

# week::seven

## Introduction to HTML Forms

### HTML Form Components

#### <form> parent element

- Begins the HTML form; Must be closed with a </form> tag.
- It is a block-level element, and some browsers apply padding and margin by default. Remove this using a reset style sheet.
  - Can have many forms on a single page, though forms should not be nested within each other.
- `<form action="url" method="post">...</form>`

#### Methods: GET vs. POST

- **get**
  - Contents of the form submission become part of the URL
  - Is arranged in a `name=value` format
  - All items are preceded by a single question mark (?) and are separated by an ampersand (&).
  - Form submissions can be bookmarked (a good thing).
  - Example: `http://www.google.com/search?client=safari&q=santa+claus`
- **post**
  - Content of the form submission are sent in the body of the request and are hidden from view to the end user (they are not placed in the URL).
  - Post is the preferred method for most form transactions.
  - Best for server-processed data (things going into a database).
  - Also more secure since field values are not displayed in the URL.

#### Comparing the two HTTP methods: GET and POST.

- Source: [http://www.w3schools.com/tags/ref\\_httpmethods.asp](http://www.w3schools.com/tags/ref_httpmethods.asp)

	GET	POST
<b>Back button/Reload</b>	Harmless	Data will be re-submitted
<b>Bookmarked</b>	Can be bookmarked	Cannot be bookmarked
<b>Cached</b>	Can be cached	Not cached
<b>Visibility</b>	Data is visible to everyone in the URL	Data is not displayed in the URL
<b>History</b>	Parameters remain in browser history	Parameters are not saved in browser history
<b>Restrictions on data length</b>	Yes, when sending data, the GET method adds the data to the URL; and the length of a URL is limited (2048 characters)	No restrictions
<b>Restrictions on data type</b>	Only ASCII characters allowed	No restrictions. Binary data is also allowed
<b>Security</b>	GET is less secure compared to POST because data sent is part of the URL. Never use GET when sending passwords or other sensitive information!	POST is a little safer than GET because the parameters are not stored in browser history or in web server logs

**<fieldset> & <legend> tags**

- **Fieldset** represents a logical grouping of input tags.
- **Legend** is a label for a fieldset.
- `<fieldset>`
  - `<legend>Contact Information</legend>`
  - `<input type="text" />`
- `</fieldset>`

**<label> tag**

- Links a piece of HTML text to a specific form field.
- Good practice for accessibility reasons, though not technically required.
- `<label for="input_id">My Label Text</label>`

**Text fields**

- General-purpose text input fields.
- Only allows for one visible line of text.
- `<input type="text" name="name" id="name" placeholder="Name" />`

**Password fields**

- Same as Text fields, but hides from view what is being typed.
- `<input type="password" name="name" id="name" value="" />`

**Text areas**

- Allow for large and wide text input.
- Good for writing paragraphs of information.
- Can set its size by specifying the `cols` and `rows` attributes.
- `<textarea name="name" id="name" cols="##" rows="##"></textarea>`

**Radio buttons**

- Creates inputs with pre-determined values.
- User can select **only one** of the listed options per set.
  - Usage: "Select One of the following:"
- All radio buttons in a set must share the same name value.
- `<input type="radio" name="radioset" id="radioset01" value="data_one" />`  
`<input type="radio" name="radioset" id="radioset02" value="data_two" />`

**Checkboxes**

- Creates inputs with pre-determined values.
- User can select **multiple checkboxes** within a set.
  - Usage: "Select One or More of the following:"
- Checkboxes must have different name values in order to work correctly.
- `<input type="checkbox" name="name_one" id="check01" value="data_one" />`  
`<input type="checkbox" name="name_two" id="check02" value="data_two" />`

**Hidden fields**

- Passes values to the Web site server that are not seen by the end user.
- `<input type="hidden" name="name" id="name" value="" />`

### Pull-down menus

- Creates input with pre-determined values.
- User can select **one or multiple** within a set, depending on the settings.
  - Usage: “Select One of the following:” or “Select One or More of the following:”
  - Default allows for only one item to be selected.
- Select menu shares a single name value.
- Set the size to more than 1 to allow for multiple selections.
- ```
<select name="name" id="name" size="1">
  <option value="value">Name here</option>
</select>
```

### Submit Button

- Provides a way for the user to submit their data to the Web site.
- Can be replaced by a submit graphic (see second code sample below).
- ```
<input type="submit" name="name" value="message" />
<input type="image" src="art/go_arrow.gif" alt="Go" />
```

### Styling of Form Elements

- Form elements can be styled using CSS.
- Full-featured reset style sheets clear out most form styling, requiring you to bring back some basic design elements so that they appear on the page.

#### Common CSS properties using in styling form elements

- Borders
- Border radius
- Background colors and images
- Font size, family, color, and weight

### JavaScript & Form Elements

#### Hiding existing content on focus:

- ```
<input type="text" value="First Name" onfocus="this.value='';" />
```

### Form Features of HTML5

#### Useful new features

- Placeholder text; Required fields; Autocomplete control; Datalist.

#### New Inputs

- Search
- Contact details
- Sliders
  - ```
<input type="range" id="slide" name="slide" min="0" max="100" value="100" />
```
- Spinners
- Date and times

The screenshot shows a web form titled "What would you build?". It contains several input fields: "My name is", "My email address is", "Company/Institution - Name\*", "Company/Institution - Street Address\*", "City", "State" (with "United States" selected), and "Zip Code". There are also radio buttons for "I am primarily interested in developing:" (Applications, Algorithms, Games, Something Else) and "I am willing to contribute source code back to the project under Apache 2.0" (Yes, No). A text box asks "Here is what I would build:". A blue "Sign Up" button is at the bottom. Small text at the bottom right states: "By submitting this form, I agree to the Terms and Conditions and acknowledge that my information will be used in accordance with Google's Privacy Policy."