

23CSCI03P

Web Programming

Dr. Nivin Atef, Ph.D.

Nivine.atef@bue.edu.eg

Chapter 10

Form Basics

Key Concepts

Copyright © 2013

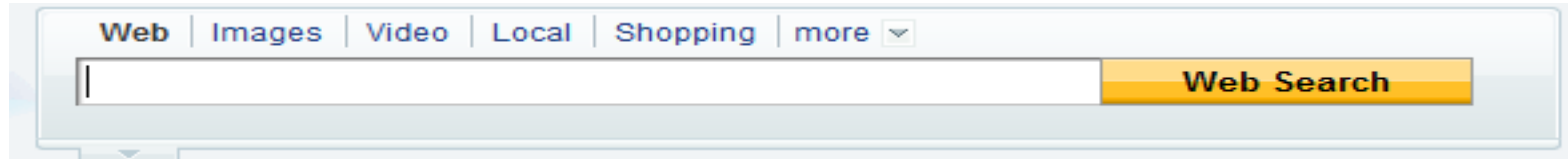
Terry Ann Morris, Ed.D

Learning Outcomes

- Describe **common uses** of **forms** on web pages
- Create forms on web pages using the form, input, textarea, and select elements
- Associate **form controls** and **groups** using label, fieldset, and legend elements
- Configure **new HTML5** form **controls** and **attributes**

Overview of Forms

- Forms are **used** all over the Web to
 - **Accept information**
 - **Provide interactivity**



- Types of forms:
 - **Search form, Order form, Newsletter sign-up form, Survey form, Add to Cart form, and so on...**

Two Components of Using Forms

1. The **web page form**

An **HTML element** that **contains** and **organizes form controls** such as text boxes, check boxes, and buttons that can accept information from website visitors.

2. The **server-side processing**

Server-side processing works with the form data and sends e- mail, writes to a text file, **updates a database**, or performs some other **type of processing** on the server.

Sample Form HTML

A screenshot of a Firefox browser window. The title bar shows 'Firefox' and standard window controls. The address bar contains 'Form Example'. The main content area displays a form titled 'Join Our Newsletter' in a large, bold, serif font. Below the title, the text 'E-mail:' is followed by a text input field. At the bottom of the form, there are two buttons: 'Sign Me Up!' and 'Reset'.

`<form>`

E-mail: `<input type="text" name="email" id="email" >`

`

`

`<input type="submit" value="Sign Me Up!"> <input type="reset">`

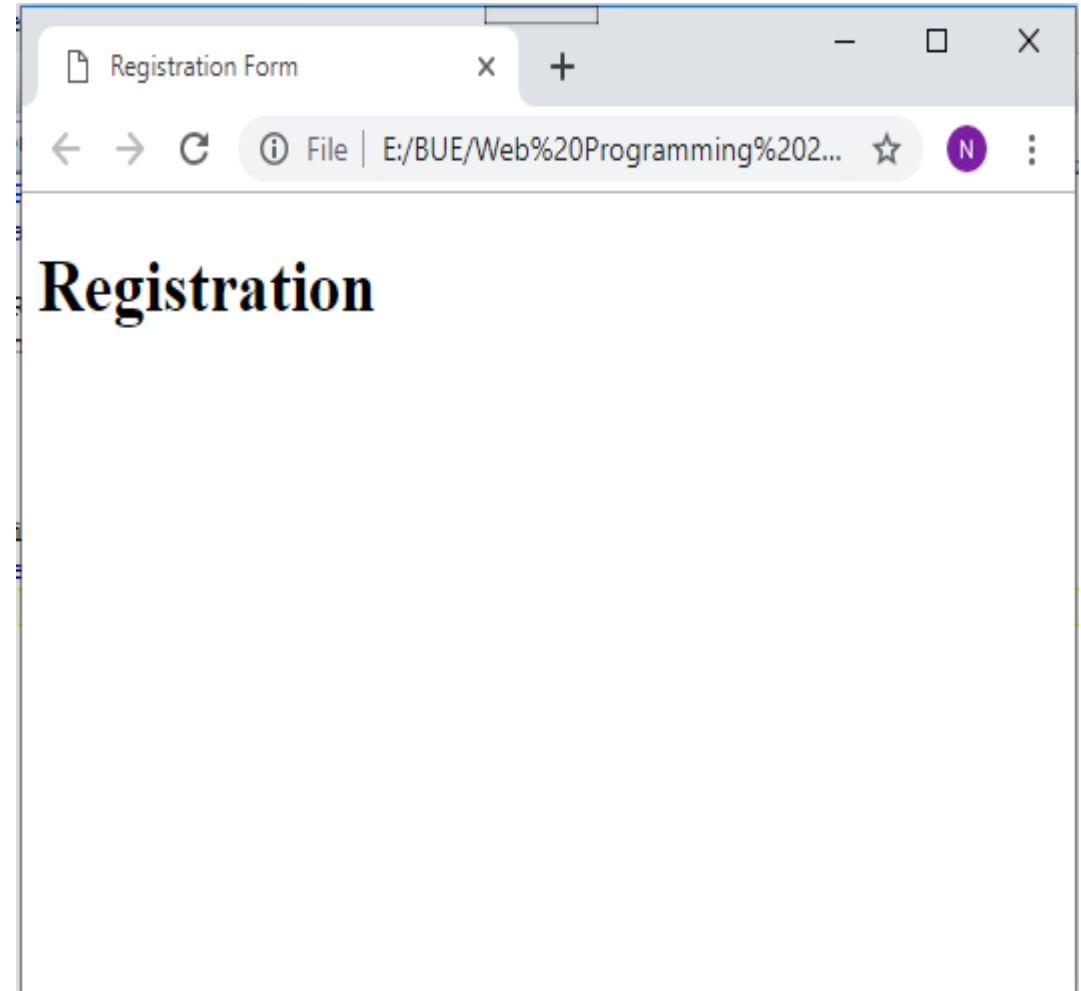
`</form>`

The Form Element <form>

- **Container Tag**
- The form element attributes:
 - **Action:** Specifies the **server-side program** or **script** that will process your form data
 - Specifies **where to send the form-data** when a form is submitted
 - **Method:**
 - get – default value, form data passed in URL
 - post – more secure, form data passed in HTTP Entity Body
 - **Name:** Identifies the form

Form Example

```
<body>  
<h1>Registration</h1>  
<form method="get">  
</form>  
</body>
```



The Input Element Text Box <input>

- **Accepts** text information
- Common Attributes:
 - Type="text"
 - Name: **unique** to be accessed by the **server side**
 - Id: **unique** to be used for **javascript**
 - Size: configures **width** of textbox in the browser
 - Maxlength: max no of characters as input
 - Value: default value
 - Required: (HTML5)
 - Placeholder: tip or hint to user (HTML5)



A sample text box with the label "Email:" and an empty input field.

Sample Text Box

Email:

<input type="text" > Example

```
<body>
```

```
<h1>Registration</h1>
```

```
<form method="get">
```

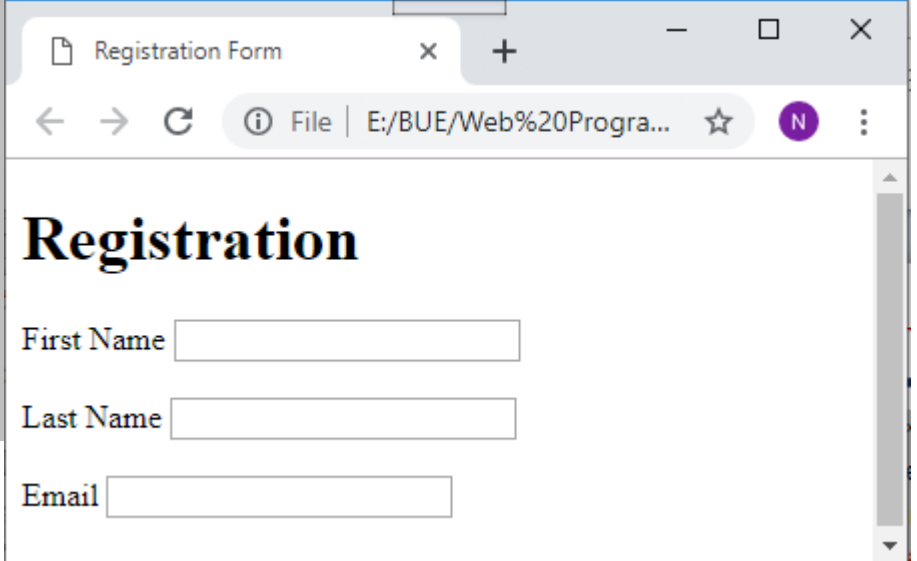
```
First Name <input type="text" name="FName" id="FName"><br><br>
```

```
Last Name <input type="text" name="LName" id="LName"><br><br>
```

```
Email <input type="text" name="email" id="email"><br><br>
```

```
</form>
```

```
</body>
```



The screenshot shows a web browser window titled "Registration Form". The address bar displays "File | E:/BUE/Web%20Progra...". The page content features a heading "Registration" followed by three text input fields labeled "First Name", "Last Name", and "Email".

The Input Element Password Box <input>

- **Accepts text information** that needs to be **hidden** as it is entered.
- **Common Attributes:**
 - type="password"
 - name
 - id
 - size
 - maxlength
 - value
 - required (HTML5)
 - placeholder (HTML5)



A sample password box with a light gray background and a subtle drop shadow. It features the text "Sample Password Box" in a bold, black, serif font at the top. Below this, the label "Password:" is followed by a text input field. The input field contains ten small black dots, representing a hidden password, and has a vertical cursor at the end.

Form Example

```
<body>
<h1>Registration</h1>
<form method="get">
```

.....

**Password <input type="password" name="passw" id="passw">

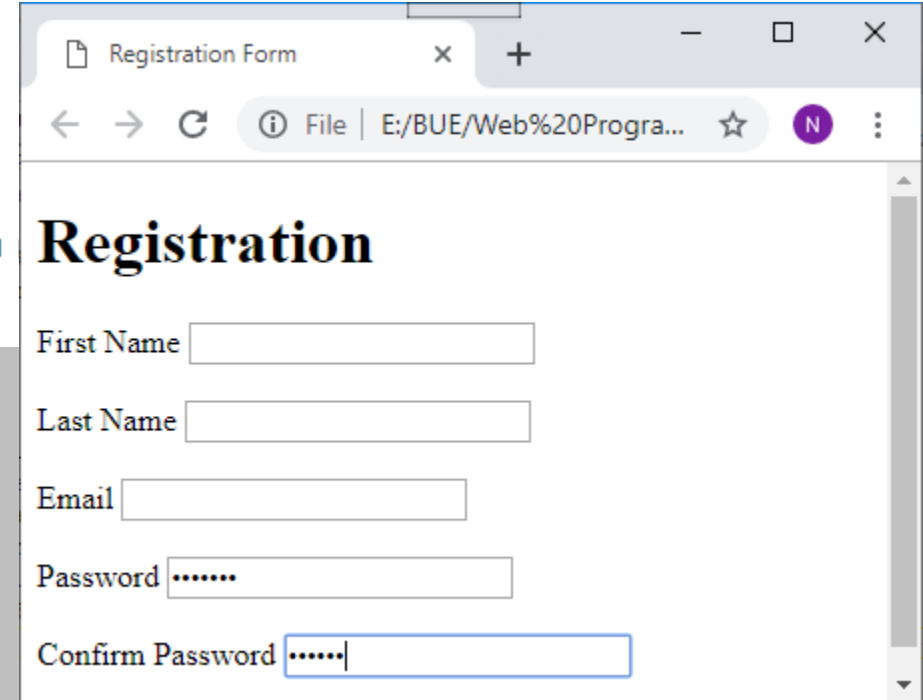
**

**Confirm Password <input type="password" name="cpass" id="cpass">

**

```
</form>
```

```
</body>
```

A screenshot of a web browser window displaying a registration form. The browser's address bar shows the file path "E:/BUE/Web%20Progra...". The form is titled "Registration" in a large, bold, black serif font. Below the title, there are five input fields: "First Name", "Last Name", "Email", "Password", and "Confirm Password". The "Password" and "Confirm Password" fields are filled with dots, indicating they are password inputs. The "Confirm Password" field has a blue border, suggesting it is the active input field. The browser window includes standard navigation buttons (back, forward, refresh) and a tab labeled "Registration Form".

The Input Element Check box <input>

- Allows the user to **select one or more** of a group of **predetermined items**

- Common Attributes:

- type="checkbox"
- name
- id
- Checked: **Specifies** that an <input> **element** should be **pre-selected** when the **page loads**
- value

Sample Check Box

Choose the browsers you use:

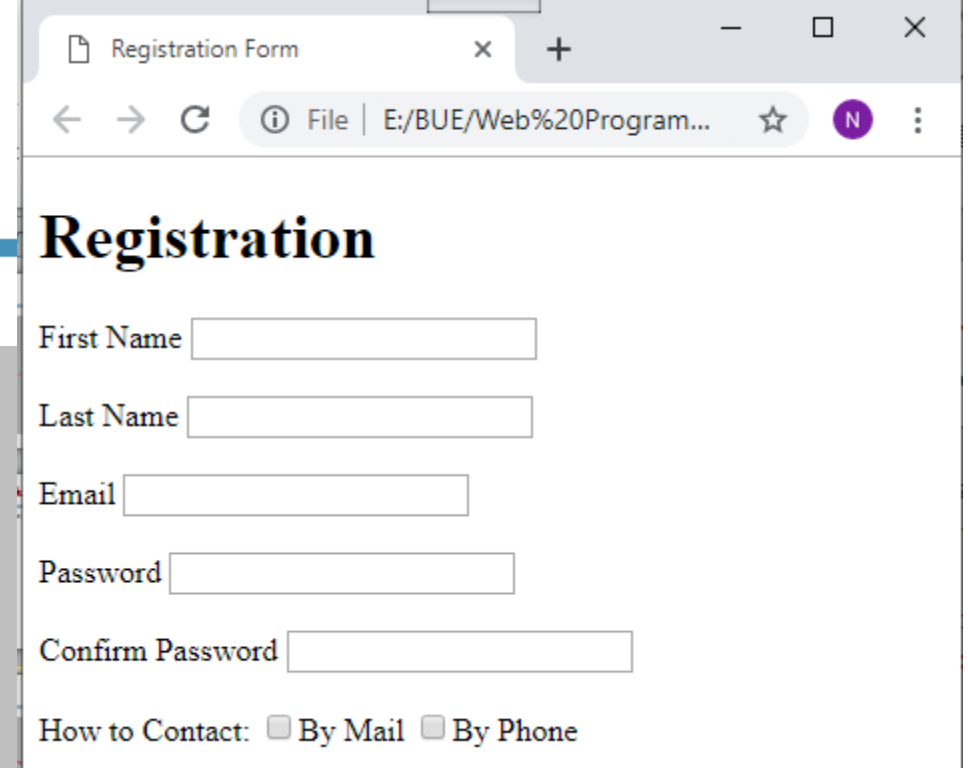
- ☐ Internet Explorer
- ☐ Firefox
- ☐ Opera

Form Example

```
<body>
<h1>Registration</h1>
<form method="get">
```

.....

```
How to Contact: <input type="checkbox" name="bymail" id="bymail" value="mail">By Mail
                <input type="checkbox" name="byphone" id="byphone" value="phone">By Phone
<br>
</form>
</body>
```

A screenshot of a web browser window displaying a registration form. The browser's address bar shows the file path "E:/BUE/Web%20Program...". The form is titled "Registration" and contains several input fields: "First Name", "Last Name", "Email", "Password", and "Confirm Password". At the bottom, there is a "How to Contact:" section with two radio buttons labeled "By Mail" and "By Phone".

The Input Element Radio Button <input>

- **Allows** the user to **select exactly one** from a group of **predetermined items**
- Each radio button in a group is **given the same name** and a **unique value**
- **Comon Attributes:**
 - type="radio"
 - name
 - id
 - checked
 - value

Sample Radio Buttons

Select your favorite browser:

- ☐ Internet Explorer
- ☐ Firefox
- ☐ Opera

Form Example

```
<body>
```

```
<h1>Registration</h1>
```

```
<form method="get">
```

```
.....
```

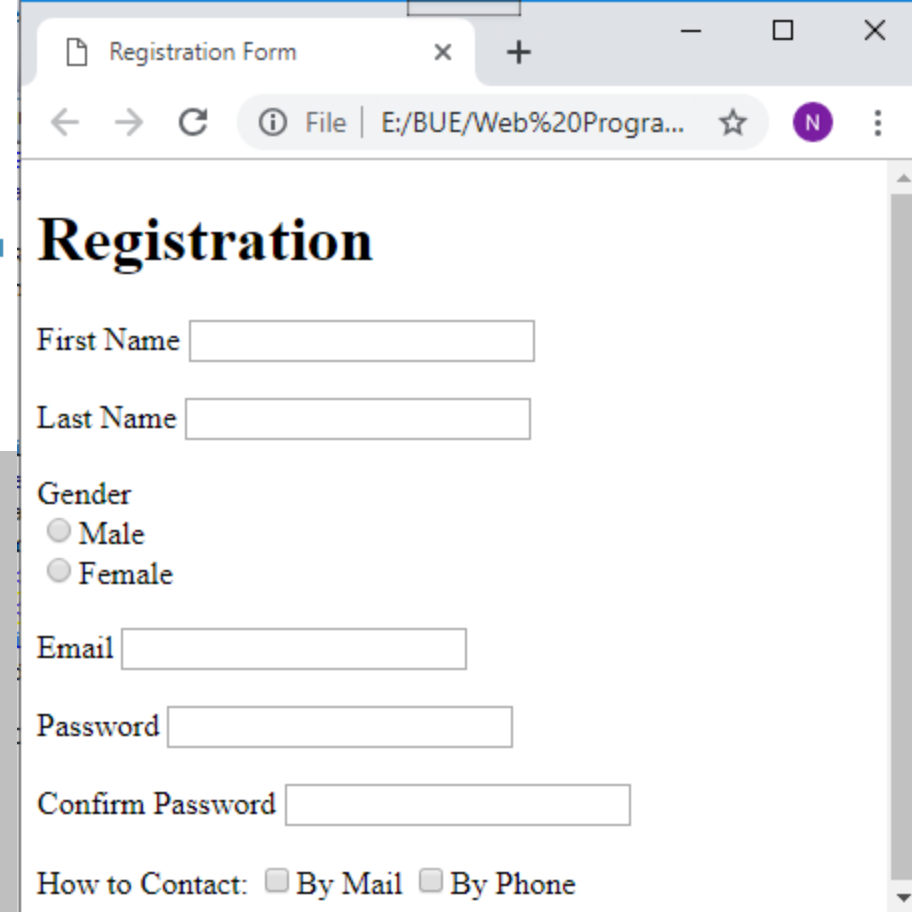
```
Gender <br>
```

```
<input type="radio" name="Gender" id="Male" value="male">Male<br>
```

```
<input type="radio" name="Gender" id="Female" value="female">Female<br><br>
```

```
</form>
```

```
</body>
```

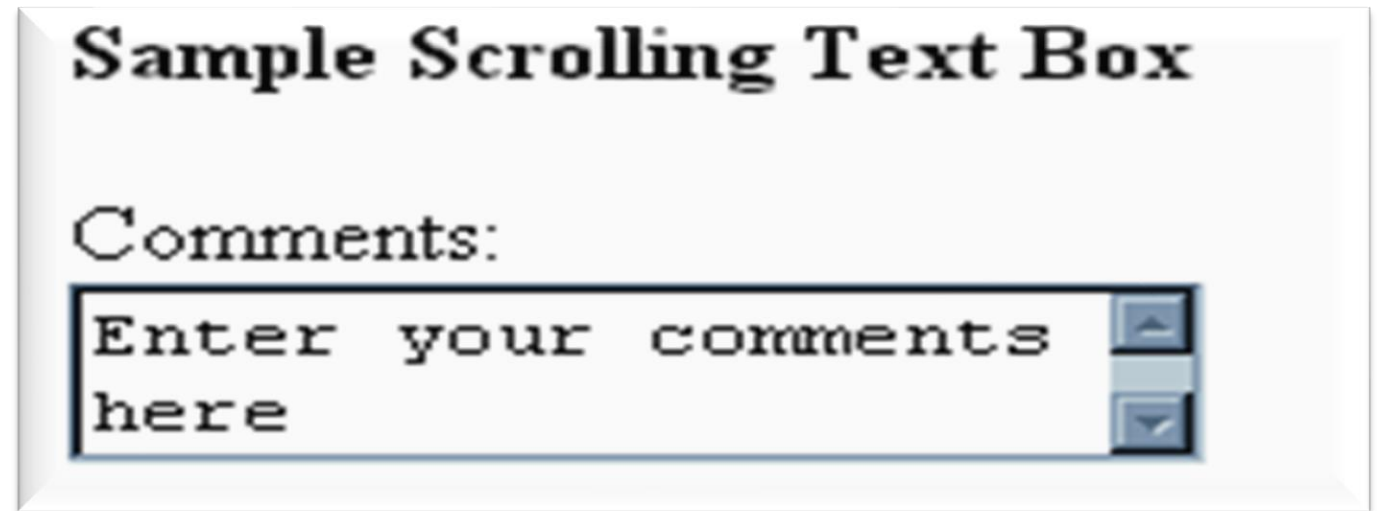


The screenshot shows a web browser window with a single tab titled "Registration Form". The address bar shows a file path: "E:/BUE/Web%20Progra...". The page content is titled "Registration" in a large, bold, black serif font. Below the title, the form contains the following fields and controls:

- "First Name" followed by a text input field.
- "Last Name" followed by a text input field.
- "Gender" section with two radio buttons: "Male" and "Female".
- "Email" followed by a text input field.
- "Password" followed by a text input field.
- "Confirm Password" followed by a text input field.
- "How to Contact:" followed by two checkboxes: "By Mail" and "By Phone".

The Textarea Element <textarea>

- **Configures a scrolling text box**
- Common Attributes:
 - name
 - id
 - cols
 - rows



Form Example

```
<body>
```

```
<h1>Registration</h1>
```

```
<form method="get">
```

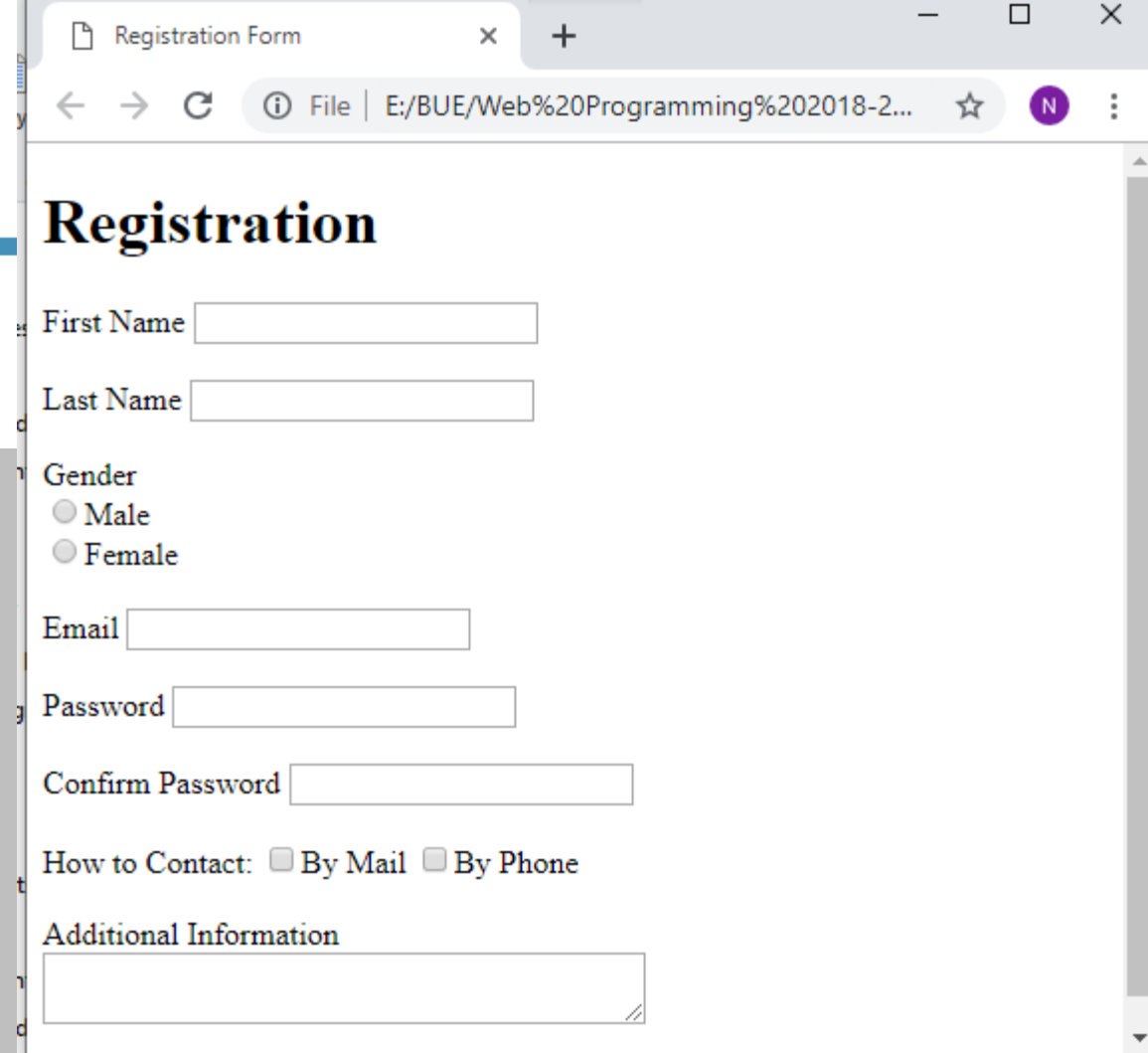
.....

```
Additional Information <br>
```

```
<textarea name="Add" id="Add" cols="40"
rows="2"></textarea><br><br>
```

```
</form>
```

```
</body>
```

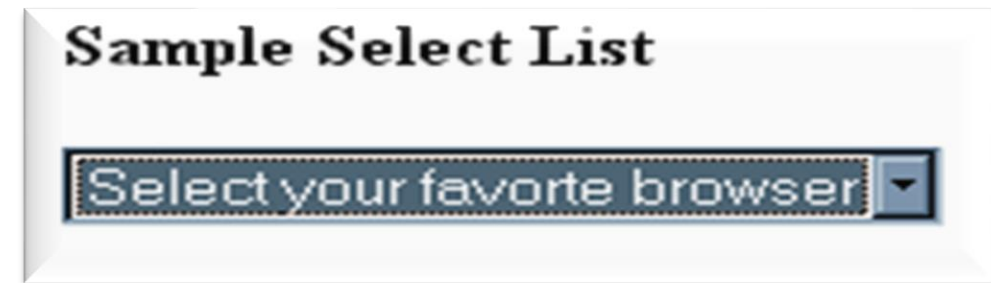


The screenshot shows a web browser window with the title "Registration Form". The address bar displays the file path "E:/BUE/Web%20Programming%202018-2...". The form itself is titled "Registration" and contains the following fields and controls:

- First Name:
- Last Name:
- Gender: ☐ Male ☐ Female
- Email:
- Password:
- Confirm Password:
- How to Contact: ☐ By Mail ☐ By Phone
- Additional Information:

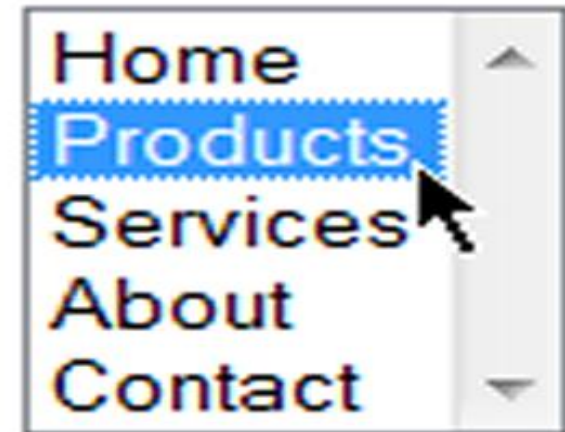
The Select Element <select>

- **Configures a select list** (along with <option> tags)
- Also known as: Select Box, Drop-Down List, Drop-Down Box, and Option Box.
- Allows the user to **select one or more items** from a list of predetermined choices.
- Common Attributes:
 - name
 - id
 - Size: **Defines** the **number of visible options** in a drop-down list



The Option Element <option>

- **Configures the options in a Select List**
- **Attributes:**
 - value



The Select Element <select> Example

```
<body>
<h1>Registration</h1>
<form method="get">
.....
Country <select size="1" name="country" id="country">
  <option>Select Your Country</option>
  <option value="Egypt">Egypt</option>
  <option value="US">United States</option>
  <option value="china">China</option>
  <option value="India">India</option>
</select><br><br>
</form>
</body>
```

Registration

First Name

Last Name

Gender

☐ Male

☐ Female

Country

Email

Password

Confirm Password

How to Contact: ☐ By Mail ☐ By Phone

Additional Information

The Input Element Submit Button <input>

- **Submits the form information**
- When clicked:
 - **Triggers** the **action** method on the <form> tag
 - **Sends the form data** (the name=value pair for each form element) to the web server.
- **Attributes:**
 - type="submit"
 - name
 - id
 - value


Sample Submit Button

A rectangular button with a light gray background and a thin black border. The text "Submit Query" is centered on the button in a black, sans-serif font.

The Input Element Reset Button <input>

- Resets the form fields to their initial values
- Attributes:
 - type="reset"
 - name
 - id
 - value

Sample Reset Button



Reset

Form Example

```
<body>
<h1>Registration</h1>
<form method="get">
.....
<input type="submit" name="submit"
      id="submit" value="Submit">
<input type="reset">
</form>
</body>
```

Registration

First Name

Last Name

Gender
☐ Male
☐ Female

Country

Email

Password

Confirm Password

How to Contact: ☐ By Mail ☐ By Phone

Additional Information

The Label Element <label>

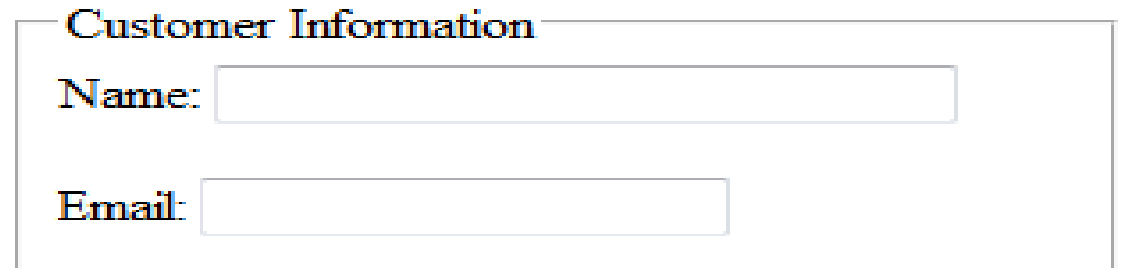
- Associates a **text label** with a **form control**
- Two Different Formats:
`<label>Email: <input type="text" name="CustEmail" id="CustEmail"></label>`

Or

```
<label for="email">Email: </label>  
<input type="text" name="CustEmail" id="email" />
```

The Fieldset & Legend Elements

- The Fieldset Element <fieldset>
 - **Container tag**
 - **Creates a visual group** of form controls on a web page
- The Legend Element <legend>
 - **Container tag**
 - **Creates a text label** within the **fieldset**



```
<fieldset><legend>Customer Information</legend>
  <label>Name:
  <input type="text" name="CName" id="CName" size="30"></label> <br><br >
  <label>Email:
  <input type="text" name="CEmail" id="CEmail"></label>
</fieldset>
```

The Fieldset & Legend Elements Example

```
<body>
<h1>Registration</h1>
<form method="get">
.....

<fieldset><legend>Gender</legend>
<input type="radio" name="Gender" id="Male"
value="male">Male<br>
<input type="radio" name="Gender" id="Female"
value="female">Female
</fieldset><br>
</form>
</body>
```

Registration

First Name

Last Name

Gender

- ☐ Male
☐ Female

Country

Email:

Password

Confirm Password

How to Contact: ☐ By Mail ☐ By Phone

Additional Information

HTML5: Email Text Box <input>

- **Accepts** text information in **e-mail address format**
- **Common Attributes:**
 - type="email"
 - name
 - id
 - size
 - maxlength
 - value
 - placeholder
 - required



HTML5: URL Text Box <input>

- **Accepts** text information in **URL format**
- **Common Attributes:**
 - type="url"
 - name
 - id
 - size
 - maxlength
 - value
 - placeholder
 - required



HTML5: Datalist Control

```
<label for="country">Country</label>
<input type="text" name="country" id="country"
  list="countries" >
<datalist id="countries">
  <option value="Egypt">
  <option value="US">
  <option value="china">
  <option value="India">
</datalist><br><br>
```

Registration

First Name

Last Name

Gender

☐ Male ☐ Female

Country

Email

Password

Confirm Password

How to Contact ☐ By Mail ☐ By Phone

Country

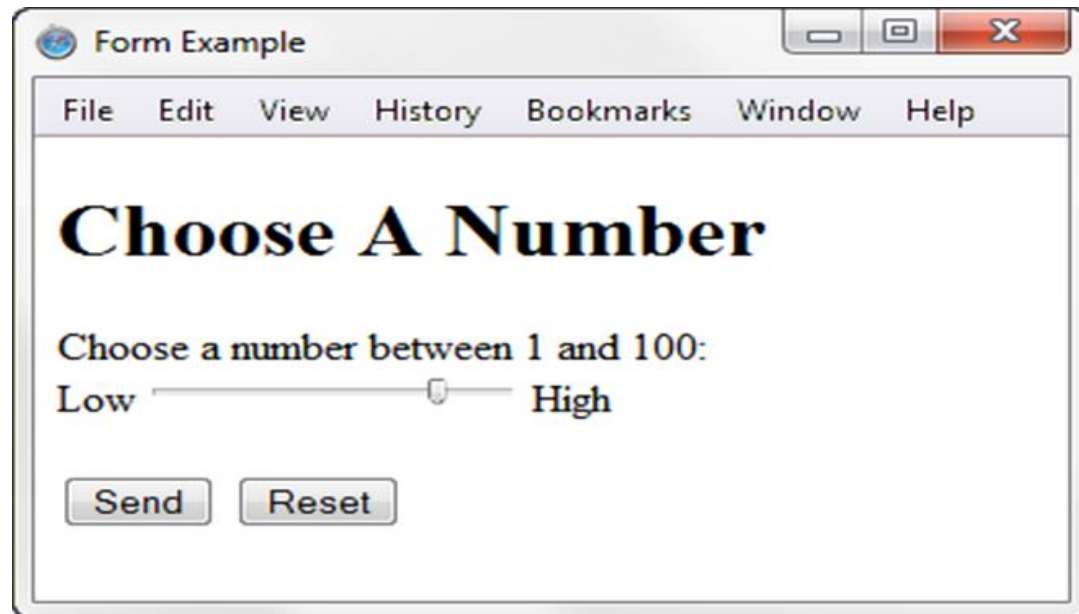
Email

HTML5: Slider Control <input>

<label for="myChoice">

Choose a number between 1 and 100:</label>

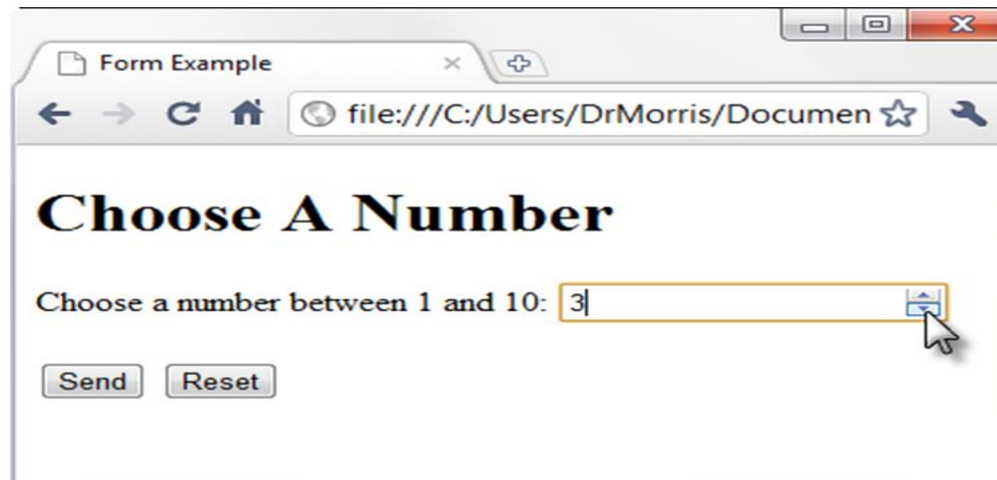
Low <input type="range" name="myChoice" id="myChoice" min="0" max="100" > High



The screenshot shows a web browser window titled "Form Example". The browser has a menu bar with "File", "Edit", "View", "History", "Bookmarks", "Window", and "Help". The main content area displays the text "Choose A Number" in a large, bold, black serif font. Below this, the text "Choose a number between 1 and 100:" is shown in a smaller black serif font. Underneath is a horizontal slider control. The slider has a horizontal line with a small vertical handle in the middle. The word "Low" is to the left of the line and "High" is to the right. At the bottom of the form, there are two buttons: "Send" and "Reset", both with a light gray background and a thin border.

HTML5: Spinner Control <input>

```
<label for="myChoice">Choose a number between 1 and 10:</label>  
<input type="number" name="myChoice" id="myChoice"  
      min="1" max="10">
```



The screenshot shows a web browser window with the title 'Form Example'. The address bar displays 'file:///C:/Users/DrMorris/Documen'. The main content area has the heading 'Choose A Number'. Below the heading is the text 'Choose a number between 1 and 10:' followed by a spinner input field. The input field contains the number '3' and has a small blue spinner icon on its right side. A mouse cursor is hovering over the spinner icon. Below the input field are two buttons: 'Send' and 'Reset'.

HTML5: Calendar Control <input>

```
<body>
<h1>Registration</h1>
<form method="get">

.....

<label for="BDate">Birth Date</label>
<input type="date" name="BDate"
id="BDate"><br><br>

</form>
</body>
```

The screenshot shows a web form titled "Registration" with the following fields:

- First Name:
- Last Name:
- Gender: ☐ Male ☐ Female
- Birth Date: (with a date picker calendar open)
- Country:
- Email:
- Password:
- Confirm Password:
- How to Contact: ☐ By Mail ☐ By Phone
- Additional Information:

The date picker calendar is open, showing the month of April 2019. The calendar grid is as follows:

Sun	Mon	Tue	Wed	Thu	Fri	Sat
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4

At the bottom of the form are "Submit" and "Reset" buttons.

HTML5 Color Control

```
<body>
```

```
<h1>Registration</h1>
```

```
<form method="get">
```

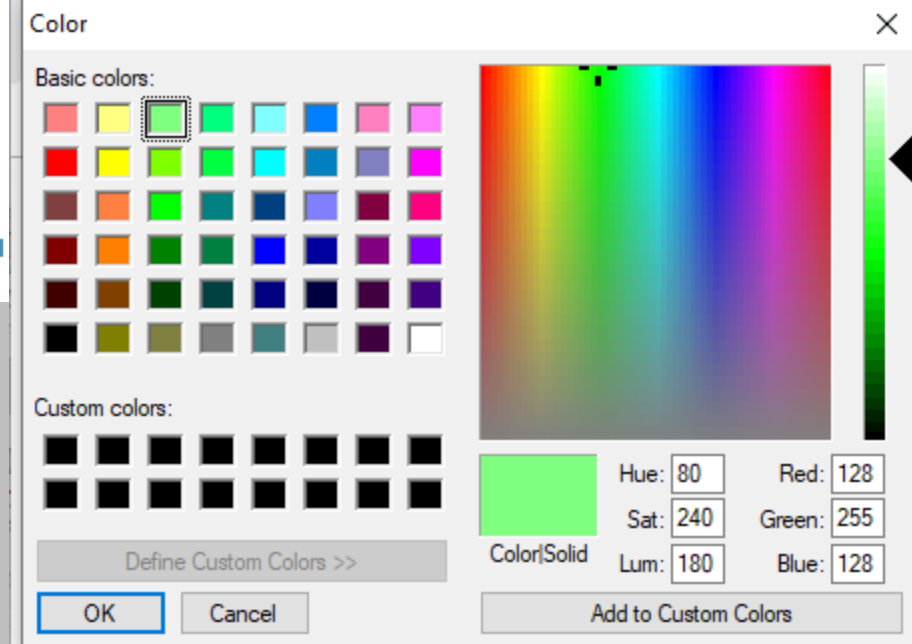
.....

```
<label for="myColor">Your Favorite Color</label>
```

```
<input type="color" name="myColor"  
id="myColor">
```

```
</form>
```

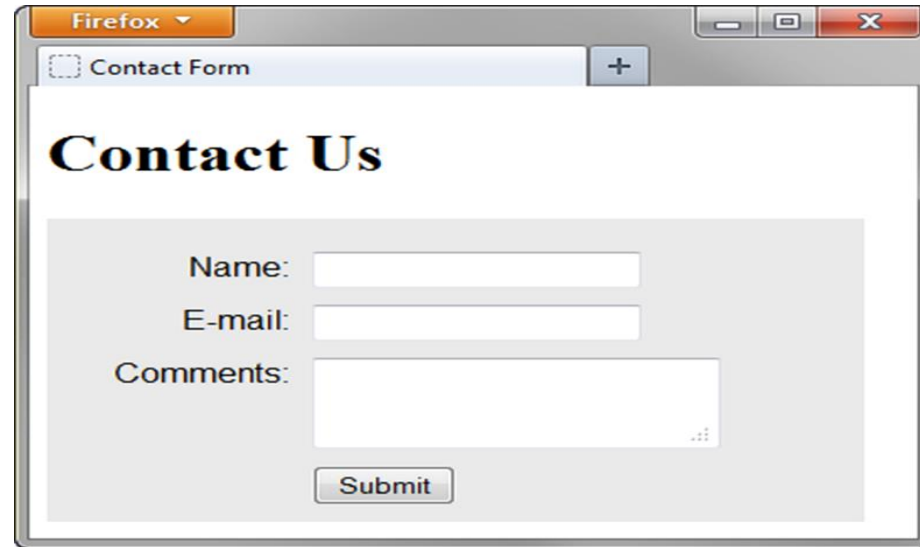
```
</body>
```



Using CSS to Style a Form

form

label	text box
label	text box
label	scrolling text box
submit button	



A screenshot of a web browser window titled "Contact Form" in the address bar. The page has a light gray background and a title "Contact Us" in a bold, serif font. Below the title, there is a form with three input fields: "Name:", "E-mail:", and "Comments:". Each field is preceded by its label. The "Comments:" field is a larger text area. At the bottom of the form, there is a "Submit" button.

```
form { background-color:#eaeaea; font-family: Arial, sans-serif;
        width: 350px; padding: 10px;}
label { float: left; clear: left; display: block; width: 100px;
        text-align: right; padding-right: 10px; margin-top: 10px; }
input, textarea { margin-top: 10px; display: block;}
#mySubmit { margin-left: 110px; }
```

Summary

- This chapter introduced the use of forms on web pages.
- You learned about how to configure form elements and provide for accessibility.
- You also learned about new HTML5 form controls and attributes.

Chapter 11

Media and Interactivity Basics

Key Concepts

Copyright © 2013

Terry Ann Morris, Ed.D

Learning Outcomes

- Describe **types** of **multimedia** files used on the Web
- Configure **hyperlinks** to **multimedia** files
- Configure **audio** and **video** on a web page with HTML5 elements
- Use the **CSS3 transform** and **transition** properties

Helper Applications & Plug-ins

- **Helper Application**

- A program that can be designated to **handle** a particular **file type** (such as .wav or.mpg) to allow the user to view or otherwise utilize the special file.
- The helper application runs in a **separate window** from the browser.

- **Plug-In**

- A newer and more common method
- Plug-ins **run right in** the browser window so that media objects can be **integrated** directly into the web page.

Commonly Used Plug-ins

- Adobe Reader
- Windows Media Player
- Apple Quicktime

Common Audio File Types

- .wav Wave File
- .aiff Audio Interchange File Format
- .mid Musical Instrument Digital Interface (MIDI)
- .au Sun UNIX sound file
- .mp3 MPEG-1 Audio Layer-3
- .ogg Ogg Vorbis (open-source)
- .m4a MPEG 4 Audio.
This audio-only MPEG-4 format is supported by Quicktime, iTunes, and iPods.

Common Video File Types

- .mov Quicktime
- .avi Microsoft Audio Video Interleaved
- .wmv Windows Media File
- .flv Flash Video File
- .mpg MPEG (Motion Picture Experts Group)
- .m4v .mp4 (MPEG-4)
- .ogv Ogg Theora (open-source)
- .webm VP8 codec (open video format, free)

Copyright Issues

- **Only publish** web pages, images, and other media that you have **personally created** or have obtained the rights or license to use.
- **Ask permission** to use media created by another person instead of simply “grabbing” it.
- All work (including web pages) are **automatically copyrighted** even if there is not copyright mark or date.
- **Fair Use Clause of the Copyright Act**

Configure Audio & Video

- Most basic method to provide audio or video files:
 - Hyperlink

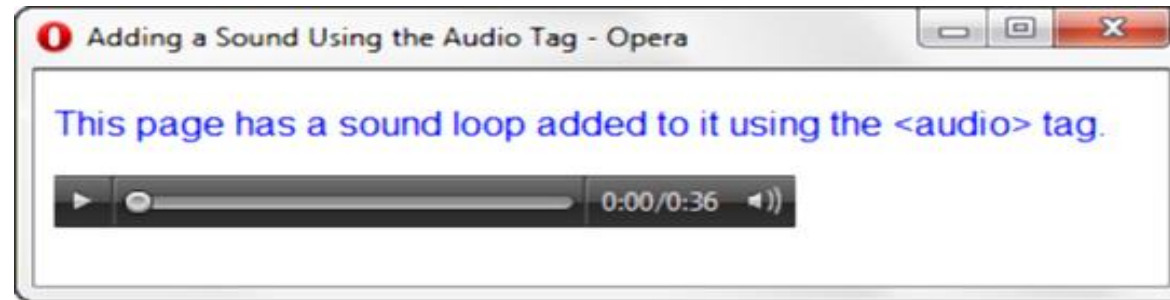
```
<a href="wdfpodcast.mp3" title="Web Design Podcast">Web Design Podcast</a>
```



Multimedia & Accessibility

- Provide alternate content
 - Transcript (for audio)
 - Captions (for video)
 - Text format

HTML5 Audio & Source Elements



```
<audio controls="controls">
```

```
  <source src="soundloop.mp3" type="audio/mpeg">
```

```
  <source src="soundloop.ogg" type="audio/ogg">
```

```
  <a href="soundloop.mp3">Download the Audio File</a> (MP3)
```

```
</audio>
```

HTML5 Video & Source Elements



```
<video controls="controls" poster="sparky.jpg"
  width="160" height="150">
  <source src="sparky.m4v" type="video/mp4">
  <source src="sparky.ogv" type="video/ogg">
  <a href="sparky.mov">Sparky the Dog</a> (.mov)
</video>
```

Use the iframe element to embed YouTube Video

```
<iframe src="http://www.youtube.com/embed/VIDEO_ID" width="640" height="385">
```

View the

```
<a href="http://www.youtube.com/watch?v=VIDEO_ID">YouTube Video</a>
</iframe>
```



CSS3 Transform Property

- Allows you to rotate, scale, skew, or move an element
- Example:
 - `rotate(3deg);`
 - `scale (x,y)`
 - `scaleX(x)`
 - `scaleY(y)`
 - `translate (x,y)`
 - `translateX(x)`
 - `translateY(y)`



CSS3 Transition Property

- Provides for changes in **property values** to display in a smoother manner over a **specified time**.

- Example:

```
nav a:hover { color: #869dc7; background-color: #eaeaea;  
transition: background-color 2s linear;}
```

Transition {Transition property;

Transition Duration ;

Transition-Timing-Function}



HTML5 Details and Summary Element

- The **details element** and **summary element** are used together to configure an **interactive widget** that **hide** and **show** content.
- **Details Element** : To **configure** the browser to render an **interactive widget** that contains **one summary element** and **detailed information**.

`<details></details>`

- **Summary Element:** is coded within the details element. The purpose of the summary element is to **contain** the **summary text shown in the interactive widget**.

`<summary></summary>`

Principles of Visual Design

- ▶ Repetition
- ▶ Contrast
- ▶ Proximity
- ▶ Alignment

Principles of Visual Design

▼ Repetition

Repeat visual components throughout the design

- ▶ Contrast
- ▶ Proximity
- ▶ Alignment

Summary

- This chapter introduced the HTML techniques and technologies used to configure **sound**, **video**, and **interactivity** on web pages.
- Issues related to accessibility and copyright were also discussed.
- The number one reason for visitors to leave web pages is too long of a download time. When using multimedia, be careful to minimize this issue.