

# Tutorial 6

---

CREATING WEB FORMS

# Example Survey Page

## Survey Form

Figure 6-1 Initial survey page

The screenshot shows a customer survey page for Red Ball Pizza. At the top, there's a navigation bar with links for home, menu, directions, coupons, orders, catering, and reviews. Below the navigation is a cartoon illustration of a chef holding a pizza. To the right of the chef is a photo of a smiling woman holding a slice of pizza. The main title is "Customer Survey". A message thanks participants for their responses and informs them they are entered into a monthly drawing for a prize. It also states that surveys are private and confidential. At the bottom, there's contact information for Red Ball Pizza and credits for the images used.

survey form will go here

Required values are marked by an asterisk (\*)

Customer Survey

Thank you for taking our customer survey. Your response helps Red Ball Pizza maintain the tradition that has made us the top-rated pizzeria in the metro area.

All participants are automatically entered into a monthly drawing to receive a Red Ball Express PizzaFest containing two large pizzas, a 2-liter soda, and a side order of chicken wings. Check your e-mail inbox for contest results.

Surveys are private and confidential. Red Ball Pizza will not share your contact information with third parties, ever.

Red Ball Pizza • 811 Beach Drive • Ormond Beach, FL 32175 • (386) 555 - 7499

Timea/Shutterstock.com  
Ariwasabi/Shutterstock.com

## Proposed Survey Form

Figure 6-2 Proposed survey form

The proposed survey form is a clean, modern design. It starts with a header stating "Required values are marked by an asterisk (\*)." The form is divided into two columns. The left column contains fields for "Customer Information" such as Name\*, Street address, City, State, Postal code, Phone number, E-mail\*, and Where did you hear about us?. It also includes questions like "How many times do you dine out per month?" and "What's your favorite Red Ball special dish?". The right column is titled "Share Your Experience at Red Ball Pizza" and includes fields for Date of visit, Receipt number\*, Order type, and responses to service and food quality. There are also rating scales for Overall food quality and Overall service quality, and a general comments section.

Required values are marked by an asterisk (\*)

Customer Information

Name \*

Street address

City

State

Postal code

Phone number

E-mail \*

Where did you hear about us?

How many times do you dine out per month?

What's your favorite Red Ball special dish?

Share Your Experience at Red Ball Pizza

Date of visit

Receipt number \*

Order type

Was the service friendly?

Was the order correct?

Was the food hot?

Overall food quality

Overall service quality

General comments

# Instruction No. 2

---

- Link pizza.css file to order.htm file
  - i.e., <link href=.... />

# Form Elements

**Figure 6-4** Inserting a form element

```
<section>
  <p>Required values* are marked by an asterisk (*)</p>
  <form id="survey" name="survey">
  </form>
</section>
```

**Figure 6-5** Values of the enctype attribute

Value	Description
application/x-www-form-urlencoded	The default format. In this format, form data is transferred as a long text string in which spaces are replaced with the + character and nontext characters (such as tabs and line breaks) are replaced with their hexadecimal code values. Field names are separated from their field values with an = symbol.
multipart/form-data	Used when sending files to a server. In this format, spaces and nontext characters are preserved, and data elements are separated using delimiter lines. The action type of the <code>form</code> element must be set to <code>post</code> for this format.
text/plain	Form data is transferred as plain text with no encoding of spaces or nontext characters. This format is most often used when the action type of the <code>form</code> element is set to <code>mailto</code> .

#3

# Form Attributes

Figure 6-6 Setting the form attributes

```
<form id="survey" name="survey"  
      action="http://www.redballpizza.com/cgi-bin/survey"  
      method="post">
```

A set of form attributes specifies where to send the form data and how to send it. You indicate this information by adding the *action*, *method*, and *enctype* attributes to the form element, as follows: `<form action="url" method="type" enctype="type">....</form>`

Where *url* specifies the filename and location of the program that processes the form, the *method* attribute specifies how Web browsers should send data to the server, and the *enctype* attribute specifies the format of the data stored in the fields. (See Figure 6-5)

The method attribute has two possible values: **get** and **post**. The **get method**, the default, appends the form data to the end of the URL specified in the action attribute so the url can be bookmarked for future searching (results in long URLs). The **post method** sends form data in a separate data stream.

# Javascript file

**Figure 6-7** Linking to the `formsubmit.js` file

```
<meta charset="UTF-8" />
<title>Customer Survey</title>
<script src="modernizr-1.5.js"></script>
<script src="formsubmit.js"></script>

<link href="rb.css" rel="stylesheet" />
```

#4

# Field Sets

Figure 6-8 Inserting field sets

```
<form id="survey" name="survey"
      action="http://www.redballpizza.com/cgi-bin/survey"
      method="post">
  <fieldset id="custInfo">
  </fieldset>
  <fieldset id="experience">
  </fieldset>
</form>
```

Figure 6-9 Inserting field set legends

```
<form id="survey" name="survey"
      action="http://www.redballpizza.com/cgi-bin/survey"
      method="post">
  <fieldset id="custInfo">
    <legend>Customer Information</legend>
  </fieldset>
  <fieldset id="experience">
    <legend>Share Your Experience at Red Ball Pizza</legend>
  </fieldset>
</form>
```

Figure 6-10 Viewing field sets and legends



# Input Box Data Types

Figure 6-11 Input box data types

Type	Displays	General Appearance
button	A button that can be clicked to perform an action from a script	<input type="button" value="Run Program"/>
checkbox	A check box that can be clicked by the user	<input type="checkbox"/> <input checked="" type="checkbox"/>
file	A Browse button to locate and select a file	<input type="file" value="C:\survey.htm"/> <input type="button" value="Browse..."/>
hidden	A hidden field, not viewable on the form	
image	An inline image that can be clicked to perform an action from a script	
password	An input box that hides text entered by the user	<input type="password" value="*****"/>
radio	An option button that can be clicked by the user	<input type="radio"/> <input checked="" type="radio"/>
reset	A button that resets the form when clicked	<input type="button" value="Cancel Form"/>
submit	A button that submits the form when clicked	<input type="button" value="Submit Form"/>
text	An input box that displays text entered by the user	<input type="text" value="Alice Nichols"/>

#5

# Inserting Input Boxes

Figure 6-12 Inserting input boxes to record customer information

```
<fieldset id="custInfo">
  <legend>Customer Information</legend>
  Name *
  <input name="custname" id="custname" />
  Street address
  <input name="street" id="street" />
  City
  <input name="city" id="city" />
  State (abbr.)
  <input name="state" id="state" />
  Postal code
  <input name="zip" id="zip" />
  Phone number
  <input name="phone" id="phone" />
  E-mail *
  <input name="email" id="email" />
</fieldset>
```

descriptive text

text input box

field name

id for the input box control

# Input Boxes

Figure 6-13 Inserting input boxes to record the customer's experience

```
<fieldset id="experience">
  <legend>Share Your Experience at Red Ball Pizza</legend>
  Date of visit
  <input name="visitdate" id="visitdate" />
  Receipt number *
  <input name="receipt" id="receipt" />
</fieldset>
```

Figure 6-14 Input boxes with sample customer name

Required values are marked by an asterisk (\*)

Customer Information

Name *	Street address	City	State (abbr.)
Postal code	Phone number	E-mail *	
Share Your Experience at Red Ball Pizza			
Date of visit	Receipt number *		

descriptive text

input text box with sample text

The diagram illustrates the mapping of descriptive text and input text box with sample text to form fields. A green box labeled 'descriptive text' has an arrow pointing to the 'Name \*' field. Another green box labeled 'input text box with sample text' has an arrow pointing to the 'Share Your Experience at Red Ball Pizza' text area.

#4,5,6,9,14,18,

# Form Labels

Figure 6-15 Adding form labels

```
<fieldset id="custInfo">
    <legend>Customer Information</legend>
    <label for="custname">Name *</label>
    <input name="custname" id="custname" />

    <label for="street">Street address</label>
    <input name="street" id="street" />

    <label for="city">City</label>
    <input name="city" id="city" />

    <label for="state">State (abbr.)</label>
    <input name="state" id="state" />

    <label for="zip">Postal code</label>
    <input name="zip" id="zip" />

    <label for="phone">Phone number</label>
    <input name="phone" id="phone" />

    <label for="email">E-mail *</label>
    <input name="email" id="email" />
</fieldset>

<fieldset id="experience">
    <legend>Share Your Experience at Red Ball Pizza</legend>
    <label for="visitdate">Date of visit</label>
    <input name="visitdate" id="visitdate" />

    <label for="receipt">Receipt number *</label>
    <input name="receipt" id="receipt" />
</fieldset>
```

label associated with the visitdate input box

label element

#22

# Form Field Sets

Figure 6-16 Styles for the form field sets

adds a 1-pixel-wide solid red border to each field set

floats each field set on the left

sets the background color of each field set to a medium yellow

```
/* Field set styles */  
fieldset{  
background-color: rgb(255, 246, 205);  
border: 1px solid rgb(233, 69, 0);  
float: left;  
margin: 10px 0px 10px 2.5%;  
width: 46%;  
}
```

sets the margins around the field sets

sets the width of each field set to 46% of the total page width

# Field Set Legends

Figure 6-17 Styles for the field set legends

```
Legend {  
background-color: rgb(233, 69, 0);  
color: white;  
padding: 3px 0px;  
text-indent: 5px;  
width: 100%;  
}
```

displays the legends in white text on a red background

sets the text indent and the padding space around the legend

sets the width of each legend to 100% of the width of the field set

# Field Label Styles

Figure 6-18 Styles for the field labels

```
/* Label styles */
label {
    clear: left;
    display: block;
    float: left;
    font-size: 0.9em;
    margin: 7px 4% 7px 5px;
    width: 40%;}
```

displays the label only when the left margin is clear

displays each label as a block floated on the left

sets the font size of each field label

sets the margin around each label

sets the label width to 40% of the width of the field set

# Input Control Styles

Figure 6-19    Styles for input controls

```
/* Input control styles */  
input {  
    display: block;  
    float: left;  
    font-size: 0.9em;  
    margin: 7px 0px;  
    width: 50%;  
}
```

displays each input control as a block, floated on the left

sets the font size of the text in the input control

sets the margin space around the input control

sets the width of each input control to 50% of the width of the field set

# Example of Revised Format of Survey Form

Figure 6-20    Revised format of the survey form

The diagram illustrates the revised format of a survey form. It features two main sections: 'Customer Information' and 'Share Your Experience at Red Ball Pizza'. A legend at the top left defines terms: 'legend' points to the text 'Required values are marked by an asterisk (\*)'; 'field set' points to the 'Customer Information' section; 'field label' points to the label 'Date of visit' under the second section; and 'input box' points to the text input field next to the 'Receipt number \*' label.

**Customer Information**

Required values are marked by an asterisk (\*)

Name *	<input type="text"/>
Street address	<input type="text"/>
City	<input type="text"/>
State (abbr.)	<input type="text"/>
Postal code	<input type="text"/>
Phone number	<input type="text"/>
E-mail *	<input type="text"/>

**Share Your Experience at Red Ball Pizza**

Date of visit	<input type="text"/>
Receipt number *	<input type="text"/>

field set

field label

input box

# Input Box and Default Values

## Setting Width to Input Box

Figure 6-21 Setting the width of the state input box

```
input#state {  
    width: 50px;  
}
```

sets the width of  
the state input  
box to 50 pixels

## Defining Default Values

Figure 6-22 Defining default values for the city and state fields

```
<label for="city">City</label>  
<input name="city" id="city" value="Ormond Beach" />  
  
<label for="state">State (abbr.)</label>  
<input name="state" id="state" value="FL" />
```

default value for the city field

default value for the state field

Figure 6-24 Viewing default values and placeholder text

Customer Information		Share Your Experience at Red Ball Pizza	
Name *	first and last name	Date of visit	
Street address		Receipt number *	re-nnnnnn
City	Ormond Beach		
State (abbr.)	FL		
Postal code	nnnn (-nnnn)		
Phone number	(nnn) nnn-nnnn		
E-mail *			

default value for  
the state field

placeholder text for  
the receipt input box

#7

# Adding Placeholders to Form

Figure 6-23 Adding placeholders to the survey form

```
<fieldset id="custInfo">
    <legend>Customer Information</legend>
    <label for="custname">Name *</label>
    <input name="custname" id="custname"
        placeholder="first and last name" />
    <label for="street">Street address</label>
    <input name="street" id="street" />
    <label for="city">City</label>
    <input name="city" id="city" value="Ormond Beach" />
    <label for="state">State (abbr.)</label>
    <input name="state" id="state" value="FL" />
    <label for="zip">Postal code</label>
    <input name="zip" id="zip"
        placeholder="nnnn (-nnnn)" />
    <label for="phone">Phone number</label>
    <input name="phone" id="phone"
        placeholder="(nnn) nnn-nnnn" />
    <label for="email">E-mail *</label>
    <input name="email" id="email" />
</fieldset>
<fieldset id="experience">
    <legend>Share Your Experience at Red Ball Pizza</legend>
    <label for="visitdate">Date of visit</label>
    <input name="visitdate" id="visitdate" />
    <label for="receipt">Receipt number *</label>
    <input name="receipt" id="receipt"
        placeholder="re-nnnnnn"/>
</fieldset>
```

placeholder text  
for the phone field

Figure 6-24 Viewing default values and placeholder text

Customer Information		Share Your Experience at Red Ball Pizza
Name *	first and last name	Date of visit
Street address		Receipt number *
City	Ormond Beach	re-nnnnn
State (abbr.)	FL	
Postal code	nnnn (-nnnn)	
Phone number	(nnn) nnn-nnnn	
E-mail *		

A placeholder is a text string that appears within the control element and provides users with information about the kind of information accepted by the field.

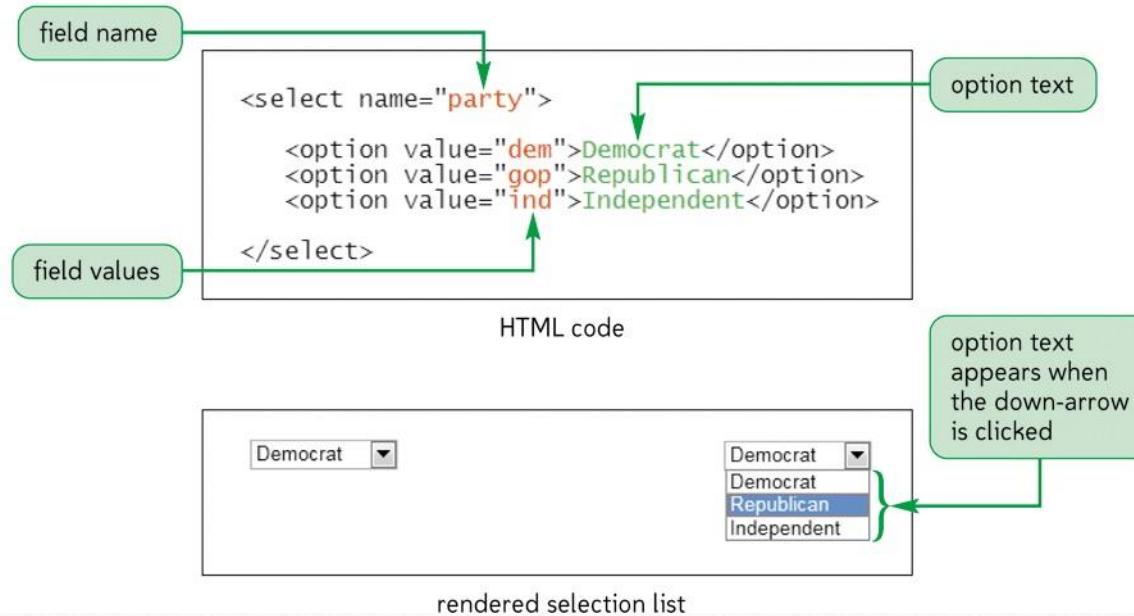
For text: `placeholder="text"`

For phone numbers:

`<input name="phone" placeholder="(nnn)nnn-nnnn" />`

# Selection List

Figure 6-25 Creating a selection list



# Order Type Selection List

Figure 6-26 Creating the ordertype selection list

```
label text  
<label for="ordertype">Order type</label>  
<select name="ordertype" id="ordertype">  
    <option value="type1">Carry out</option>  
    <option value="type2">Delivery</option>  
    <option value="type3">Dine in</option>  
    <option value="type4">Take and bake</option>  
</select>  
</fieldset>  
</form>
```

selection list options

Figure 6-29 Specifying the selected option

```
<label for="ordertype">Order type</label>  
<select name="ordertype" id="ordertype">  
    <option value="type1">Carry out</option>  
    <option value="type2">Delivery</option>  
    <option value="type3" selected="selected">Dine in</option>  
    <option value="type4">Take and bake</option>  
</select>
```

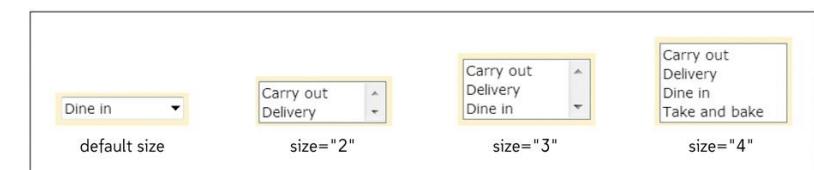
the Dine in option will appear selected by default

Figure 6-28 Order type selection list

Share Your Experience at Red Ball Pizza

Date of visit	<input type="text"/>
Receipt number *	<input type="text"/>
Order type	<input type="button" value="Carry out"/>

Figure 6-30 Setting the size of the selection list



# Select Element Style Rule

Figure 6-27 Style rule for the select element

```
input#state {  
    width: 50px;  
}  
  
/* Selection list styles */  
  
select {  
    display: block;  
    float: left;  
    font-size: 0.9em;  
    margin: 7px 0px;  
}
```

displays the selection list as a block

floats the selection list on the left

sets the font size to 0.9 em

sets the margin space around the selection list

# InfoSrc Selection List

## Creating List

Figure 6-31 Creating the infoSrc selection list

```
<label for="email">E-mail *</label>
<input name="email" id="email" />

<label>Where did you hear about us?</label>
<select name="infoSrc" id="infoSrc" size="5">
    <option value="internet">Internet</option>
    <option value="mag">Magazine</option>
    <option value="news">Newspaper</option>
    <option value="word">Word of Mouth</option>
    <option value="other">Other</option>
</select>
```

setting the size of the selection list to five options

Figure 6-32 List box for listing the information source

A screenshot of a web form. It includes an input field labeled "E-mail \*", a text area, and a dropdown menu labeled "Where did you hear about us?". The dropdown menu contains five options: "Internet", "Magazine", "Newspaper", "Word of Mouth", and "Other". A green callout box points to the dropdown menu with the text "five options are displayed in the selection list".

## Multiple Selections

Figure 6-33 Allowing for multiple selections

```
<label>Where did you hear about us? <br />(select all that apply)</label>
<select name="infoSrc" id="infoSrc" size="5" multiple="multiple">
    <option value="internet">Internet</option>
    <option value="mag">Magazine</option>
    <option value="news">Newspaper</option>
    <option value="word">Word of Mouth</option>
    <option value="other">Other</option>
</select>
```

revised label text

users can select multiple options

#19

# Option Groups in Selection List

Figure 6-34 Organizing a selection list with option groups

HTML code

```
<label for="party">Candidate</label>
<select name="party">

    <optgroup label="Democrat">
        <option value="d1">Tim Harris</option>
        <option value="d2">Gary Nielsen</option>
        <option value="d3">Kate Paulteny</option>
    </optgroup>

    <optgroup label="Republican">
        <option value="r1">Barbara Alt</option>
        <option value="r2">Peter Trudeau</option>
        <option value="r3">Maria Sandoval</option>
    </optgroup>

</select>
```

rendered selection list

Candidate

Tim Harris

<b>Democrat</b>
<b>Tim Harris</b>
Gary Nielsen
Kate Paulteny
<b>Republican</b>
Barbara Alt
Peter Trudeau
Maria Sandoval

#19

# Option Button Group

## Creating Option Button Group

Figure 6-35 Creating a group of option buttons

```
HTML code <fieldset>
    <legend>Party Affiliation</legend>
    <label for="demOption">Democrat</label>
    <input type="radio" name="party" id="demOption" value="dem" />
    <label for="repOption">Republican</label>
    <input type="radio" name="party" id="repOption" value="rep" />
    <label for="indOption">Independent</label>
    <input type="radio" name="party" id="indOption" value="ind" />
</fieldset>
```

rendered option buttons

Party Affiliation  
Democrat  Republican  Independent

## Creating serviceFriendly field

Figure 6-36 Option button group for the serviceFriendly field

```
<label value="type4">Take and bake</option>
</select>
<label>Was your service friendly?</label>
<fieldset class="optionGroup">
    <label for="sYes">Yes</label>
    <input type="radio" name="serviceFriendly" id="sYes" value="yes" />
    <label for="sNo">No</label>
    <input type="radio" name="serviceFriendly" id="sNo" value="no" />
</fieldset>
```

field set containing the label and option buttons

label associated with the option button control

option button for the serviceFriendly field

Figure 6-37 Rendered option buttons

Order type	Dine in <input type="button" value="▼"/>
Was your service friendly?	<input checked="" type="radio"/> Yes <input type="radio"/> No

# Option Button Group Style Rules

Figure 6-38 Style rules for the option button group

```
/* Option button styles */  
fieldset.optionGroup {  
    border-width: 0px;  
}  
  
fieldset.optionGroup label {  
    display: inline;  
    float: none;  
    margin: 0px 3px 0px 0px;  
    width: 30px;  
}  
  
fieldset.optionGroup input {  
    display: inline;  
    float: none;  
    margin: 0px 20px 0px 0px;  
    width: 20px;  
}
```

removes the border from the field set

displays the labels inline without floating; sets the right margin to 3 pixels and the width to 30 pixels

displays the input controls inline without floating; sets the right margin and width to 20 pixels

Figure 6-39 Revised appearance of the option button group

Order type Dine in ▾

Was your service friendly? Yes  No

# Option Button

## Before Style Rule

Figure 6-37 Rendered option buttons

A screenshot of a web form. At the top, there is a dropdown menu labeled "Order type" with the value "Dine in". Below it is a question "Was your service friendly?" followed by two radio buttons. The "Yes" radio button is selected, indicated by a blue outline.

## After Style Rule

Figure 6-39 Revised appearance of the option button group

The same web form as Figure 6-37, but after applying CSS style rules. The two radio buttons for "Was your service friendly?" are now displayed side-by-side, no longer stacked vertically.

Figure 6-38 Style rules for the option button group

```
/* Option button styles */
fieldset.optionGroup {
    border-width: 0px;
}

fieldset.optionGroup label {
    display: inline;
    float: none;
    margin: 0px 3px 0px 0px;
    width: 30px;
}

fieldset.optionGroup input {
    display: inline;
    float: none;
    margin: 0px 20px 0px 0px;
    width: 20px;
}
```

removes the border from the field set

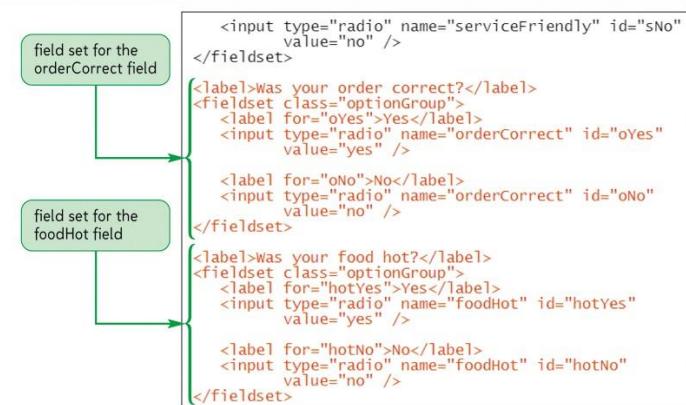
displays the labels inline without floating; sets the right margin to 3 pixels and the width to 30 pixels

displays the input controls inline without floating; sets the right margin and width to 20 pixels

# Option Button Groups

orderCorrect & foodHot fields

Figure 6-40 Option button groups for the orderCorrect and foodHot fields



Rendered Option Button Group

Figure 6-41 Completed option button groups

The screenshot shows a rendered web page with three groups of radio buttons. The first group, 'serviceFriendly', has 'Yes' and 'No' options. The second group, 'orderCorrect', has 'Yes' and 'No' options. The third group, 'foodHot', also has 'Yes' and 'No' options. Each group is labeled with its respective field name: 'serviceFriendly field', 'orderCorrect field', and 'foodHot field'.

Order type	Dine in
Was your service friendly?	<input checked="" type="radio"/> Yes <input type="radio"/> No
Was your order correct?	<input checked="" type="radio"/> Yes <input type="radio"/> No
Was your food hot?	<input checked="" type="radio"/> Yes <input type="radio"/> No

# Text Area Control

## Text Area Control

Figure 6-42

Creating a text area control

```
<label for="hotNo">No</label>
<input type="radio" name="foodHot" id="hotNo"
      value="no" />
</fieldset>

<label for="comments">Tell us more about your experience!</label>
<textarea name="comments" id="comments"></textarea>
</fieldset>
```

comments field

## Text Area Box Style Rule

Figure 6-43

Style rule for the text area box

```
/* Text area styles */
textarea {
    display: block;
    font-size: 0.9em;
    float: left;
    height: 150px;
    margin: 10px 0px;
    width: 50%;
}
```

sets the font size to 0.9 em

floats the text area box on the left

makes the text area box 150 pixels high with a width of 50% of the field set; sets the top and bottom margins to 10 pixels

displays the text area box as a block

Figure 6-44 Text area box

A screenshot of a web page showing a form. At the top, there are two radio buttons labeled "Yes" and "No". Below the radio buttons is a text area with the placeholder text "Tell us more about your experience!". The entire form is contained within a light yellow box.

# Check Box

## Creating a Check Box

Figure 6-45 Creating a check box for the newscb field

```
<label for="comments">Tell us more about your experience!</label>
<textarea name="comments" id="comments"></textarea>

</fieldset>

<label id="newsletter">
  <input type="checkbox" name="newscb" />
  E-mail me your newsletter for great coupons and specials!
</label>

</form>
```

## Check Box Style Rule

Figure 6-46 Style rules for the label and check box

```
/* Check box styles */
#newsletter {
  color: rgb(233, 69, 0);
  float: none;
  margin: 10px auto;
  text-align: center;
  width: 90%;
}

#newsletter input {
  display: inline;
  float: none;
  width: 20px;
}
```

displays the label text in medium red

displays the label text without floating

sets the top/bottom margin to 10 pixels, centers the label text, and sets the width to 90% of the page width

displays the check box inline within the label and without floating, and sets the width to 20 pixels

# Sample Survey Form

Figure 6-47 Current appearance of the survey form

Required values are marked by an asterisk (\*)

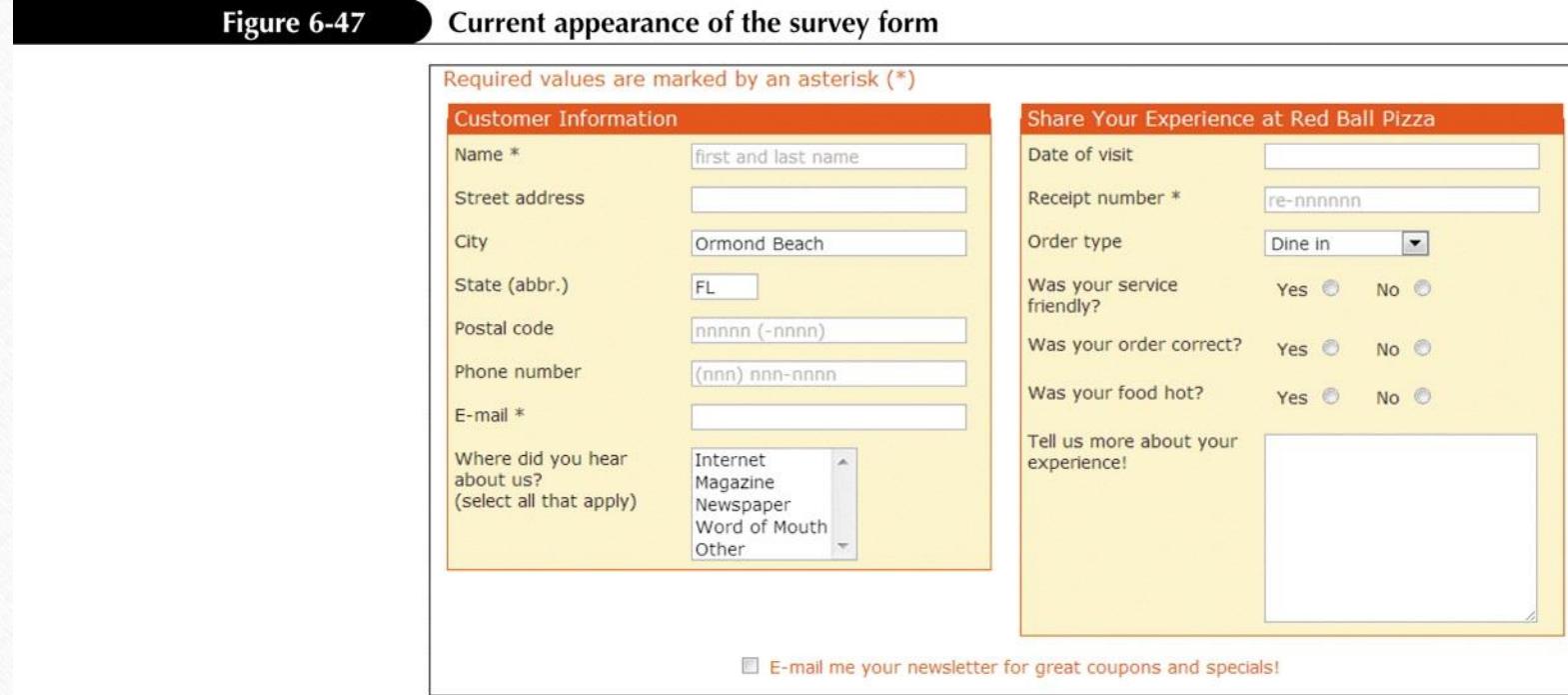
**Customer Information**

Name *	first and last name
Street address	
City	Ormond Beach
State (abbr.)	FL
Postal code	nnnn (-nnnn)
Phone number	(nnn) nnn-nnnn
E-mail *	
Where did you hear about us? (select all that apply)	Internet Magazine Newspaper Word of Mouth Other

**Share Your Experience at Red Ball Pizza**

Date of visit	
Receipt number *	re-nnnnnn
Order type	Dine in
Was your service friendly?	Yes <input checked="" type="radio"/> No <input checked="" type="radio"/>
Was your order correct?	Yes <input checked="" type="radio"/> No <input checked="" type="radio"/>
Was your food hot?	Yes <input checked="" type="radio"/> No <input checked="" type="radio"/>
Tell us more about your experience!	

E-mail me your newsletter for great coupons and specials!



# Data Types

Figure 6-48 HTML5 data types		
Type	Description	General Appearance
color	An RGB color value that can be selected from a color picker dialog box	
date	A date (year, month, day) with no specified time zone	
datetime	A date and time (year, month, day, hour, minute, second, fraction of a second) with the time zone set to Coordinated Universal Time (UTC)	
datetime-local	A date and time (year, month, day, hour, minute, second, fraction of a second) with no specified time zone	
email	An e-mail address or list of e-mail addresses	nichols@redballpizza.com
month	A date consisting of a year and a month	
number	A numeric value	5 <input type="button" value="."/>
range	A numeric value selected from a defined range of values	<input type="range" value="5"/>
search	A text string usually used for performing searches	local pizza <input type="button" value="X"/>
tel	A telephone number	(365) 555 - 7499
time	A time value (hour, minute, seconds, fractional seconds)	01:45 <input type="button" value="."/>
url	A URL of a Web site or Internet resource	http://www.redballpizza.com
week	A date consisting of a year number and a week number	

Figure 6-49 Virtual keyboards for different data types



Figure 6-50 Applying the tel and email data types

```

 input box contains a telephone number
<label for="phone">Phone number</label>
<input name="phone" id="phone" type="tel" placeholder="(nnn) nnn-nnnn" />

 input box contains an e-mail address
<label for="email">E-mail *</label>
<input name="email" id="email" type="email" />

```

Figure 6-51 Applying the date data type

```

<fieldset id="experience">
  <legend>Share Your Experience at Red Ball Pizza</legend>
  <label for="visitdate">Date of visit</label>
  <input name="visitdate" id="visitdate" type="date" />
</fieldset>
 input box contains a date

```

# Calendar Widget & Number Data Type

Figure 6-52 Calendar widget in the Opera browser

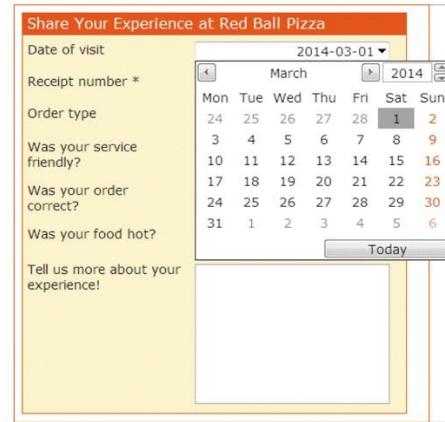


Figure 6-53 Applying a number data type

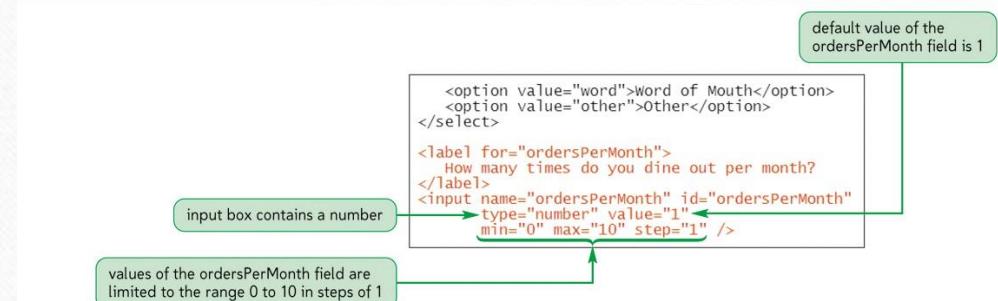


Figure 6-54 Style rule for the ordersPerMonth field

```
/* Number input box styles */
#ordersPerMonth {
    width: 70px;
}
```

A style rule for the "ordersPerMonth" input field. The rule is preceded by a comment /\* Number input box styles \*/. The selector #ordersPerMonth is followed by a brace {}, which is connected by a callout to the text "sets the width of the input box to 70 pixels".

# Number Spin Box

## Number Spin Box

Figure 6-55 Number spin box in the Opera browser

A screenshot of a web form in the Opera browser. The form includes a question "Where did you hear about us? (select all that apply)" with options: Internet, Magazine, Newspaper, Word of Mouth, and Other. Below it is another question "How many times do you dine out per month?" followed by a number spin box with the value "1". A green callout box points to the spin box with the text "click the spin arrows to increase and decrease the numeric value between 0 and 10".

## Input Element w/ Range Data Type

Figure 6-56 Adding input elements with the range data type

```
<label>Rate the overall service<br />(0 = poor; 10 = great)</label>
<input name="service" id="service" type="range" value="5" min="0" max="10" step="1" />

<label>Rate the food quality<br />(0 = poor; 10 = great)</label>
<input name="quality" id="quality" type="range" value="5" min="0" max="10" step="1" />

<label for="comments">Tell us more about your experience!</label>
<textarea name="comments" id="comments"></textarea>
```

#13

# Range Slider

Figure 6-57 Range slider in the Opera browser

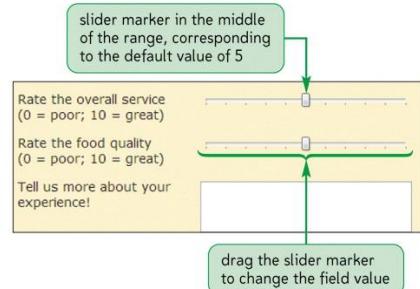


Figure 6-58 Adding descriptive labels to the range slider

```
<label>Rate the overall service<br />
  (0 = poor; 10 = great)</label>
<label class="sliderLabel">0</label>
<input name="service" id="service" type="range" value="5"
  min="0" max="10" step="1" />
<label class="sliderLabel">10</label>

<label>Rate the food quality<br />
  (0 = poor; 10 = great)</label>
<label class="sliderLabel">0</label>
<input name="quality" id="quality" type="range" value="5"
  min="0" max="10" step="1" />
<label class="sliderLabel">10</label>
```

Figure 6-59 Style rules for the range slider

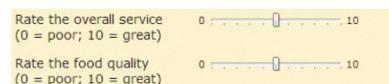
```
/* Range slider styles */
label.sliderLabel {
  clear: none;
  font-size: 0.7em;
  margin: 10px 0px;
  text-align: center;
  width: 10px
}

input[type="range"] {
  width: 150px;
}
```

Diagram illustrating style rules for the range slider:

- displays without waiting for the margins to be clear of floats
- reduces the font size to 0.7 em
- sets the top/bottom margin to 10 pixels; centers the text; and sets the width to 10 pixels
- sets the width of the range slider to 150 pixels

Figure 6-60 Range slider in the Opera browser



#10-12

# Data List & Submit & Reset Buttons

## Adding a Data List

Figure 6-61 Adding a data list

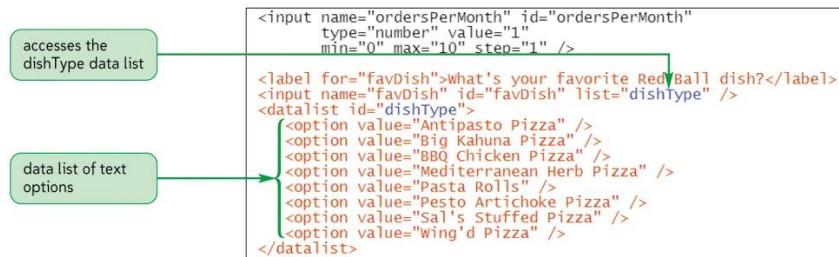
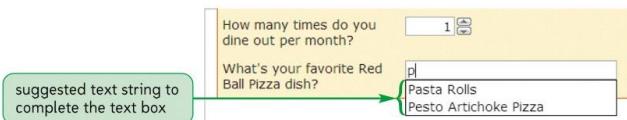


Figure 6-62 Accessing a data list of options



## Submit & Reset Buttons

#20

Figure 6-63 Creating submit and reset buttons

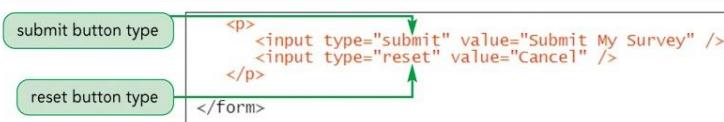
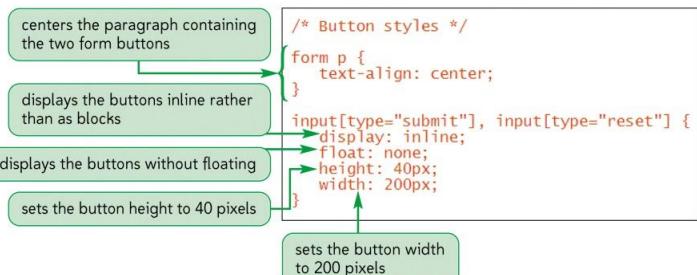


Figure 6-64 Style rules for the submit and reset buttons



# Final Layout of Example

Figure 6-65 Final layout of the survey page as displayed by Opera

The screenshot shows a customer survey page for Red Ball Pizza. At the top, there's a cartoon illustration of a chef holding a pizza and a banner that says "Red Ball". To the right of the banner is a photo of a smiling woman holding a pizza slice. Below the illustration and banner, there's a brief introduction: "Thank you for taking our customer survey. Your response helps Red Ball Pizza maintain the tradition that has made us the top-rated pizzeria in the metro area. All participants are automatically entered into a monthly drawing to receive a Red Ball Express PizzaFest containing two large pizzas, a 2-liter soda, and a side order of chicken wings. Check your e-mail inbox for contest results." A note below states: "Surveys are private and confidential. Red Ball Pizza will not share your contact information with third parties, ever."

**Customer Information**

Name \* first and last name  
Street address \_\_\_\_\_  
City Ormond Beach  
State (abbr.) FL  
Postal code nnnnn (-nnnn)  
Phone number (nnnn) nnnn-nnnn  
E-mail \_\_\_\_\_  
Where did you hear about us? (select all that apply)  
Internet Magazine Newspaper Word of Mouth Other  
How many times do you dine out per month? 1  
What's your favorite Red Ball dish? \_\_\_\_\_

**Share Your Experience at Red Ball Pizza**

Date of visit \_\_\_\_\_  
Receipt number \* re-nnnnnn  
Order type Dine in  
Was your service friendly? Yes  No   
Was your order correct? Yes  No   
Was your food hot? Yes  No   
Rate the overall service (0 = poor; 10 = great) 0           10  
Rate the food quality (0 = poor; 10 = great) 0           10  
Tell us more about your experience! \_\_\_\_\_

E-mail me your newsletter for great coupons and specials!

submit button → Submit My Survey ← Cancel ← reset button

Red Ball Pizza • 811 Beach Drive • Ormond Beach, FL 32175 • (386) 555 - 7499

Timea/Shutterstock.com  
Ariwasabi/Shutterstock.com

# Custom Button

Figure 6-66 Creating a custom button

HTML code

```
<button type="button">  
      
    Return to the <br /> Home Page  
</button>
```

custom button



# Required Fields

Figure 6-67 Marking required fields

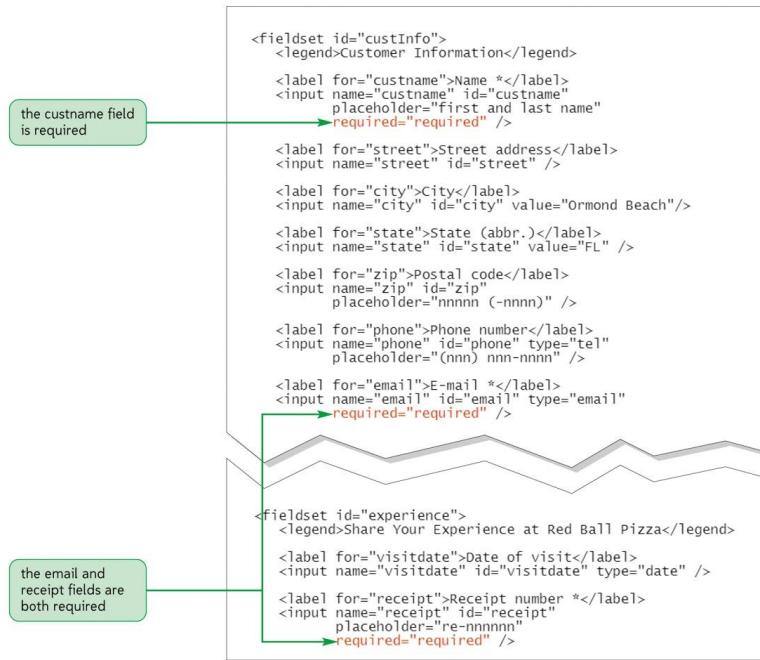
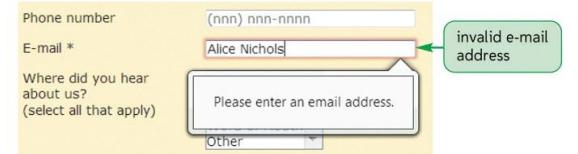


Figure 6-68 Data validation error message in Google Chrome



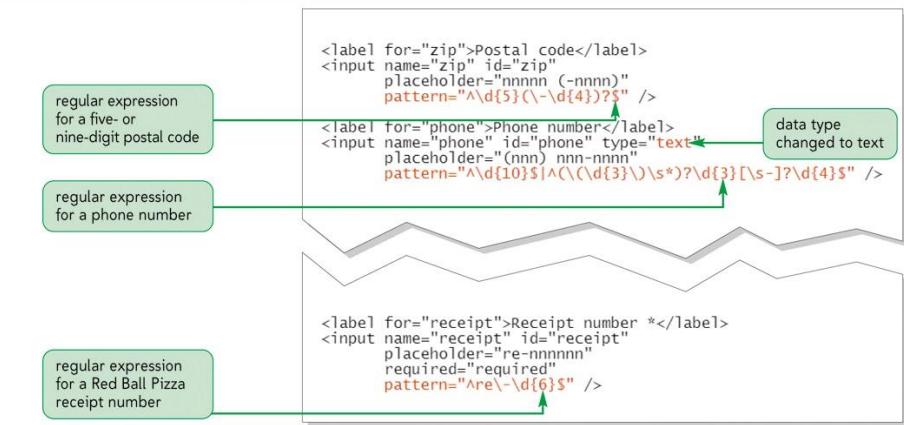
Figure 6-69 Entering an invalid e-mail address



#7,8

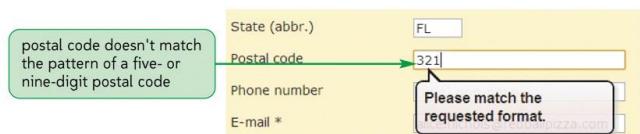
# Regular Expressions & Entering Invalid Code

Figure 6-70 Specifying character patterns with regular expressions



See `regex.txt` file in your Data Files folder for regular expressions used in this tutorial.

Figure 6-71 Entering an invalid postal code



A browser supports the date data type and rejects invalid dates if they are not in the form as yyyy-mm-dd. Same goes for telephone numbers and zip codes. The input element for the zip field, insert the regular expression pattern that tests for the presence of a five- or nine-digit postal code and the input element for a phone field would test for a valid phone number pattern.

# Pseudo-classes & Style Rules for Elements

Figure 6-72 Pseudo-classes for Web form controls

Pseudo-Class	Matches
checked	Check boxes or options that are checked
default	The default user control element
disabled	Control elements that are disabled
enabled	Control elements that are enabled
focus	Control elements that have the focus (are actively selected) in the form
indeterminate	Check boxes or option buttons whose toggle states (checked or unchecked) cannot be determined
in-range	Control elements whose values are within each field's range of values (between the min and max attribute values)
invalid	Control elements whose values fail validation tests
optional	Control elements that are optional (not required) in the Web form
out-of-range	Control elements whose values are outside each field's range of values (outside of the min and max attribute values)
required	Control elements that are required in the Web form
valid	Control elements whose values pass validation tests

Figure 6-73 Style rule for elements that have the focus

```
/* Validation styles */  
input:focus, select:focus, textarea:focus {  
    background-color: #ccffff;  
}
```

selector for input, select, and  
textarea elements that have the focus

changes the  
background color  
to light green

Figure 6-74 Changed background color for element that has the focus

Required values are marked by an asterisk (\*)

Customer Information

Name *	Alice Nichols
Street address	
City	Ormond Beach
State (abbr.)	FL

active input element has the focus  
with a light green background

# Invalid Field Values

Figure 6-75 Style rules for valid and invalid field values

```
/* validation styles */  
input:focus, select:focus, textarea:focus {  
background-color: #e0f2e0;  
}  
  
input:focus:valid {  
background: #e0f2e0 url(go.png) bottom right no-repeat;  
-o-background-size: contain;  
-moz-background-size: contain;  
-webkit-background-size: contain;  
background-size: contain;  
}  
  
input:focus:invalid {  
background: #ffeb3b url(stop.png) bottom right no-repeat;  
-o-background-size: contain;  
-moz-background-size: contain;  
-webkit-background-size: contain;  
background-size: contain;  
}
```

style rule for valid data values  
style rule for invalid data values  
displays the *stop.png* image file in the bottom-right corner of the input box on a light red background

displays the *go.png* image file in the bottom-right corner of the input box on a light green background  
resizes the image so that it is contained within the input box

resizes the image so that it is contained within the input box

Figure 6-76 Inline validation on the postal code

initial text does not match a valid postal code	Postal code	321	
five-digit postal code is valid	Postal code	32175	
as the user continues to type, the postal code becomes invalid again	Postal code	32175-61	
final nine-digit postal code is valid	Postal code	32175-6136	

# Completed Example

Figure 6-77 Build Your Own Pizza form

The screenshot shows a web-based ordering form for Red Ball Pizza. At the top, there's a navigation bar with links for home, menu, directions, coupons, orders, catering, and reviews. Below the navigation is a cartoon illustration of a chef holding a pizza and a delivery person holding a pizza box. The main title "Online Ordering" is centered above the form area.

**Required values are marked by an asterisk (\*)**

**Customer Information:**

- Name \*: Alice Nichols
- Delivery Address \*: 811 Beach Drive  
Ormond Beach, FL
- Phone \*: (386) 555-7499
- Delivery Time (leave blank for immediate delivery): 9:35 PM

**Build Your Own Pizza**

Select Your Pizza Size (10, 12, or 14 inch): 12

Choose Your Crust: Stuffed

Quantity (call for quantities larger than 10 pizzas): 2

Special Instructions: Add red pepper flakes on the side and please include extra napkins and paper plates.

**Meat Toppings:**

Location	Pepperoni	Ham	Pork	Sausage	Chicken
Top Left	●	●	●	●	●
Top Middle	●	●	●	●	●
Top Right	●	●	●	●	●
Middle Left	●	●	●	●	●
Middle Middle	●	●	●	●	●
Middle Right	●	●	●	●	●
Bottom Left	●	●	●	●	●
Bottom Middle	●	●	●	●	●
Bottom Right	●	●	●	●	●

Add Double Cheese:

Add Double Sauce:

**Vegetable Toppings:**

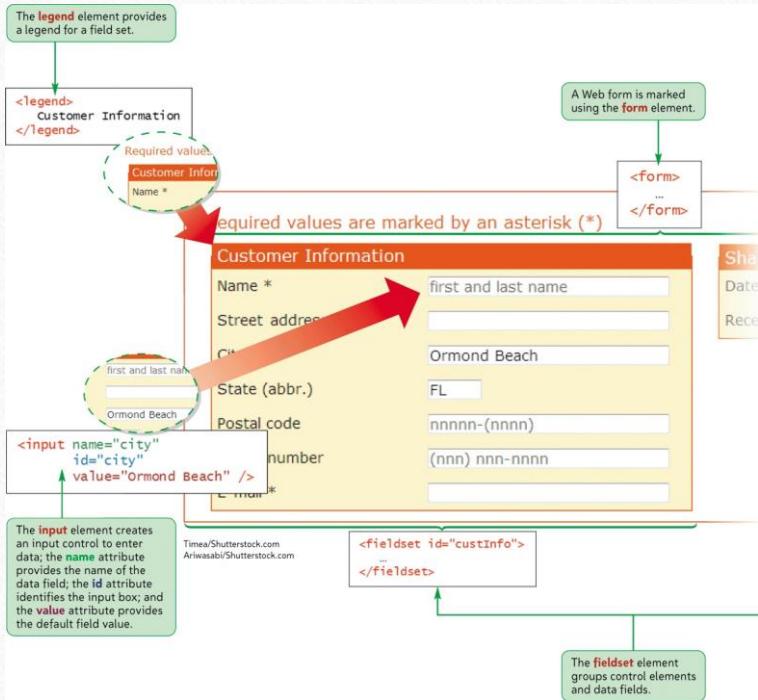
Location	Mushrooms	Green Peppers	Onions	Tomatoes	Jalapenos
Top Left	●	●	●	●	●
Top Middle	●	●	●	●	●
Top Right	●	●	●	●	●
Middle Left	●	●	●	●	●
Middle Middle	●	●	●	●	●
Middle Right	●	●	●	●	●
Bottom Left	●	●	●	●	●
Bottom Middle	●	●	●	●	●
Bottom Right	●	●	●	●	●

**Next**

Red Ball Pizza • 811 Beach Drive • Ormond Beach, FL 32175 • (386) 555 - 7499

Timea/Shutterstock.com  
Stephen Coburn/Shutterstock.com

# Page 394



# Page 395

The diagram illustrates a web form for sharing an experience at Red Ball Pizza. The form includes fields for Date of visit and Receipt number, both enclosed in a fieldset.

**HTML Structure:**

```
<label for="visitdate">Date of visit</label>
  

Fieldset:
<fieldset id="experience">
    ...
</fieldset>
```

**Annotations:**

- Date of visit:** A label with a green dashed circle around it. A callout box states: "The **label** element provides a text description for a control element." Below the label is the corresponding HTML code: `<label for="visitdate">Date of visit</label>`.
- for attribute:** A callout box states: "The **for** attribute identifies the id of the control element." It points to the `for="visitdate"` attribute in the label tag.
- Receipt number:** A label with a green dashed circle around it. A callout box states: "The **placeholder** attribute provides a hint about the type of data accepted in the field." Below the label is the corresponding HTML code: `<label for="receipt" placeholder="re-nnnnnn">Receipt number *</label>`.
- Placeholder:** A callout box states: "The **placeholder** attribute provides a hint about the type of data accepted in the field." It points to the `placeholder="re-nnnnnn"` attribute in the input tag.

# Page 420

The screenshot shows a web form for "Customer Information" on a Red Ball Pizza website. The form includes fields for Name, Street address, City, State, Phone number, E-mail, and a dropdown menu for "Where did you hear about us?". A checkbox at the bottom allows users to receive newsletters.

**HTML Code Examples:**

- Select Element:** `<select name="infosrc" id="infosrc" size="5" multiple="multiple">`
- Input Type Checkbox:** `<input name="newsrb" type="checkbox" />`

**Annotations:**

- A callout box points to the select element with the text: "The `size` attribute specifies the number of options displayed in the selection list, and the `multiple` attribute allows a user to make multiple selections."
- A callout box points to the checkbox with the text: "The `checkbox` type displays a check box control element in the Web form."
- A red arrow points from the "Where did you hear about us?" dropdown to the explanatory text above it: "Required values are marked by an asterisk (\*)".
- A red arrow points from the "E-mail me your newsletter for great..." checkbox to the explanatory text above it: "Required values are marked by an asterisk (\*)".

# Page 421

The screenshot shows a survey form titled "Share Your Experience at Red Ball Pizza". The form includes fields for Date of visit, Receipt number, Order type (a dropdown menu), and service and order satisfaction (radio buttons). A large text area for comments is also present.

**Order type:**

```
<select name="ordertype">
    <option value="type1">Carry out</option>
    <option value="type2">Delivery</option>
    <option value="type3" selected>Dine in</option>
    <option value="type4">Take and bake</option>
</select>
```

A callout box explains the `select` element: "The select element creates a selection list; the option elements provide the options in the list; option text is contained within the option element; and the default value is indicated by the selected attribute."

**Order correct:**

```
<input type="radio" name="ordercorrect" value="yes" />
<input type="radio" name="ordercorrect" value="no" />
```

A callout box explains the `radio` type: "The radio type displays a radio or option button; the name attribute defines the data field; and the value attribute provides the field value associated with the button."

**Comments:**

```
<textarea>...</textarea>
```

A callout box explains the `textarea` element: "The textarea element creates a text area box for multiple lines of text."

# pizza.css

---

22)

```
fieldset {  
}
```

```
#custInfo {  
}
```

```
#buildPizza {  
}
```

```
#mean, #vegetables {  
}
```

23)

```
Legend {  
}
```

24)

```
Label {  
}
```

# pizza.css

---

25)

```
input {  
}
```

```
textarea{  
}
```

26)

```
form img{  
}
```

27)

```
#delTime {  
}
```

28)

```
#sizeimage {  
}  
  
#size {  
}
```

# pizza.css

---

29)

```
#crust {  
}
```

30)

```
#quantity{  
}
```

31)

```
fieldset.optionGroup{  
}
```

32)

```
input[type="radio"]{  
}
```

```
input[type="checkbox"]{  
}
```

33)

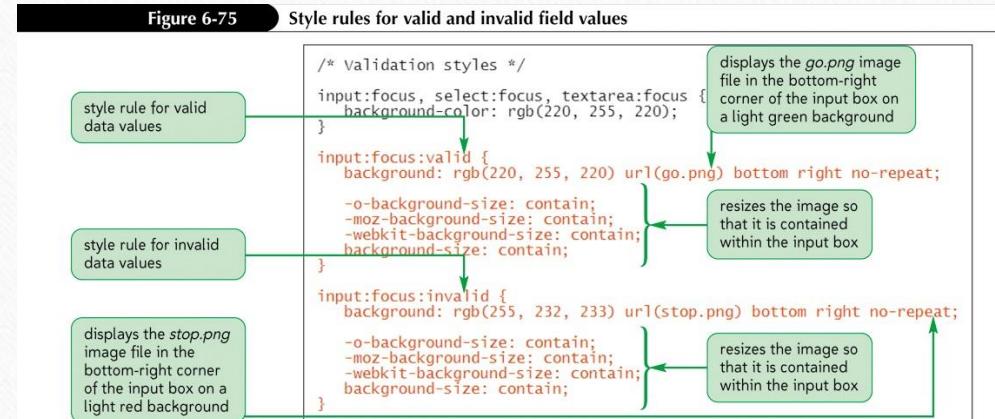
```
input[type="submit"]{  
}
```

# pizza.css

```
34)  
input:focus, select:focus, textarea:focus {  
}  
  
35)  
input:focus:valid {  
}  
  
36)  
Input:focus:invalid {  
}
```

Add to both #35 and #36:

```
-o-background-size:contain;  
-moz-background-size:contain;  
-webkit-background-size:contain;  
background-size: contain;
```



**Figure 6-76 Inline validation on the postal code**

Validation State	Postal Code	Visual Feedback
initial text does not match a valid postal code	321	
five-digit postal code is valid	32175	
as the user continues to type, the postal code becomes invalid again	32175-61	
final nine-digit postal code is valid	32175-6136	

# Page 442

The focus, valid, and invalid pseudo-elements can be used for **inline validation**, in which data errors are highlighted as they occur during data entry.

Required values are marked by an asterisk (\*)

Customer Information

Alice Nichols  
811 Beach Drive  
Ormond Beach  
FL  
32175  
(386) 555-7499  
alice.nichols@redballpizza.com

City  
State (abbr.)  
Postal code  
Phone number  
E-mail \*

Where did you hear about us? (optional)  
Other  
Internet Magazine Newspaper Word of Mouth Other

How many times do you dine out per month?  
3

What's your favorite Red Ball Pizza dish?  
Mediterranean Herb Pizza

E-mail me your newsletter for great deals!

Aniwasabi/Shutterstock.com

`<input name="name" required="required" />`

The **required** data type indicates that a value is required for the field, and the form will be rejected without it.

`<input name="orders" type="number" min="0" max="10" step="1" />`

The **number** data type creates a spin box; the **min** and **max** attributes define the minimum and maximum values, respectively, the **step** attribute defines the amount the value increases or decreases with each click of the spin arrow.

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

The **submit** data type displays a submit button used to submit the form to the server; the **value** attribute specifies the button text.

# Page 443

Share Your Experience at Red Ball Pizza

Date of visit  `<input name="visitdate" type="date" />`

Receipt number \*

Order type

Was your service friendly?  Yes  No

Was your order correct?  Yes  No

Was your food hot?  Yes  No

Rate the overall service (0 = poor; 10 = great)

Rate the food quality (0 = poor; 10 = great)

Tell us more about your experience!  
  
`<input name="quality" type="range" min="0" max="10" step="1" />`

Get our newsletter for great coupons and specials!

The `date` data type identifies field input as pertaining to dates; some browsers will display a calendar picker to allow users to easily insert date values.

The `range` data type displays a horizontal range slider between a minimum and maximum value with tick marks set at intervals set by the `step` attribute.

The `reset` data type displays a reset button to reset the Web form to its original values.