

Buổi 3

HTML Form

HTML5

I. Tổng quan về Form

Biểu mẫu (*form*) là một tập hợp các trường dữ liệu (*field*) mà người dùng có thể nhập vào nhằm mục đích thu thập thông tin, dữ liệu từ phía người dùng. Trong phần mềm, những trường dữ liệu này sẽ được biểu diễn bằng các *control*.

Trong HTML, để tạo 1 form, chúng ta dùng thẻ `<form>` với cú pháp như sau:

```
<form method="get" action="code.php">  
    <!-- Các control của form -->  
</form>
```

Trong đó:

- Thuộc tính `method` quy định cách gửi dữ liệu từ trình duyệt lên máy chủ là `get` hay `post`.
- Thuộc tính `action` cho biết dữ liệu sẽ được gửi đến trang web nào.

Thao tác gửi dữ liệu lên máy chủ được gọi là thao tác *submit*.

Ảnh minh họa 1 form đăng kí tài khoản:



Tạo Tài khoản Google

Tiếp tục tới Gmail

Bạn có thể sử dụng chữ cái, số và dấu chấm

Sử dụng 8 ký tự trở lên và kết hợp chữ cái, chữ số và biểu tượng

[Đăng nhập](#)

[Tiếp theo](#)



Một tài khoản. Hoạt động trên mọi sản phẩm và dịch vụ của Google.

II. Các control trong form

Control là những thứ dùng để thu thập dữ liệu hoặc hành vi từ người dùng khi tương tác với trang web, ví dụ như `TextBox`, `Button`, `RadioButton`, `CheckBox`...

1. Thẻ <input>

Thẻ <input> dùng để tạo 1 control tùy thuộc vào giá trị thuộc tính type:

Tên control	type	Cú pháp ví dụ
CheckBox	checkbox	<input type="checkbox"> Nhớ mật khẩu
ColorPicker	color	<input type="color">
DatePicker	date	<input type="date">
FileUpload	file	<input type="file">
ImageButton	image	<input type="image">
MonthPicker	month	<input type="month">
Numeric	number	<input type="number">
RadioButton	radio	<input type="radio"> Nam
Range	range	<input type="range">
TextBox	text	<input type="text">
	password	<input type="password">
	email	<input type="email">
TimePicker	time	<input type="time">
WeekPicker	week	<input type="week">

Snippet để tạo nhanh thẻ <input> là input:* (với * là giá trị thuộc tính type muốn tạo, ví dụ: input:text, input:password...).

Một số thuộc tính của thẻ <input>:

Tên thuộc tính	Ý nghĩa
autocomplete	Dữ liệu của control được lưu lại để autocomplete
autofocus	Control được focus khi load trang (TextBox)
checked	Control được chọn sẵn khi load trang (CheckBox hoặc RadioButton)
disabled	Control bị khóa, không thể sử dụng
max, min	Giá trị tối đa và tối thiểu được nhập
maxlength	Số lượng kí tự tối đa được nhập
name	Quy định tên (tự đặt) của trường dữ liệu khi được gửi lên máy chủ.
pattern	Quy định biểu thức RegEx ¹ để kiểm tra dữ liệu
placeholder	Text gợi nhớ khi nhập liệu (TextBox)
readonly	Dữ liệu của control không thể sửa được
required	Bắt buộc phải nhập dữ liệu cho control
size	Chiều rộng của control
step	Bước tăng của số (Numeric hoặc Range)
value	Giá trị mặc định của control, cũng là giá trị sẽ được gửi lên máy chủ.

Lưu ý: Chỉ có những control có thuộc tính name mới được gửi giá trị lên máy chủ khi form được submit. Mỗi control sẽ được gửi 1 cặp name – value.

¹ Regular Expression (RegEx) sẽ được học ở phần JavaScript

2. Thẻ <select>

Thẻ <select> tạo ra 1 menu xổ xuống (*drop-down list*) chứa nhiều giá trị có thể chọn được.

Cú pháp:

```
<select name="list">
  <option value="a">Giá trị 1</option>
  <option value="b">Giá trị 2</option>
  <option value="c">Giá trị 3</option>
  <option value="d">Giá trị 4</option>
</select>
```

Có thể thêm thuộc tính `selected` cho một trong số thẻ <option> để mặc định chọn giá trị đó, hoặc thuộc tính `multiple` cho thẻ <select> để cho phép chọn nhiều lựa chọn.

3. Thẻ <textarea>

Thẻ <textarea> tạo ra một vùng nhập liệu nhiều dòng.

Cú pháp:

```
<textarea name="message" rows="10" cols="30">
</textarea>
```

Trong đó: Thuộc tính `rows` và `cols` quy định kích thước số dòng và số cột.

4. Thẻ <button>

Thẻ <button> tạo ra 1 button, cũng có 3 dạng là `button`, `reset`, `submit` tương tự thẻ <input>.

Cú pháp:

```
<button type="button">Click Me!</button>
```

Sự khác nhau giữa thẻ <button> và thẻ <input> đó là thẻ <button> có thể mở và thẻ đóng, do đó có thể viết nội dung cho button đó, thậm chí chèn thêm các thẻ khác ở giữa thẻ <button> (ví dụ như hình ảnh), còn thẻ <input> chỉ có thẻ mở nên không làm được như vậy.

5. Thẻ <fieldset> và <legend>

Thẻ <fieldset> dùng để gom nhóm các control lại với nhau về mặt giao diện. Thẻ <legend> dùng để đặt tiêu đề cho nhóm đó.

Cú pháp:

```
<fieldset>
  <legend>Thông tin tài khoản</legend>
  <!-- Các control muốn cài đặt -->
</fieldset>
```

III. Submit dữ liệu lên máy chủ: GET vs. POST

Dữ liệu được gửi từ form của trình duyệt ở máy khách (*client*) lên máy chủ (*server*) có thể theo 2 cách: GET và POST.

1. GET

Dữ liệu được gửi theo dạng GET sẽ được chèn vào ngay trong URL của trang web đích theo từng cặp `name=value`, cách nhau bởi dấu `&`, cách URL chính bởi dấu `?`.

Ví dụ: Sau khi gõ từ khóa tìm kiếm là "home" vào khung tìm kiếm của Google, chọn ngôn ngữ tìm kiếm là tiếng Anh, Google chuyển hướng sang 1 trang web khác có URL như sau:

<https://www.google.com/search?q=home&lang=en>

Dữ liệu được gửi theo dạng GET có những đặc điểm sau:

- Không bảo mật.

- Có giới hạn (URL dài tối đa khoảng 3000 kí tự).
- Hữu ích khi người dùng muốn lưu lại địa chỉ trang web.

Snippet để tạo form dạng GET là `form:get`.

2. POST

Dữ liệu được gửi theo dạng POST không chèn vào URL của trang web đích mà được gửi bằng các gói tin qua mạng.

Dữ liệu được gửi theo dạng POST có những đặc điểm sau:

- Bảo mật.
- Không giới hạn.
- Không thể lưu lại địa chỉ trang web.

Snippet để tạo form dạng POST là `form:post`.

Lưu ý: Cho dù là theo cách nào, chỉ có những control có thuộc tính name mới được gửi dữ liệu lên máy chủ khi form được submit.

IV. Tổng quan về HTML5

Với các phiên bản cũ của HTML (4.01 trở về trước), việc thêm các đối tượng đa phương tiện như đồ họa, âm thanh, video cần phải có plug-in của hãng thứ ba, điển hình là Adobe Flash Player. Plug-in này đã để lộ nhiều nhược điểm:

- Tốn nhiều tài nguyên hệ thống
- Dễ bị khai thác bởi hacker
- Không hỗ trợ trên các thiết bị iOS như iPhone, iPad.

Do đó, HTML5 ra đời để khắc phục những nhược điểm trên, đồng thời bổ sung một số tính năng mới cũng như loại bỏ các thẻ HTML ít được sử dụng.

Một số API mới do HTML5 cung cấp:

- HTML Geolocation
- HTML Drag and Drop
- HTML Local Storage
- HTML Application Cache
- HTML Web Workers
- ...

Một số thay đổi về các thẻ HTML:

HTML 4.01	HTML5
<acronym>	<abbr>
<applet>	<object>
<basefont>, <big>, <center>, , <strike>, <tt>	Sử dụng CSS để định dạng
<dir>	
<frame>, <frameset>, <noframes>	Bị loại bỏ

Một số thẻ, thuộc tính và giá trị mới:

- Các thẻ ngữ nghĩa (*semantic tag*): <header>, <footer>, <section>, <article>...
- Các control của form: <datalist>, <keygen>, <output>...

- Các kiểu dữ liệu nhập (*input type*) cho form control: number, date, time, range...
- Các thuộc tính cho form control: autocomplete, autofocus, formaction, required...
- Thẻ đồ họa: <svg> và <canvas>
- Thẻ đa phương tiện: <audio>, <video>, <embed>

V. Đồ họa với HTML5

1. Thẻ <svg>

Thẻ <svg> cho phép vẽ các đối tượng đồ họa đơn giản theo dạng Scalable Vector Graphic:

- Hình tròn:

```
<svg width="100" height="100">
  <circle cx="50" cy="50" r="40" stroke="green" stroke-width="4"
    fill="yellow" />
</svg>
```

Trình duyệt không hỗ trợ SVG!

- Hình chữ nhật:

```
<svg width="400" height="100">
  <rect width="400" height="100" style="fill:rgb(0,0,255); stroke-
    width:10; stroke:rgb(0,0,0)" />
</svg>
```

- Hình chữ nhật bo góc:

```
<svg width="400" height="180">
  <rect x="50" y="20" rx="20" ry="20" width="150" height="150"
    style="fill:red;stroke:black;stroke-width:5;opacity:0.5" />
</svg>
```

- Hình ngôi sao:

```
<svg width="300" height="200">
  <polygon points="100,10 40,198 190,78 10,78 160,198"
    style="fill:lime;stroke:purple;stroke-width:5;fill-
    rule:evenodd;" />
</svg>
```

- Logo đơn giản:

```
<svg height="130" width="500">
  <defs>
    <linearGradient id="grad1" x1="0%" y1="0%" x2="100%"
      y2="0%">
      <stop offset="0%" style="stop-
        color:rgb(255,255,0);stop-opacity:1" />
      <stop offset="100%" style="stop-
        color:rgb(255,0,0);stop-opacity:1" />
    </linearGradient>
  </defs>
  <ellipse cx="100" cy="70" rx="85" ry="55" fill="url(#grad1)" />
  <text fill="#ffffff" font-size="45" font-family="Verdana" x="50"
    y="86">SVG</text>
</svg>
```

2. Thẻ <canvas>

Thẻ <canvas> dùng để chứa các đối tượng đồ họa. Việc vẽ các đối tượng này phải sử dụng mã (thường là JavaScript). Canvas là 1 vùng hình chữ nhật trên trang web, mặc định không có nội dung và không có viền.

Cách vẽ một số đối tượng đồ họa đơn giản bằng <canvas>:

- Hình tròn:

```
<canvas id="myCanvas" width="200" height="100" style="border:2px solid #000000;">
</canvas>
<script>
    var c = document.getElementById("myCanvas");
    var ctx = c.getContext("2d");
    ctx.beginPath();
    ctx.arc(95, 50, 40, 0, 2 * Math.PI);
    ctx.stroke();
</script>
```

- Hình chữ nhật:

```
<canvas id="myCanvas" width="200" height="100" style="border:2px solid #000000;">
</canvas>
<script>
    var c = document.getElementById("myCanvas");
    var ctx = c.getContext("2d");
    ctx.fillStyle = "#FF0000";
    ctx.fillRect(0, 0, 150, 75);
</script>
```

- Văn bản:

```
<canvas id="myCanvas" width="200" height="100" style="border:2px solid #000000;">
</canvas>
<script>
    var c = document.getElementById("myCanvas");
    var ctx = c.getContext("2d");
    ctx.font = "30px Cambria";
    ctx.fillText("Hello World",10,50);
</script>
```

VI. Đa phương tiện với HTML5

HTML5 hỗ trợ các thẻ <audio>, <video> và <embed> để chèn các nội dung đa phương tiện (*multimedia*) vào trang web mà không cần thêm plug-in của bên thứ ba.

1. Thẻ <audio>

Thẻ <audio> dùng để chèn file âm thanh vào trang web.

Cú pháp:

```
<audio src="url" controls></audio>
```

Trong đó:

- Thuộc tính controls để hiển thị trình phát audio.
- Thuộc tính src là đường dẫn đến file audio.

Ví dụ: Xem ví dụ 3.1.

2. Thẻ <video>

Thẻ <video> dùng để chèn file video vào trang web.

Cú pháp:

```
<video src="url" width="600" height="450" controls></video>
```

Trong đó:

- Thuộc tính controls để hiển thị trình phát video.
- Thuộc tính src là đường dẫn đến file video.

Ví dụ: Xem ví dụ 3.2².

3. Thẻ <embed>

Thẻ <embed> dùng để chèn đối tượng vào trang web (thường là file flash hoặc video YouTube).

Cú pháp:

```
<embed src="url" width="w" height="h">
```

Trong đó:

- Thuộc tính src là đường dẫn đến đối tượng cần chèn.
- Thuộc tính width và height là chiều rộng và chiều cao (pixel) của vùng hiển thị đối tượng.

Ví dụ:

- Dùng thẻ <embed> chèn file flash:

```
<embed src="Hello.swf">
```

- Dùng thẻ <embed> để chèn video từ YouTube:

```
<embed width="1024" height="576" src="http://www.youtube.com/embed/ID">
```

với ID là mã của video, ví dụ: <http://www.youtube.com/watch?v=9jboscGmhZQ>

Ví dụ: Xem ví dụ 3.3.

Lưu ý: Ngoài cách chèn bằng thẻ <embed>, video từ YouTube còn có thể chèn bằng thẻ <iframe> hoặc thẻ <object>.

VII. Bài tập

Thiết kế 1 form đăng kí tài khoản gồm các thông tin sau (SV tự chọn loại control phù hợp với từng mục bên dưới):

- Tên tài khoản.
- Mật khẩu và xác nhận mật khẩu.
- Họ tên.
- Email.
- SĐT.
- Địa chỉ: Gồm số nhà, phường/xã, quận/huyện, tỉnh/thành phố.
- Nghề nghiệp: Gồm có Học sinh – Sinh viên, Nhân viên văn phòng, Công nhân, Khác.
- Ngày sinh.
- Giới tính.
- Avatar (chọn file ảnh để upload).
- Thông tin thêm.

² Trong ví dụ này có thêm một số nút chức năng được cài thêm bằng JavaScript. SV có thể bỏ qua các nút này.