

# Web Design with HTML5 & CSS3

CHAPTER 8

CREATING FORMS

# Form Controls

- ▶ An example of a form with several input controls

The image shows a 'Reservation Request' form with three main sections: 'General Information', 'Preferences', and 'Contact Information'. Each section contains different types of input controls, which are labeled with callouts:

- General Information:**
  - Arrival date:  (date controls)
  - Nights:  (date controls)
  - Adults aged:  (select controls)
  - Children:  (select controls)
  - Comments:  (textarea control)
- Preferences:**
  - Room type:  Standard  Business  Suite (radio controls)
  - Bed type:  King  Double Double (radio controls)
  - Smoking (checkbox control)
- Contact Information:**
  - Name:  (text control)
  - Email:  (email control)
  - Phone:  (tel control)
- Buttons:**
  - Submit Request (submit control)
  - Clear Form (reset control)

Additional labels on the left side of the form include 'labels' pointing to the text labels, and 'submit control' pointing to the 'Submit Request' button.

# How Forms Work

VOTE FOR YOUR FAVORITE JAZZ  
MUSICIAN OF ALL TIME

Username:

I vote for:

- Ella Fitzgerald
- Herbie Hancock
- John Coltrane
- Miles Davis
- Thelonius Monk

**1: User fills in form and presses button to submit info to server**

# How Forms Work

VOTE FOR YOUR FAVORITE JAZZ MUSICIAN OF ALL TIME

Username:

I vote for:

- Ella Fitzgerald
- Herbie Hancock
- John Coltrane
- Miles Davis
- Thelonius Monk

.....		.....
.....		.....
.....		.....
.....		.....
.....		.....
.....		.....
.....		.....
.....		.....
.....		.....
.....		.....
.....		.....
.....		.....
.....		.....
.....		.....
.....		.....

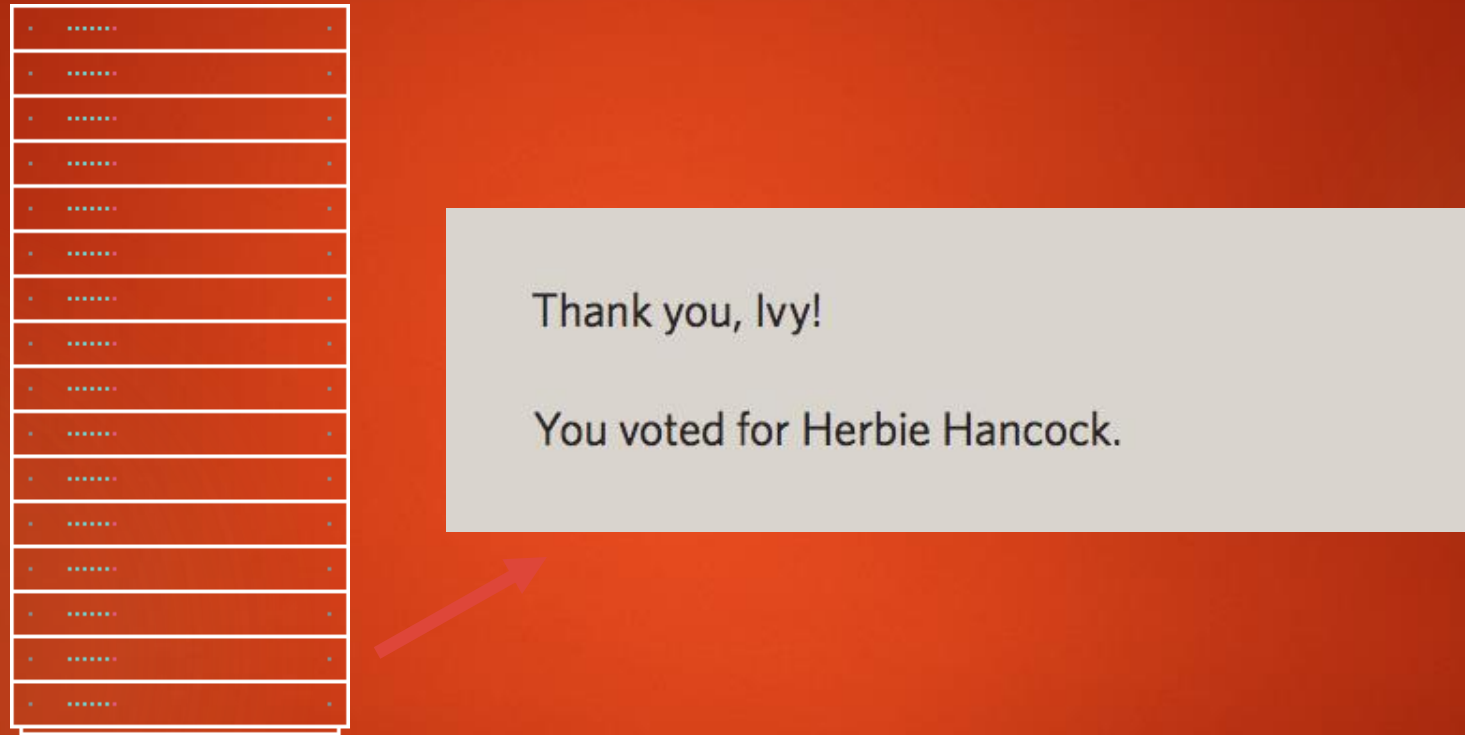
2: Name of each form control sent with value user entered

# How Forms Work



**3: Server processes information using programming language**

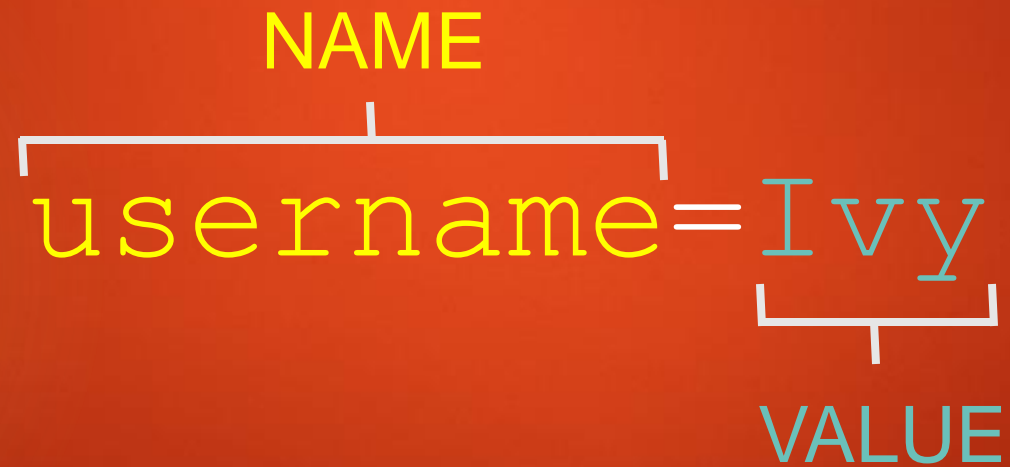
# How Forms Work



**4: Server send data (usually JSON) back to browser**

# Name & Value Pairs

- ▶ Data from a form is send to server in name/value pairs



# Form Processing

```
<form method="POST" action="newCustomer.aspx">
```

- ▶ **action** - Specifies the browser's action when submitting the form
- ▶ **method** - Specifies how to send the data entered in the form to the server to be processed
  - ▶ GET
  - ▶ POST



# Form Processing

- ▶ **GET** method - appends the name-value pairs to the URL indicated in the action attribute

- ▶ `<form method="GET" action="newCustomer.aspx">`

`https://www.mycompany.com/newcustomer.aspx?fname=Peter&lname=Smith`

- ▶ **POST** method - sends a separate data file with the name-value pairs to the URL indicated in the action attribute

- ▶ `<form method="POST" action="newcustomer.aspx">`

# Form Structure

10

```
<form  
  action="http://example.com/join.php"  
  method="get">
```

This is where the form controls  
will appear.

```
</form>
```

# Form Controls

- ▶ Each input control has attributes that are used more frequently than the others:
  - ▶ **name** – It identifies the specific information that is being sent when the form is submitted for processing. All controls have a name
  - ▶ **id** – It provides a unique ID for the element. Use the `id` attribute with input controls
  - ▶ **value** – It specifies the value of an `input` element and varies depending on input type

# <input> Element

- ▶ Most important form element
- ▶ Can be displayed in several ways, depending on the type attribute:

Type	Description
<code>&lt;input type="text"&gt;</code>	Defines a one-line text input field
<code>&lt;input type="radio"&gt;</code>	Defines a radio button (for selecting one of many choices)
<code>&lt;input type="submit"&gt;</code>	Defines a submit button (for submitting the form)

# <input type="text">

- ▶ Defines a one-line input field for text input
- ▶ If the **type** attribute is omitted, the input field gets the default type: "text"

# <input type="text">

14

```
<!DOCTYPE html>
<html>
<body>

<h2> Input Text </h2>
<form>
First name: <input type="text" name="firstname" id="fName" value="Mickey"><br>

Last name: <input type="text" name="lastname" id="lName" value="Mouse"><br><br>

<input type="submit" value="Submit">
</form>

</body>
</html>
```

## Input Text

First name:

Last name:

14

# <input type="text">

15

```
<!DOCTYPE html>
<html>
<body>

<h2> Input Text </h2>
<form>
First name:<br>
<input type="text" name="firstname" id="fName" value="Mickey"><br><br>

Last name:<br>
<input type="text" name="lastname" id="lName" value="Mouse"><br><br>

<input type="submit" value="Submit">
</form>

</body>
</html>
```

## Input Text

First name:

Last name:

15

# <input type="text"> - Label for

16

```
<!DOCTYPE html>
<html>
<body>

<h2> Input Text </h2>
<form>
<label for="fName"> First Name </label>
<input type="text" name="firstname" id="fName" value="Mickey"><br><br>

<label for="lName"> Last Name </label>
<input type="text" name="lastname" id="lName" value="Mouse"><br><br>

<input type="submit" value="Submit">
</form>

</body>
</html>
```

## Input Text

First Name

Last Name

16



# <input type="radio">

- ▶ Defines a radio button
- ▶ Select ONLY ONE from several choices

```
<!DOCTYPE html>
<html>
<body>

<h2> Radio Butons </h2>

<form>
  <input type="radio" name="gender" id="male" value="male"> Male<br>
  <input type="radio" name="gender" id="female" value="female" checked> Female<br>
  <input type="radio" name="gender" id="other" value="other"> Other<br><br>
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

## Radio Butons

- Male
- Female
- Other

# `<input type="radio">` - Label for

- ▶ Select ONE of a limited number of choices:

```
<!DOCTYPE html>
<html>
<body>

<h2> Radio Butons </h2>

<form>
  <label for="male">Male</label>
  <input type="radio" name="gender" id="male" value="male"><br>

  <label for="female">Female</label>
  <input type="radio" name="gender" id="female" value="female" checked><br>

  <label for="other">Other</label>
  <input type="radio" name="gender" id="other" value="other"><br><br>

  <input type="submit" value="Submit">
</form>

</body>
</html>
```

## Radio Butons

Male

Female

Other

# <input type="checkbox">

- ▶ Defines a checkbox
- ▶ Select ZERO or MORE options from several choices

```
<input type="checkbox" name="smoking" id="smoking">Smoking  
<input type="checkbox" name="nonsmoking" id="nonsmoking" class="left">Balcony  
<input type="checkbox" name="petfriendly" id="petfriendly" class="left">Pet-friendly<br>
```

Smoking  Balcony  Pet-friendly

# <select> Element

- ▶ Defines a drop-down list, which contains one or more **<option>** elements that define an option that can be selected

```
<!DOCTYPE html>
<html>
<body>

<label>Pick Car</label>
<select name="car" id="car">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="opel">Opel</option>
  <option value="audi">Audi</option>
</select>

</body>
</html>
```

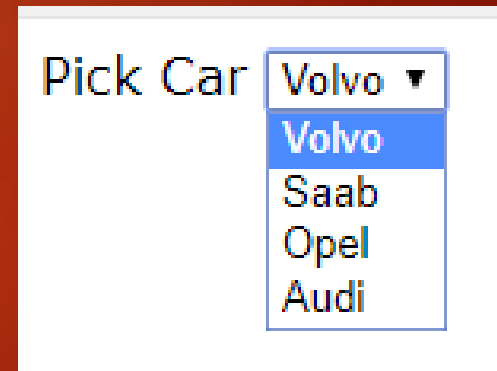
OR

```
<!DOCTYPE html>
<html>
<body>

<label> Pick a car</label>
<select name="car" id="car">
  <option value="1">Volvo</option>
  <option value="2">Saab</option>
  <option value="3">Opel</option>
  <option value="4">Audi</option>
</select>

</body>
</html>
```

Result:



# <select> Element

- ▶ By default, the first item in the drop-down list is selected.
- ▶ To define a pre-selected option, add the **selected** attribute to the option:

```
<!DOCTYPE html>
<html>
<body>

<label>Pick Car</label>
<select name="car" id="car">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="opel">Opel</option>
  <option value="audi" selected>Audi</option>
</select>

</body>
</html>
```

Pick Car

# <select> Element – size attribute

- ▶ Use the **size** attribute to specify the number of visible values:

```
<label>Adults:</label>
<select name="adults" size="3" id="adults">
  <option value="1">1</option>
  <option value="2">2</option>
  <option value="3">3</option>
  <option value="4">4</option>
</select><br>
<label>Children:</label>
<select name="children" size="4" id="children">
  <option value="0">0</option>
  <option value="1">1</option>
  <option value="2">2</option>
  <option value="3">3</option>
  <option value="4">4</option>
</select><br>
```

Adults:

Children:

# <select> Element – **multiple** attribute

- ▶ Use the **multiple** attribute to allow the user to select more than one value
- ▶ Hold down the Ctrl (windows) / Command (Mac) button to select multiple options.

```
<label>Adults aged:</label>
<select name="adults" size="4" multiple id="adults">
  <option value="1">18-25</option>
  <option value="2">26-40</option>
  <option value="3">41-64</option>
  <option value="4">65+</option>
</select><br>
```

Adults aged:

18-25	▲
26-40	
41-64	
65+	▼

# <textarea> Element

- ▶ The **<textarea>** element defines a multi-line input field (**a text area**)
- ▶ The **rows** attribute specifies the visible number of lines in a text area.
- ▶ The **cols** attribute specifies the visible width of a text area

```
<label>Comments:</label>  
  <textarea name="comments" rows="5" cols="24">This is where a  
  description could be placed.</textarea><br>
```

Comments:

This is where a  
description could be  
placed.



# <textarea> Element

- ▶ The size of the text area can be defined by using CSS:

```
<label>Comments:</label>  
  <textarea name="comments" style="width:300px; height:100px">This  
  is where a description could be placed.</textarea><br>
```

Comments:

# <input type="password">

- ▶ Defines a **password field**
- ▶ The characters in a password are masked (shown as asterisks or circles)

```
<label for="username">Username:</label>  
<input type="text" name="username" id="username" required><br>  
<label for="password">Password:</label>  
<input type="password" name="password" id="password" required><br>
```

Username:	<input type="text" value="John Doe"/>
Password:	<input type="password" value="*****"/>

# <input type="submit">

- ▶ Defines a button for submitting form data to a form-handler
- ▶ The form-handler is specified in the form's action attribute
- ▶ When "Submit Request" is clicked, the name of each control and its data value are sent to the server (response.html) for processing

```
<form action="response.html">
  <fieldset>
    <legend>Contact Information</legend>
    <label for="name">Name:</label>
    <input type="text" name="name" id="name"><br>
    <label for="email">Email:</label>
    <input type="text" name="email" id="email"><br>
    <label for="phone">Phone:</label>
    <input type="text" name="phone" id="phone" placeholder="999-999-9999"
    ><br>
  </fieldset>

  <input type="submit" id="submit" value="Submit Request">
</form>
```

**Contact Information**

Name:

Email:

Phone:

# <input type="reset">

- ▶ Defines a reset button that resets all form values to their default values
- ▶ If the "Clear Form" button is clicked, the form-data is reset to the default values

```
<form action="response.html">
  <fieldset>
    <legend>Contact Information</legend>
    <label for="name">Name:</label>
    <input type="text" name="name" id="name"><br>
    <label for="email">Email:</label>
    <input type="text" name="email" id="email"><br>
    <label for="phone">Phone:</label>
    <input type="text" name="phone" id="phone" placeholder="999-999-9999"><br>
  </fieldset>

  <input type="submit" id="submit" value="Submit Request">
  <input type="reset" id="reset" value="Clear Form"><br>
</form>
```

The screenshot shows a web form titled "Contact Information" with a blue border. It contains three text input fields: "Name:" with the value "John Doe", "Email:" with the value "johndoe@johndoe.com", and "Phone:" with the value "999-999-9999". Below the input fields are two buttons: "Submit Request" and "Clear Form".


# <input type="button">

- ▶ Defines a clickable **button**. The button has an **onclick** action that is executed when clicked.

```
<!DOCTYPE html>
<html>
<body>

<input type="button" onclick="alert('Hello World!')" value="Click Me!">

</body>
</html>
```



Click Me!


# Input Type Attribute - **required**

30

- ▶ The **required** attribute specifies that an input field must be filled out before submitting the form.
- ▶ An error is shown when trying to submit without filling out the required field and the submission does not happen

```
<label for="email">Email Address:</label>  
<input type="email" name="email" id="email" required>
```

Email Address:

Phone Number:  Please fill out this field.

Referral Source:

# Form Validation

- ▶ Two valid ways to write it.

```
<label for="username">Username:</label>  
<input type="text" name="username" required="required" />
```

```
<label for="password">Password:</label>  
<input type="password" name="password" required />
```

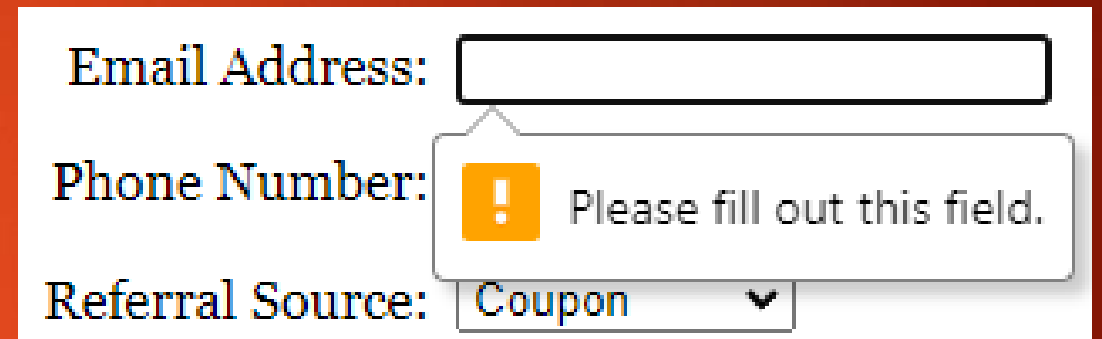
```
<input type="submit" value="Submit" />
```

# Default error display may look different on different browsers



Username:   
Password:

Please fill out this field.



Email Address:   
Phone Number:   
Referral Source:

! Please fill out this field.

- ▶ Custom validation messages are possible:

```
<input type="text" id="username" required  
oninvalid="this.setCustomValidity('Enter Username Here')"  
oninput="this.setCustomValidity('')"/>
```



# <input type="email" required>

- ▶ By changing the input type to "email" while using the required attribute, the browser can be used to validate (with some limitations) email addresses.

```
<legend>Contact Information</legend>
<label for="name">Name:</label>
<input type="text" name="name" id="name" required><br>
<label for="email">Email:</label>
<input type="email" name="email" id="email" required><br>
<label for="phone">Phone:</label>
<input type="text" name="phone" id="phone" placeholder="999-999-9999"><br>
```

Contact Information

Name:

Email:

⚠ Please include an '@' in the email address. 'doe' is missing an '@'.

Contact Information

Name:

Email:

⚠ Please enter a part following '@'. 'doe@' is incomplete.

Contact Information

Name:

Email:

Phone:

⚠ '.' is used at a wrong position in '.com'.

# `<input type="tel" required pattern="[0-9]{3}-[0-9]{3}-[0-9]{4}">`

- ▶ By changing the input type to `"tel"` and using the `pattern` attribute while using the `required` attribute, the browser can be used to validate phone numbers. Note this pattern excludes country code.

```
<legend>Contact Information</legend>
<label for="name">Name:</label>
<input type="text" name="name" id="name" required><br>
<label for="email">Email:</label>
<input type="email" name="email" id="email" required><br>
<label for="phone">Phone:</label>
<input type="tel" name="phone" id="phone" placeholder="999-999-9999" required pattern=
"[0-9]{3}-[0-9]{3}-[0-9]{4}"><br>
```

The screenshot shows a web form titled "Contact Information" with three input fields: "Name:" (containing "John Doe"), "Email:" (containing "john@doe.com"), and "Phone:" (containing "519-"). A red error message box is displayed over the "Phone:" field, containing a red exclamation mark icon and the text "Please match the requested format." Below the form is a "Submit" button.

# <input type="date" required>

- ▶ By changing the input type to “date” and using the required attribute, the browser can be used to input specific dates.

```
<legend>General Information</legend>
<label for="arrival_date">Arrival date:</label>
<input type="date" name="arrival_date" id="arrival_date" autofocus required><br>
<label for="nights">Departure:</label>
<input type="date" name="nights" id="nights" required><br>
```

**General Information**

Arrival date:

Departure:

**General Information**

Arrival date:

Departure:

Adults aged:

Children:

Comments:

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30	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

**General Information**

Arrival date:

Departure:

Adults aged:

Children:

Please fill out this field.

# Academic Integrity Reminder

# This week - Forms

37

- ▶ Forms quiz
- ▶ Forms homework
- ▶ Should be on to week 11 term project work.
- ▶ Add training.html to your term project
  
- ▶ Next week:
- ▶ Chapter 9 - Audio & Video