Essentials of Web Design

CSCI 1210

Print to PDF

Forms

Forms

Forms

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
- Registering for an online site/service

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

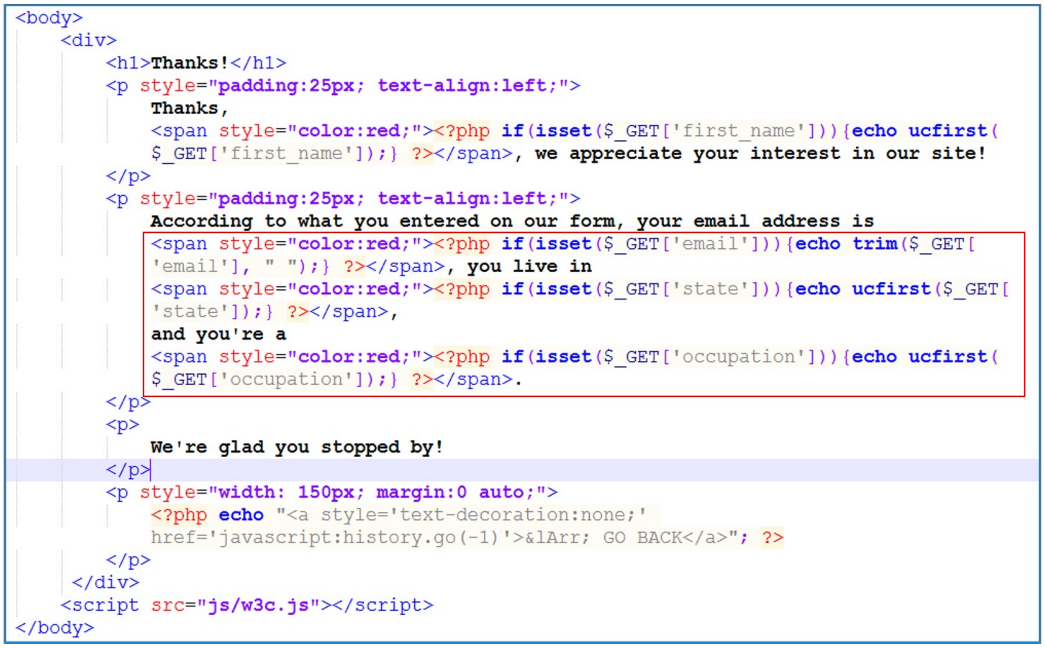
Different types of form fields exist to permit collecting different types of information

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

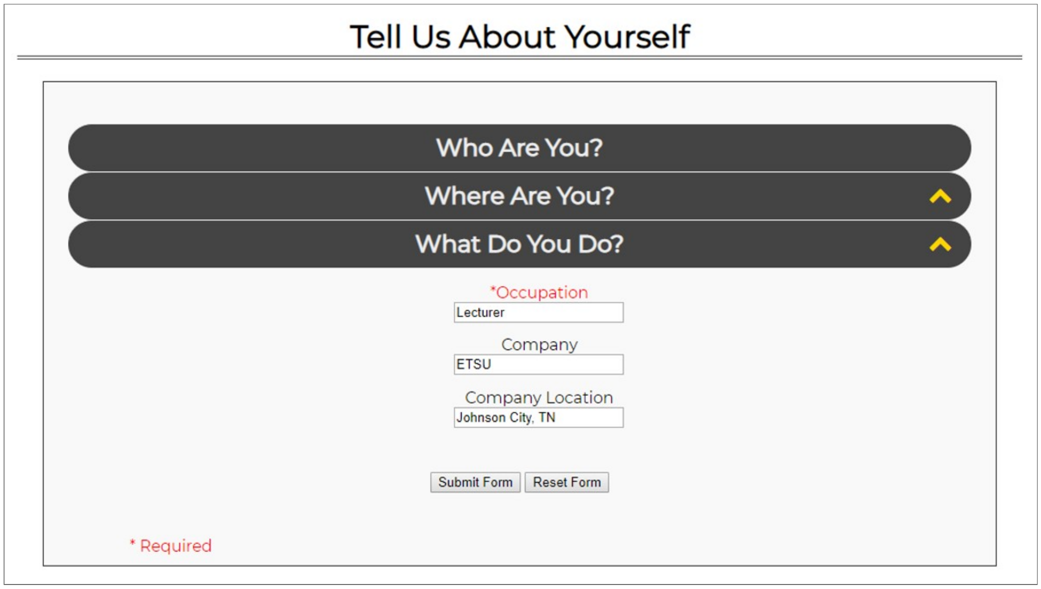
The program processes the information and replies in a fashion dictated by its programming

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>
```

Sending Data from Client to Server



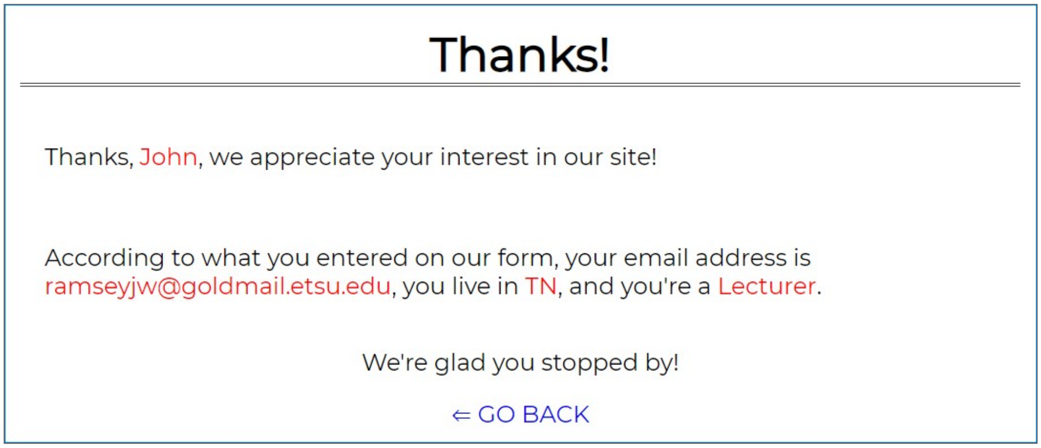
*Occupation: Lecturer

Company: ETSU

Company Location: Johnson City, TN

* Required

Sending Data from Client to Server



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!

Creating a Form

Form Element

```
<form>
    <!-- form elements -->
</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

`method` - specifies the method to be used for transferring the information

```
<form action="php/myprog.php"
method="post">
    <!-- form elements -->
</form>
```

Method

`GET`

For the attribute method, we have two possibilities:

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

The information is contained in the page header (meta-data that is sent to the server along with the request)

Still not secure, but information is hidden from the casual observer, or 'shoulder-surfer'

GET vs POST

When should we use `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 `POST`?

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

Basic Form Example (Search Google)

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

Form Fields

Number field
URL field
File selector
Data lists
Checkboxes
Radio button
Submit button
Reset button
Multi-line textbox
Select list

Input Element

The `<input>` element is used ot create (most of) the fields in a form

The `<input>` element is an inline, standalone element

Two required attributes:

name - Needed to reference the field when the form has been submitted

type - Identifies which kind of input field is being created (textbox, password, checkbox, radio button, etc)

Input Element

```
<form method="get"
action="https://www.google.com/search.php">
    <input type="text" name="q">
    <input type="submit" name="“submit”">
</form>
```

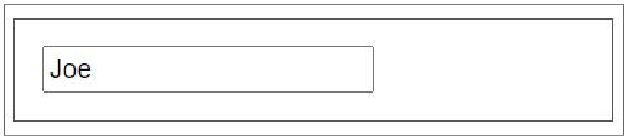
Single Line Textbox

```
            <input type="text" name="firstName">
```

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

We must add the descriptive text to instruct users:

```
First Name: <input type="text" name="firstName">
```



Joe

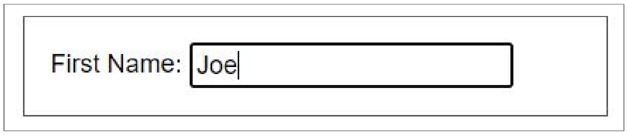
Single Line Textbox

Attributes:

value - assigns a 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 field as a prompt. Once the user begins typing in the field, the placeholder disappears

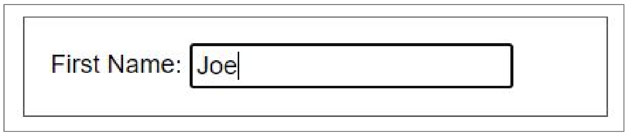

First Name: Joe

Textbox - Other Attributes

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

maxlength – specifies the maximum number of characters

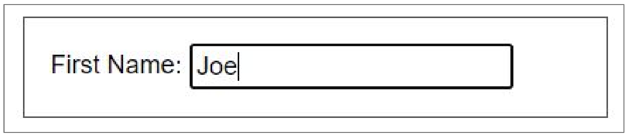
disabled – specifies if the input element is visible, but the end user is not allowed to fill out the box


First Name: Joe

Textbox - Other Attributes

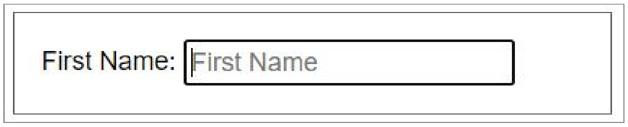
autofocus – specifies the element on the page that should receive focus when the form loads

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



First Name: Joe

Textbox - Example



First Name: First Name

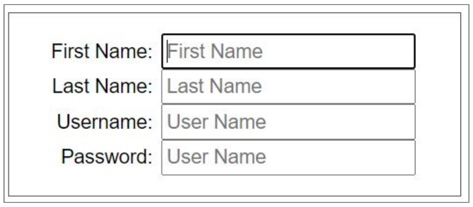
```
<input type="text" name="firstName"
placeholder="First Name" size="20" maxlength="30"
autofocus="" required="">
```

Password

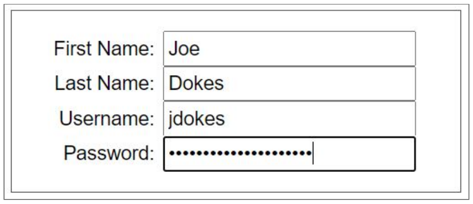
```
<input type="password" name="password">
```

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



First Name: First Name
Last Name: Last Name
Username: User Name
Password: User Name



First Name: Joe
Last Name: Dokes
Username: jdokes
Password: •••••••••••••••••••••

Password Example

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

(I don't know why my application insists on adding the `=""` to the `required` attribute. It isn't necessary)

Checkboxes

```
<input type="checkbox" name="sauce" value="yes">
```

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

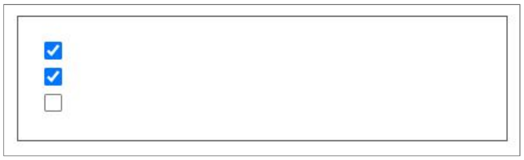
With a checkbox, the name/value pair is only submitted if the end user selects the checkbox

Attribute:

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

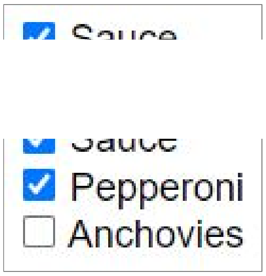
Checkbox Examples

```
<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>
```



☑
☑
☐

Checkbox Example

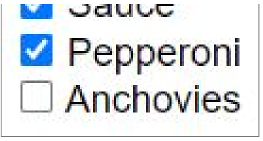


☑ Sauce
☑ Pepperoni
☐ Anchovies

```
<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

Checkboxes - Labels



☑ Sauce
☑ Pepperoni
□ Anchovies

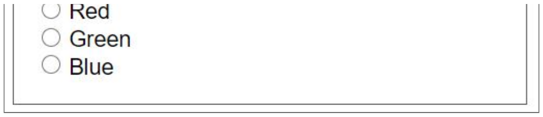
```
<input type="checkbox" name="sauce" id="sa" value="yes"
checked="">
<label for="sa">Sauce</label><br>
<input type="checkbox" name="pepperoni" id="pep" value="yes"
checked="">
<label for="pep">Pepperoni</label><br>
<input type="checkbox" name="anchovies" id="anc"
value="yes">
<label for="anc">Anchovies</label><br>
```

Checkboxes - Labels

The value for `for=""` in the `<label>` element must match the value for `id=""` in the `<input>` element

Now, the checkbox will be selected when either the box itself or its label is clicked

Radio Buttons



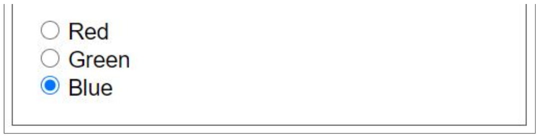
○ Red
○ Green
○ Blue

```
<input type="radio" name="favoriteColor"
value="red"> Red<br>
<input type="radio" name="favoriteColor"
value="green"> Green<br>
<input type="radio" name="favoriteColor"
value="blue"> Blue<br>
```

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



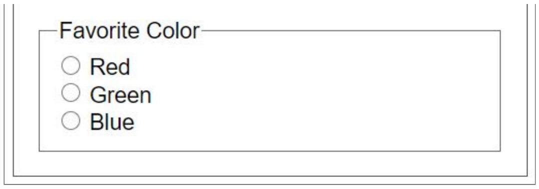
○ Red
○ Green
● Blue

```
<input type="radio" name="favoriteColor" value="red"
id="r">
<label for="r">Red</label><br>

<input type="radio" name="favoriteColor" value="green"
id="g">
<label for="g">Green</label><br>

<input type="radio" name="favoriteColor" value="blue"
id="b">
<label for="b">Blue</label><br>
```

Fieldset



```
<fieldset>
    <legend>Favorite Color</legend>
    <input type="radio" name="favoriteColor" value="red"
id="r">
    <label for="r">Red</label><br>
    <input type="radio" name="favoriteColor"
value="green" id="g">
    <label for="g">Green</label><br>
    <input type="radio" name="favoriteColor" value="blue"
id="b">
    <label for="b">Blue</label><br>
</fieldset>
```

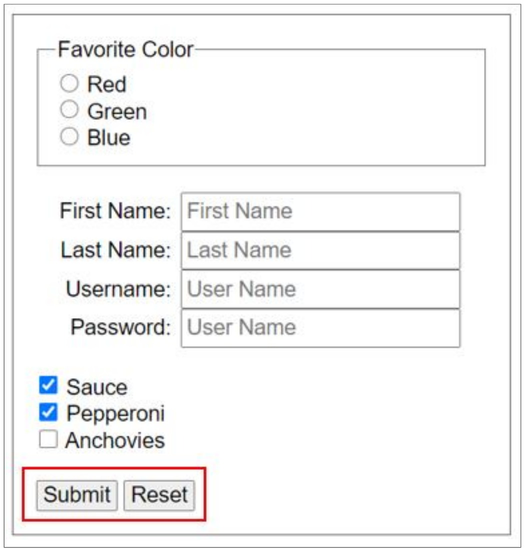
Submit & Reset

```
<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 form to the default values assigned for each field

Submit & Reset



Favorite Color

○ Red
○ Green
○ Blue

First Name: First Name
Last Name: Last Name
Username: User Name
Password: User Name

☑ Sauce
☑ Pepperoni
□ Anchovies

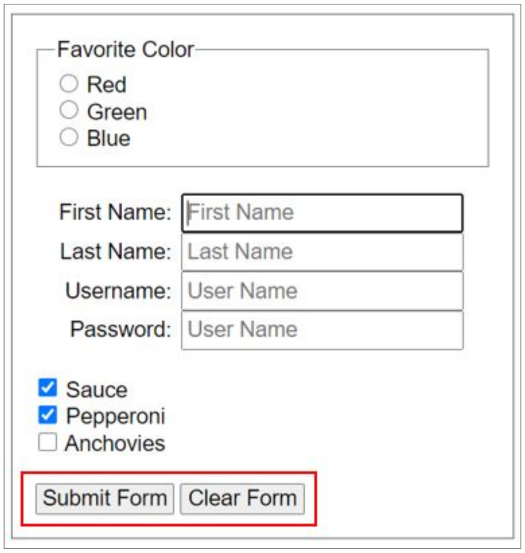
Submit & Reset

```
    <input type="submit" value="Submit
Form">
    <input type="reset" value="Clear
Form">
```

Submit & Reset have one optional attribute: `value`

This will modify the display text in the button(s)

Submit & Reset



Favorite Color

○ Red
○ Green
○ Blue

First Name: First Name
Last Name: Last Name
Username: User Name
Password: User Name

☑ Sauce
☑ Pepperoni
□ Anchovies

Textarea

```
<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>`, use the `value` attribute

Textarea attributes

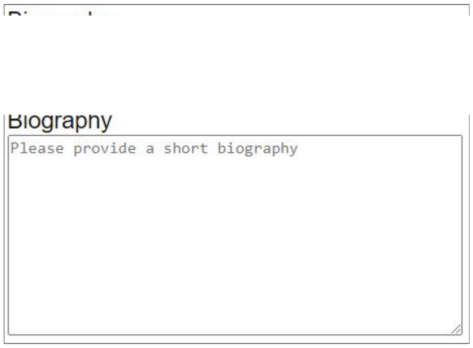
`cols` - width of the element

`maxlength` - maximum number of characters

`rows` - number of rows displayed

`readonly` - user cannot change the contents

Textarea - Example



```
Biography<br>
<textarea cols="50" name="bio" maxlength="600"
rows="10" id="ta" placeholder="Please provide a
short biography" value="none"></textarea>
```

Textarea - How it Works

Here's a little JavaScript that shows how a `<textarea>` works

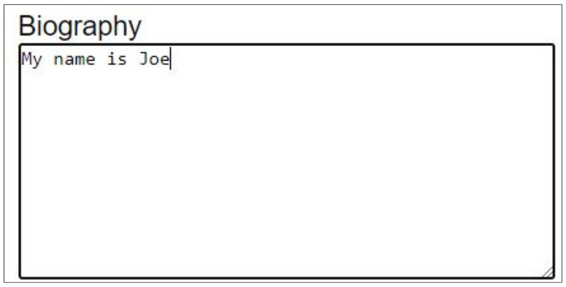
```
<script>
    var ta =
document.getElementById("ta");
    ta.addEventListener("input",
function() {
        console.log(ta.value);
    });
</script>
```

Textarea - How it Works

What this does is tell the browser to output the value of the text area each time the user enters something

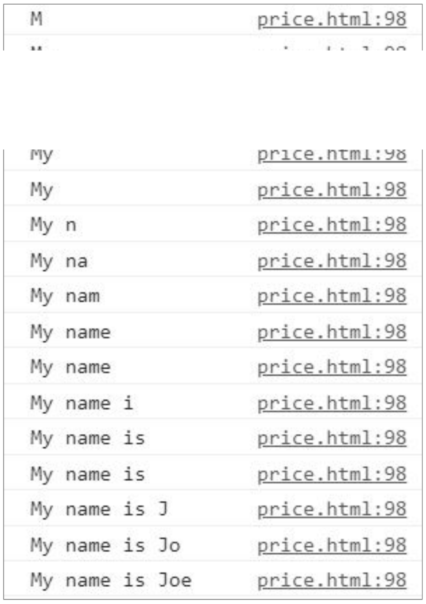
```
<script>
    var ta =
document.getElementById("ta");
    ta.addEventListener("input",
function() {
        console.log(ta.value);
    });
</script>
```

Textarea - How it Works



```
My name is Joe
```

Console Output



```
M                    price.html:98
M                    price.html:98
My                   price.html:98
My                   price.html:98
My n                 price.html:98
My na                price.html:98
My nam               price.html:98
My name              price.html:98
My name              price.html:98
My name i            price.html:98
My name is           price.html:98
My name is           price.html:98
My name is J         price.html:98
My name is Jo        price.html:98
My name is Joe       price.html:98
```

Select List

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

```
<select name="select">

</select>
```

Select List

Each list option is created using the `<option>`…`</option>` element

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

Attributes for `<select>`

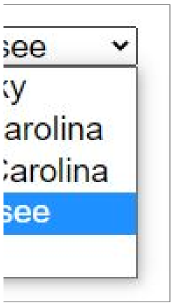
- `disabled` - The list is visible, but the user cannot select from it
- `size` - The number of visible options
- `multiple` - User can select more than one option

Attribute for the `<option> element`

- `selected` - Specified element is selected when the page loads (user can change)

Select List - Example

```
<select name="select">
    <option value="KY">Kentucky</option>
    <option value="NC">North
Carolina</option>
    <option value="SC">South
Carolina</option>
    <option value="TN"
selected="">Tennessee</option>
    <option value="VA">Virginia</option>
</select>
```



Questions?

Reading for this week

Chapter 7 - Forms

Lecture Quiz

1. What attribute do we use to tell the browser how a form should be submitted?

   A. `action`

   B. `post`

   C. `method`

   D. `get`

Lecture Quiz

2. (T/F) `<input>` elements are block elements

   A. True

   B. False

Lecture Quiz

3. When a form is submitted, what happens?

   A. It is processed by a server-side script

   B. The browser processes it and displays the result

   C. It is converted to an executable file and executed by the server

   D. It is converted to an executable file and executed by the browser

Lecture Quiz

4. What form control offers two or more options, but only allows the user to select one?

A. checkbox

B. text

C. textarea

D. radio button

Lecture Quiz

5. If we want a checkbox to be selected by default when the page loads, what binary attribute do we use?

A. selected

B. checked

C. default

D. picked

Lecture Quiz

6. What does the `reset` control do?

A. Returns all of the form fields to their default values

B. Reloads the page, which clears all of the form values

C. Clears the value from the last selected field

D. Clears only the fields that have a default value specified

Lecture Quiz

7. Which of the following is a server-side scripting language?

A. JavaScript

B. jQuery

C. PHP

D. C++

Lecture Quiz

8. (T/F) We can modify the display of form elements using CSS

   A. True

   B. False

Lecture Quiz

9. How do we create a multi-line text input field?

   A. `<input class="text">`

   B. `<textarea></textarea>`

   C. `<input class="textarea">`

   D. `input class="multiline">`

Lecture Quiz

10. Which form method should we use to enable bookmarking?

A. GET

B. POST

C. FETCH

D. RETRIEVE