



PHÁT TRIỂN ỨNG DỤNG WEB

BÀI 8

CÁC PHƯƠNG THỨC TRUYỀN DỮ LIỆU TRONG PHP



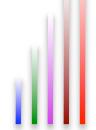
Nội dung



- ❖ Đặt vấn đề
- ❖ Cơ chế truyền và nhận dữ liệu giữa các trang Web
- ❖ Các cách truyền và nhận dữ liệu
- ❖ Hoạt động nhóm
- ❖ Yêu cầu bài nộp 3
- ❖ Bài tập



Đặt vấn đề



- ❖ Website “vatgia.com”

Pin Dell Inspiron XPS M1330 11.1 Volt Li-ion (LED224)

Giá: **340.000đ** (Có VAT)

Cập nhật: 1 ngày 21 giờ trước

- ✓ Tặng 3.400đ khi mua online
- ✓ Giá Gốc - Hàng Chính Hãng - Bảo Hành(1 đổi 1)

Giao hàng đến Chọn thành phố - quận huyện ▾

Số lượng: - 1 + Hiện có 10 sản phẩm



Mua ngay



Thêm vào giỏ hàng

Đặt vấn đề

Giỏ hàng của tôi + Chọn thêm sản phẩm

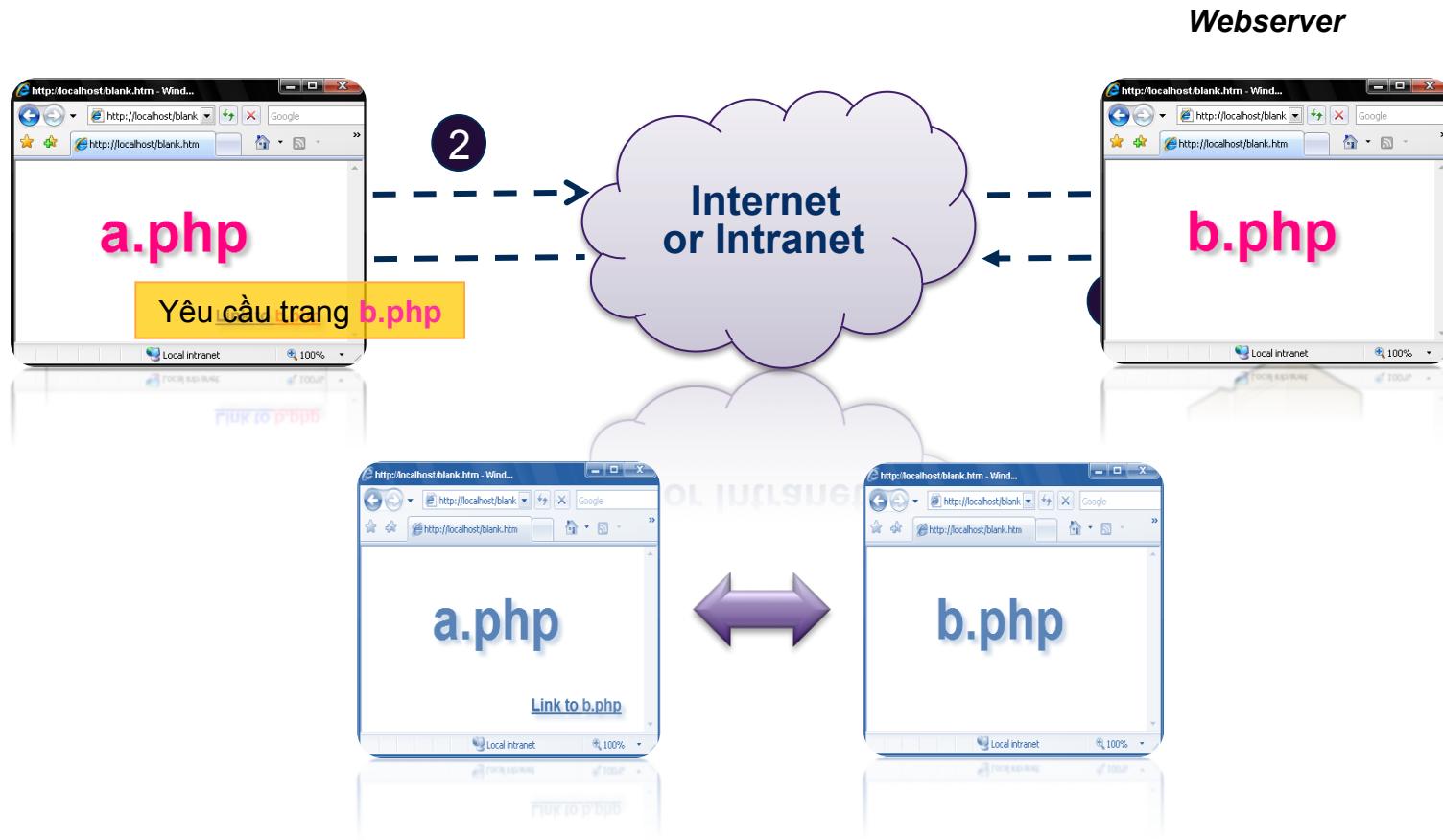
Sản phẩm	Đơn giá	Số lượng	Thành tiền
Đại Lý Phụ Kiện Máy Văn Phòng 			
Pin Dell Inspiron XPS M1330 11.1 Volt...	340.000đ	3	1.020.000đ
Xóa			

Tổng: 1.020.000đ

✓ Đặt hàng



Cơ chế truyền nhận dữ liệu



Cơ chế truyền nhận dữ liệu (tt)



The image shows two side-by-side web browser windows. Both are running on a local intranet (http://localhost) and are displaying results from a login attempt.

Left Browser Window: The title bar says "http://localhost/da...". It displays a "Đăng nhập hệ thống" (Login system) form. The "Tên đăng nhập" (Login name) field contains "admin" and the "Mật khẩu" (Password) field contains "*****". A large red starburst with a question mark is overlaid on the right side of this window.

Right Browser Window: The title bar says "http://localhost/xID...". It displays a "Kết quả đăng nhập" (Login result) page. It shows the input values: "Tên đăng nhập là: admin" and "Mật khẩu là: phpAdmin".



Cơ chế truyền nhận dữ liệu (tt)



Cơ chế truyền nhận dữ liệu (tt)



- ❖ Dữ liệu của người dùng từ trình duyệt sẽ được gửi lên máy chủ dưới dạng từng cặp biến=giá_trị và có thể đi theo 4 cách khác nhau
- ❖ Tuỳ theo từng cách gửi lên máy chủ mà máy chủ có các cách tương ứng để lấy dữ liệu được gửi lên
- ❖ 4 cách truyền dữ liệu:
 - GET, POST và
 - COOKIES, Session.
- ❖ Vậy GET, POST và COOKIES, Session?



Truyền dữ liệu bằng phương thức



- ❖ GET
- ❖ POST



Cơ chế truyền nhận dữ liệu



❖ Trang web nhập dữ liệu:

- Sử dụng đối tượng <form>
- Nhập dữ liệu thông qua các <control>
- Thực hiện việc truyền dữ liệu thông qua nút nhấn Submit

```
<form name="form1" method="post/get"  
action="URL">  
    <input type="submit" name="Submit"  
    value="Submit">  
</form>
```



Cơ chế truyền nhận dữ liệu (tt)

- ❖ Trang nhận dữ liệu (URL) sử dụng các biến toàn cục của PHP

`$_POST[“tên control”]`

`$_GET[“Tên control”]`

`$_REQUEST[“tên control”];`



Truyền dữ liệu bằng phương thức Get



- ❖ Tham số truyền đi qua địa chỉ URL
 - `http://domain/pathfile?`
`fieldname1=value1&fieldname2=value2`
- ❖ **Ví dụ:**
[http://www.uit.edu.vn/xemdiem.php?
mssv=11520123](http://www.uit.edu.vn/xemdiem.php?mssv=11520123)



Truyền dữ liệu bằng phương thức Get



- ❖ Truyền thông qua form
 - ✧ Form có thuộc tính method=“**GET**”
 - ✧ Nhận dữ liệu thông qua mảng toàn cục:
 - **`$_GET[“tên control”]`**
 - **`$_REQUEST[“tên control”]`**



Ví dụ - giao diện trang nhập dữ liệu

A screenshot of a web browser window displaying a login page. The address bar shows the URL: `localhost/get_post_session_cookie/get/login.php`. The page contains two text input fields: one for "User name" containing "admin" and another for "Pass word" containing "1234567". Below the inputs is a "Login" button.

← → ⌛ localhost/get_post_session_cookie/get/login.php

User name admin

Pass word 1234567

Login

Ví dụ - trang nhận dữ liệu

A screenshot of a web browser window. The address bar shows the URL: localhost/get_post_session_cookie/get/index.php?USER=admin&PASS=1234567&Login=Login. The main content area displays the text "user name:admin" and "Pass word:1234567".



Trang nhập dữ liệu (login.php)

```
<form name="form1" method="get" action="index.php">
<p>
    <label>User name
    <input name="USER" type="text" id="USER">
    </label>
</p>
<p>
    <label>Pass word
    <input name="PASS" type="text" id="PASS">
    </label>
</p>
<p>
    <input type="submit" name="Submit" value="Login">
</p>
</form>
```



Trang nhận dữ liệu (index.php)

```
<?php  
    echo "user name:". $_GET["USER"]. "</br>";  
    echo "Pass word:". $_GET["PASS"]. "</br>";  
?>
```



Truyền dữ liệu bằng phương thức Get



- Truyền dữ liệu bằng phương thức GET không thông qua form mà truyền trực tiếp thông qua địa chỉ URL
- **Ví dụ:**
<http://localhost/chitietsanpham.php/msp=001>



Truyền dữ liệu bằng phương thức Get

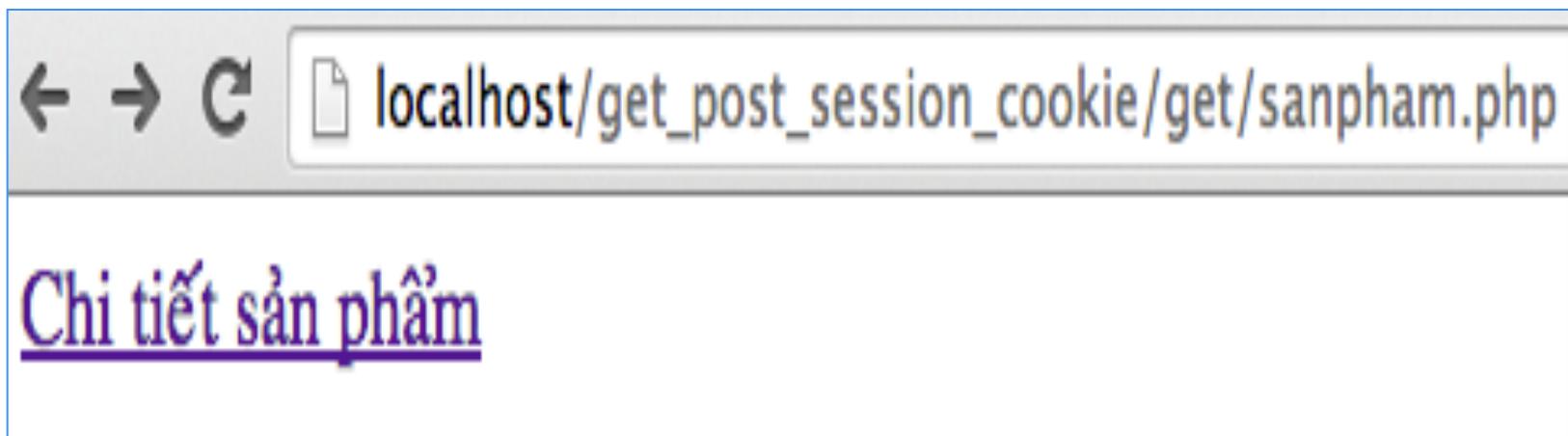
- ❖ Ta cũng có thể đưa lên nhiều cặp **tên biến=giá_trị** bằng cách phân cách chúng bởi dấu &:
[http://localhost/chitietsanpham.php?
masp=7&mansx=12](http://localhost/chitietsanpham.php?masp=7&mansx=12)
- ❖ Với địa chỉ URL trên, ta sẽ gửi lên 2 cặp biến=giá_trị theo phương thức GET, đó là:
masp=7, mansx=12
- ❖ Lấy giá trị thông qua mảng toàn cục:
 - **\$_GET**[“tên biến”]
 - **\$_REQUEST**[“tên biến”]



Ví dụ



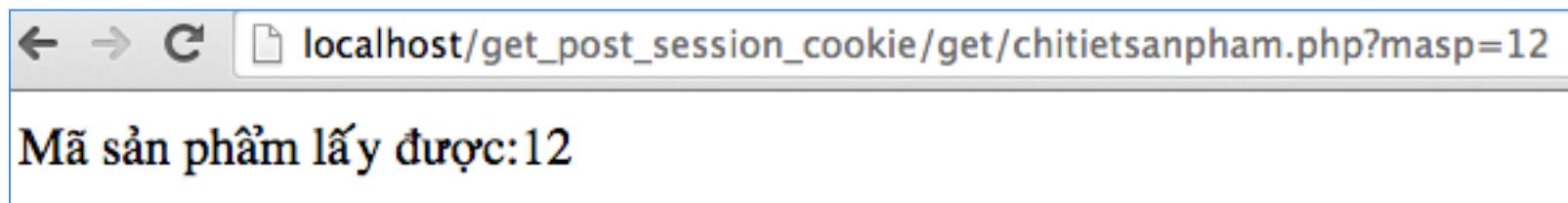
- Xét trang “*sanpham.php*” có giao diện sau:



Ví dụ (tt)



- Khi click chọn link “chi tiết sản phẩm” thì trang “*chitietsanpham.php*” được mở lên. Trang “*chitietsanpham.php*” lấy dữ liệu từ trang “*sanpham.php*” và hiển thị lên màn hình



Ví dụ 1



❖ Trang “sanpham.php”

```
<html>
<head>
<meta charset="UTF-8">
<title>Sản phẩm</title>
</head>
<body>
<a href="chitietsanpham.php?masp=12">Chi tiết sản
    phẩm </a>
</body>
</html>
```



Ví dụ (tt)

❖ Trang lấy dữ liệu (chitietsanpham.php)

```
<html>
<head>
<meta charset="UTF-8">
<title>Thông tin chi tiết sản phẩm</title>
</head>
<body>
<?php
    $masp=$_REQUEST["masp"];
    echo "Mã sản phẩm lấy được:".$masp;
?>
</body>
</html>
```

Phương thức POST



- ❖ Form có thuộc tính method ="POST"
- ❖ Tham số truyền đi được ẩn bên trong FORM
- ❖ Lấy dữ liệu:
 - `$_POST["tên control"]` ;
 - `$_REQUEST["tên control"]`;



Ví dụ

- Xét trang đăng ký tài khoản có giao diện sau:

Tài khoản đăng nhập

Mật khẩu đăng nhập

Nhập lại mật khẩu

Giới tính

Đăng ký



Ví dụ (tt)

- Khi nhập chọn dữ liệu trên form đăng ký và nhấn chọn nút đăng ký thì website mở trang và hiển thị thông tin đã nhập trong trang “thongtindangky.php”

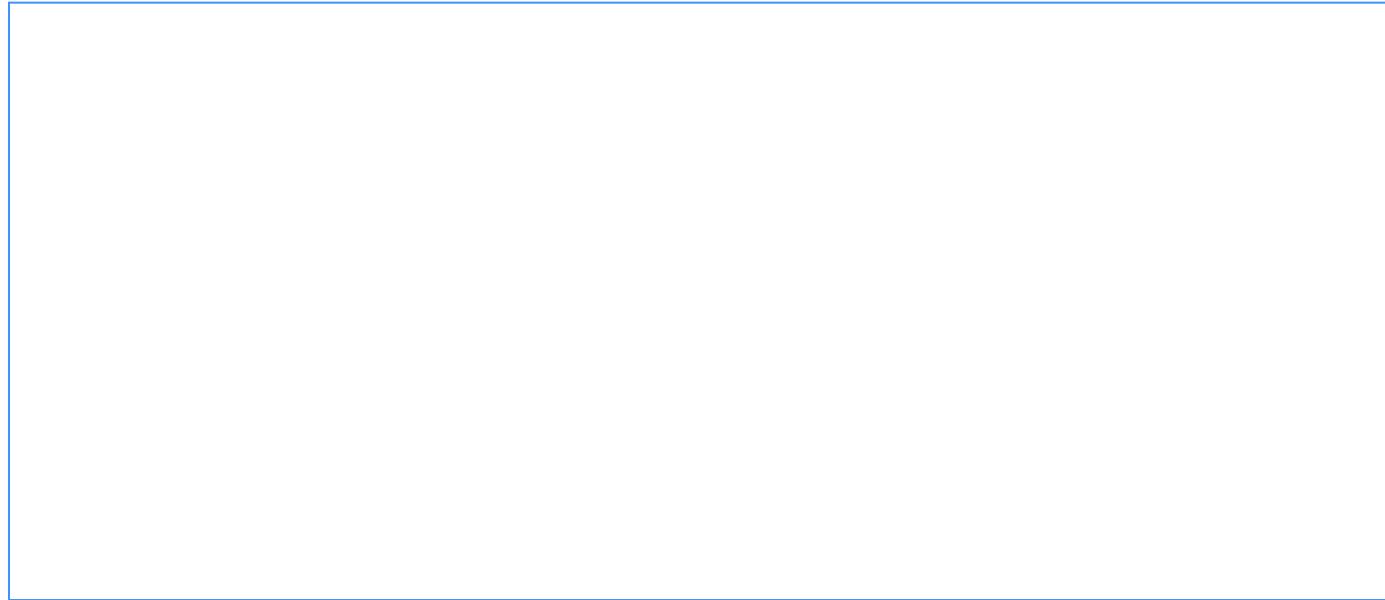
The screenshot shows a registration form with the following fields:

- A text input field containing "vantai".
- A password input field containing ".....".
- A password confirmation input field containing ".....".
- A dropdown menu set to "Nam".
- A blue rectangular button labeled "Đăng ký".



Ví dụ (tt)

- Nội dung được hiển thị trong trang “thongtindangky.php”



Ví dụ - trang “dangky.php”

```
<body>
<table align="center" bgcolor="#F0F0F0" cellspacing="0" border="0"
       width="250" cellpadding="15">
<tr><td>
<form method="POST" action="thongtindangky.php">
<input type="text" placeholder="Tài khoản đăng nhập" name="user"
       size="40"><br><br>
<input placeholder="Mật khẩu đăng nhập" type="password" name="pass"
       size="40"><br><br>
<input placeholder="Nhập lại mật khẩu" type="password" name="repass"
       size="40"><br><br>
<select name="gt">
    <option selected>Giới tính </option>
    <option value="Nam"> Nam</option>
<option value="Nu"> Nữ</option>
</select><br><br>
<input type="Submit" name="Submit" value="Đăng ký">
</form>  </td></tr>
</table>
</body>
```

Ví dụ - trang “thongtindangky.php”

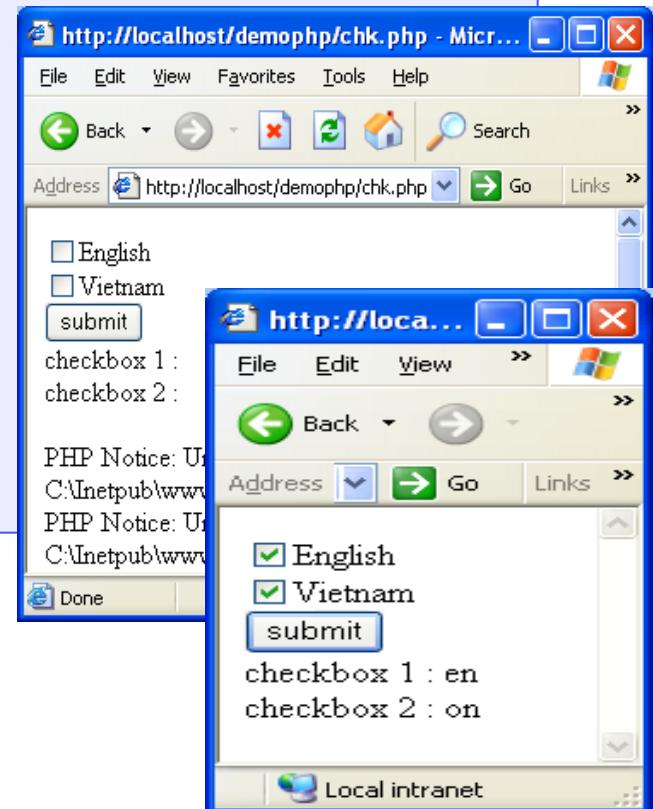
```
<body>
<?php
    $user=$_POST["user"];
    $pass=$_POST["pass"];
    $gt=$_POST["gt"];
    echo "<h3>Bạn đã đăng ký với tài khoản sau:</h3>";
    echo "Tên đăng nhập:".$user."<br>";
    echo "Mật khẩu đăng nhập:".$pass."<br>";
    echo "Giới tính:".$gt."<br>";
?
</body>
```



Truyền/Nhận dữ liệu từ Checkbox

File: Checkbox.php

```
<html><body>
<form method="get" action="checkbox.php">
<input type="checkbox" name="chk1" value="en">English <br>
<input type="checkbox" name="chk2">Vietnam<br>
<input type="submit" value="submit"><br>
</form>
<?php
    echo "checkbox 1 : " . $_REQUEST[ 'chk1' ];
    echo "checkbox 2 : " . $_REQUEST[ 'chk2' ];
?>
</body></html>
```



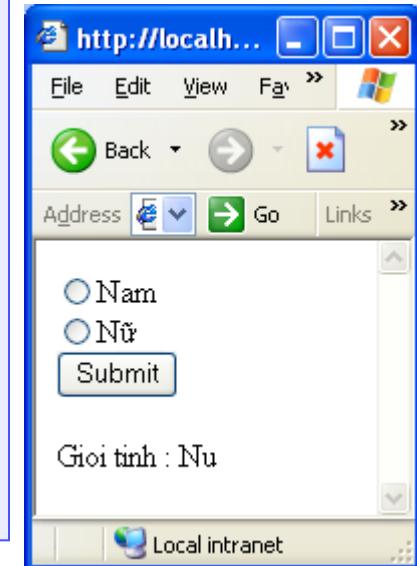
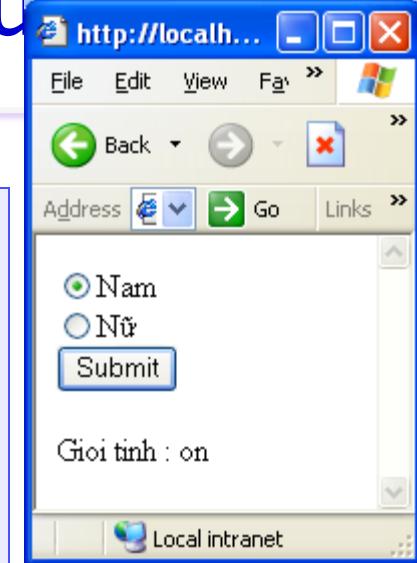
Truyền/Nhận dữ liệu từ Radio Button

File: RADIO.PHP

```
<html><body>

<form action="radio.php" method="GET">
<input type=RADIO" NAME="radGT">Nam<br>
<input type=RADIO" NAME="radGT" VALUE="Nu">Nữ<br>
<input type="SUBMIT" VALUE="Submit">
</form>

<?php
    if (isset($_GET[ 'radGT'])) {
        echo "Gioi tinh : " . $_GET[ 'radGT'];
    }
?>
</body></html>
```



Truyền/Nhận dữ liệu từ ComboBox

File: select.php

```
<body>
<form method="POST" action="select.php">
<select name="noicongtac">
    <option value="TPHCM">Thành Phố Hồ Chí Minh</option>
    <option value="HN">Hà nội</option>
    <option>Đà Nẵng</option>
</select>
<input type="submit" name="submit" value="Chọn"/>
</form>
<?php
    if (isset($_POST['noicongtac']))
    {
        echo "Bạn đã chọn:". $_POST['noicongtac'] . "<br/>";
    }
?>
</body>
```

The screenshot shows three separate dropdown menus, each with a selected option and a 'Chọn' button. The first menu has 'Thành Phố Hồ Chí Minh' selected, with the message 'Bạn đã chọn:TPHCM'. The second menu has 'Hà nội' selected, with the message 'Bạn đã chọn:HN'. The third menu has 'Đà Nẵng' selected, with the message 'Bạn đã chọn:Đà Nẵng'.

Thành Phố Hồ Chí Minh	Chọn
Bạn đã chọn:TPHCM	
Hà nội	Chọn
Bạn đã chọn:HN	
Đà Nẵng	Chọn
Bạn đã chọn:Đà Nẵng	

Truyền/Nhận dữ liệu từ Listbox - sử dụng Mảng

File: listbox.PHP

```
<body>
<form method="POST" action="listbox.php">
<select name="noicongtac[]" multiple>
<option value="TPHCM">Thành Phố Hồ Chí Minh</option>
<option value="HN">Hà nội</option>
<option>Đà Nẵng</option>
</select>
<input type="submit" name="submit" value=>
</form>
<?php
if(isset($_POST['submit'])&&($_POST['submit']=="Chọn"))
{
    echo "Bạn đã chọn:<br>";
    foreach($_POST['noicongtac'] as $nct)
        echo $nct."<br>";
}
?></body>
```

Thành Phố Hồ Chí Minh
Hà nội
Đà Nẵng

Chọn

Bạn đã chọn:
HN
Đà Nẵng

Khi nào dùng GET, POST



❖ Phương thức GET:

- **Ưu điểm:**

- **Ưu điểm:**
 - ❖ Dữ liệu được truyền đi nhanh hơn POST
 - ❖ Có thể giả lập phương thức GET để truyền dữ liệu mà không cần dùng FORM

- **Nhược điểm:**

- **Nhược điểm:**
 - ❖ Không thích hợp để truyền dữ liệu có tính bảo mật
 - ❖ Dữ liệu truyền đi bị giới hạn (2048 ký tự)



Khi nào dùng GET, POST (tt)



❖ Phương thức POST:

- **Ưu điểm:**

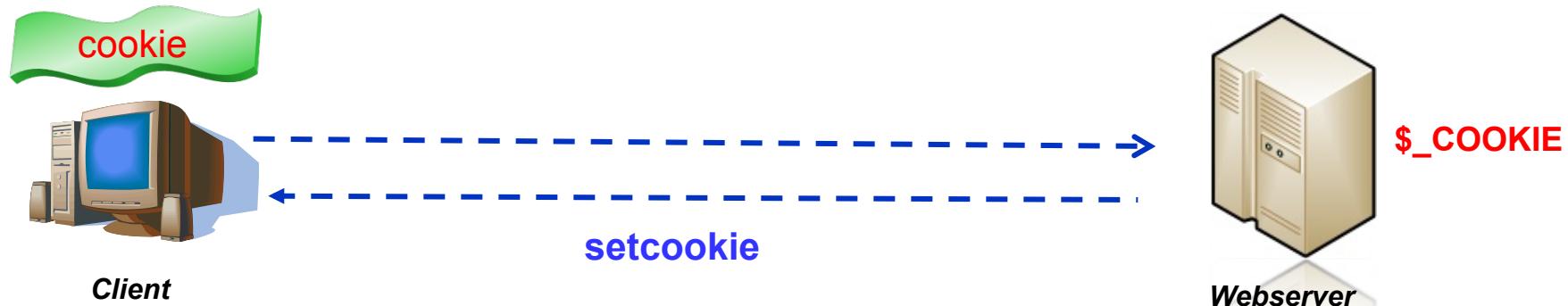
- ✧ Bảo mật hơn phương thức GET
 - ✧ Không giới hạn dữ liệu truyền đi

- **Nhược điểm:**

- ✧ Có thể gây ra lỗi nếu người dùng muốn quay lại trang kết quả (khi nhấn nút back, hoặc refresh) do bị expired
 - ✧ Dữ liệu truyền đi chậm hơn phương thức GET



Cookie – Giới thiệu



- ❖ Là 1 tập tin được server **lưu xuống máy client**
- ❖ Mỗi lần client gửi request 1 trang web, đồng thời sẽ gửi kèm file cookie đã lưu lần trước lên server
- ❖ Việc **xử lý thông tin (lưu, lấy)** trong cookie do server thực hiện
- ❖ Thường được sử dụng để **lưu thông tin cá nhân** của client



Cookie – Giới thiệu



- ❖ 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
- ❖ 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.



Cookie – Cú pháp sử dụng



❖ Lệnh ghi cookie

setcookie(name, value, expire, path, domain);

- name: Tên cookie
- value: Giá trị cookie
- expire: Thời điểm mà cookie hết hiệu lực
- path : Đường dẫn trên server mà cookie có hiệu lực
- domain : Xác định tên miền mà cookie được gửi đi
- *Bắt buộc phải xuất hiện trước thẻ <html>*



Cookie – Cú pháp sử dụng



- Lấy giá trị cookie

```
echo $_COOKIE[“cookieName”];
```

- Xóa cookie

```
setcookie(“cookieName”, “”, time() –  
3600);
```

Ví dụ: tạo trang “a.php”

```
<?php $t="1111";
      setcookie("a",$t);

?>
<html>
<body>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8">
<title>Trang a</title>
</head>
gia tri ghi len cookies:
<?php
      echo $t;
?>
<a href ="b.php"> qua trang b
</body>
</html>
```

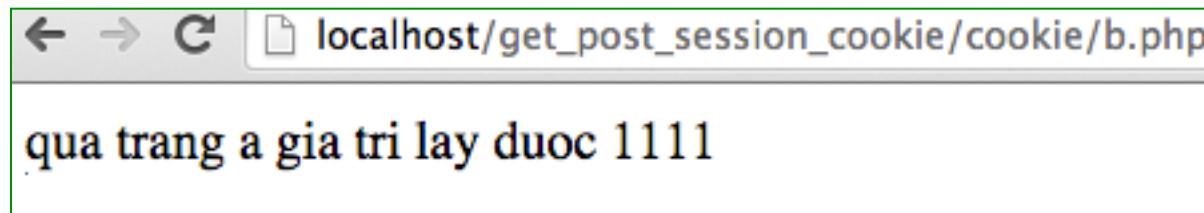


Kết quả

- Trang “*a.php*”



- Trang “*b.php*”



Tạo trang “b.php”

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=windows-1252">
<title>Trang b</title>
</head>
<body>
<a href="a.php"></a> qua trang a
<?
    if (isset($_COOKIE['a']))
    {
        echo "gia tri lay duoc ".$_COOKIE['a'];
    }
    else
        echo "khong lay duoc";
?>
</body>
</html>
```

Session



- Là đoạn dữ liệu được lưu trên server, khi browser có yêu cầu lấy dữ liệu từ session thì server cung cấp.
- Mỗi session lưu định danh duy nhất cho từng client
- Mục đích lưu biến dữ liệu dùng chung cho nhiều trang trong 1 phiên làm việc của client



Session – Cú pháp sử dụng



- Khởi động Session

`session_start();`

■ *Bắt buộc phải xuất hiện trước thẻ <html>*

- Ghi giá trị Session

`$_SESSION[“sessionVar”] = $value ;`

- Đọc giá trị từ Session

`$value=$_SESSION[“sessionVar”] ;`

```
if (isset($_SESSION[“sessionVar”])  
    echo $_SESSION[“sessionVar”];
```

- Hủy biến trong Session

`unset($_SESSION[“sessionVar”]);`

- Hủy cả Session

`session_destroy();`



Ví dụ tạo trang s_a.php

```
<?php
session_start();
$_SESSION['username'] = 'guest';
$_SESSION [password] = '12345';
?>
<html>
<head><title>a.php</title></head>
<body>
<a href="s_b.php">vào day</a> de kiem tra.
</body>
</html>
```



Tạo trang “s_b.php”

```
<?php
    session_start();
?>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Untitled Document</title>
</head>
<body>
Giá trị session lấy được
<a href="a.php">file a.php</a>:<br>
username = <?php echo $_SESSION['username']; ?><br>
time = <?php echo $_SESSION['password']; ?>
</body>
</html>
```

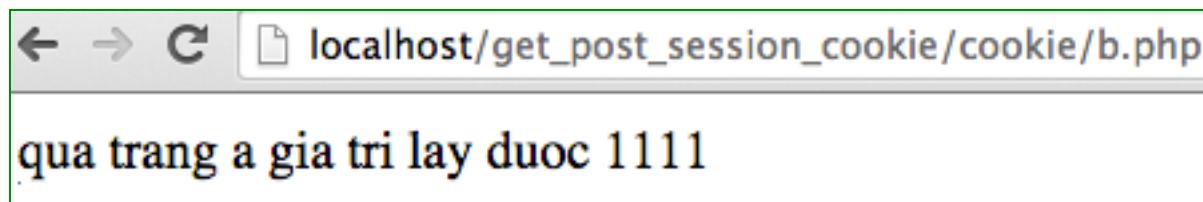


Kết quả

- Trang “*s_a.php*”



- Trang “*s_b.php*”



So sánh giữa cookie và session



- ❖ Cookie và Session đều có chung mục đích là lưu giữ data để truyền từ 1 trang web sang 1 trang web khác (trên cùng website).
- ❖ Phương thức lưu trữ và quản lý data của Cookie và Session có phần khác nhau.
- ❖ Cookie sẽ được lưu trữ tại browser, do browser quản lý và browser sẽ tự động truyền cookie ngược lên server mỗi khi truy cập vào 1 trang web trên server.
- ❖ Dữ liệu lưu trữ trong Session được webserver quản lý
- ❖ Browser chỉ truyền ID của session lên server để lấy dữ liệu khi cần



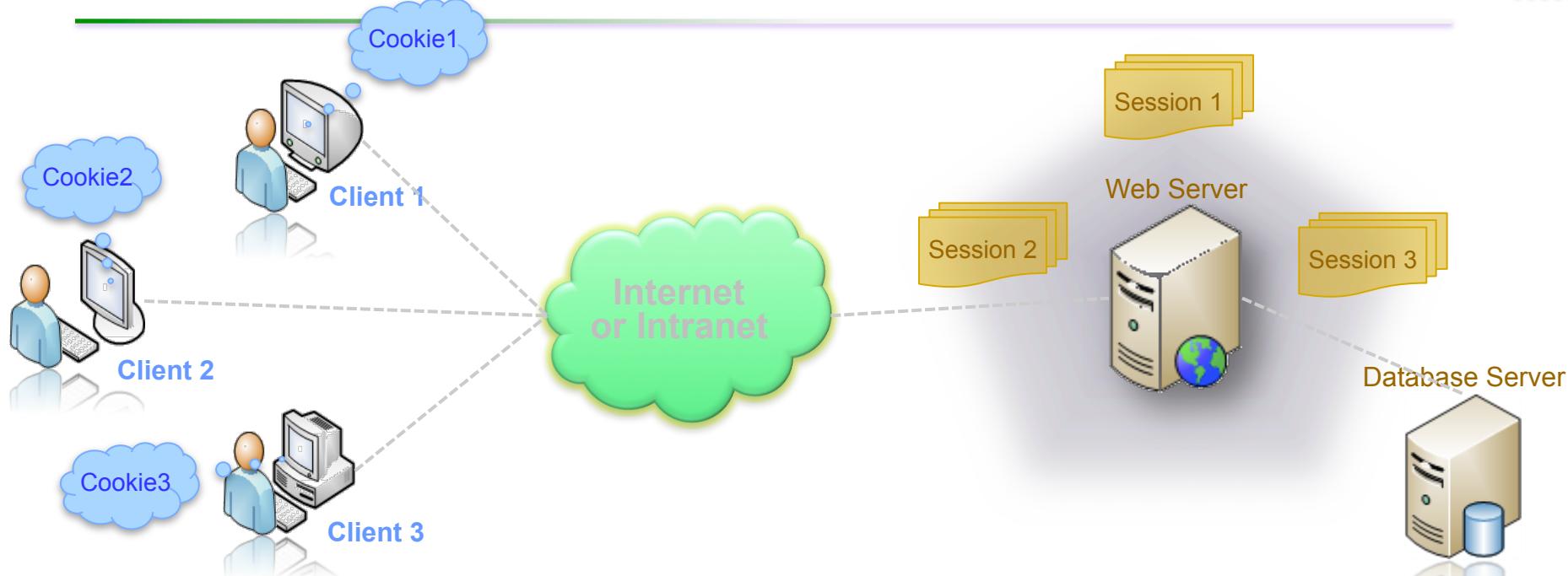
Sử dụng Cookie hay Session?



- ❖ Sử dụng Session hoặc Cookie là tùy vào lựa chọn của Lập trình viên, tuy nhiên Session thường được ưa chuộng hơn Cookie vì một số lý do sau:
 - Session vẫn sử dụng được trong trường hợp vùng nhớ Cookie bị chặn.



So sánh : Session – Cookie – Database



Thông tin chia sẻ	Vị trí lưu trữ	Thời gian tồn tại	Số client	Số trang web sử dụng
Database/File	Server	Dài	Nhiều	Nhiều
Session	Server	Ngắn	1	Nhiều
Cookies	Client	Dài	1	Nhiều



Sử dụng Cookie hay Session?

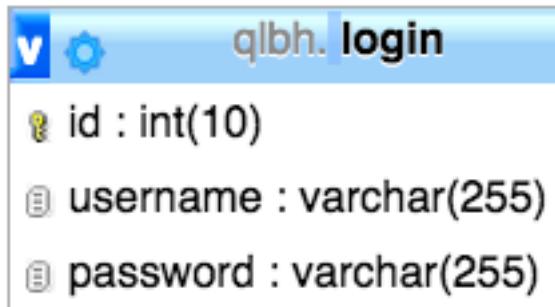
- Lượng data truyền tải giữa browser và server: chỉ mỗi **session** ID được truyền giữa browser và **server**, data thực sự được **website** lưu trữ trên server.
- **Bảo mật**: Càng ít thông tin được truyền tải qua lại giữa browser và client càng tốt, và càng ít thông tin được lưu trữ tại client càng tốt.



Hoạt động nhóm



- Giả sử ta có bảng lưu thông tin tài khoản của người dùng như sau:



- Các nhóm thảo luận và đưa ra quy trình để triển khai trang *login* và *logout* bằng cách dùng session. Mô tả các công việc cần thực hiện trong từng trang
- Viết code cho các trang web đó



Hoạt động nhóm (tt)



- Giao diện trang “*index.php*”

Login Form

UserName :

Password :

Login





Hoạt động nhóm (tt)

- Giao diện trang “*profile.php*” khi đăng nhập thành công như sau

The screenshot shows a user interface with a blue header bar. On the left, it says "Chào mừng: admin". On the right, there is a "Log Out" button. A green rectangular box highlights the text "Tên đăng nhập vào hệ thống" (Login name into the system), which is positioned below the welcome message. An arrow points from the text "Tên đăng nhập vào hệ thống" down towards this highlighted area.

Chào mừng: **admin**

Log Out

Tên đăng nhập vào hệ thống



Session - Ứng dụng cho Login



Bước 1: Tạo trang **index.php** yêu cầu người dùng đăng nhập.

Bước 2: Tạo trang **login.php** xử lý thông tin đăng nhập từ trang **index.php**

- Kết nối với CSDL, kiểm tra thông tin đăng nhập có hợp lệ hay không ?
 - Nếu **không hợp lệ** thì cho redirect về trang **index.php**.
 - Nếu **hợp lệ** thì
 - Dùng một biến trong **Session** để lưu tên người dùng đã đăng nhập
 - Redirect đến trang “**profile.php**”

Bước 3: Trang “**profile.php**” lấy Session xuống và kiểm tra trong CSDL có tồn tại user này hay không. Nếu không có thì redirect về trang **index.php**

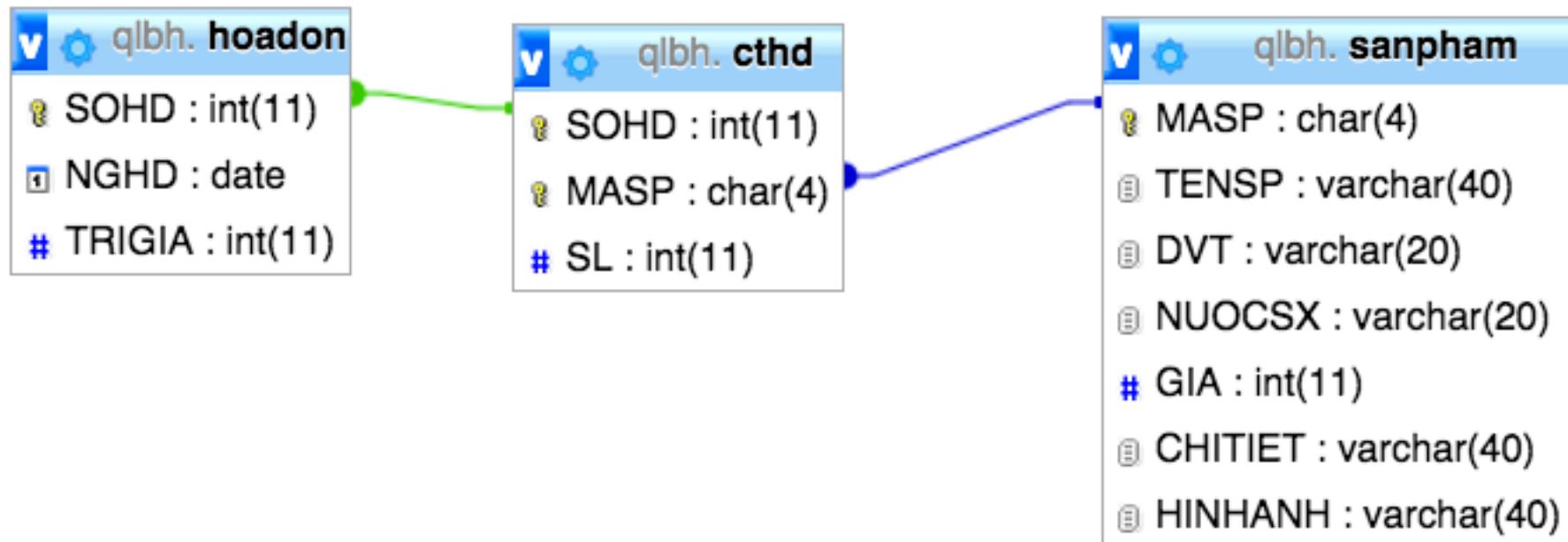
Bước 4: Tạo trang **logout.php**

- Hủy tất cả các biến trong session
- Redirect về trang **index.php**



Hoạt động nhóm

- ❖ Xét CSDL gồm các lược đồ quan hệ sau



Hoạt động nhóm



- ❖ Các nhóm đưa ra quy trình lưu trữ và xử lý các chức năng “*tiếp tục mua hàng*” và “*thanh toán*” liên quan đến “*giỏ hàng*”
- ❖ Giao diện của các trang như sau:
 - ❖ Trang “[index.php](#)”
 - ❖ Trang “[giohang.php](#)”
 - ❖ Trang “[thanhtoan.php](#)”



Bài tập nộp 3



- ❖ Mỗi sinh viên tiếp tục hoàn thiện chức năng “Xóa” và “Cập nhật” trên giỏ hàng
- ❖ Viết trang Admin cho phần xử lý giỏ hàng như sau:

STT	Ngày lập	Tổng giá	Chức năng		
1	2015-05-05	7000	Xem	Xóa	Sửa
2	2015-05-06	29000	Xem	Xóa	Sửa
3	2015-05-06	215000	Xem	Xóa	Sửa
4	2015-05-06	5000	Xem	Xóa	Sửa

- ❖ Thời hạn nộp trên moodle, kể từ hôm nay đến hết ngày 14/5/2015



Bài tập nộp 3(tt)



- ❖ Trang “*xem.php*” có giao diện sau

Tên sản phẩm	Số lượng	Đơn giá/SP	Tổng giá
Laptop Charger	2	100000	200000
Atech Mouse	3	5000	15000
Tổng tiền			215000

- ❖ Trang “*sua.php*” có giao diện sau

Giỏ hàng				
Tên sản phẩm	Giá	Số lượng	Thành tiền	Xóa
IBM CDROM Drive	7000	1	7000	Xóa
Tổng cộng: 7000				

Cập nhật





Bài tập

- ❖ Thực hiện các bài tập từ 1 đến 10 trong chương 5 của tập tin bài tập đã được cung cấp từ buổi đầu tiên của môn học

