

Web Development & Design Foundations with HTML5 8th Edition

CHAPTER 9 KEY CONCEPTS

Learning Outcomes

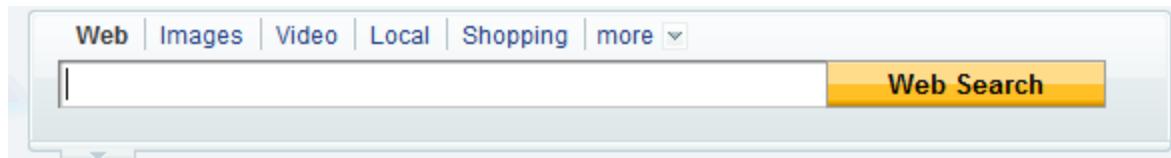
In this chapter, you will learn how to ...

- Describe common uses of forms on web pages
- Create forms on web pages using the form, input, textarea, and select elements
- Create forms that provide additional accessibility features using the accesskey and tabindex attributes
- Associate form controls and groups using the label, fieldset, and legend elements
- Create custom image buttons and use the button element
- Use CSS to style a form
- Configure new HTML5 form controls including the email, URL, datalist, range, spinner, calendar, and color controls
- Describe the features and common uses of server-side processing
- Invoke server-side processing to handle form data
- Find free server-side processing resources on the Web

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...

Overview of Forms (2)

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.

Shipping Address Entry

Name:	<input type="text"/>
Company:	<input type="text"/>
Address Line 1:	<input type="text"/>
Address Line 2:	<input type="text"/>
City:	<input type="text"/>
State:	<input type="text"/>
Zip Code:	<input type="text"/> - <input type="text"/>
Country:	<input type="text"/> United States <input type="button" value="▼"/>
<input type="button" value="Continue"/> <input type="button" value="Clear All"/>	

Two Components of Using Forms

1. The HTML form
the web page user interface

and

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.

HTML Using Forms

<form>

- Contains the form elements on a web page
- Container tag

<input>

- Configures a variety of form elements including text boxes, radio buttons, check boxes, and buttons
- Stand alone tag

<textarea>

- Configures a scrolling text box
- Container tag

<select>

- Configures a select box (drop down list)
- Container tag

<option>

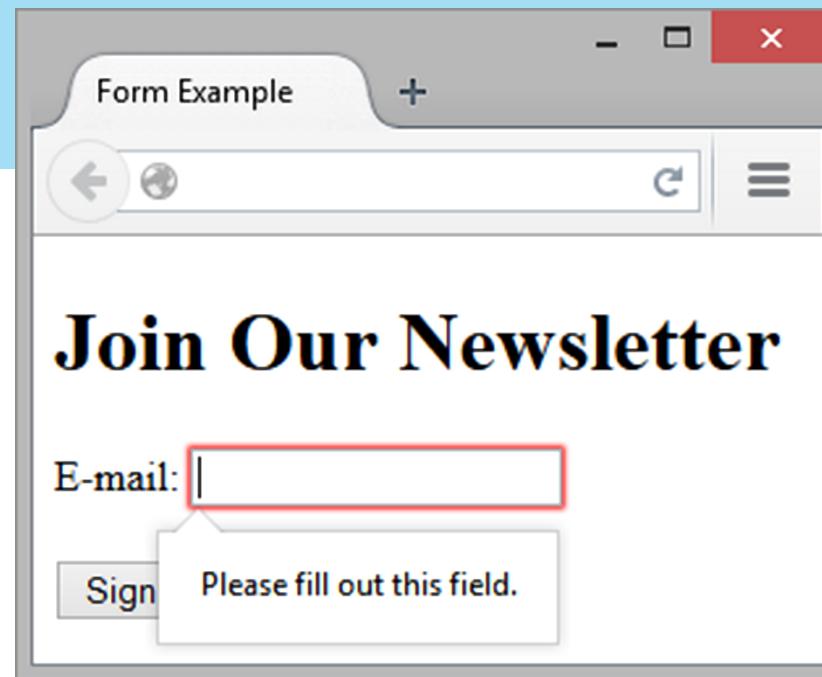
- Configures an option in the select box
- Container tag

Sample Form HTML

```
<form>
```

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

```
</form>
```



HTML form element

- The form element attributes:

- action
 - Specifies the server-side program or script that will process your form data
- method
 - get – default value,
form data passed in URL
 - post – more secure,
form data passed in HTTP Entity Body
- name
 - Identifies the form
- id
 - Identifies the form

Input Text box

<input>

Accepts text information

- Attributes:
 - type="text"
 - name
 - id
 - size
 - maxlength
 - value

Sample Text Box

Email:

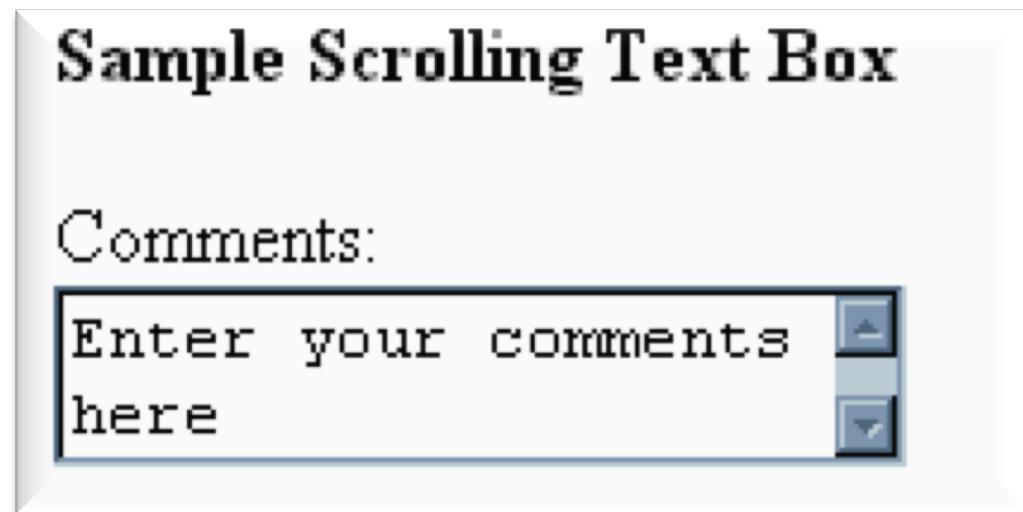
textarea Scrolling Text Box

<textarea> </textarea>

Configures a scrolling text box

Attributes:

- name
- id
- cols
- rows



input 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

Submit Query

input Reset Button

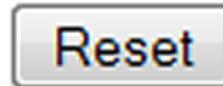
<input>

Resets the form fields to their initial values

Attributes:

- type="reset"
- name
- id
- value

Sample Reset Button



Hands-On Practice

Contact Form

Contact Us

First Name:

Last Name:

E-mail:

Comments:

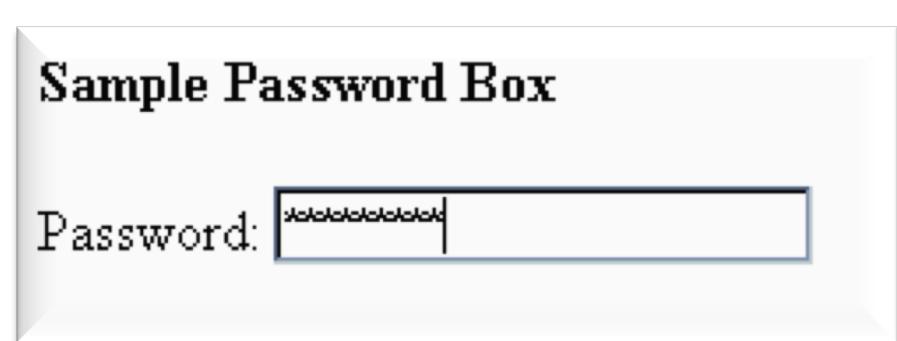
Contact Reset

input Password box

<input>

Accepts text information that needs to be hidden as it is entered

- Attributes:
 - type="password"
 - name
 - id
 - size
 - maxlength
 - value



input Check box

<input>

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

- Attributes:
 - type="checkbox"
 - name
 - id
 - checked
 - value

Sample Check Box

Choose the browsers you use:

- Internet Explorer
- Firefox
- Opera

input 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

- Attributes:
 - type="radio"
 - name
 - id
 - checked
 - value

Sample Radio Buttons

Select your favorite browser:

- Internet Explorer
- Firefox
- Opera

input Hidden form data

<input>

This form control is *not* displayed on the web page.

Hidden form fields

- Can be accessed by both client-side and server-side scripting
- Sometimes used to contain information needed as the visitor moves from page to page.

Attributes:

- type="hidden"
- name
- id
- value

Select List

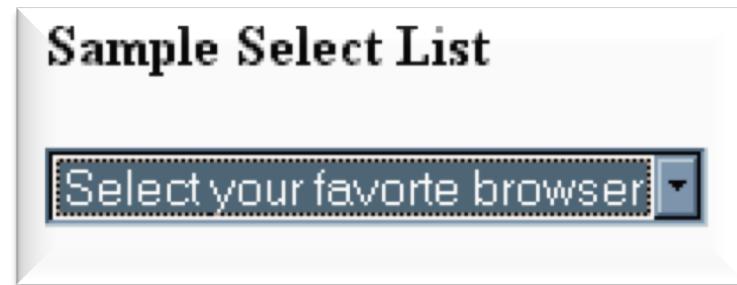
<select></select>

Configures a select list (along with option elements)

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.

- Attributes:
 - name
 - id
 - size
 - multiple



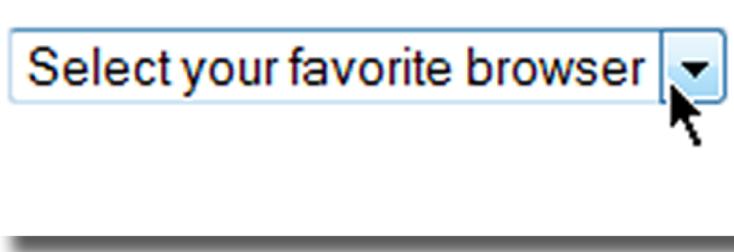
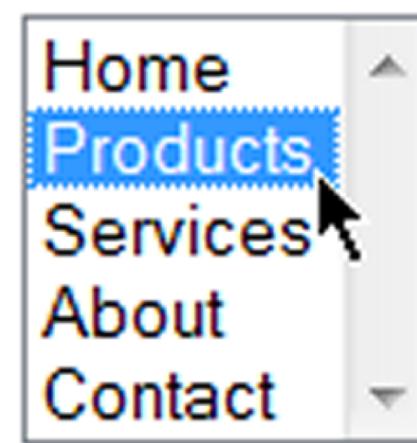
Options in a Select List

<option></option>

Configures the options in a Select List

Attributes:

- value
- selected



Checkpoint

1. You are designing a web site for a client who sells items in a retail store.

They want to create a customer list for e-mail marketing purposes. Your client sells to consumers and needs a form that accepts their customer's name and e-mail address.

Would you recommend using two input boxes
(one for the name and one for the e-mail)
or three input boxes
(for the first name, last name, and email address)?
Explain your answer.

Checkpoint

2. You are designing a survey form for a client. One of the questions has 10 possible responses.
Only one response can be selected per question.

What type of form control would you use to configure this question on the web page?

3. True or False.
In a radio button group, the value attribute is used by the browser to process the radio buttons as a group.

Input Image Button

<input>

Submits the form

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="image"
- name
- id
- src

The image shows a user interface for a login page. It consists of a white rectangular box containing two horizontal text input fields. The first field is labeled "Name:" and the second is labeled "Password:". Below these fields is a large, rounded rectangular button with a green gradient background and the words "Log In" in white. The entire form is set against a dark grey background.

Button Element

<button></button>

A container tag

When clicked, its function depends on the value of the type attribute.

Can contain a combination of text, images, and media

- Attributes:

- type="submit", type="reset", type="button"
- name
- id
- alt
- value

The image shows a user interface for a newsletter sign-up. It features two input fields: one for 'Name' and one for 'E-mail'. Below these fields is a large, prominent green button with the text 'Sign Up' in white. Underneath the button, there is a smaller text link that says 'Sign up for free newsletter'.

Accessibility & Forms

Label Element

Fieldset Element

Legend Element

Tabindex Attribute

Accesskey Attribute

Label element

<label></label>

Associates a text label with a form control

Two Different Formats:

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

Or

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

Fieldset and Legend Elements

The Fieldset Element

- Container tag
- Creates a visual group of form elements on a web page

The Legend Element

- Container tag
- Creates a text label within the fieldset

The screenshot shows a window titled "Contact Form" with a tab labeled "Contact Us". Inside, there's a section titled "Customer Information" containing three text input fields for "First Name", "Last Name", and "E-mail". Below this is a larger text area for "Comments". At the bottom are two buttons: "Contact" and "Reset".

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

tabindex attribute

Attribute that can be used on form controls and anchor tags

Modifies the default tab order

Assign a numeric value

```
<input type="text" name="CustEmail" id="CustEmail"  
tabindex="1">
```

accesskey attribute

Attribute that can be used on form controls and anchor tags

Create a “hot-key” combination to place the focus on the component

Assign a value of a keyboard letter

On Windows use the CTRL and the “hot-key” to move the cursor

```
<input type="text" name="CustEmail"  
id="CustEmail" accesskey="E">
```

Checkpoint

1. Describe the purpose of the fieldset and legend elements.
2. Describe the purpose of the accesskey attribute and how it supports accessibility.
3. When designing a form, should you use the standard submit button, an image button, or a button tag? Are these different in the way that they provide for accessibility? Explain your answer.

Using CSS to Style a Form

form

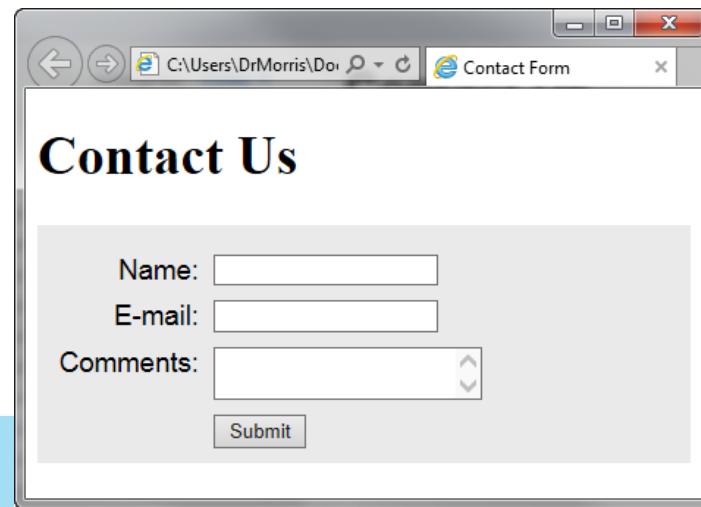
label text box

label text box

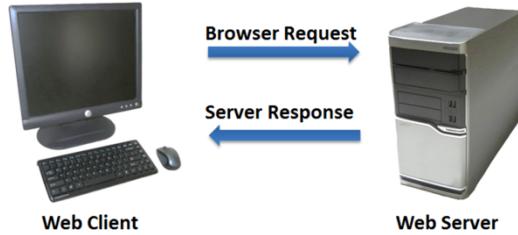
label scrolling text box

submit button

Student Files: formcss.html



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

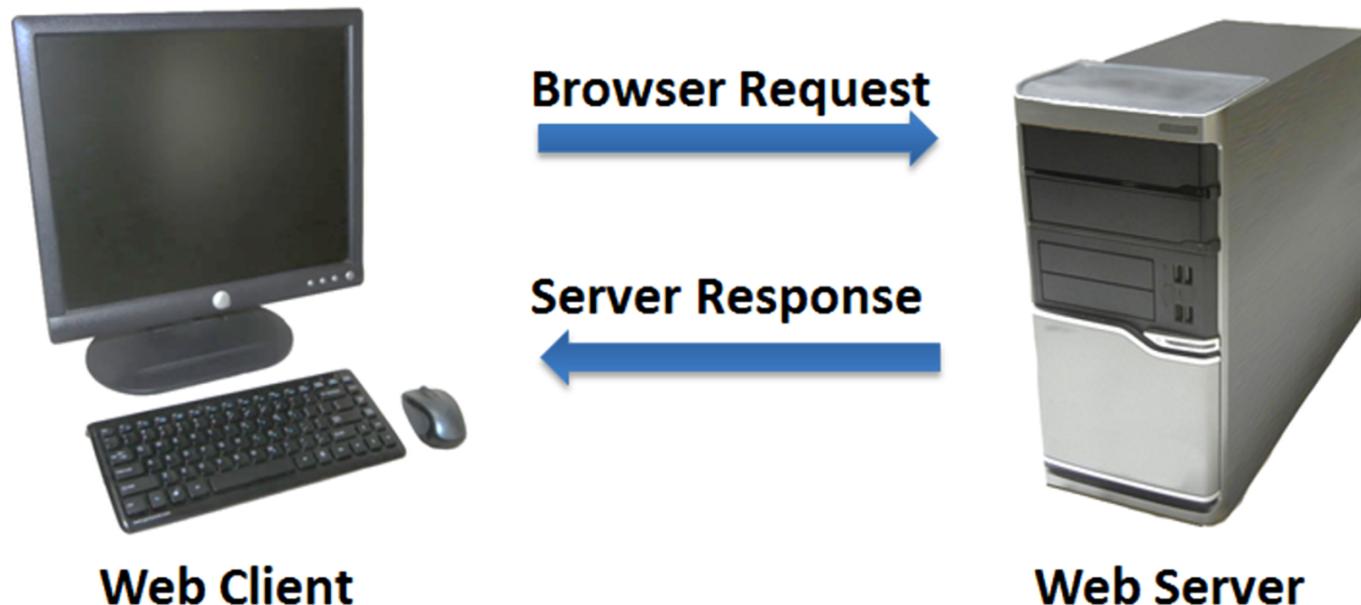


Server-Side Processing

- Your web browser requests web pages and their related files from a web server.
- The web server locates the files and sends them to your web browser.
- The web browser then renders the returned files and displays the requested web pages for you to use.

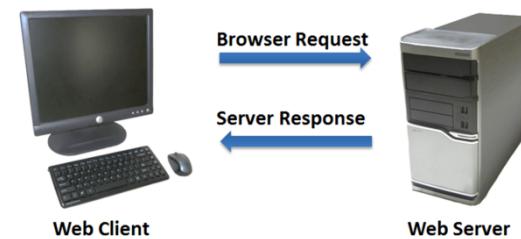
CGI Common Gateway Interface

A protocol for a web server to pass a web page user's request to an application program and accept information to send to the user.



Server-Side Scripting

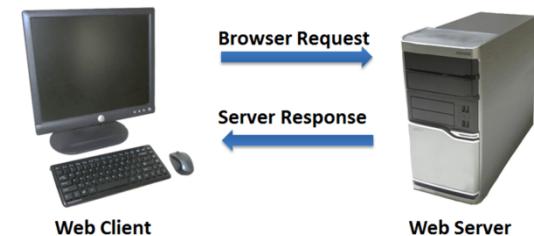
- One of many technologies in which a server-side script is embedded within a Web page document saved with a file extension such as:
 - .php (PHP)
 - .asp (Active Server Pages)
 - .cfm (Adobe ColdFusion)
 - .jsp (Sun JavaServer Pages)
 - .aspx (ASP.Net).



- Uses direct execution — the script is run either by the web server itself or by an extension module to the web server.

Steps in Utilizing Server-Side Processing

1. Web page invokes server-side processing by a form or hyperlink.
2. Web server executes a server-side script.
3. Server-side script accesses requested database, file, or process.
4. Web server returns web page with requested information or confirmation of action.



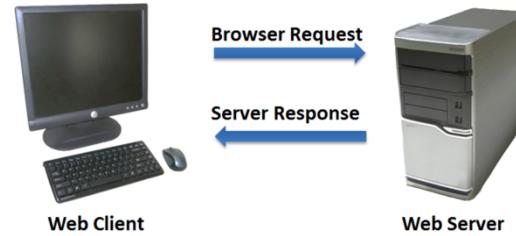
Common Uses of Server-Side Scripting

Search a database

Place an order at an online store

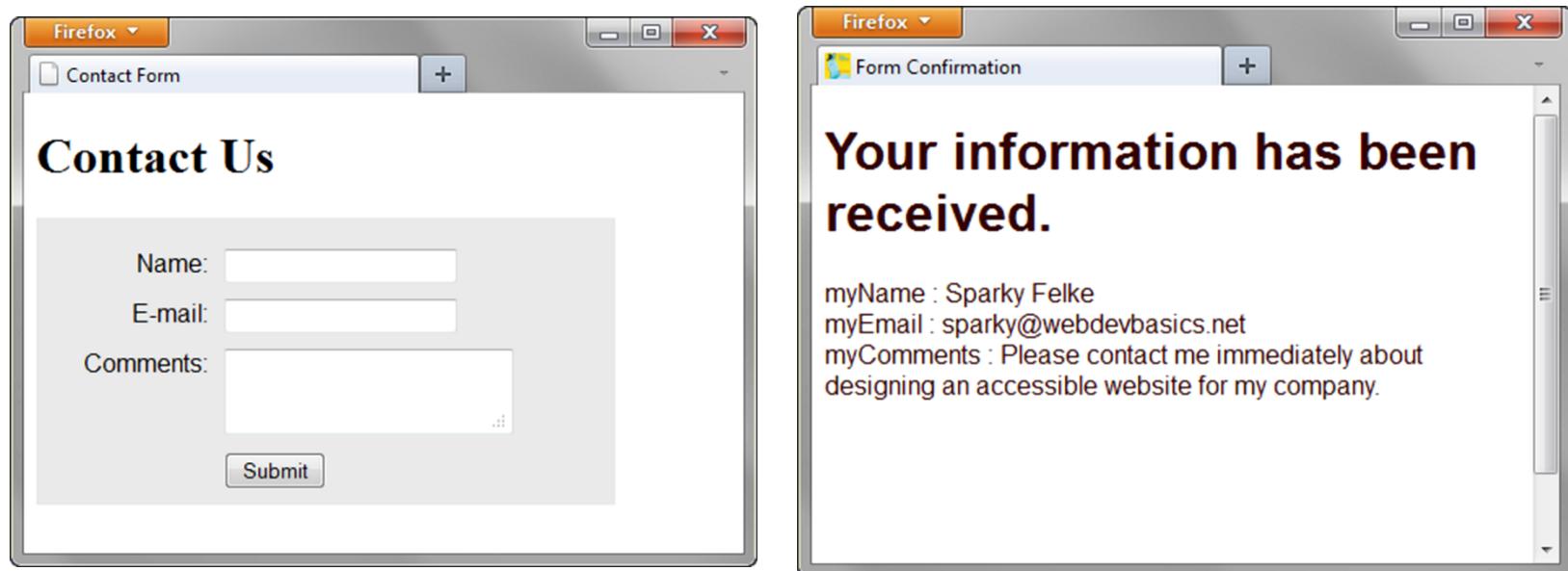
Send a web page to a friend

Subscribe to a newsletter



Sending information to a Server-side Script

```
<form method="post"  
action="http://webdevbasics.net/scripts/demo.php ">
```



Sources of Free Server-Side Processing

Many web host providers offer free scripts for their clients. Contact their support area or FAQ to learn more about their services.

Some web sites that offer FREE remotely hosted scripts (in return for displaying an ad).

- <http://formbuddy.com>
- <http://response-o-matic.com>
- <http://master.com>
- <http://www.formmail.com>
- <http://wufoo.com>
- <http://formassembly.com>

Server-Side Scripting Technologies

- JavaServer Pages

<http://www.oracle.com/technetwork/java/javaee/jsp>

- ColdFusion

<http://www.adobe.com/products/coldfusion>

- PHP

<http://www.php.net>

- Ruby on Rails

<http://www.rubyonrails.org>

- Microsoft's .NET Framework

<http://www.microsoft.com/net>

- Microsoft Active Server Pages

<http://msdn.microsoft.com/en-us/library/ms972337.aspx>

Checkpoint

1. Describe server-side processing.
2. Describe why communication is needed between the developer of a server-side script and the web page designer.

HTML5: Email Text Box

<input>

Accepts text information in e-mail address format

- Common Attributes:
 - type="email"
 - name
 - id
 - size
 - maxlength
 - value
 - placeholder
 - required

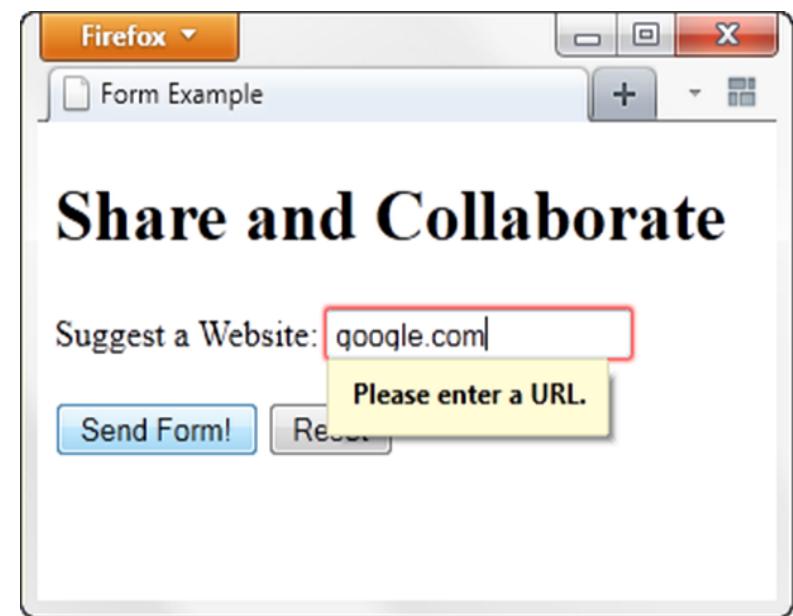


HTML5: URL Text Box

<input>

Accepts text information in URL format

- Common Attributes:
 - type="url"
 - name
 - id
 - size
 - maxlength
 - value
 - placeholder
 - required



HTML5: Telephone Number Text Box

<input>

Accepts text information in telephone number format

- Common Attributes:
 - type="tel"
 - name
 - id
 - size
 - maxlength
 - value
 - placeholder
 - required

HTML5: Search Text Box

<input>

Accepts search terms

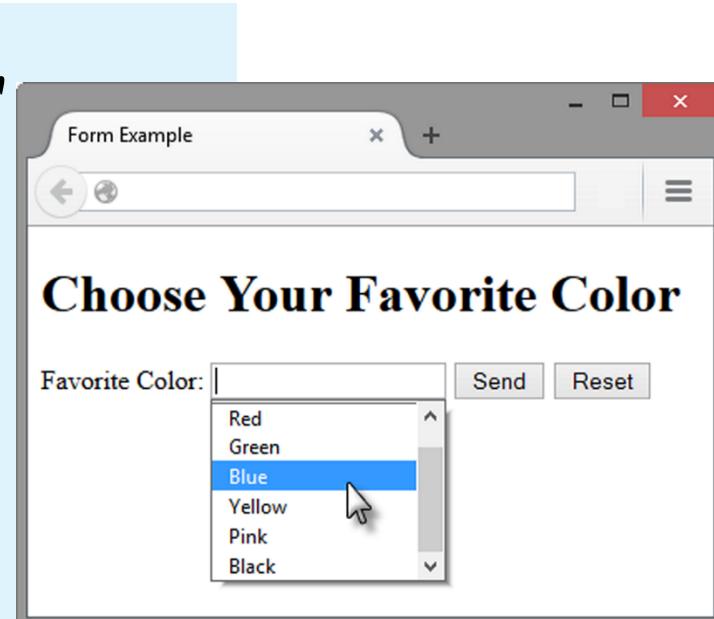
- Common Attributes:

- type="search"
- name
- id
- size
- maxlength
- value
- placeholder
- required

HTML5: Datalist Control

```
<label for="color">Favorite Color:</label>
<input type="text" name="color" id="color"
list="colors" >

<datalist id="colors">
  <option value="red" label="Red">
  <option value="green" label="Green">
  <option value="blue" label="Blue">
  <option value="yellow" label="Yellow">
  <option value="pink" label="Pink">
  <option value="black" label="Black">
</datalist>
```

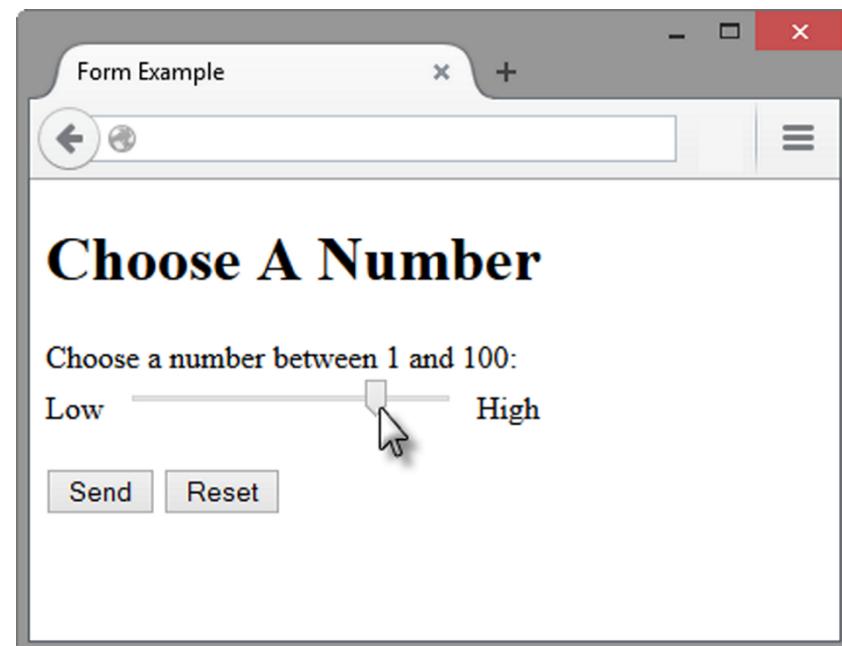


HTML5: Slider Control

```
<label for="myChoice">
```

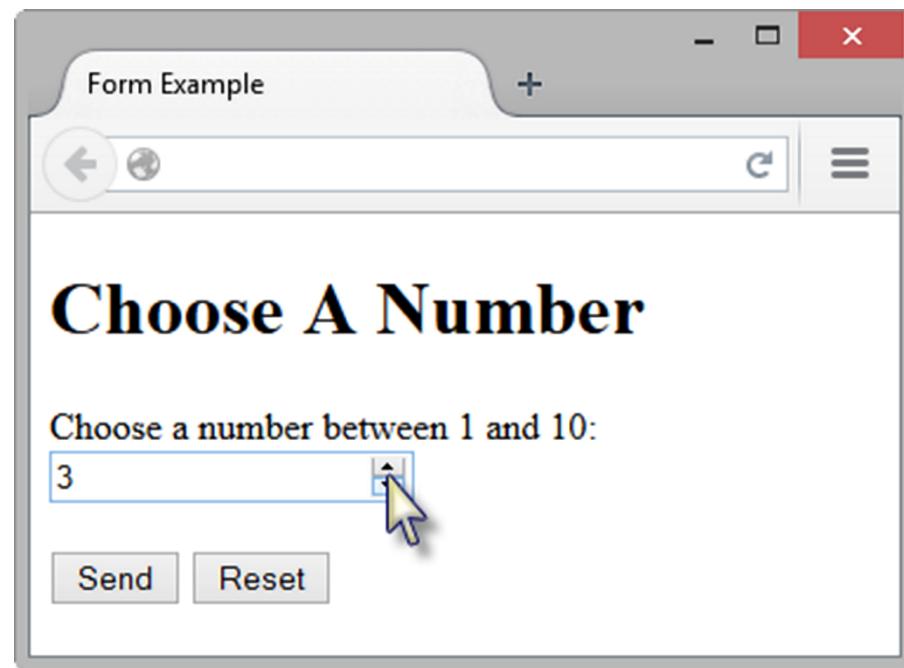
Choose a number between 1 and 100:</label>

Low <input type="range"
name="myChoice"
id="myChoice"
min="1"
max="100"> High



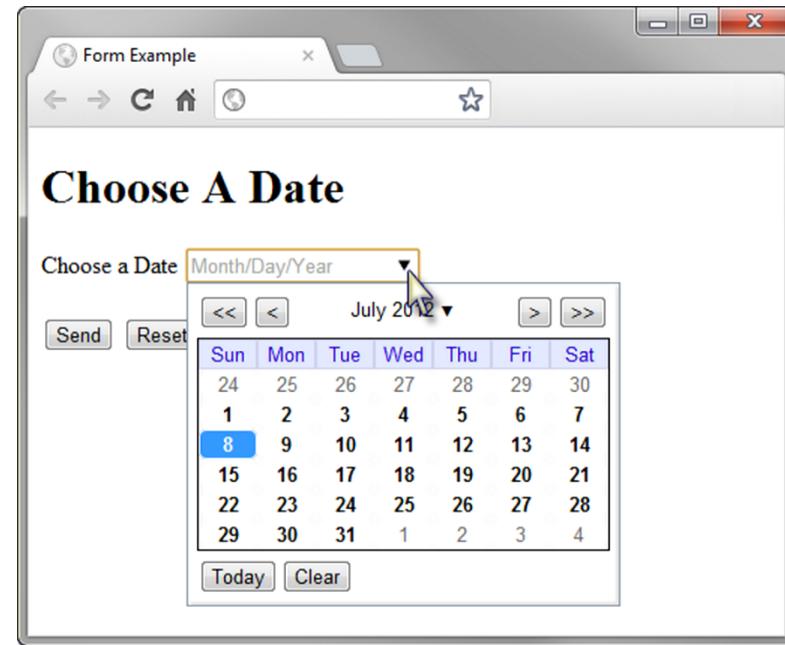
HTML5: Spinner Control

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



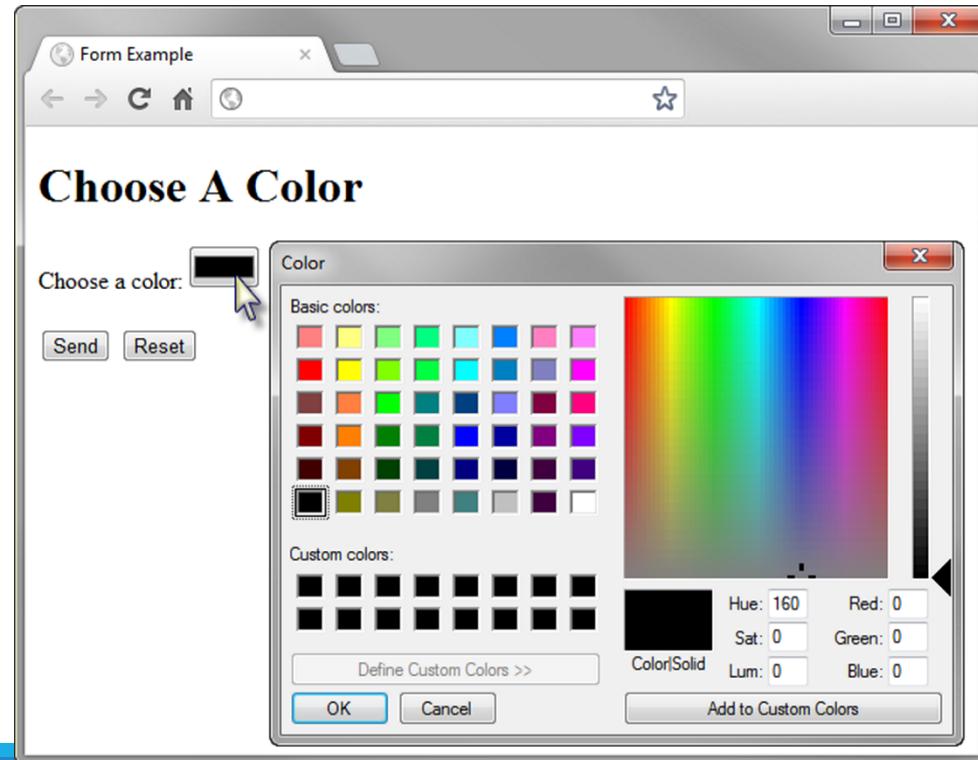
HTML5: Calendar Control

```
<label for="myDate">Choose a Date</label>
<input type="date" name="myDate" id="myDate">
```



HTML5: Color Well Control

```
<label for="myColor">Choose a color:</label>
<input type="color" name="myColor" id="myColor">
```



Practice with an HTML5 Form

The form display and functioning varies with browser support.

Comment Form

Send Us Your Comments

Required fields marked with an asterisk *

*Name:

*E-mail:

Rating (1 - 10):

*Comments:

Submit

Comment Form

Send Us Your Comments

Required fields marked with an asterisk *

*Name:

*E-mail:

Rating (1 - 10):

*Comments:

Submit

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 how to configure a form to access server-side processing.

In addition, you learned about new HTML5 form controls.