

# CHƯƠNG 5

# HTML FORM VÀ XỬ LÝ FORM

# NỘI DUNG CHƯƠNG 5

**5.1. FORM VÀ CÁC THUỘC TÍNH**

**5.2. CÁC KIỂU PHẦN TỬ TRÊN FORM**

**5.3. TRUY CẬP DỮ LIỆU ĐƯỢC GỬI TỪ FORM**

# Form và các thuộc tính

## FORM

- Nhận các giá trị mà người dùng nhập vào
- Có thể gửi dữ liệu và Server xử lý, thông qua phương thức GET, POST.

Đăng ký thành viên

Tên đăng nhập: (\*)

Mật khẩu: (\*)  [Kiểm tra](#)

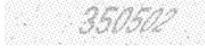
Xác nhận mật khẩu: (\*)

Địa chỉ email: (\*)

Xác nhận email: (\*)

Ngày sinh: (\*)  1  1  1910

Giới tính: (\*)  Nam  Nữ

Mã xác nhận: (\*) 

Nhập mã xác nhận: (\*)

Vui lòng đọc kỹ thỏa thuận sử dụng bên dưới để tiếp tục quá trình đăng ký.

Nghe nhạc Online, chắc hẳn nhiều bạn trong chúng ta đã quá quen với hình thức nghe nhạc này rồi phải không? Vừa chat, vừa tung tảng vào các website trên mạng, lại vừa nghe những bản nhạc hay thì còn gì bằng? ^^ Đặc biệt là với sự ra đời của dịch vụ ADSL thì việc nghe nhạc Online lại càng trở nên dễ dàng hơn...

Tôi hoàn toàn chấp nhận [Thỏa thuận sử dụng](#) của NgheNhac.info

[Đăng ký](#) [Nhập lại](#)

**TÌM NHANH**

**Tìm kiếm theo:**

- Toàn bộ nhóm hàng -
- Toàn bộ nhóm hàng -
- Thời trang Nam
- Thời trang trẻ em
- Thời trang nữ**
- Đồng phục
- Trang phục thể thao
- Phụ Trang

» Xem giỏ hàng » Thanh toán

# Form và các thuộc tính

- Thẻ <FORM> được sử dụng để tạo một vùng trên trang như một biểu mẫu
- Ví dụ:

```
<form action="form_action.php" method="get" name="main_form">
    First name: <input type="text" name="fname" /><br />
    Last name: <input type="text" name="lname" /><br />
    Your image: <input type="file" name="pic" id="pic" /><br />
    <input type="submit" value="Submit" />
</form>
```

# FORM VÀ CÁC THUỘC TÍNH

## ■ Thuộc tính action

- Thuộc tính này định nghĩa hành động (action) được thực thi khi dữ liệu form được gửi đi. Thông thường, dữ liệu form được gửi đến 1 trang web trên server khi người dùng nhấn nút submit
- Nếu thuộc tính action bị trống (khuyết), thì giá trị của thuộc tính sẽ thiết lập là trang hiện tại chứa form
- Trong ví dụ dưới đây, dữ liệu form sẽ được gửi đến trang xuly.php ở server. Trang này chứa 1 số đoạn kịch bản để xử lý dữ liệu form.

```
<form action="xuly.php">
```

# Form và các thuộc tính

## ■ Thuộc tính method

□ Thuộc tính method mô tả phương thức HTTP (GET hoặc POST) được dùng khi gửi dữ liệu form

□ GET:

- Phương thức mặc định khi gửi dữ liệu form là GET. Khi dùng GET, dữ liệu form sẽ hiển thị trên địa chỉ URL,
- Ví dụ như:

<http://www.dammio.com/.../xuly.php?ho=Helmi&ten=Dammio>

# Form và các thuộc tính

## ■ Thuộc tính method

### □ GET:

- Cặp thuộc tính name/value tương ứng ở form sẽ hiển thị ở URL, chẳng hạn như (name/ho) và (value/Helmi)
- Chiều dài của URL tối đa khoảng 3000 ký tự
- Không dùng GET để gửi dữ liệu nhạy cảm (ví dụ số tài khoản ngân hàng, mật khẩu)
- GET dùng tốt hơn cho dữ liệu không cần bảo mật, như các chuỗi truy vấn ở Google

# Form và các thuộc tính

## ■ Thuộc tính method

### □ POST

- Phương thức POST không hiển thị ở địa chỉ URL
- Nếu có 1 form chứa các dữ liệu nhạy cảm hoặc cần bảo mật, hãy dùng POST
- POST không có giới hạn kích thước gửi đi và có thể dùng để gửi lượng lớn dữ liệu.

# Form và các thuộc tính

## ■ Thuộc tính name

- Khai báo một tên riêng cho form

# Form và các thuộc tính

- **Gom nhóm dữ liệu form với fieldset:** Phần tử fieldset dùng để gom nhóm các dữ liệu liên quan trong 1 form. Phần tử legend dùng để định nghĩa nhãn cho phần tử fieldset.

```
<form action="xuly.php">
  <fieldset>
    <legend>Thông tin cá nhân:</legend>
    Tên:<br>
    <input type="text" name="ten" value="Dammio"><br>
    Họ:<br>
    <input type="text" name="họ" value="Helmi"><br><br>
    <input type="submit" value="Submit">
  </fieldset>
</form>
```

# Các phần tử của Form

- Phần tử form định nghĩa 1 form (mẫu) được dùng để thu thập dữ liệu từ người dùng.
- Định dạng:

<form>

Các phần tử form

</form>

- Một HTML form chứa nhiều phần tử form bên trong. Phần tử form có nhiều loại khác nhau như các dạng phần tử input, checkbox, radio, submit,...

# Các phần tử của Form

**ĐĂNG KÝ THÀNH VIÊN**  
**THÔNG TIN TÀI KHOẢN**

Username:

Password:

Nhập lại password:

**THÔNG TIN CÁ NHÂN**

Họ và tên:

Giới tính:

Địa chỉ:

Ngày sinh:

Email:

Tình trạng hôn nhân

độc thân  đã đính hôn  
 ly dị  quá phụ

# Các phần tử của Form

## ■ Phần tử INPUT

Thuộc tính	Mô tả
TYPE	Có thể chọn một trong các loại phần tử như: TEXT, PASSWORD, CHECKBOX, RADIO, SUBMIT, RESET, FILE, HIDDEN và BUTTON. Mặc định là TEXT.
VALUE	Đây là thuộc tính tùy chọn, dùng để thiết lập giá trị khởi tạo của điều khiển.
SIZE	Thuộc tính này xác định độ rộng ban đầu của điều khiển. Đối với kiểu là TEXT hay PASSWORD thì kích thước được xác định theo ký tự. Đối với các loại phần tử nhập khác, độ rộng được xác định bằng điểm (pixels)

# Các phần tử của Form

Thuộc tính	Mô tả
MAXLENGTH	Để xác định số ký tự lớn nhất có thể nhập vào phần tử TEXT hoặc PASSWORD. Mặc định là không giới hạn.
CHECKED	Thuộc tính logic để xác định phần tử có được chọn hay không. Thuộc tính này được sử dụng khi kiểu nhập là RADIO hay CHECKBOX.
NAME	Thuộc tính này xác định tên của điều khiển.

# Các phần tử của Form

- **Input văn bản:** Sử dụng type="text" để định nghĩa trường văn bản 1 dòng

Tên:

Họ:

```
<form>
    Tên:<br>
    <input type="text" name="firstname"><br>
    Họ:<br>
    <input type="text" name="lastname">
</form>
```

# Các phần tử của Form

- **Input mật khẩu:** Sử dụng type="password" để định nghĩa input dạng password

Họ tên:

Dammio

Mật khẩu:

.....|

```
<form>
    Họ tên:<br>
    <input type="text" name="username"><br>
    Mật khẩu:<br>
    <input type="password" name="psw">
</form>
```

# Các phần tử của Form

## ■ Input submit:

- Sử dụng type="submit" để định nghĩa nút nhấn cho việc gửi (submit) dữ liệu form đến một xử lý form (form-handler).
- Form-handler điển hình là một trang web server với một đoạn mã kịch bản để xử lý dữ liệu đầu vào và được đặc tả ở thuộc tính action của form.

# Các phần tử của Form

Tên:

Họ:

```
<form action="/form_action.php">
Tên:<br>
<input type="text" name="firstname" value="Dammio"><br>
Họ:<br>
<input type="text" name="lastname" value="Ta"><br><br>
<input type="submit" value="Gửi">
</form>
```

# Các phần tử của Form

## ■ Input reset:

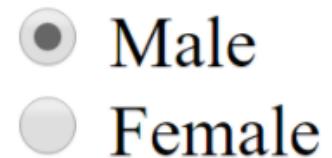
- Định nghĩa một nút thiết lập lại tất cả dữ liệu form về giá trị mặc định.

```
<form action="/action_page.php">  
Họ:<br>  
<input type="text" name="firstname" value="Dammio"><br>  
Tên:<br>  
<input type="text" name="lastname" value="Ta"><br><br>  
<input type="submit" value="Submit">  
<input type="reset">  
</form>
```

# Các phần tử của Form

## ■ Input radio:

- Để định nghĩa kiểu radio thì có thể dùng input type="radio"
- Các radio cho phép người dùng chọn 1 trong các lựa chọn



```
<form>
  <input type="radio" name="gender" value="nam" checked> Male<br>
  <input type="radio" name="gender" value="nữ"> Female<br>
</form>
```

# Các phần tử của Form

## ■ Input checkbox:

- Để định nghĩa kiểu checkbox thì có thể dùng input type="checkbox"
- Các checkbox cho phép người dùng không chọn hoặc nhiều lựa chọn

- Blue color
- Red color

```
<form>
  <input type="checkbox" name="color1" value="blue"> Blue color<br>
  <input type="checkbox" name="color2" value="red"> Red color
</form>
```

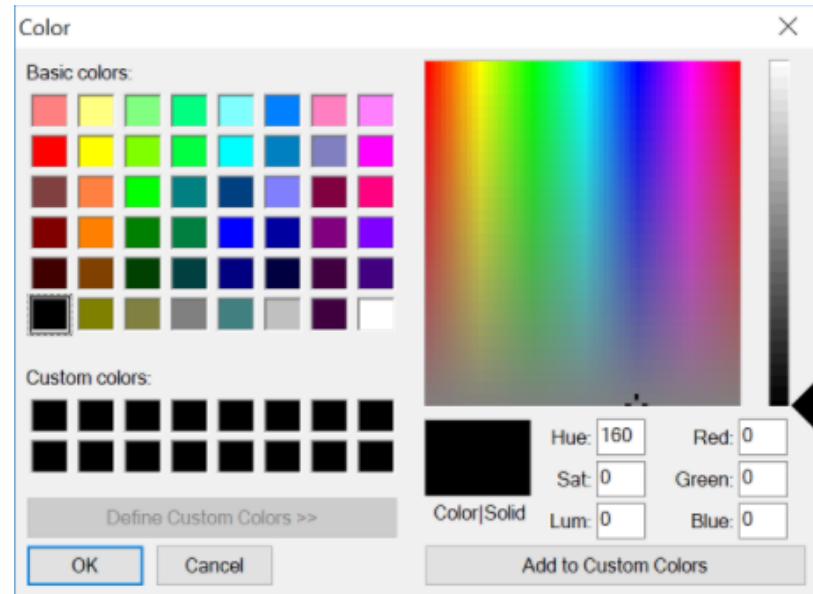
# Các phần tử của Form

- **Input button:** Để định nghĩa một nút nhấn, có thể dùng input type="button"

```
<form>
  <input type="button" onclick="alert('Xin chào!')" value="Nhấn nút!">
</form>
```

# Các phần tử của Form

- **Input color:** Dạng input này được dùng cho các trường nhập liệu chứa một màu sắc nào đó.



```
<form>
  <input type="color" name="favcolor">
</form>
```

# Các phần tử của Form

- **Input date:** Để hiển thị ngày, bạn hãy dùng đoạn mã input type="date"

```
<form>
    Chọn ngày trước 1980-01-01:
    <input type="date" name="bday" max="1979-12-31"><br>
    Chọn ngày trước 2000-01-01:
    <input type="date" name="bday" min="2000-01-02"><br>
</form>
```

# Các phần tử của Form

- **Input datetime-local:** Để đặc tả trường nhập liệu chứa ngày và giờ mà không có múi giờ, bạn hãy dùng input type="datetime-local".



11/11/1999 09:23 SA ✕ ▲ ▼

```
<form>
<input type="datetime-local" name="bdaytime">
</form>
```

# Các phần tử của Form

- **Input month:** Tương tự ngày, đoạn mã input type="month" cho phép người dùng chọn tháng và năm.

```
<form>
<input type="month" name="bdaymonth">
</form>
```

# Các phần tử của Form

- **Input email:** Để định nghĩa một trường nhập liệu chỉ chứa địa chỉ email, hãy dùng input type="email".

```
<form>  
<input type="email" name="email">  
</form>
```

# Các phần tử của Form

## ■ Input number

- Nếu muốn trường nhập liệu chỉ là kiểu số (number), có thể định nghĩa input type="number".
- Bạn cũng có thể thiết lập các giới hạn vùng số được chấp nhận. Ví dụ chỉ cho phép người dùng nhập các số từ 1 đến 10.

```
<form>
<input type="number" name="quantity" min="1" max="10">
</form>
```

# Các phần tử của Form

## ■ Các giới hạn nhập liệu

- disabled: vô hiệu hóa trường nhập liệu
- max: giá trị tối đa của trường nhập liệu
- maxlength: số ký tự tối đa của trường nhập liệu
- min: giá trị tối thiểu của trường nhập liệu
- readonly: đặc tả trường nhập liệu chỉ đọc (không được thay đổi)
- required: đặc tả trường nhập liệu phải có dữ liệu
- size: đặc tả chiều rộng (số ký tự) của trường nhập liệu
- step: đặc tả khoảng số hợp lệ của trường nhập liệu
- value: đặc tả giá trị mặc định

# Các phần tử của Form

- **Ví dụ:** Hiển thị một trường nhập liệu số với giá trị từ 0 đến 100, mỗi bước cách nhau 10 đơn vị. Giá trị mặc định là 30.

```
<form>
<input type="number" name="points" min="0" max="100" step="10"
value="30">
</form>
```

# Các phần tử của Form

- **Input tel:** Đoạn mã input type="tel" để định nghĩa trường nhập liệu chứa số điện thoại. Dạng tel hiện chỉ hỗ trợ ở trình duyệt Safari 8.

```
<form>  
<input type="tel" name="usrtel">  
</form>
```

# Các phần tử của Form

## ■ Phần tử TextArea

Phần tử này tạo ra một vùng cho phép nhập văn bản trên nhiều dòng so với hộp văn bản nhập một dòng. Ta phải xác định kích thước của TextArea. Ta cũng phải xác định số dòng, số cột trong TextArea. Phải kết thúc phần tử với thẻ đóng </TEXTAREA>

## ■ Ví dụ:

```
<TEXTAREA NAME = "Text1" COLS = 20 ROWS = 7>  
</TEXTAREA>
```

# Các phần tử của Form

Thuộc tính	Mô tả
COLS	Truy xuất độ rộng của TextArea
ROWS	Thiết lập hoặc truy xuất số hàng ngang trong <TEXTAREA>
SIZE	Thiết lập hoặc truy xuất kích thước của điều khiển
TYPE	Truy xuất loại điều khiển, sử dụng giá trị <TEXTAREA>
VALUE	Thiết lập hoặc truy xuất giá trị của TEXTAREA

# Các phần tử của Form

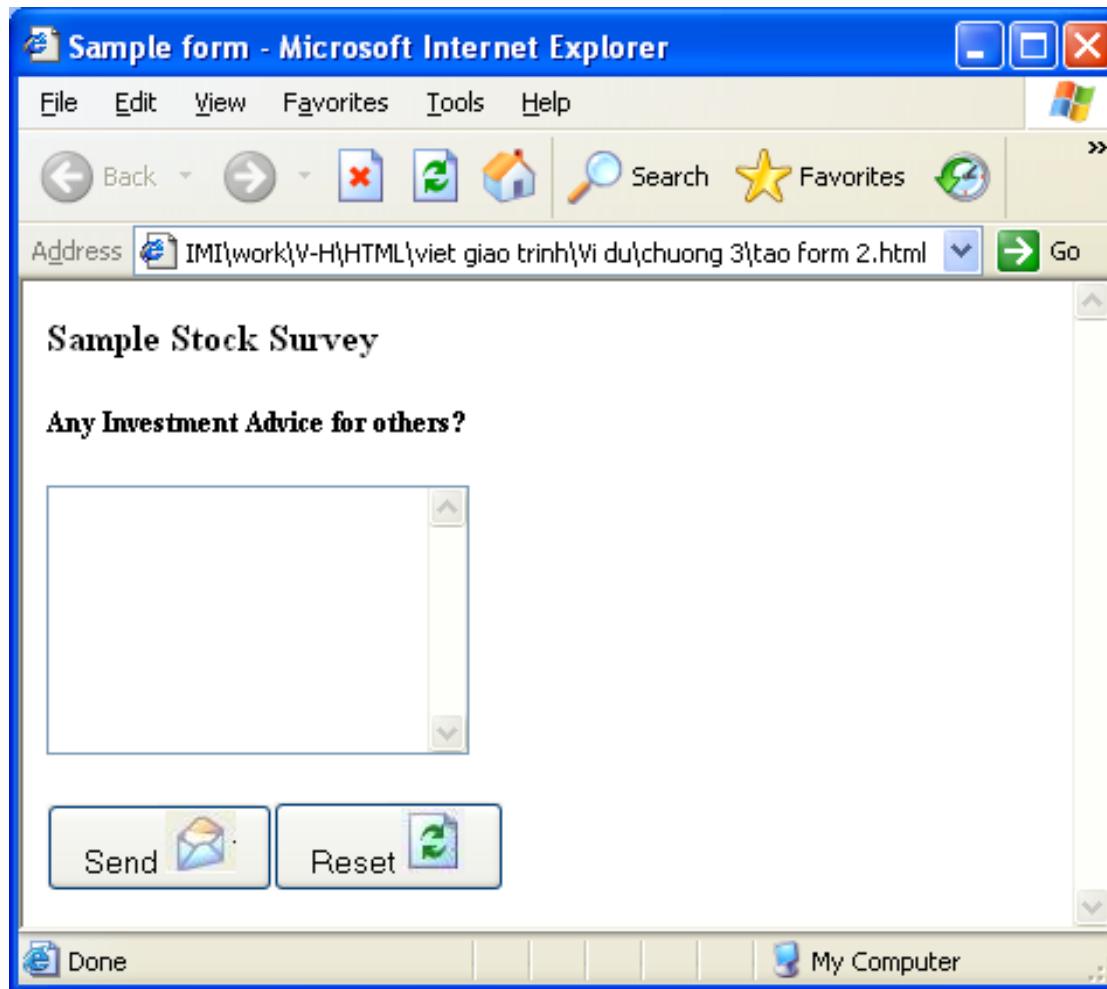
## ■ Phần tử Button

- Phần tử này tạo ra điều khiển nút giống như khi sử dụng phần tử INPUT với thuộc tính type=button.
- Nếu phần tử BUTTON có thuộc tính type=submit thì lại giống với phần tử INPUT khi nhận thuộc tính type=submit.
- Sự khác nhau ở chỗ phần tử INPUT có dạng một ảnh “phẳng” trong khi phần tử BUTTON thì hiển thị như một nút có hiệu ứng lên/xuống khi nhấp vào.

Ví dụ:

```
<BUTTON TYPE = “submit” NAME = “Submit” VALUE =  
“Submit”> Send <IMG src = “send.jpg”> </BUTTON>  
<BUTTON TYPE = “reset” NAME = “Reset” VALUE = “Reset”>  
Reset <IMG src=“reset.jpg”> </BUTTON>
```

# Các phần tử của Form



# Các phần tử của Form

## ■ Phần tử Select

- Phần tử SELECT được sử dụng để hiển thị một danh sách các lựa chọn cho người dùng. Mỗi lựa chọn được biểu diễn bởi phần tử OPTION.
- Một phần tử SELECT phải chứa ít nhất một phần tử OPTION. Thành phần được chọn lựa sẽ hiển thị với màu khác so với các thành phần còn lại.

```
<SELECT NAME = "Select example">
    <OPTION></OPTION>
    <OPTION>1. Online</OPTION>
    <OPTION>2. Touch Tone Trading</OPTION>
    <OPTION>3. Broker Assisted</OPTION>
    <OPTION>Other</OPTION>
</SELECT>
```

# Các phần tử của Form



# Các phần tử của Form

Thuộc tính	Mô tả
NAME	Gán tên cho phần tử. Khi biểu mẫu được gửi đi, thuộc tính <i>tên</i> được gán với giá trị chọn lựa.
SIZE	Nếu có nhiều sự chọn lựa, người dùng sử dụng chức năng cuộn, thuộc tính này xác định <i>số dòng trong danh sách</i> được hiển thị
MULTIPLE	Là thuộc tính logic cho phép người dùng chọn <i>một hoặc nhiều</i> chọn lựa.

# Các phần tử của Form

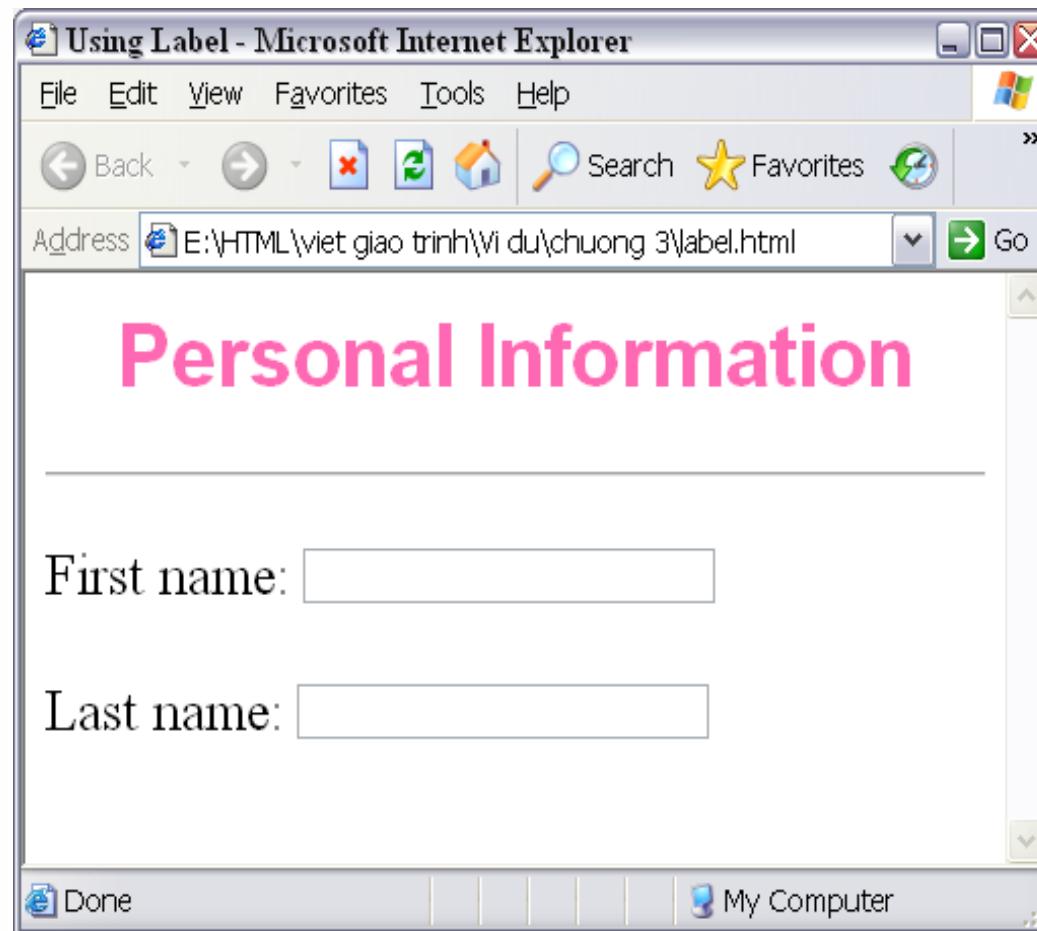
## ■ Phần tử Label

- Phần tử LABEL được sử dụng để gán nhãn cho các phần tử điều khiển. Chẳng hạn, ta có thể gắn nhãn vào phần tử TEXT khi trang hiển thị.
- Ta phải xác định thuộc tính ID của điều khiển mà nó được gắn vào.

Ví dụ:

```
<LABEL for="firstname"> First name:</LABEL>
<INPUT type="text" id="firstname"> <BR><BR>
<LABEL for="lastname"> Last name: </LABEL>
<INPUT type="text" id = "lastname"> <BR>
```

# Các phần tử của Form



# Điều khiển các phần tử trên Form

## ■ Thiết lập thứ tự tab

- Thuộc tính tabindex của một phần tử xác định trình tự phần tử nhận tiêu điểm thông qua bàn phím, bao gồm cả các phần tử được lồng vào các phần tử khác.
- Giá trị có thể là bất cứ số nào giữa 0 và 32767. Nếu gán trùng tabindex, các phần tử nhận tiêu điểm theo thứ tự mà nó xuất hiện trong tài liệu.
- Nếu phần tử nào không hỗ trợ thuộc tính tabindex, nó sẽ nhận tiêu điểm cuối cùng. Nếu ta vô hiệu hóa một phần tử, nó sẽ có thứ tự tab và sẽ không nhận được tiêu điểm.

### Ví dụ:

```
<INPUT tabindex=2 TYPE=RADIO NAME="CONTROL1"  
      VALUE="0" CHECKED> Web Designer  
<INPUT tabindex=6 TYPE=SUBMIT VALUE=OK>
```

# Truy cập dữ liệu được lấy từ Form

- PHP cho phép ta lấy giá trị dựa vào 2 phương thức POST và GET.
- Đối với Form sử dụng method là POST ta lấy giá trị của Form bằng : \$giatri = \$\_POST['Field\_Name'].
- Đối với Form sử dụng method là GET ta lấy giá trị của Form bằng : \$giatri = \$\_GET['Field\_Name'].
- Cũng là một kiểu lấy thông tin từ người dùng. \$\_REQUEST bao hàm luôn cả \$\_GET và \$\_POST.  
Lưu ý: khi dùng cái này thì bất kỳ gì được truyền qua POST hay GET đều được \$\_REQUEST thu thập.

# GET

## Truy cập dữ liệu được lấy từ Form

```
<?php
    if( $_GET["name"] || $_GET["age"] )
    {
        echo "Chào mừng ". $_GET['name']. "<br />";
        echo "Bạn ". $_GET['age']. " tuổi.";
        exit();
    }
?>
<html>
    <body>
        <form action="<?php $_PHP_SELF ?>" method="GET">
            Họ tên: <input type="text" name="name" />
            Tuổi: <input type="text" name="age" />
            <input type="submit" />
        </form>
    </body>
</html>
```

# POST

# Truy cập dữ liệu được lấy từ Form

```
<?php
    if( $_POST["name"] || $_POST["age"] )
    {
        if (preg_match("/[^A-Za-z'-]/",$_POST['name'] ))
        {
            die ("Biến name không hợp lệ - nên là các chữ cái");
        }
        echo "Chào mừng ". $_POST['name']. "<br />";
        echo "Bạn ". $_POST['age']. " tuổi.";
        exit();
    }
?>
<html>
    <body>
        <form action="<?php $_PHP_SELF ?>" method="POST">
            Họ tên: <input type="text" name="name" />
            Tuổi: <input type="text" name="age" />
            <input type="submit" />
        </form>
    </body>
</html>
```

# Truy cập dữ liệu được lấy từ Form

## REQUEST

```
<?php  
    if( $_REQUEST["name"] || $_REQUEST["age"] )  
    {  
        echo "Chào mừng ". $_REQUEST['name']. "<br />";  
        echo "Bạn ". $_REQUEST['age']. " tuổi.";  
        exit();  
    }  
?>  
<html>  
    <body>  
        <form action="<?php $_PHP_SELF ?>" method="POST">  
            Họ tên: <input type="text" name="name" />  
            Tuổi: <input type="text" name="age" />  
            <input type="submit" />  
        </form>  
    </body>  
</html>
```

# Truy cập dữ liệu được lấy từ Form

## ■ Xây dựng Form

```
2 <html>
3 <form action = "processform.php" method = post>
4 Please type your name here : <BR>
5 <input type = text name = "username"><BR><BR>
6 <input type = submit value = "Submit Data">
7 </form>
8 </html>
9
```

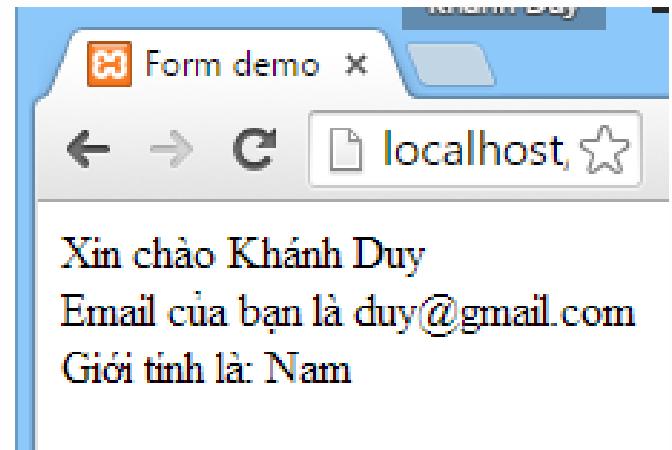
Khi submit dữ liệu sẽ chuyển đến file processform.php. Để lấy giá trị chúng ta sử dụng:

```
$username = $_POST['username'];
```

# Truy cập dữ liệu được lấy từ Form

## ■ Xây dựng Form Html

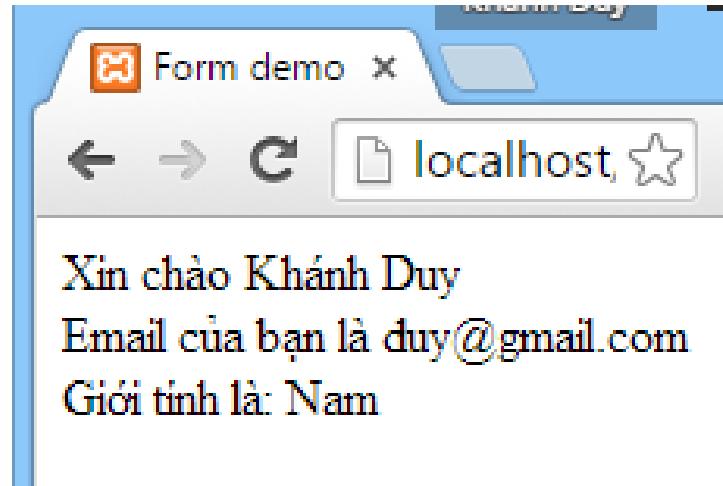
```
<html>
<head>
<meta charset="utf-8" />
<title>Demo form</title>
</head>
<body>
<form method="POST" action="xuly.php">
Họ và tên: <input type="text" name="hoten" /><br />
Email: <input type="text" name="email" /><br />
Giới tính: <input type="radio" name="gioitinh" value="Nam" /> Nam
<input type="radio" name="gioitinh" value="Nữ" /> Nữ<br />
<input type="submit" value="Gửi" />
</form>
</body>
</html>
```



# Truy cập dữ liệu được lấy từ Form

## ■ Xây dựng mã lệnh xuly.php

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<title>Form demo</title>
</head>
<body>
Xin chào <?php echo $_POST['hoten']; ?><br />
Email của bạn là <?php echo $_POST['email']; ?><br />
Giới tính là: <?php echo $_POST['gioitinh']; ?>
</body>
</html>
```



# Truy cập dữ liệu được lấy từ Form

## ■ Xây dựng mã lệnh xuly.php

```
<td>
<input type="submit" name="save" class="noborder" id="save" value="Save" alt="Save"
tabindex="4" />

</td>

<td>

<input type="submit" name="publish" class="noborder" id="publish" value="Publish"
alt="Publish" tabindex="5" />

</td>
```

# Truy cập dữ liệu được lấy từ Form

## \$ \_GET

Được sử dụng khi dùng URL để gửi yêu cầu lên server.  
Có thể không cần Form

Lấy thông tin một trang web (Web page), nếu trên thanh  
địa chỉ có cặp biến = giá trị . Ví  
dụ *index.php?loaitin=thethao*, nếu ta  
thay *thethao* bằng *tinthegioi* thì sẽ lấy được các loại tin  
thuộc *tinthegioi*

Được dùng để gửi dữ liệu trong các form đến một trang  
nào đó. Dữ liệu gửi đi là một phần trong URL với dạng  
“&bién=gia\_trí”

Dữ liệu trong form phải là chuỗi hoặc số, không gửi  
được file hình

Dữ liệu gửi đi là hạn chế vì phụ thuộc vào trình duyệt  
hỗ trợ tối đa URL có độ dài bao nhiêu.

Có thể thêm dữ liệu gửi đi trực tiếp vào URL  
Bảo mật kém.

## \$ \_POST

Bắt buộc phải có Form. Vì thế không thể gửi một  
yêu cầu Post từ lần truy cập đầu tiên vào trang web.

Dữ liệu gửi đi không hiển thị. Do vậy, không thể  
thay giá trị trên thanh địa chỉ được.

Dữ liệu gửi đi không hiển thị trên thanh địa chỉ, nó  
được chuyển đến server âm thầm.

Nếu muốn gửi hình lên server buộc phải dùng POST

Dữ liệu gửi đi lớn. Các kiểu File, Ảnh, Binary đều có  
thể gửi đi được, dung lượng của chúng phụ thuộc vào  
cấu hình của server.

Dữ liệu được gửi đi như trong form của nó  
Bảo mật hơn Get

# BÀI THỰC HÀNH

## ■ Bài 1:

- Xây dựng Form login
- Nếu nhập user và pass là admin và 123456 thì thông báo “Xin chào Admin”
- Nếu user và pass khác với admin và 123456 thì thông báo “Đăng nhập không hợp lệ”

# BÀI THỰC HÀNH

## ■ Bài 2:

- Xây dựng Form tính toán
- Nhập số thứ 1, chọn phép toán, nhập số thứ 2
- Tính toán và in kết quả