

Forms

Essentials of Web Design
CSCI 1210

*East Tennessee State University
Department of Computing*

*CSCI 1210
Essentials of Web Design*

1

Two way communication

One way communication:

So far we have created HTML5 that the server sends to the client for display in a browser

Two way communication: How can the client send information back to the server?

Supplying name, address, and payment information for an online purchase.

Adding your name to a online guestbook

*East Tennessee State University
Department of Computing*

*CSCI 1210
Essentials of Web Design*

2

Sending Data from Client to Server

A form is used to collect data from the client for transmission

Form gathers all the information and submits it as a single "bundle" of information

Different types of form fields exist to permit collecting different types of information

Sending Data from Client to Server

When a form is submitted, it must be **processed** by the server using a program called a server-side script.

The program processes the information and replies in a fashion dictated by its programming

Sending Data from Client to Server

```

<body>
  <div>
    <h1>Thanks!</h1>
    <p style="padding:25px; text-align:left;">
      Thanks,
      <span style="color:red;"><?php if(isset($_GET['first_name'])) {echo ucfirst(
        $_GET['first_name']); } ?></span>, we appreciate your interest in our site!
    </p>
    <p style="padding:25px; text-align:left;">
      According to what you entered on our form, your email address is
      <span style="color:red;"><?php if(isset($_GET['email'])) {echo trim($_GET[
        'email'], " "); } ?></span>, you live in
      <span style="color:red;"><?php if(isset($_GET['state'])) {echo ucfirst($_GET[
        'state']); } ?></span>,
      and you're a
      <span style="color:red;"><?php if(isset($_GET['occupation'])) {echo ucfirst(
        $_GET['occupation']); } ?></span>.
    </p>
    <p>
      We're glad you stopped by!
    </p>
    <p style="width: 150px; margin:0 auto;">
      <?php echo "<a style='text-decoration:none;'
        href='javascript:history.go(-1)'\>&lArr; GO BACK</a>"; ?>
    </p>
  </div>
  <script src="js/w3c.js"></script>
</body>

```

5

Tell Us About Yourself

Who Are You?

Where Are You? ⬆

What Do You Do? ⬆

*Occupation

Company

Company Location

* Required

6

Sending Data from Client to Server

Thanks!

Thanks, **John**, we appreciate your interest in our site!

According to what you entered on our form, your email address is **ramseyjw@goldmail.etsu.edu**, you live in **TN**, and you're a **Lecturer**.

We're glad you stopped by!

[⇐ GO BACK](#)

Creating a form

Form Element

```
<form></form>
```

All items in the form must be contained in a form element

There is only one set of `<form>` tags containing all the fields rather than a form tag around each item

Creating a form

Required form attributes:

action--specifies the URL that is to receive the transmitted information

method--specified the method to be used for transferring the information

```
<form action='php/process_form.php' method='get'>  
  
</form>
```

Method

For the attribute method, we have two possibilities:

get

The value get sends form data as a part of the URL

It is added as a variable/value pair at the end of the URL after the question mark:

<https://www.google.com/search?q=dogs>

Multiple variable/value pairs are separated by the ampersand symbol in the URL:

https://www.google.com/search?q=dogs&gws_rd=ssl

Method

For the attribute method, we have two possibilities:

post

Sends the data to the server in a manner that is not visible to the end user

Get vs Post

When should we use the value get?

When should we use the value post?

Get vs Post

When should we use the value get?

When we would like the user to be able to bookmark the link so that they can return to that exact page without submitting the form.

When the data is not sensitive data

When should we use the value post?

Get vs Post

When should we use the value get?

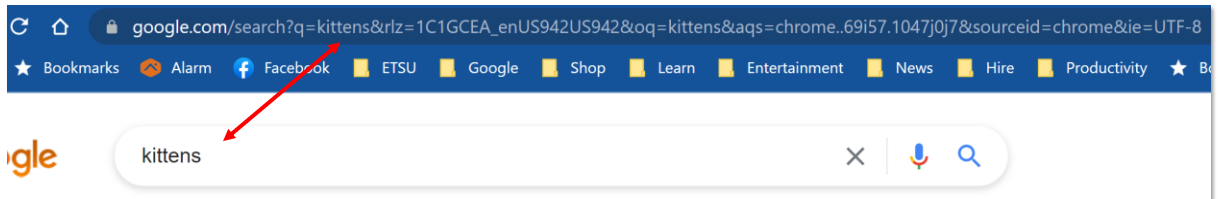
When we would like the user to be able to bookmark the link so that they can return to that exact page without submitting the form.

When the data is not sensitive data

When should we use the value post?

When the data is sensitive data (social security number, username/password, credit card information, etc)

Get



East Tennessee State University
Department of Computing

CSCI 1210
Essentials of Web Design

15

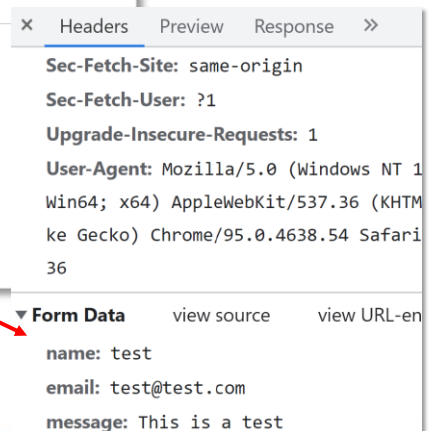
Post

Thank you

Thanks, test. I'll get right on it

Your Message:

This is a test



East Tennessee State University
Department of Computing

CSCI 1210
Essentials of Web Design

16

Basic Form Example (Search Google)

```
<body>
  <form action='https://www.google.com/search' method='get'>
    <input type='text' name='q'>
    <input type='submit' name='submit'>
  </form>
</body>
```

A visual representation of the HTML code above, showing a search form with a text input field and a 'Submit' button.

Form Fields

Single line textbox

Number Field

Radio Button

Password textbox

URL Field

Submit Button

Date Selector

File Selector

Reset Button

Email Address

Datalists

Multiple line textbox

Hidden Fields

Checkboxes

Select List

*Today's lecture / example

Input tag

```
<input>
```

The input tag is an **inline, standalone** tag

Input tag

Two critical attributes:

name – the attribute name is what is needed in order to reference the field once the form has been submitted

type – the attribute type identifies which input field is being created (textbox, password, checkbox, radio button, etc)

Input tag

```
<body>
  <form action='https://www.google.com/search' method='get'>
    <input type='text' name='q'>
    <input type='submit' name='submit'>
  </form>
</body>
```

Single Line Textbox

```
<input type='text' name='q'>
```

By default there is no label to direct users on what to input

We must add the descriptive text to instruct users:

```
Search for: <input type='text' name='q'>
```

Single Line Textbox

Attributes:

value – value allows you to assign a default value to the field

If the form is submitted without being changed by the end user, this value will be submitted

placeholder – this shows text in the box. Once the user begins typing in the field, the words are removed



Search for:

Single Line Textbox

Attributes:

size – specifies the size of the box, in number of characters

maxlength – specifies the maximum number of characters

disabled – specifies if the input element is visible, but the end user is not allowed to fill out the box

Single Line Textbox

Attributes:

autofocus – specifies the element on the page that should receive focus when the form loads

required – specifies that the field must be completed before the form can be submitted

Single Line Textbox

Example:

```
<input type="text" name="firstName" placeholder="First Name" size="20"
maxlength="30" autofocus required>
```



First Name Submit

Password

```
First Name: <input type='text' name='firstName'><br>
Last Name: <input type='text' name='lastName'><br>
Username: <input type='text' name='userName'><br>
Password: <input type='password' name='password'><br>
```

First Name:	<input type="text"/>
Last Name:	<input type="text"/>
Username:	<input type="text"/>
Password:	<input type="password"/>
<input type="submit" value="Submit"/>	

First Name:	Joseph
Last Name:	Dokes
Username:	jdokes
Password:	*****
<input type="submit" value="Submit"/>	

The password field works the same as the textbox field and can utilize the same attributes

The difference between the two is that the password field masks the user input

Password

```
<input type="password" name="password" placeholder="Password"
size="20" maxlength="30" required>
```

First Name:	First Name
Last Name:	Last Name
Username:	User Name
Password:	Password
<input type="submit" value="Submit"/>	

First Name:	Joe
Last Name:	Dokes
Username:	jdokes
Password:	*****
<input type="submit" value="Submit"/>	

Checkboxes

```
<input type="checkbox" name="sauce" value="yes" checked><br>
<input type="checkbox" name="pepperoni" value="yes" checked><br>
<input type="checkbox" name="anchovies" value="yes"><br>
```

The checkbox allows a user to select 0, 1, or many options

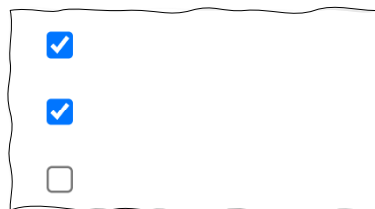
With a checkbox, the name/value pair is only submitted if the end user selects the checkbox

Attribute:

checked – this will ensure that the box is checked by default

Checkboxes

```
<input type="checkbox" name="sauce" value="yes" checked><br>
<input type="checkbox" name="pepperoni" value="yes" checked><br>
<input type="checkbox" name="anchovies" value="yes"><br>
```



Checkboxes

Example:

```
<input type="checkbox" name="sauce" value="yes" checked> Sauce<br>
<input type="checkbox" name="pepperoni" value="yes" checked> Pepperoni<br>
<input type="checkbox" name="anchovies" value="yes"> Anchovies<br>
```

Sauce
 Pepperoni
 Anchovies

However, if you click on the word, you cannot select the checkbox

Checkboxes

Example:

Sauce
 Pepperoni
 Anchovies

```
<input type="checkbox" name="sauce" id='s' value="yes" checked>
<label for='s'>Sauce</label><br>
<input type="checkbox" name="pepperoni" id='p' value="yes" checked>
<label for='p'>Pepperoni</label><br>
<input type="checkbox" name="anchovies" id='a' value="yes">
<label for='a'>Anchovies</label><br>
```

Now, a user can click on the words associated with the checkbox -> The `for=""` attribute/value pair corresponds to the `id=""` a/v pair in the input element

Radio Buttons

Yes or No?

Yes

No

```
<input type='radio' name='answer' id='y' value='yes'>
```

Radio buttons work in the same manner as checkboxes with one major difference, with radio buttons you can only select one option out of a grouping

In order to group, the attribute **name** must be the same for each option in the group

Radio Buttons

Yes or No?

Yes

No

```
<fieldset>
  <legend>Yes or No?</legend>
  <input type='radio' name='answer' id='y' value='yes'>
  <label for='y'>Yes</label><br>
  <input type='radio' name='answer' id='n' value='no'>
  <label for='n'>No</label><br>
</fieldset>
```

Radio Buttons

Yes or No?

Yes
 No

```
<fieldset>
  <legend>Yes or No?</legend>
  <input type='radio' name='answer' id='y' value='yes' >
  <label for='y'>Yes</label><br>
  <input type='radio' name='answer' id='n' value='no' >
  <label for='n'>No</label><br>
</fieldset>
```

Often, we want to display radio buttons or checkbox groups together
We can do this with the `fieldset` element

Submit and Reset

```
<input type='submit' > <input type='reset' >
```

The submit button allows the user to submit the field for processing. The location is determined by the attribute `action` in the `<form>` tag.

The reset button allows the user to clear out any user input and reset the field to the default values assigned for each field.

Attribute:

The attribute `value` can be utilized to change the text that appears on the button

Submit and Reset

Example:

```
<input type='submit'> <input type='reset'>
```

```
<input type='submit' value='Submit Form'>  
<input type='reset' value='Reset Form'>
```

Multiple Line Textbox

```
<textarea name='bio'></textarea>
```

The `<textarea>` element allows for the creation of a multiline text area

If you want a default value associated with the textarea, place that text between the opening and closing of the tag

Multiple Line Textbox

Attributes:

cols – specifies the visible width of the text area

maxlength – specifies the maximum number of characters

rows – specifies the number of visible rows of text

readonly – specifies that the text area cannot be changed

textarea

Example:

The image shows a browser window with a text area. The text area has a title 'Biography' and a placeholder text 'Please provide a short biography'. The text area is empty, and the placeholder text is visible.

```
Biography<br>
<textarea cols="50" name="bio" maxlength="600" rows="10"
placeholder="Please provide a short biography"></textarea>
```

Select List

A select list is a drop down (or scrollable) list of predefined options that a user can select

```
<select name='state'></select>
```

Select List

For each option presented to the end user, you must use the <option>...<option> tag

```
<select name='state'>  
  <option value='TN'>Tennessee</option>  
  <option value='VA'>Virginia</option>  
  <option value='NC'>North Carolina</option>  
</select>
```

The text between the opening and closing of the option tag is the text that is visible to the end user

The value associated with the selected option is what is submitted

Select List

Attributes for select tag

disabled – specifies that the drop-down list is visible but the user cannot select from the list

size – specifies the number of visible options

multiple – specifies that a user can select multiple options from the list

Attributes for option tag

selected – specifies the selected element by default when the page loads

Select List

Example

```
<select name='state'>
  <option value='GA'>Georgia</option>
  <option value='KY'>Kentucky</option>
  <option value='NC'>North Carolina</option>
  <option value='TN' selected>Tennessee</option>
  <option value='VA'>Virginia</option>
</select>
```

Questions?

*East Tennessee State University
Department of Computing*

*CSCI 1210
Essentials of Web Design*

45

Some Final Thoughts...

*East Tennessee State University
Department of Computing*

*CSCI 1210
Essentials of Web Design*

46

CSS Uniformity Across Browsers

*East Tennessee State University
Department of Computing*

*CSCI 1210
Essentials of Web Design*

47

Web Design Frustrations

One of the biggest frustrations in Web Design is browsers

Browsers will display HTML/CSS in different manners, making it difficult for the same uniform look across browsers

Two possible solutions:

- Creating CSS files specific for the various browsers

- Utilizing CSS to equalize the display

*East Tennessee State University
Department of Computing*

*CSCI 1210
Essentials of Web Design*

48

Creating Unique CSS

```
<!--[if IE]>  
<link rel="stylesheet" type="text/css" href="all-ie-only.css" />  
<![endif]-->  
  
<!--[if !IE]><!-->  
<link rel="stylesheet" type="text/css" href="not-ie.css" />  
<!--<![endif]-->
```

Not the most desired option

Not supported in IE10 and up

However, it does allow you to target specific version of IE

Normalize.css

Normalize.css is a small CSS file that provides better cross-browser consistency in the default styling of HTML elements (Gallagher 2012)

Utilized by sites such as Twitter, Bootstrap, TweetDeck, etc

Download: <http://necolas.github.io/normalize.css/>

Normalize.css

Sample from the .css file:

```
/* HTML5 display definitions
   ========================================================================= */
/**
 * Correct 'block' display not defined for any HTML5 element in IE 8/9.
 * Correct 'block' display not defined for 'details' or 'summary' in IE 10/11 and Firefox.
 * Correct 'block' display not defined for 'main' in IE 11.
 */

article,
aside,
details,
figcaption,
figure,
footer,
header,
hgroup,
main,
nav,
section,
summary {
  display: block;
}
```

Mobile Design

Mobile Technology

77% of American Adults have a smartphone*

While texting, talking, emailing and going online dominate, a majority of Americans also use their smartphones for social networking, taking photos or videos, and catching up with the news

How does your website compare? Is it mobile ready?

*<http://www.pewinternet.org/fact-sheet/mobile/>

East Tennessee State University
Department of Computing

CSCI 1210
Essentials of Web Design

53

Mobile Technology



VS



East Tennessee State University
Department of Computing

ESSENTIALS OF WEB DESIGN

54

Mobile Design Options

Three basic options:

Create a second site just for mobile users:

<http://m.foxnews.com>

<http://foxnews.mobi>

Recognize the device that is viewing the screen, and change CSS files:

CNN:

http://z.cdn.turner.com/cnn/tmpl_asset/static/www_homepage/2892/css/hplib-min.css for desktop

http://z.cdn.turner.com/cnn/tmpl_asset/static/mobile_phone/4047/css/lib-min.css for mobile

Utilize Responsive Design

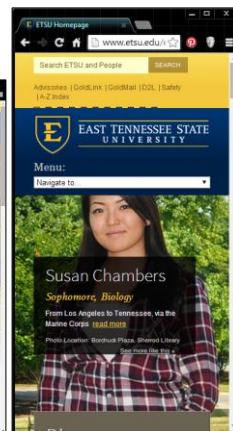
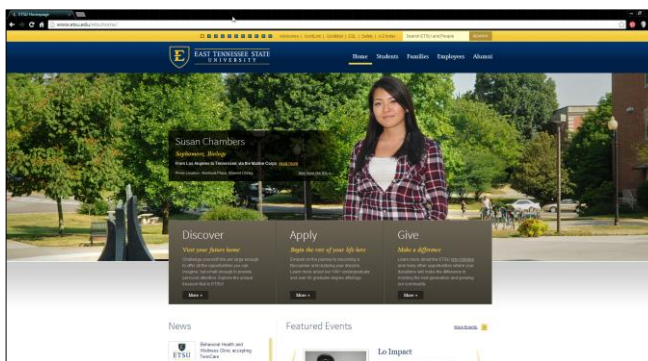
*East Tennessee State University
Department of Computing*

*CSCI 1210
Essentials of Web Design*

55

Responsive Design

Responsive Design does not change the CSS or the entire page, rather it scales the page to meet viewing window



*East Tennessee State University
Department of Computing*

*CSCI 1210
Essentials of Web Design*

56

Frameworks

What are CSS Frameworks

An CSS framework is a way to create dynamic web sites or web applications in a very quick manner

Provides a quick and easy method to create beautiful websites, standardizing components that are needed

Some examples:

Foundation - <http://foundation.zurb.com/>

HTML KickStart - <http://www.99lime.com/elements/>

HTML5 Boilerplate – <http://html5boilerplate.com>

Bootstrap – <http://getbootstrap.com>

Bootstrap

Bootstrap was originally created by team members from Twitter

One of the most popular HTML/Front-End Frameworks with a massive user community

Responsive Design built in

FREE :-)

Sources

- Gallagher, N. (Feb 2012). "About Normalize.css", NicholasGallagher.com, Retrieved from <http://nicolasgallagher.com/about-normalize-css/>
- "Gestalt Principles: How Are Your Designs Perceived?", VanseoDesign, Retrieved from <http://www.vanseodesign.com/web-design/gestalt-principles-of-perception/>
- Levin, J. (Mar 2005). "Gestalt Principles and Web Design", University of California San Diego, Retrieved from <http://teprever.ucsd.edu/~jlevin/gp/index.html>.
- "Mobile Technology Fact Sheet", Pew Research Internet Project, Retrieved from <http://www.pewinternet.org/fact-sheets/mobile-technology-fact-sheet/>
- "HTML Reference", W3Schools, Retrieved from <http://www.w3schools.com/tags/default.asp>
- "HTML Forms and Input", W3Schools, Retrieved from http://www.w3schools.com/html/html_forms.asp
- Stephen Hendrix, ETSU Department of Computing

Copyrights



Presentation prepared by and copyright of John Ramsey,
East Tennessee State University, Department of
Computing . (ramseyjw@etsu.edu)



- Microsoft, Windows, Excel, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.
- IBM, DB2, DB2 Universal Database, System i, System i5, System p, System p5, System x, System z, System z10, System z9, z10, z9, iSeries, pSeries, xSeries, zSeries, eServer, z/VM, z/OS, i5/OS, S/390, OS/390, OS/400, AS/400, S/390 Parallel Enterprise Server, PowerVM, Power Architecture, POWER6+, POWER6, POWER5+, POWER5, POWER, OpenPower, PowerPC, BatchPipes, BladeCenter, System Storage, GFFS, HACMP, RETAIN, DB2 Connect, RACF, Redbooks, OS/2, Parallel Sysplex, MVS/ESA, AIX, Intelligent Miner, WebSphere, Netfinity, Tivoli and Informix are trademarks or registered trademarks of IBM Corporation.
- Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.
- Oracle is a registered trademark of Oracle Corporation.
- HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.
- Java is a registered trademark of Sun Microsystems, Inc.
- JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.
- SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries.
- Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects S.A. in the United States and in other countries. Business Objects is an SAP company.
- ERPsim is a registered copyright of ERPsim Labs, HEC Montreal.
- Other products mentioned in this presentation are trademarks of their respective owners.

*East Tennessee State University
Department of Computing*

*CSCI 1210
Essentials of Web Design*