

CGS 3175: Internet Applications Fall 2007

Advanced XHTML & CSS – Part 1

Instructor : Dr. Mark Llewellyn
markl@cs.ucf.edu
HEC 236, 407-823-2790
<http://www.cs.ucf.edu/courses/cgs3175/fall2007>

School of Electrical Engineering and Computer Science
University of Central Florida



Things to Try Yourself

17. What happens if the CSS for WebSite3 (in this set of notes) is modified as follows:

```
#flotright {margin:10px;  
            clear: both; <!-- new property added -->  
            float:right;  
        }
```

Compare the screen shot on page 30 of Web Page Layouts – Part 2 with the screen shot on the next page.... What's the difference?



Professional Cycling 2007 - Windows Internet Explorer

E:\Courses\CGS 3175 - Internet Applications\sample code\Website3.html


File Edit View Favorites Tools Help

Google

Bookmarks 84 blocked Check AutoLink AutoFill Send to Settings

CNN.com - Breaking News, U... Professional Cycling 2007

To help protect your security, Internet Explorer has restricted this webpage from running scripts or ActiveX controls that could access your computer. Click here for options... Copy



Professional Cycling 2007

Home

Spring

Summer

Fall

Winter

The professional cycling season spans from late January to early November on the road. During the winter months the activity switches to cyclo-cross or indoor track racing. There are professional riders in more than 100 nations. The highest concentration of professional riders is in Europe where Belgians, Italians, and French riders are the most numerous. One of the more famous recent professional riders is Lance Armstrong of the US who won an unprecedented 7 Tour de France races in a row. One of the most famous professionals of all time is Belgian Eddy Merckx who won virtually every professional race on the calendar. One of the current stars of professional racing is Tom Boonen, a Belgian following in the foot steps of Eddy Merckx. Last season Boonen won 28 races.


To explore some of the action from the 2007 season use the links to the left....

ProCycling News

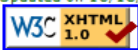
The latest news from the world of professional cycling updated daily is available at: [ProCycling News](#).

USA Cycling

To find out more about bicycle racing in the United States visit [USA Cycling](#).

Copyright © 2007 Professional Cycling Fan Club
Last Updated on 10/18/2007



Done My Computer 100%

Things to Try Yourself

18. See if you can create a screen that looks like the one on the following page.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Tiled Background Image </title>
    <style type="text/css">
      <!--
        .left { background-image:url(diluca07-1.jpg);
                background-color:black;
                background-repeat:repeat-y;
                width:100px;
                height:800px;
                position:absolute;
                top:0px;
                left:0px;
              }
      -->
    </style>
  </head>
  <body>
    <div class="left">
    </div>
  </body>
</html>
```



Things to Try Yourself

```
.right { background-image:url(diluca07-1.jpg);
        background-color:black;
        background-repeat:repeat-y;
        width:100px;
        height:800px;
        position:absolute;
        top:0px;
        left:1175px;
    }
.top { background-image:url(diluca07-1.jpg);
       background-color:black;
       background-repeat:repeat-x;
       width:1300px;
       height:100px;
       position:absolute;
       top:0px;
       left:0px;
    }
.bottom { background-image:url(diluca07-1.jpg);
          background-color:black;
          background-repeat:repeat-x;
          width:1300px;
          height:100px;
          position:absolute;
          left:0px;
          top:700px;
        }
```

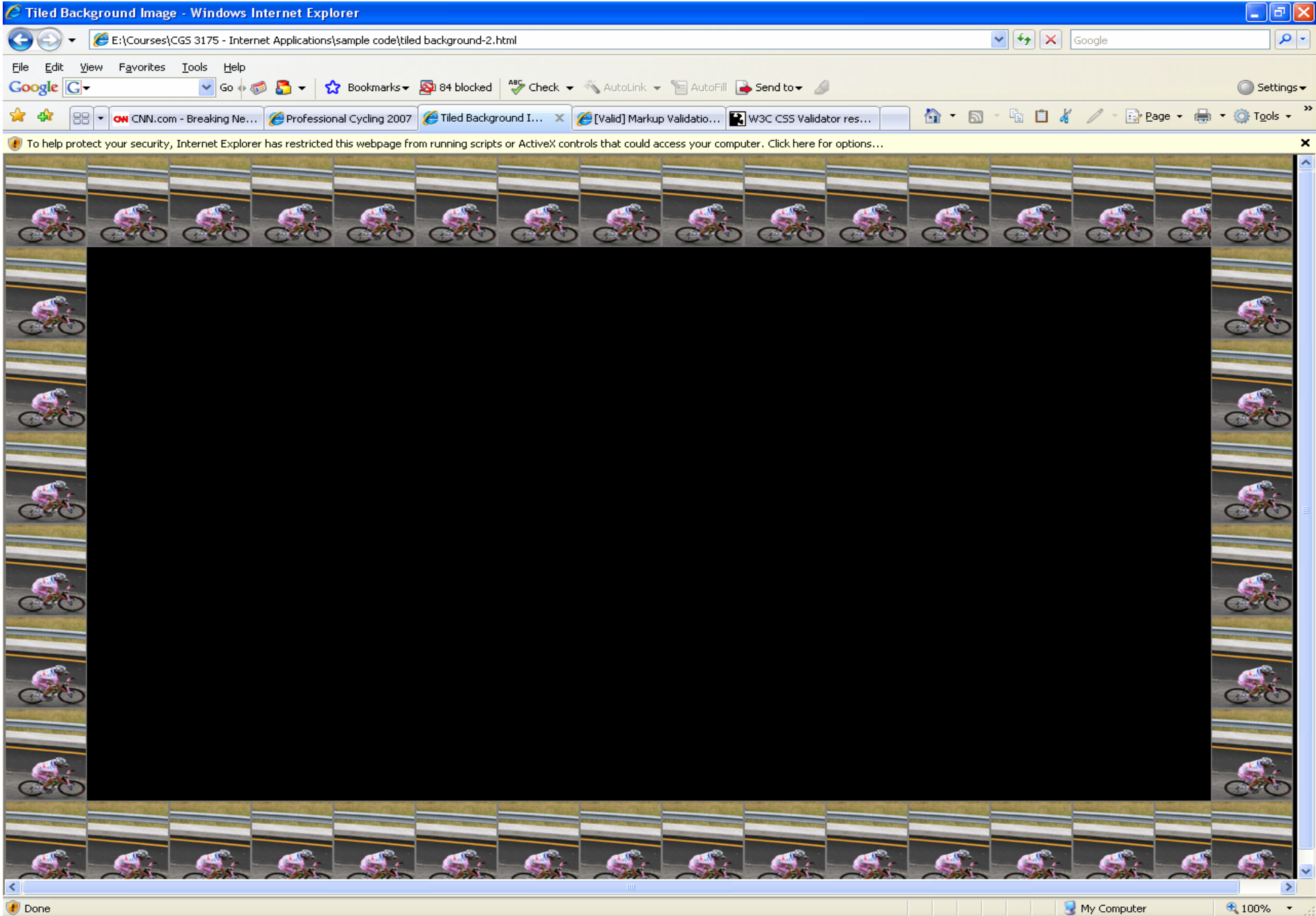


Things to Try Yourself

```
body { background-color:black;
        background-position:0px 0px;
        scrollbar-face-color:#cc66cc;
        scrollbar-arrow-color:#006600;
        scrollbar-track-color:#cccccc;
    }
-->
</style>
</head>
<body>
    <div class="left"> </div>
    <div class="top"> </div>
    <div class="bottom"> </div>
    <div class="right"> </div>

    <p>
    <a href="http://validator.w3.org/check?uri=referer"></a>
    </p>
</body>
</html>
```





Another Look At Lists

- One of the advantages of using CSS for page layout involves the use of semantically correct code. Writing semantically correct does means using the markup tag that most accurately reflects the purpose of the content. Using the various levels of heading tags for content headings and subheadings, or placing paragraphs of text within paragraph tags (rather than using line breaks) are examples of writing semantically correct code.
- This type of coding is a step in the direction to support the Semantic Web. Leading Web developers have promoted the idea of using unordered lists to configure navigation menus. After all – a navigation menu is a list of links – semantically speaking it's a much better fit than coding links in separate paragraphs of using the `display:block` property on anchor tags.
- In the last example WebSite3, we styled the navigation menu as an unordered list. You want to get in the habit of coding all of your navigation menus in this fashion and in general developing semantically correct code.
- The next few pages illustrate some of the options for using lists for navigation menus.



Another Look At Lists

- Shown below is a snippet of the XHTML code from the Website3.html document that shows how the navigation area is defined using an unordered list element:

```
<ul class="navBar">
  <li><a class="navBar" href="Website3.html">Home</a></li>
  <li><a class="navBar" href="spring.html">Spring</a></li>
  <li><a class="navBar" href="summer.html">Summer</a></li>
  <li><a class="navBar" href="fall.html">Fall</a></li>
  <li><a class="navBar" href="winter.html">Winter</a></li>
</ul>
```

- The CSS class navBar used above, is shown below:

```
.navBar{text-decoration:none;
  font-size:large;
  padding:3px;
  margin: 15px;
  list-style-type:none;
  line-height:2;
}
```

The way the navigation links look

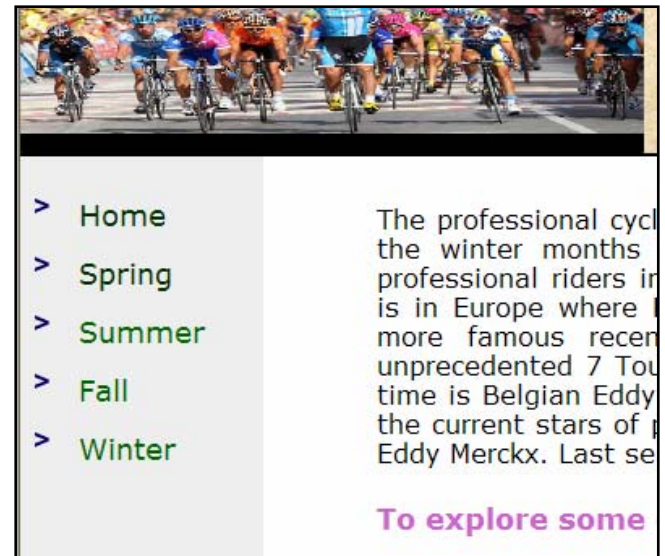


Another Look At Lists

- In addition to the standard types of “bullets” that can be applied to an unordered list with CSS, you can also specify an image to be used as the “bullet”. An example of this is shown below:
- In the CSS document, use the `list-style-image` property and specify the `url` of the image you want to use as the “bullet”.

```
.navBar{text-decoration:none;  
    font-size:large;  
    padding:3px;  
    margin: 15px;  
    list-style-image:url(arrow.gif);  
    line-height:2;  
}
```

The way the navigation links look using an image. I've put the `arrow.gif` file on the course Web page if you want to try this one yourself.

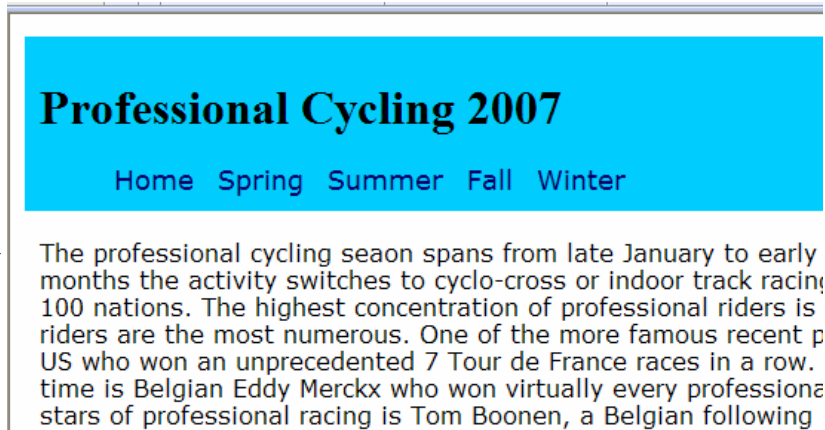


Another Look At Lists

- You might be wondering if this CSS technique for modeling navigation areas as lists will also work for navigation areas that use a horizontal layout rather than a vertical layout. The answer is yes!
- List-items are block elements, they need to be configured as in-line elements to display as a single line in a horizontal navigation area layout. Once again, the `display` property can be used to accomplish this effect with a value of `inline`.
- Shown below is the modification of the external style sheet for the Web page we built in Web Page Layouts – Part 1 redone with an inline navigation list.

```
.nav li {display:inline;
        list-style-type:none;
}
.nav {padding: 5px;
      background-color:#00CCFF;
      color:#000066;
      text-decoration:none;
      font-family:Verdana, Arial, sans-serif;
      width:90px;
      text-align:center;
      display:inline;
      list-style-type:none;
}
```

The way the navigation links look using an inline list.



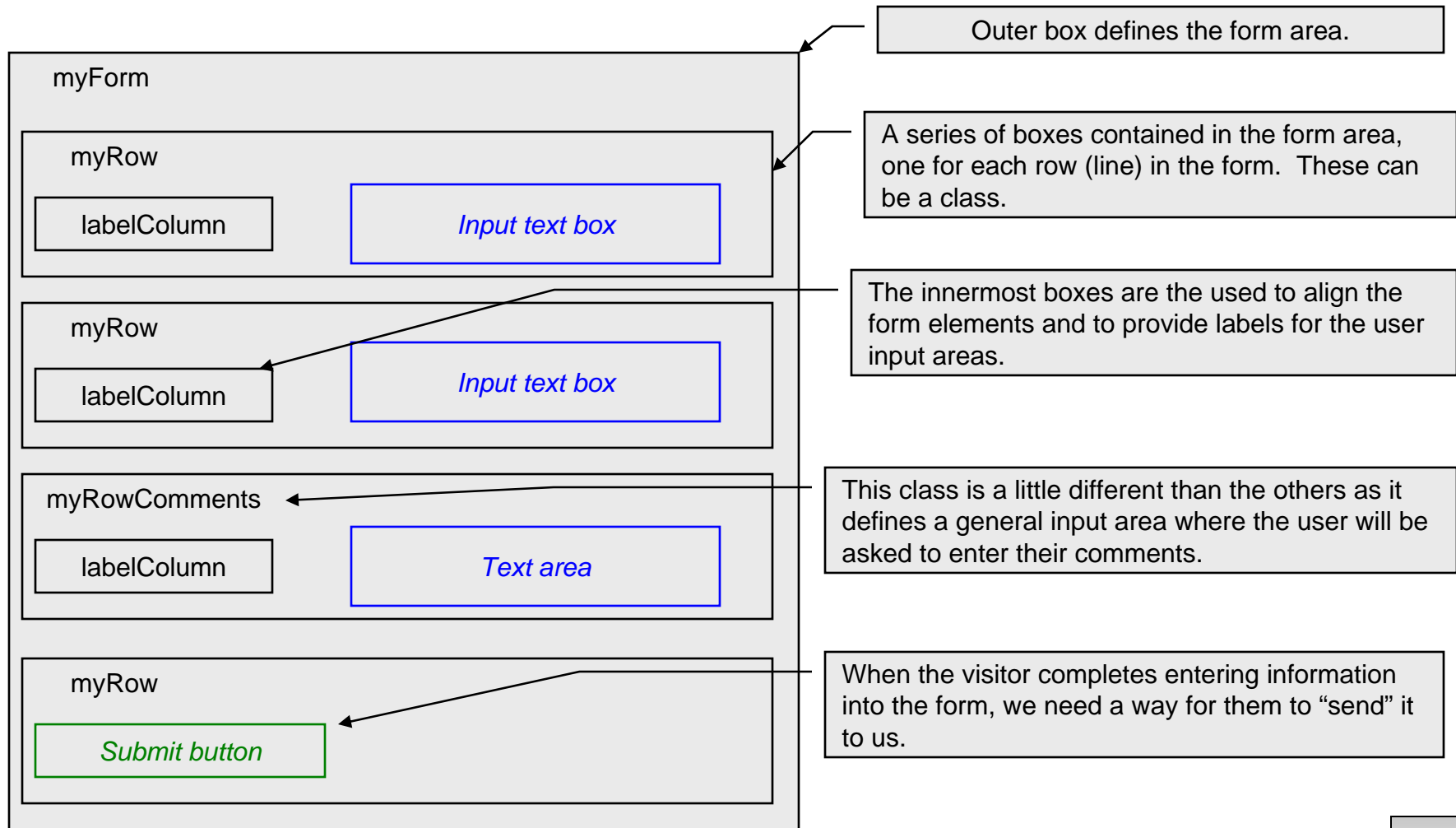
Using CSS to Style a Form

- Tables have been traditionally used to create forms in Web pages.
- Since we are learning the new style of Web page development in this course which utilizes CSS for all page layout, we'll also learn the new way of creating forms using only CSS without tables.
- To create a table using only CSS, you take advantage of the CSS box model to create a series of boxes which are the elements (rows and columns) of your table.
- As we've done with our splash pages and content pages, the best way to design a form using CSS is to first storyboard the form layout and then create the XHTML document to include the form elements and finally create the CSS document to render the form with the correct layout.



Using CSS to Style a Form

- Shown below is the storyboard for the form we want to create.

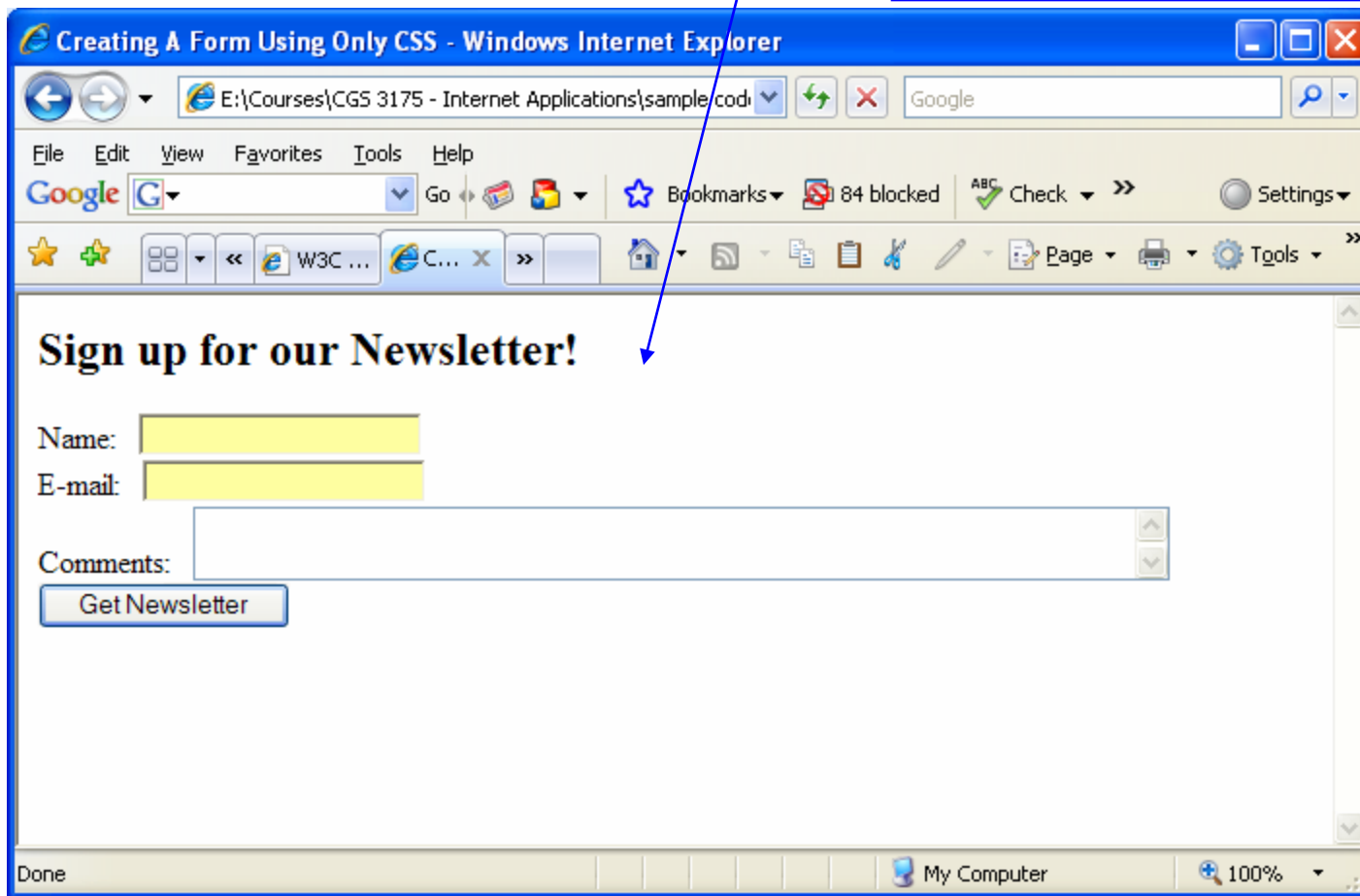


Using CSS to Style a Form

- We'll define a `myForm` id to define the properties of the entire form area, thus, `myForm` is the outermost box.
- The `myRow` class will set the height for a typical line in the form.
- The `myRowComments` class will configure a form area that is a general text input area where we will receive comments from the visitors. We'll ultimately set this up so that a 20 pixel margin is set below the scrolling text box.
- The `labelCol` class is the key to aligning the text. We'll need to set the width of this element to accommodate our largest label. We'll start out with 100 pixel width and adjust from there. All of the text in the `labelCol` elements will align to the right.
- We'll create the XHTML document first.



Browser rendering of our form elements without CSS.



New XHTML Tags: ``

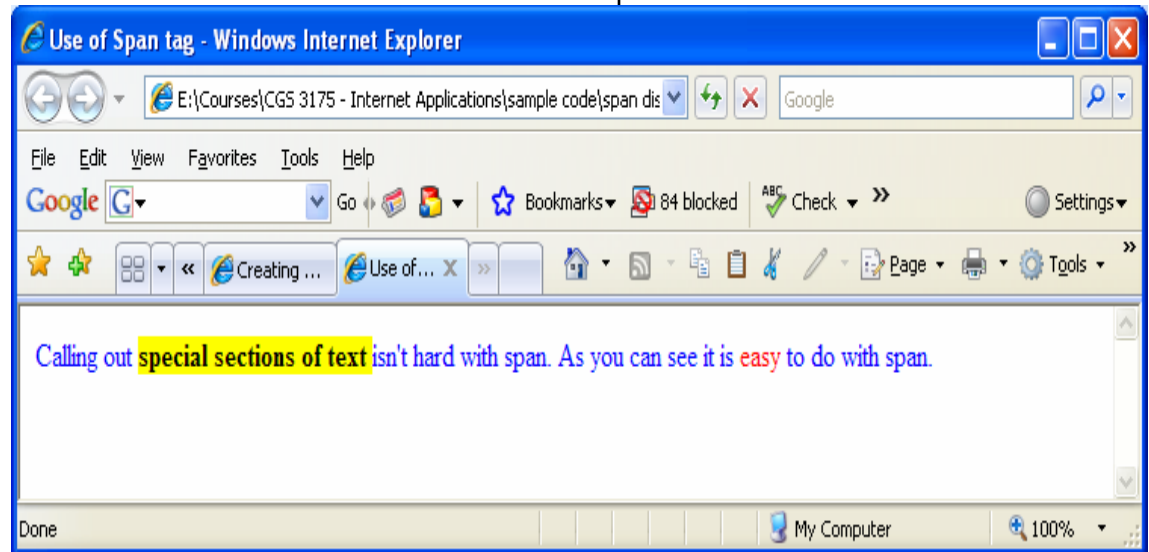
- As you look through the XHTML document on the previous page, you will see a few new XHTML tags that we have not yet encountered. Before going further, we'll look at these new tags.
- The `` tag: We've already made extensive use of the `<div>` tag. Like the `<div>` tag, the `` tag is a “generic” tag that has no default rendering. While the `<div>` tag applies to block elements, the `` tag applies to inline elements. The primary difference between the two is that `<div>` induces a return (blank line) and `` does not.
- Thus, any time you want to apply a style to a few words, or even a few letters, the best approach is to use a `` .



New XHTML Tags:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Use of Span tag </title>
<style type="text/css">
  <!--
    body {color:blue;
  }
  -->
</style>
</head>

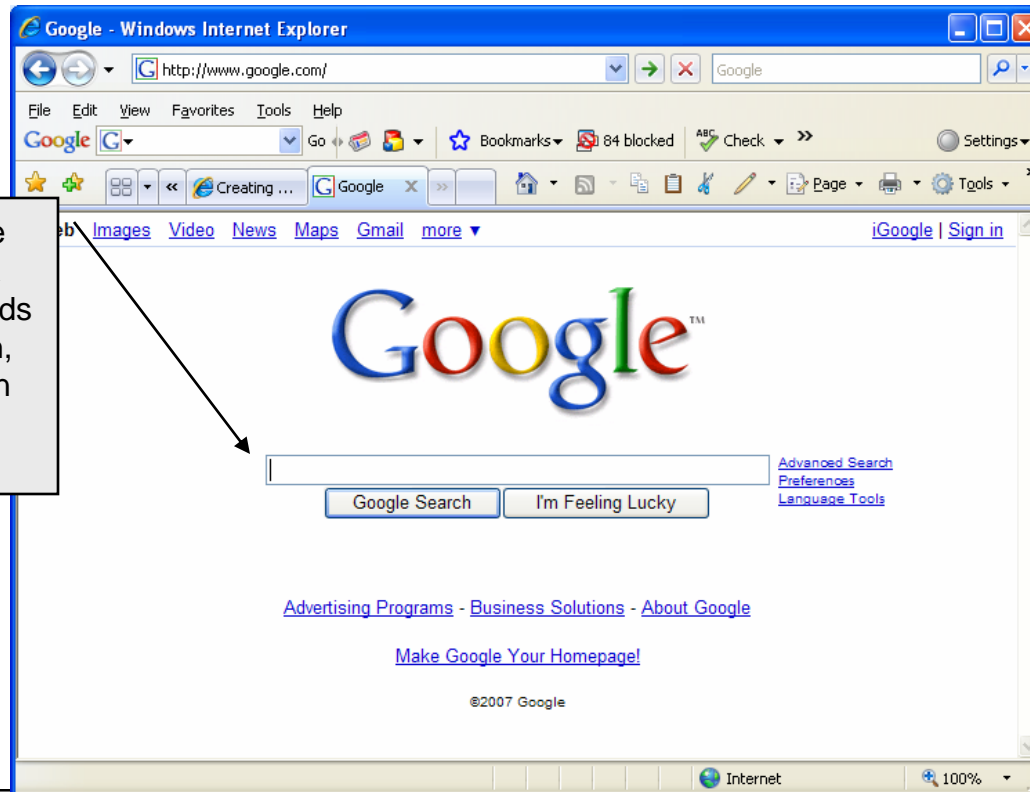
<body>
<p>
  Calling out <span style="background-color:yellow; font-
weight:bold; color:black;">
  special sections of text </span> isn't hard with span. As you
can see it is
  <span style="color:red">easy </span>to do with span.
</p>
</body>
</html>
```



New XHTML Tags: <form>

- Every time you use a search engine, place an order, or join an online mailing list, you use a **form**.
- A form is an XHTML element that contains and organizes other objects – such as text boxes, check boxes, and buttons – that can accept information from Web site visitors.

This form has only three elements. One text box that accepts the keywords to be used in the search, and two buttons that can submit the form and get the search started.



New XHTML Tags: <form>

- A form in XHTML is contained within a `form` element. The form itself contains regular text and other XHTML form elements such as check boxes, pull-down menus, and text fields.
- The W3C specification calls these form elements **controls**, but are most commonly referred to as “**form fields**”.
- There are a variety of form fields that can be inserted.
- In order to make a form work, you will need to specify two things in the form tag: (1) the address of the program that will handle the form contents using the `action` attribute and (2) the method by which the form data will be passed using the `method` attribute.
- The `action` attribute determines how the form is to be handled. The action attribute is usually set to the URL of the program that will handle the data in the form. We’ll discuss this attribute in more detail later but for now we won’t really “handle” the data in the form in any significant manner. In our example, we will simply email the data.



New XHTML Tags: <form>

- The **method** attribute determines how the form will be submitted to the address specified by the **action** attribute.
- There are only two acceptable values for this attribute: **get** and **post**. These are HTTP methods that a browser uses to talk to a server. (See Introduction – Part 2, page 33.) We won't worry about the subtle differences between these two values for the moment, but in general **post** will be preferred whenever either large amounts of data are being sent to the server or data is to be stored in a database.
- The **enctype** attribute determines how the data in the form is to be encoded when it is sent to the server. The default for most browsers is the same MIME format that is used for URLs. Particularly when using a **mailto** URL, the encoding type **text/plain** is normally used for ease of reading.
- Before we go any further, let's complete our first XHTML form by creating the CSS document to format the form.

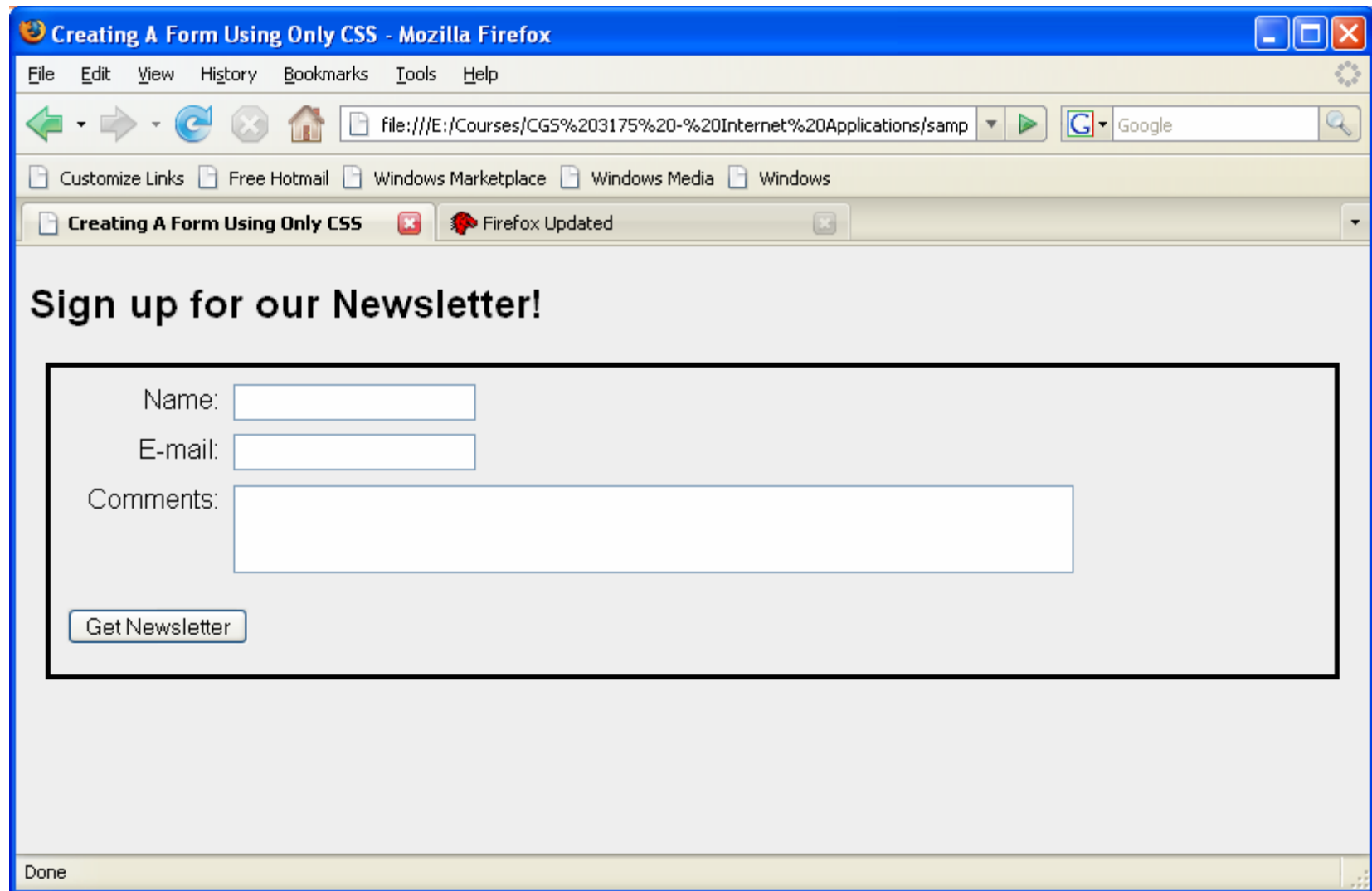


Creating The CSS For Our First Form

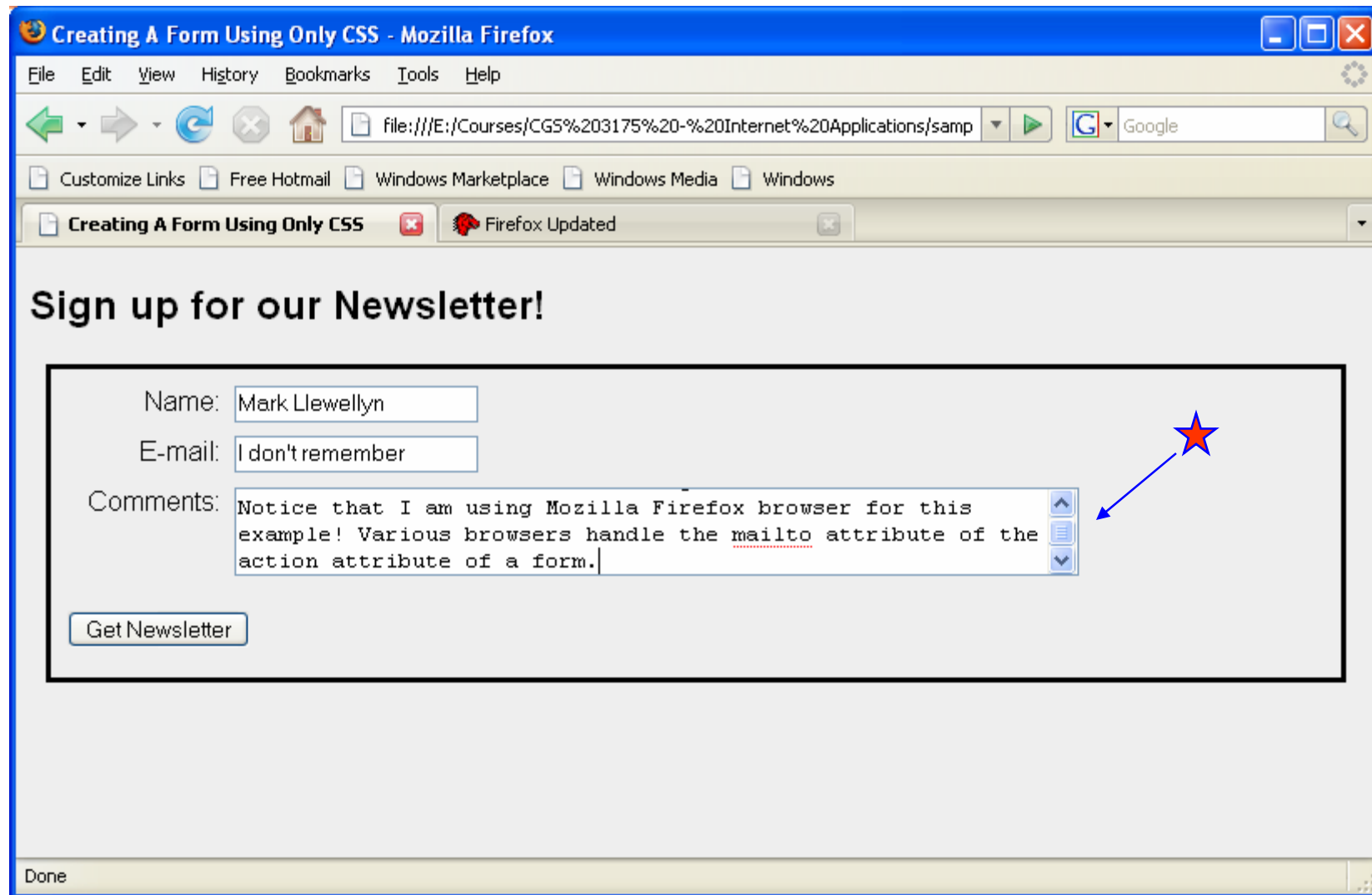
```
/* CSS for use with form2.html */  
  
body{ font-family:Arial,sans-serif;  
      background-color:#eeeeee;  
      color:#000000;  
}  
  
#myForm {border:3px solid #000000;  
         padding:10px;  
         margin:10px;  
}  
  
.myRow {height:30px;  
}  
  
.myRowComments {margin-bottom:20px;  
}  
  
.labelCol{float:left;width: 100px; text-align: right;  
}
```



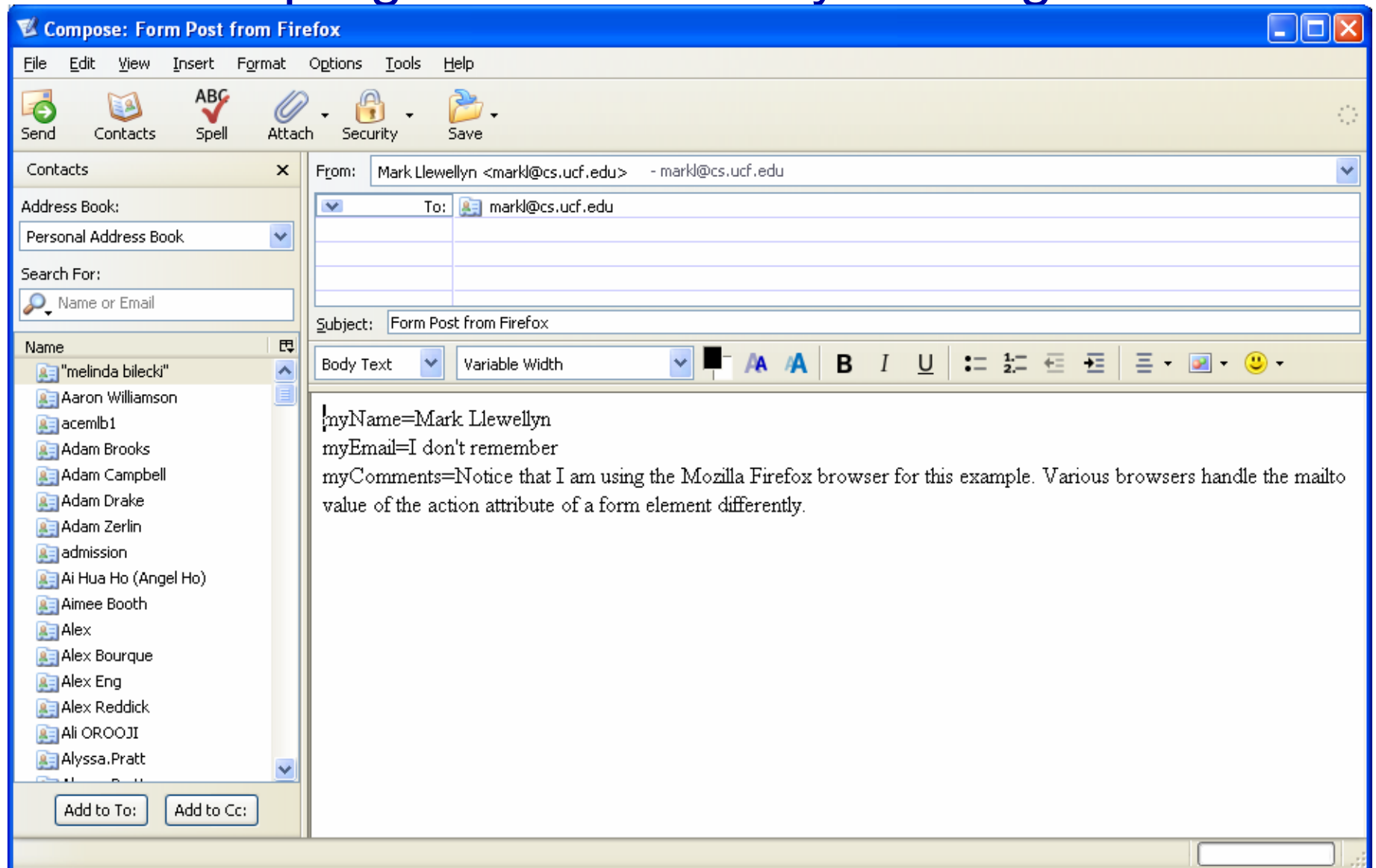
Browser Rendering Of The Form



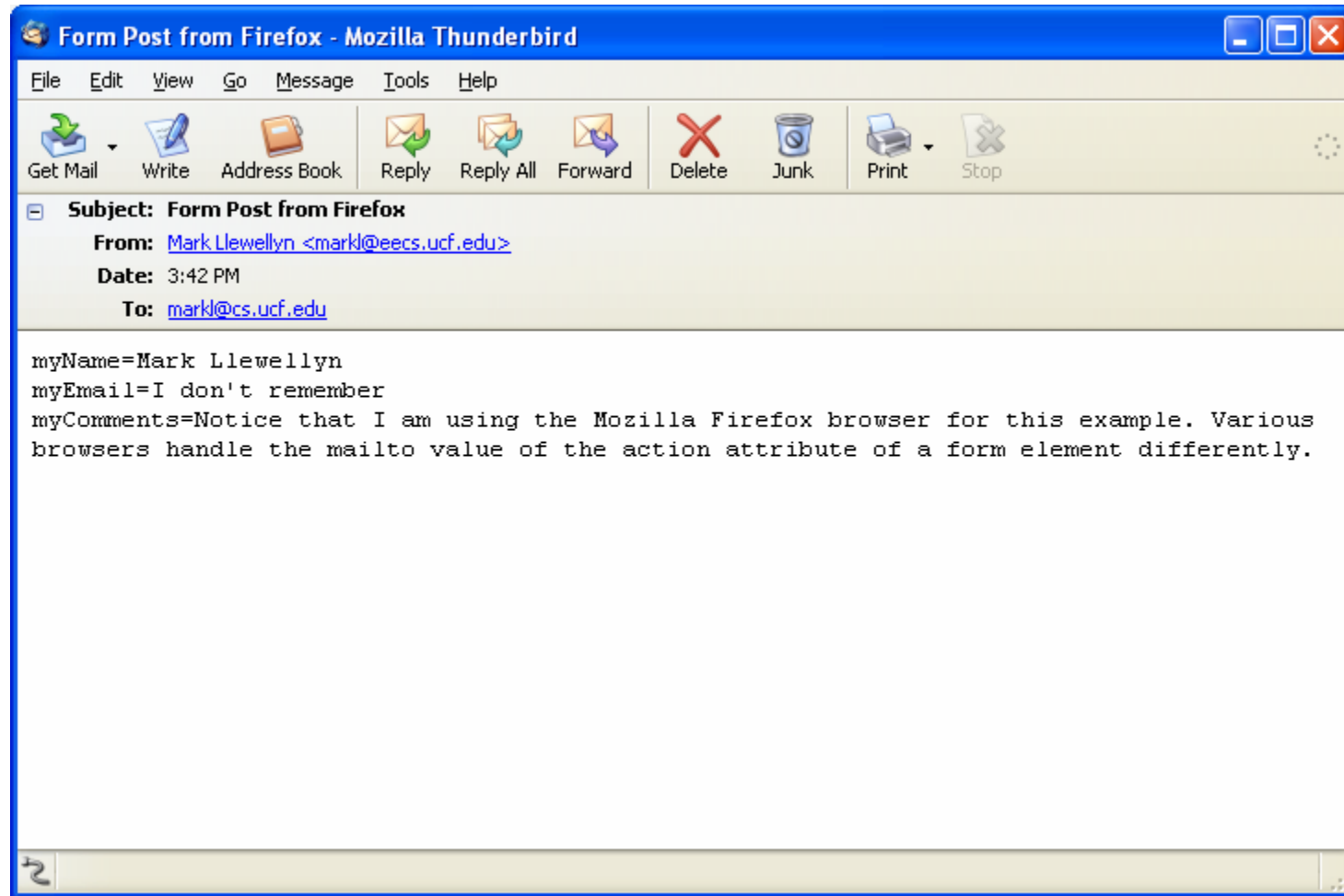
Browser Rendering Of The Form



Email program activated by clicking button



Email received by markl@cs.ucf.edu



New XHTML Tags:

- There are a number of basic form elements that are used to gather information from the Web site visitor. Some of these are:
 - Text boxes:
 - Scrolling text boxes
 - Select lists
 - Radio buttons
 - Check boxes
 - Buttons
- XHTML tags that configure these form elements include the `<input />`, `<textarea>`, `<select>`, and `<option>` tags.
- We'll look at a number of these form elements in detail now.



Things to Try Yourself

17. What happens if the CSS for WebSite3 (in this set of notes) is modified as follows:

```
#floatright {margin:10px;  
              clear: both; <!-- new property added -->  
              float:right;  
            }
```

Compare the screen shot on page 30 of Web Page Layouts – Part 2 with the screen shot on the next page.... What's the difference?

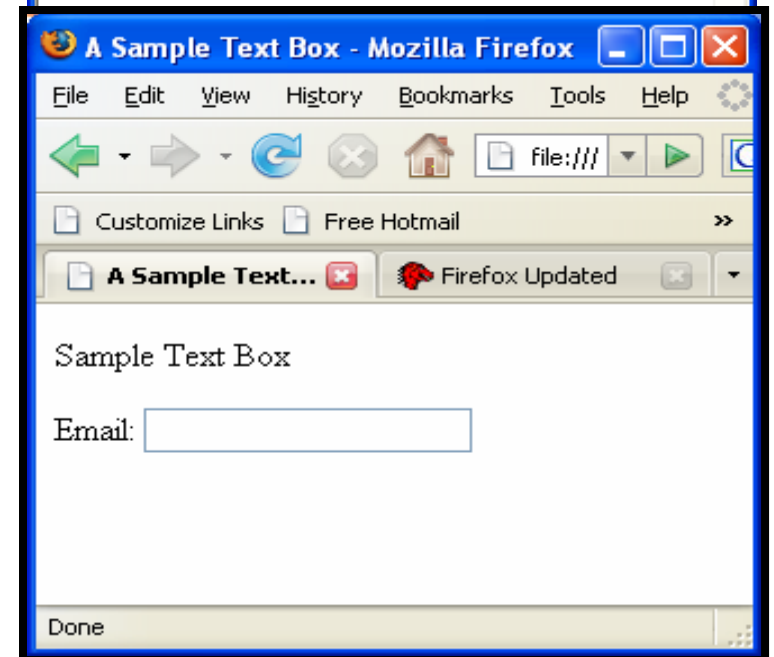
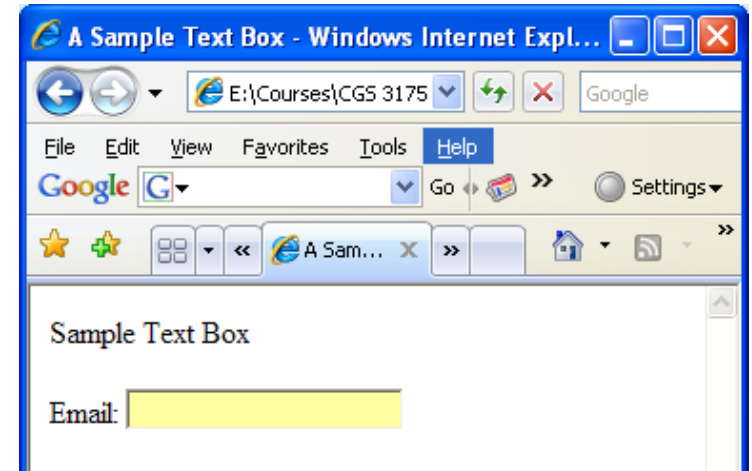


Form Element: Text Box

- Text boxes are configured by the `<input />` tag.
- Text boxes accept text or numeric information.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>A Sample Text Box</title>
</head>
<body>
<p> Sample Text Box </p>
<form method="get"
action="mailto:markl@cs.ucf.edu">
<p>Email:<input type="text" id="email" /></p>
</form>
</body>
</html>
```

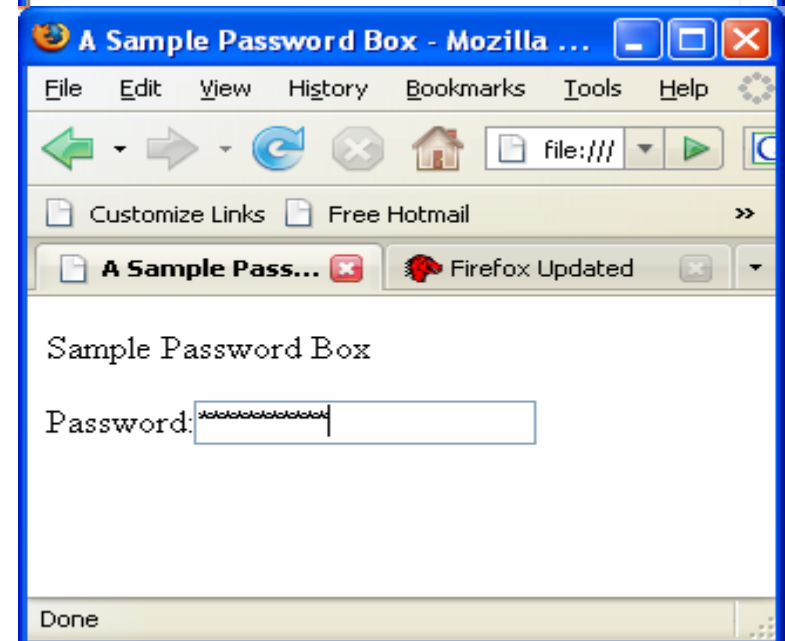
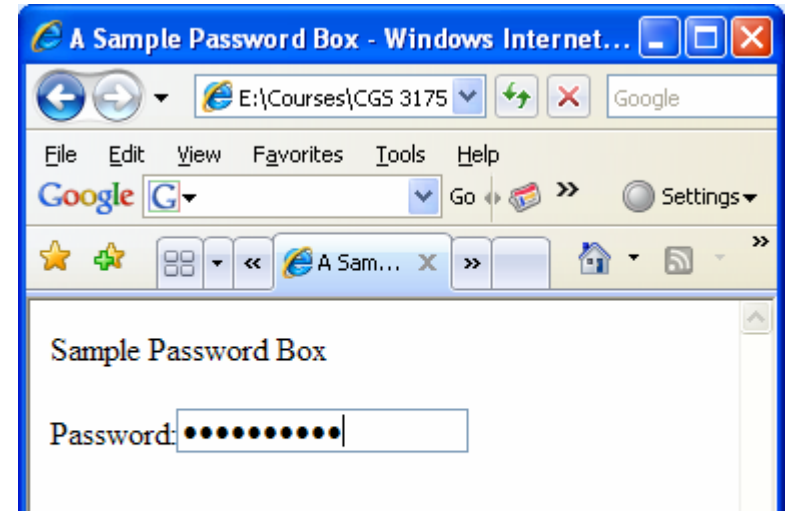


Form Element: Password Box

- Password boxes are also configured by the `<input />` tag.
- Password boxes are like text boxes except that they hide data as it is entered.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>A Sample Password Box</title>
</head>
<body>
  <p> Sample Password Box </p>
  <form method="get"
action="mailto:markl@cs.ucf.edu">
    <p>Password:<input type="password" id="psswd"
/></p>
  </form>
</body>
</html>
```



Attributes For Text and Password Boxes

Common Attributes	Values	Usage
type	"text" , "password"	Configures the text box input type.
id	Alphanumeric, no spaces, begins with a letter	Provides a unique identifier for the form element.
size	Numeric	Configures the width of the text box as displayed by the browser. If size is omitted, the browser displays the text box with its own default size.
maxlength	Numeric	Configures the maximum length of data accepted by the text box.
value	Text or numeric characters	Assigns an initial value to the text box that is displayed by the browser. Accepts the information typed in the text box. This value can be accessed by client-side scripting languages and by server-side processing.

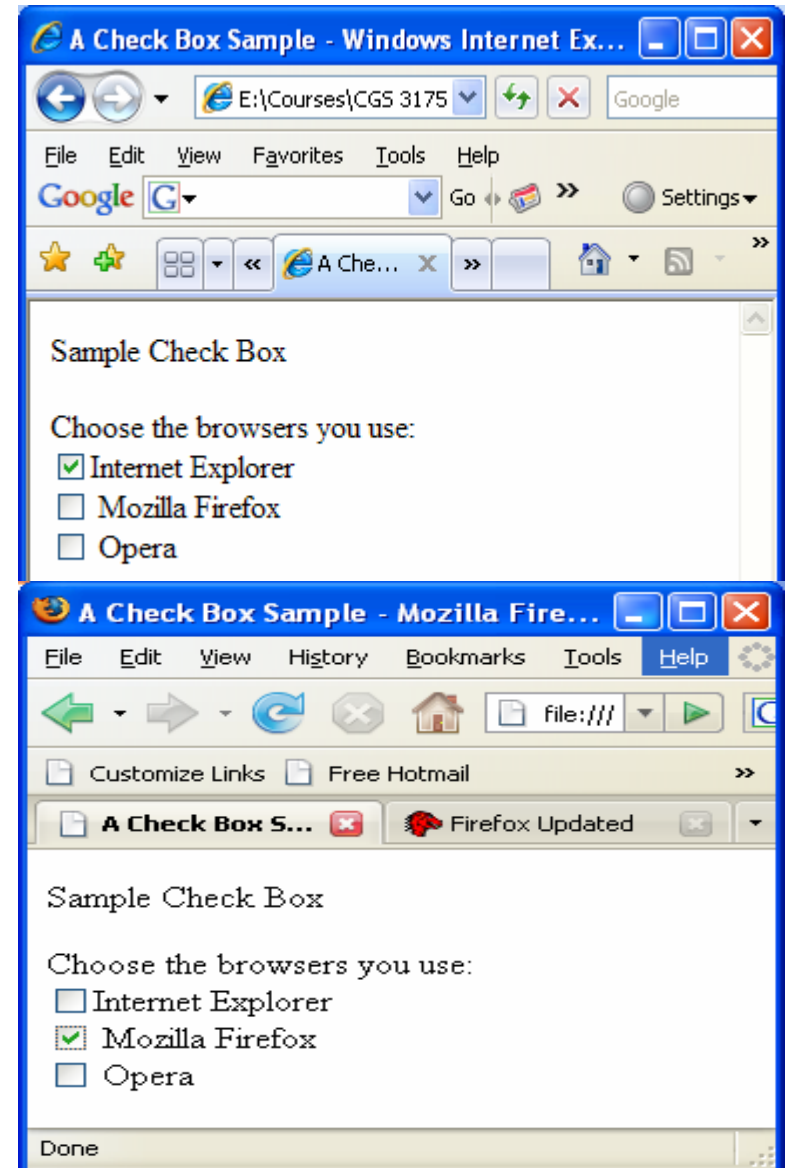


Form Element: Check Box

- Check boxes are also configured by the `<input />` tag.
- Check boxes allow the user to select one or more of a group of predetermined items.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>A Check Box Sample</title>
</head>
<body>
  <p> Sample Check Box </p>
  <form method="get" action="mailto:markl@cs.ucf.edu">
    <p>Choose the browsers you use:<br />
      <input type="checkbox" id="IE" value="yes"
/>Internet Explorer <br />
      <input type="checkbox" id="Mozilla" value="yes"
/> Mozilla Firefox <br />
      <input type="checkbox" id="Opera" value="yes"
/> Opera <br />
    </p>
  </form>
</body>
</html>
```



Attributes For Check Boxes

Common Attributes	Values	Usage
type	"checkbox"	Configures the check box.
id	Alphanumeric, no spaces, begins with a letter	Provides a unique identifier for the form element.
checked	"checkbox"	Configures the check box to be checked by default when displayed by the browser.
value	Text or numeric characters	Assigns a value to the check box that is triggered when the check box is checked. This value can be accessed by client-side and server-side processing.

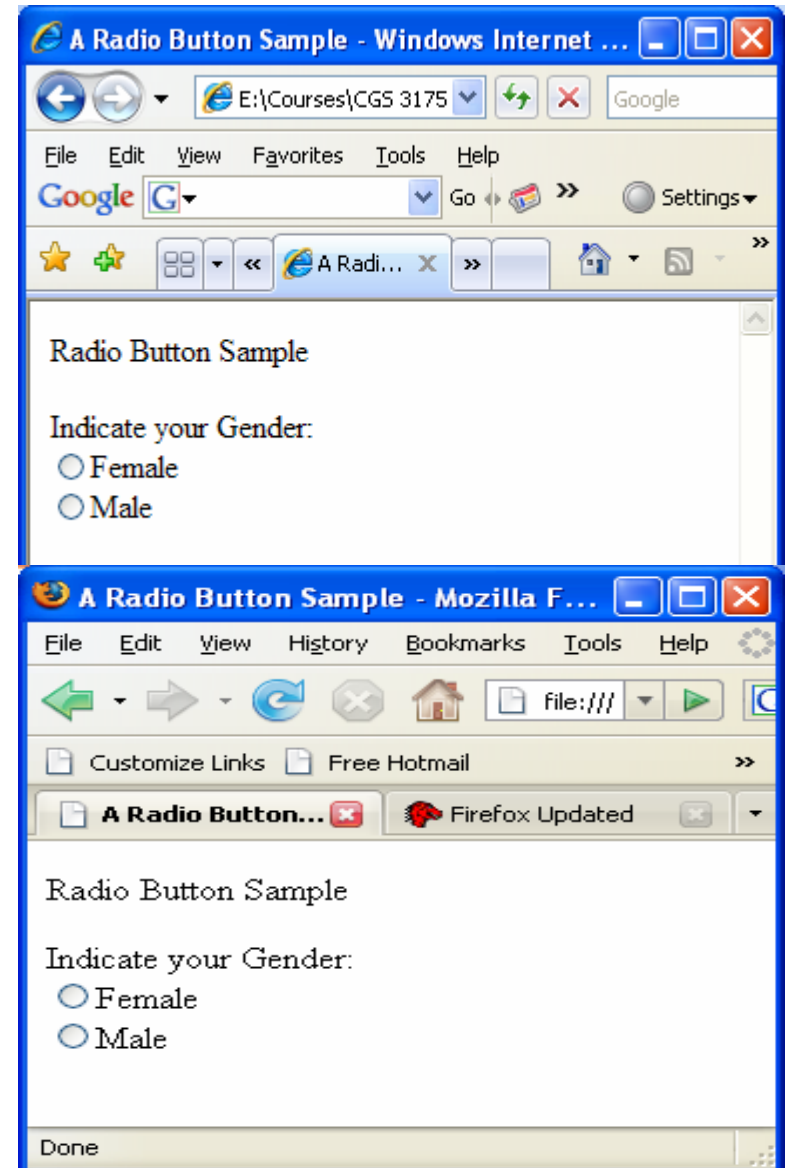


Form Element: Radio Button

- Radio buttons are also configured by the `<input />` tag.
- Radio buttons allow the user to select exactly one item from a group of predetermined items.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>A Radio Button Sample</title>
</head>
<body>
  <p> Radio Button Sample </p>
  <form method="get" action="mailto:markl@cs.ucf.edu">
    <p>Indicate your Gender:<br />
      <input type="radio" name="gender" id="gf"
value="female" />Female <br />
      <input type="radio" name="gender" id="gm"
value="male" />Male <br />
    </p>
  </form>
</body>
</html>
```



Attributes For Radio Buttons

Common Attributes	Values	Usage
type	"radio"	Configures the radio button.
name	Alphanumeric, no spaces, begins with a letter	REQUIRED – All radio buttons in a group must have the same name. This attributes also names the form element for access\by client and server side scripts.
id	Alphanumeric, no spaces, begins with a letter	Provides a unique identifier for the form element.
checked	"checked"	Configures the radio button to be selected by default when displayed by the browser.
value	Text or numeric characters	Assigns a value to the radio button that is triggered when the radio button is checked. This should be a unique value for each radio button in a group. This value can be accessed by client-side and server-side processing.

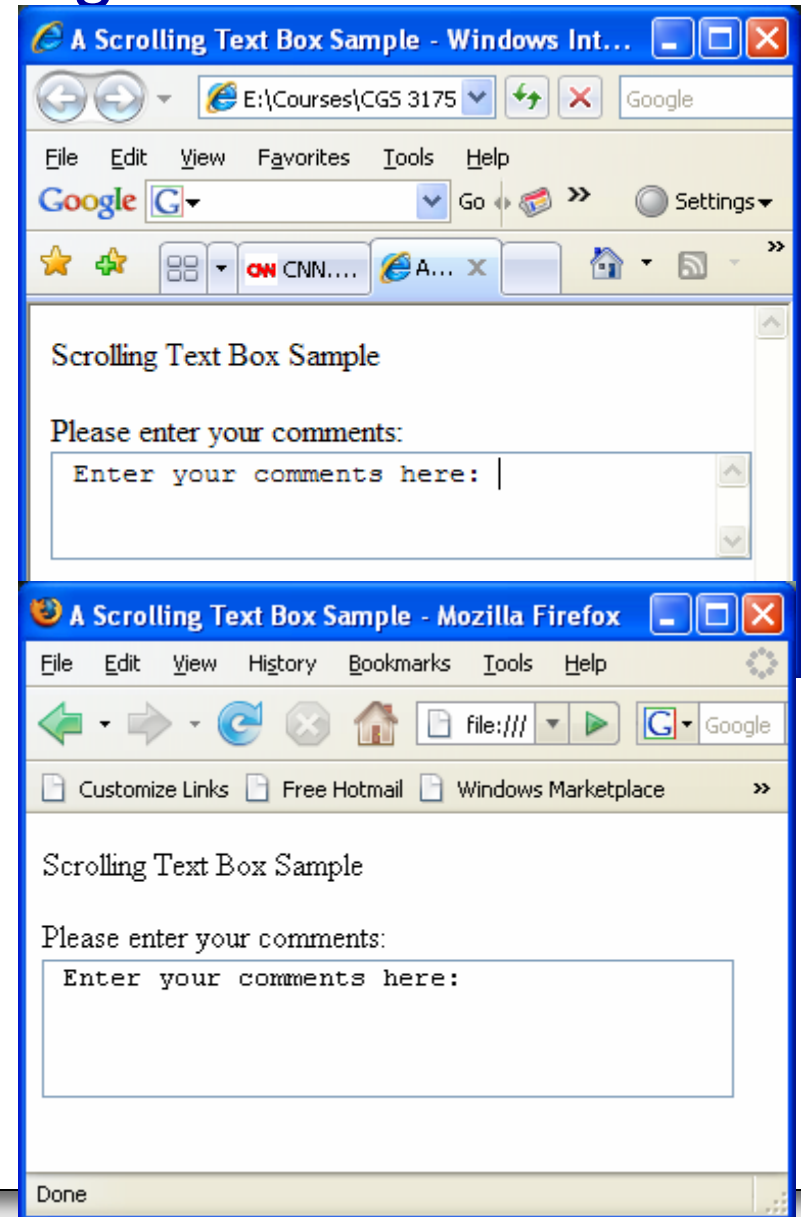


Form Element: Scrolling Text Box

- The `<textarea>` container tag configures a scrolling text box. A scrolling text box is used for accepting free-form comments, questions, or descriptions.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>A Scrolling Text Box Sample</title>
</head>
<body>
  <p> Scrolling Text Box Sample </p>
  <form method="get"
action="mailto:markl@cs.ucf.edu">
    <p>Please enter your comments:<br />
      <textarea id="comments" cols="40"
rows="3"> Enter your comments here:</textarea>
    </p>
  </form>
</body>
</html>
```



Attributes For Scrolling Text Boxes

Common Attributes	Values	Usage
id	Alphanumeric, no spaces, begins with a letter	Provides a unique identifier for the form element.
cols	numeric	Configures the width in character columns of the scrolling text box. If <code>cols</code> is omitted, the browser displays the scrolling text box with its own default width.
rows	numeric	Configures the height in rows of the scrolling text box. If <code>rows</code> is omitted, the browser displays the scrolling text box with its own default height.

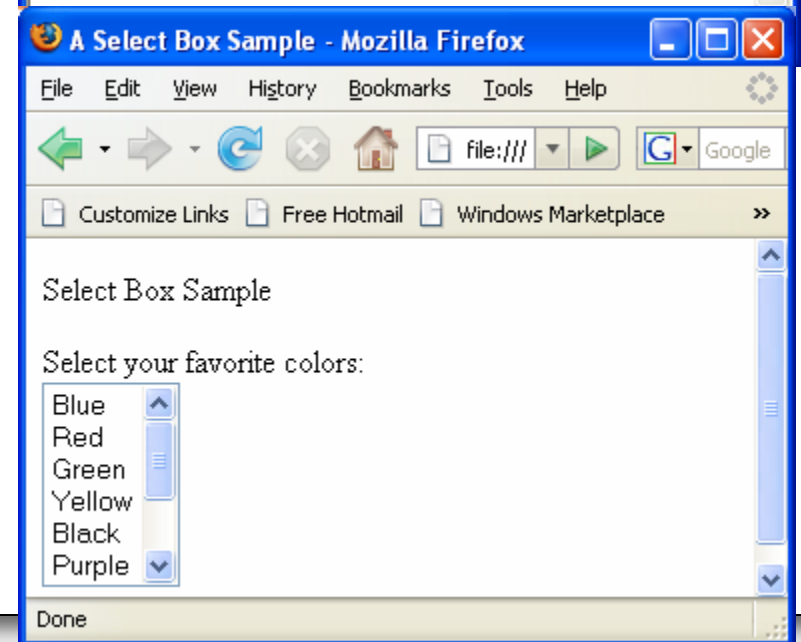
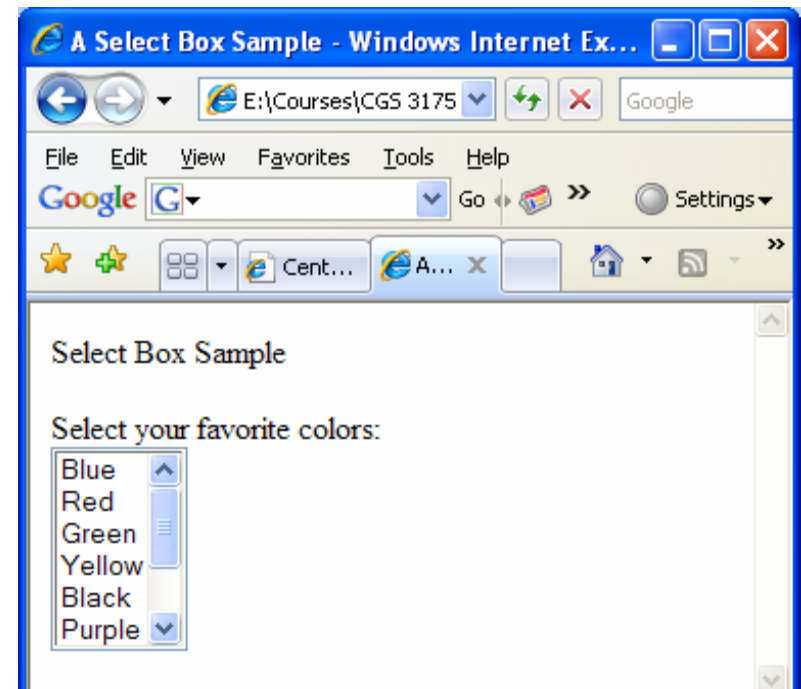


Form Element: Select List

- The `<select>` container tag (along with `<option>` tags) configures a select list. This form element has several names: select list, select box, drop-down list, drop-down box, and option box. It allows the visitor to select one or more items from a list of predetermined choices. The `<option>` container configures the choices in a select list.

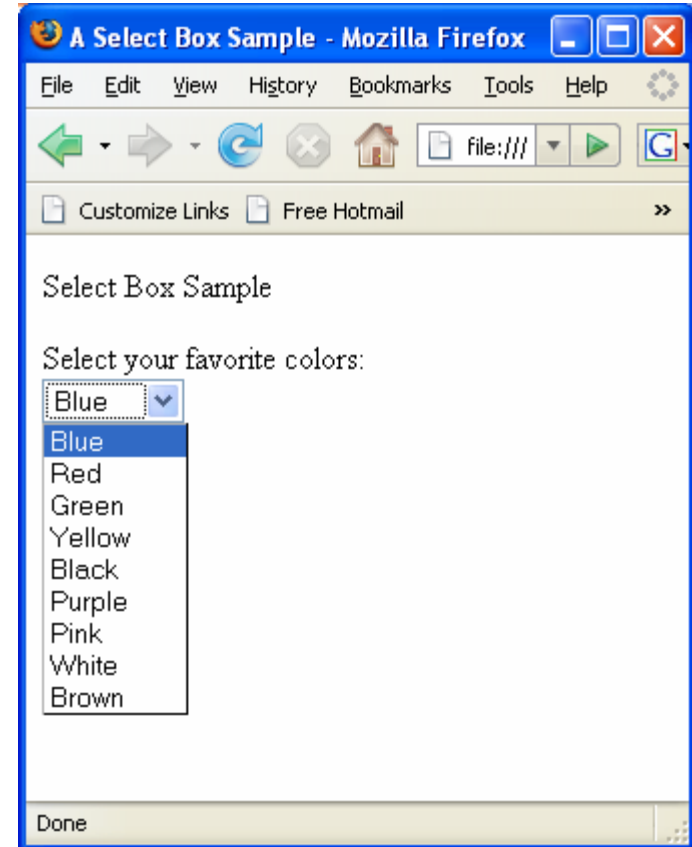
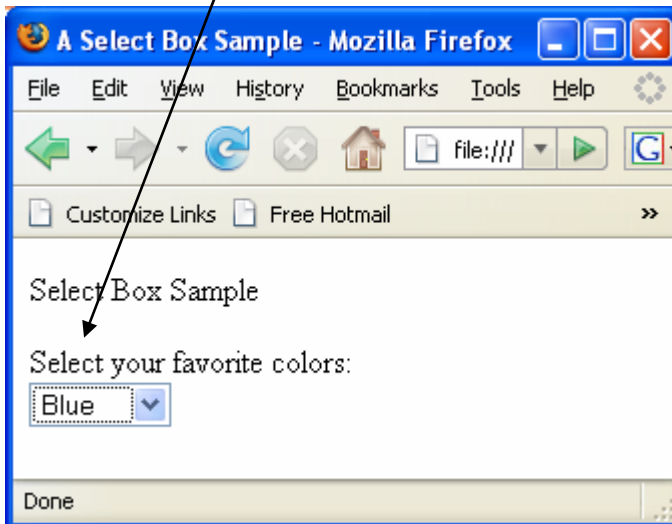
Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>A Select Box Sample</title>
</head>
<body>
  <p> Select Box Sample </p>
  <form method="get" action="mailto:markl@cs.ucf.edu">
    <p>Select your favorite colors:<br />
      <select id="favcolors" size="6">
        <option value="blue">Blue</option>
        <option value="red">Red</option>
        <option value="green">Green</option>
        <option value="yellow">Yellow</option>
        <option value="black">Black</option>
        <option value="purple">Purple</option>
        <option value="pink">Pink</option>
        <option value="white">White</option>
        <option value="brown">Brown</option>
      </select>
    </p>
  </form>
</body>
</html>
```



Form Element: Select List

With size set to 1 the select list will function as a drop-down box when the arrow is clicked.



Attributes For Select Lists

Common Attributes	Values	Usage
<select> tag		
id	Alphanumeric, no spaces, begins with a letter	Provides a unique identifier for the form element.
size	numeric	Configures the number of choices the browser will display. If set to one, the element functions as a drop-down list. Scroll bars are automatically added by the browser if the number of options exceeds the space allowed.
rows	numeric	Configures the height in rows of the scrolling text box. If rows is omitted, the browser displays the scrolling text box with its own default height.
<option> tag		
value	Text or numeric characters	Assigns a value to the option. This value can be accessed by client-side and server-side processing.
selected	"selected"	Configures an option to be initially selected when displayed by a browser.

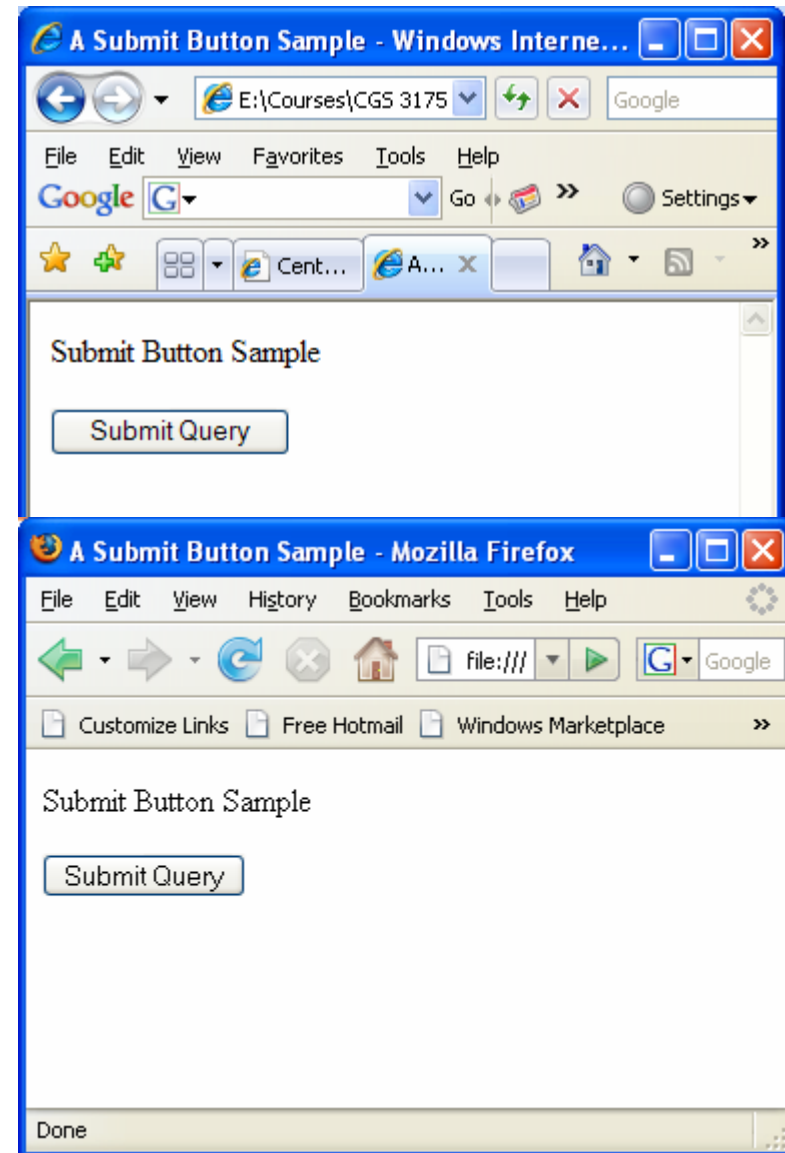


Form Element: Submit Button

- This form element is configured by the `<input />` tag and is used to submit the form. It triggers the action method on the `<form>` tag and causes the browser to send the form data to the Web server. The Web server will invoke the server-side processing listed on the form's action property.

Example:

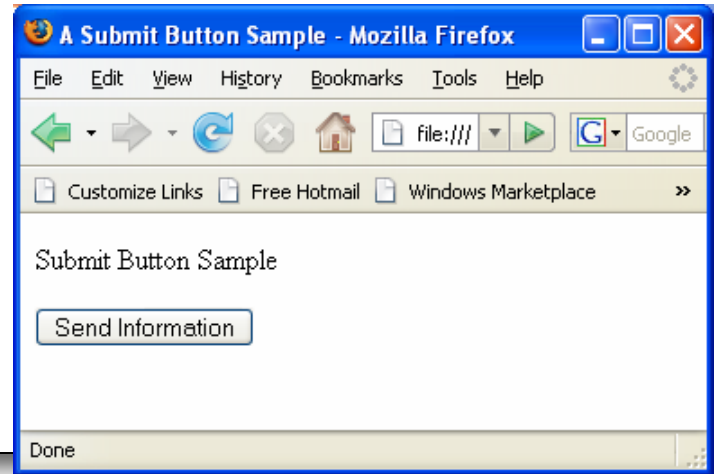
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>A Submit Button Sample</title>
</head>
<body>
<p> Submit Button Sample </p>
<form method="get"
action="mailto:markl@cs.ucf.edu">
<p><input type="submit" /></p>
</form>
</body>
</html>
```



Attributes For Submit Buttons

Common Attributes	Values	Usage
type	"submit"	Configures the submit button.
id	Alphanumeric, no spaces, begins with a letter	Provides a unique identifier for the form element.
value	Text or numeric characters	Configures the text displayed on the submit button. By default, the text "Submit Query" is displayed.

Example of a submit button with the value attribute set to "Send Information".

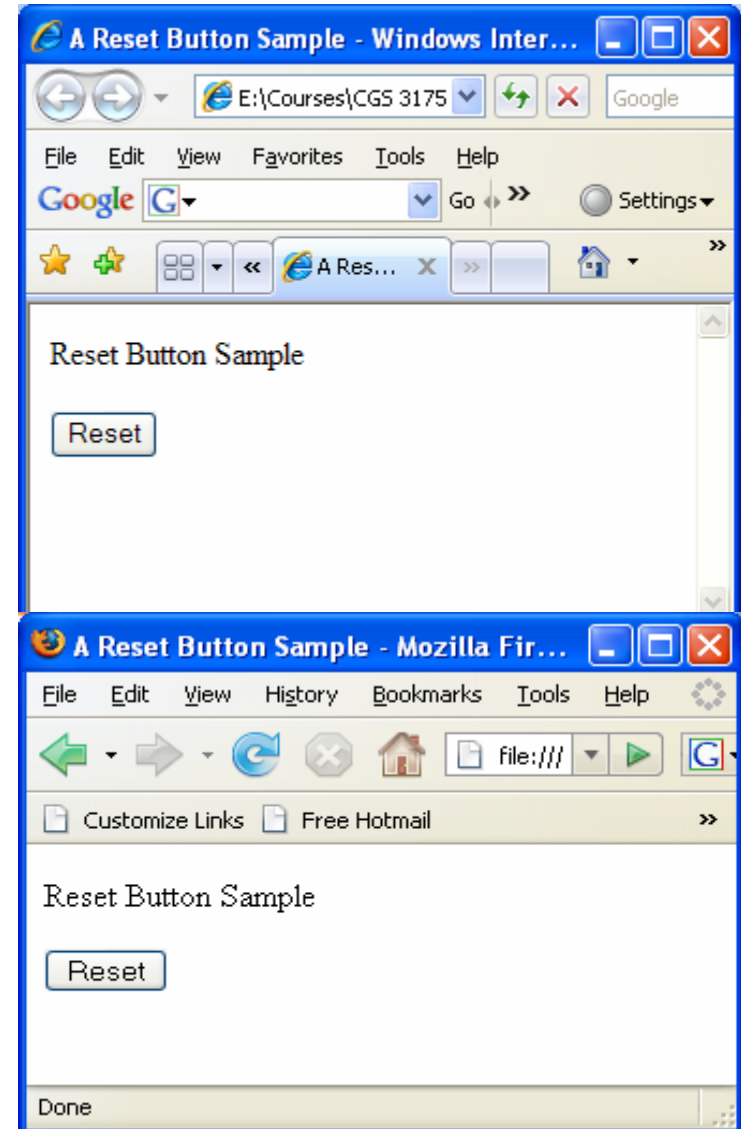


Form Element: Reset Button

- This form element is also configured by the `<input />` tag and is used to reset the form fields to their initial values.

Example:

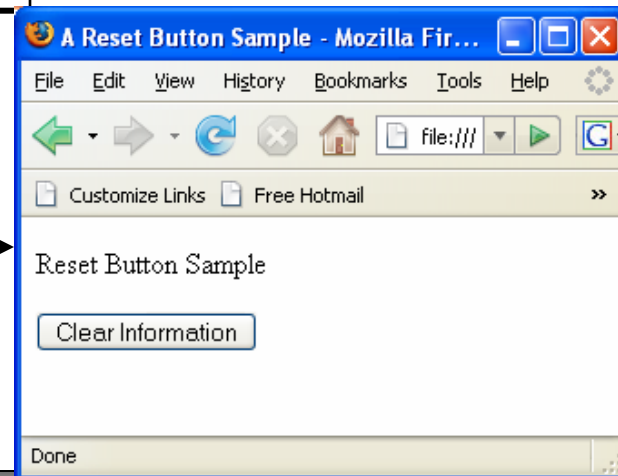
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>A Reset Button Sample</title>
</head>
<body>
  <p> Reset Button Sample </p>
  <form method="get"
action="mailto:markl@cs.ucf.edu">
    <p><input type="reset" /></p>
  </form>
</body>
</html>
```



Attributes For Reset Buttons

Common Attributes	Values	Usage
type	"reset"	Configures the reset button.
id	Alphanumeric, no spaces, begins with a letter	Provides a unique identifier for the form element.
value	Text or numeric characters	Configures the text displayed on the reset button. By default, the text "Reset" is displayed.

Example of a submit button with the value attribute set to "Clear Information".



Things to Try Yourself

19. Construct the XHTML document to create the form shown on the next page.



Google

★ ★

CNN.com - Breakin... [Valid] Markup Vali... Music Survey - ...

Music Survey

Name:

E-mail:

Select Your Favorite Types Of Music

- Pop
- Rock
- Hip-Hop
- Classical
- Country
- Folk
- Other

Select how often you purchase music CDs:

- Daily
- Weekly
- Monthly
- A few times each year
- Never

Comments:

Done My Computer 100%

