

Tạo record mới

```
INSERT INTO table_name (column1, column2, column3,...)  
VALUES (value1, value2, value3,...)
```

```
<?php  
$con = mysql_connect("localhost","peter","abc123");  
if (!$con)  
{  
    die('Could not connect: ' . mysql_error());  
}  
  
mysql_select_db("my_db", $con);  
  
mysql_query("INSERT INTO Persons (FirstName, LastName, Age)  
VALUES ('Peter', 'Griffin', '35')");  
  
mysql_query("INSERT INTO Persons (FirstName, LastName, Age)  
VALUES ('Glenn', 'Quagmire', '33')");  
  
mysql_close($con);  
  
?>
```

Truy xuất dữ liệu

SELECT column_name(s) FROM table_name

```
<?php
    $con = mysql_connect("localhost","peter","abc123");
    if (!$con)
    {
        die('Could not connect: ' . mysql_error());
    }

    mysql_select_db("my_db", $con);

    $result = mysql_query("SELECT * FROM Persons");

    while($row = mysql_fetch_array($result))
    {
        echo $row['FirstName'] . " " . $row['LastName'];
        echo "<br />";
    }
    mysql_close($con);
?>
```

Mệnh đề WHERE

```
SELECT column_name(s)
FROM table_name
WHERE column_name operator value
```

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}
mysql_select_db("my_db", $con);

$result = mysql_query("SELECT * FROM Persons
WHERE FirstName='Peter'");

while($row = mysql_fetch_array($result))
{
    echo $row['FirstName'] . " " . $row['LastName'];
    echo "<br />";
}
?>
```

Order by ...

```
SELECT column_name(s)
FROM table_name
ORDER BY column_name(s) ASC|DESC
```

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}
mysql_select_db("my_db", $con);
$result = mysql_query("SELECT * FROM Persons ORDER BY
age");
while($row = mysql_fetch_array($result))
{
    echo $row['FirstName'];
    echo " " . $row['LastName'];
    echo " " . $row['Age'];
}
mysql_close($con);
?>
```

Cập nhật giá trị trong CSDL

```
UPDATE table_name  
    SET column1=value, column2=value2,...  
    WHERE some_column=some_value
```

```
<?php  
$con = mysql_connect("localhost","peter","abc123");  
if (!$con)  
{  
    die('Could not connect: ' . mysql_error());  
}  
  
mysql_select_db("my_db", $con);  
  
mysql_query("UPDATE Persons SET Age = '36'  
WHERE FirstName = 'Peter' AND LastName = 'Griffin'");  
  
mysql_close($con);  
?>
```

Xóa record trong CSDL

```
DELETE FROM table_name  
WHERE some_column = some_value
```

```
<?php  
$con = mysql_connect("localhost","peter","abc123");  
if (!$con)  
{  
    die('Could not connect: ' . mysql_error());  
}  
  
mysql_select_db("my_db", $con);  
  
mysql_query("DELETE FROM Persons WHERE  
LastName='Griffin'");  
  
mysql_close($con);  
?>
```


Tài liệu tham khảo

- Bộ slides cũ môn Lập Trình Mạng, Khoa KH&KTMT.
- <http://www.w3schools.com>

