*Files: sb\_payments.html – sb\_validate.css – list of instructions.*

<!DOCTYPE html>

<html lang="en">

<head>

<!--

New Perspectives on HTML5 and CSS3, 8th Edition

Tutorial 7

Case Problem 2

Spice Bowl Payment Form

Author:

Date: 05/11/2023

Filename: sb\_payment.html

-->

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<title>Spice Bowl Payment Form</title>

<link href="sb\_reset.css" rel="stylesheet" />

<link href="sb\_styles.css" rel="stylesheet" />

<link href="sb\_forms.css" rel="stylesheet" />

<link href="sb\_validate.css" rel="stylesheet" />

<script src="sb\_script.js"></script>

</head>

<body>

<header>

<img src="sb\_logo.png" alt="The Spice Bowl" />

<form id="login" action="http://www.example.com/sb/login" method="post">

<input type="text" id="userBox" placeholder="username" />

<input type="text" id="pwdBox" placeholder="password" />

</form>

<nav>

<a id="navicon" href="#"><img src="sb\_navicon.png" alt="" /></a>

<ul>

<li><a href="#">Spices</a></li>

<li><a href="#">Recipes</a></li>

<li><a href="#">My Cart</a></li>

<li><a href="#">My Account</a></li>

<li><a href="#">Contact Us</a></li>

</ul>

</nav>

</header>

<section>

<h1>Payment Form</h1>

<form

id="payment" action="http://www.example.com/sb/payment" method="post">

<fieldset id="billing">

<legend>Billing Information (required)</legend>

<!-- FIRST NAME FIELD -->

<label for="fname"> First Name </label>

<input name="fname" id="firstBox" type="text" required="required" />

<!-- LAST NAME FIELD -->

<label for="lname"> Last Name </label>

<input name="lname" id="lastBox" type="text" required="required" />

<!-- STREET 1 FIELD -->

<label for="street"> Street Address</label>

<input name="street" id="streetBox" type="text" required="required" />

<!-- STREET 2 FIELD -->

<label for="street2"> Street Address (2)</label>

<input name="street2" id="streetBox2" type="text" />

<!-- CITY FIELD -->

<label for="city"> City</label>

<input name="city" id="cityBox" type="text" required="required" />

<!-- STATE FIELD -->

<label for="state">State</label>

<input name="state" id="stateBox" type="text" required="required" list="stateList"/>

<datalist id="stateList">

<option value="AL">AL</option>

<option value="AK">AK</option>

<option value="AZ">AZ</option>

<option value="AR">AR</option>

<option value="CA">CA</option>

<option value="CO">CO</option>

<option value="CT">CT</option>

<option value="DE">DE</option>

<option value="FL">FL</option>

<option value="GA">GA</option>

<option value="HI">HI</option>

<option value="ID">ID</option>

<option value="IL">IL</option>

<option value="IN">IN</option>

<option value="IA">IA</option>

<option value="KS">KS</option>

<option value="KY">KY</option>

<option value="LA">LA</option>

<option value="ME">ME</option>

<option value="MD">MD</option>

<option value="MA">MA</option>

<option value="MI">MI</option>

<option value="MN">MN</option>

<option value="MS">MS</option>

<option value="MO">MO</option>

<option value="MT">MT</option>

<option value="NE">NE</option>

<option value="NV">NV</option>

<option value="NH">NH</option>

<option value="NJ">NJ</option>

<option value="NM">NM</option>

<option value="NY">NY</option>

<option value="NC">NC</option>

<option value="ND">ND</option>

<option value="OH">OH</option>

<option value="OK">OK</option>

<option value="OR">OR</option>

<option value="PA">PA</option>

<option value="RI">RI</option>

<option value="SC">SC</option>

<option value="SD">SD</option>

<option value="TN">TN</option>

<option value="TX">TX</option>

<option value="UT">UT</option>

<option value="VT">VT</option>

<option value="VA">VA</option>

<option value="WA">WA</option>

<option value="WV">WV</option>

<option value="WI">WI</option>

<option value="WY">WY</option>

</datalist>

<label for="zip">ZIP/Postal Code</label>

<input

name="zip"

pattern="^\d{5}(-\d{4})?$"

id="zipBox"

type="text"

/>

<!-- COUNTRY FIELD -->

<label for="country">Country</label>

<input

name="country"

value="United States"

id="countryBox"

type="text"

required="required"

/>

<!-- PHONE FIELD -->

<label for="phone">Phone number</label>

<input

name="phone"

id="phoneBox"

type="tel"

pattern="^\d{10}$|^(\(\d{3}\)\s\*)?\d{3}[\s-]?\d{4}$"

required="required"

/>

<!-- CREDIT CARD INFORMATION -->

</fieldset>

<fieldset id="creditCard">

<legend>Credit Card (required)</legend>

<fieldset required="required">

<label class="cardLabel">

<input type="radio" name="cCard" />

<img src="sb\_amex.png" alt="amex" />

</label>

<label class="cardLabel">

<input type="radio" name="cCard" />

<img src="sb\_discover.png" alt="discover" />

</label>

<label class="cardLabel">

<input type="radio" name="cCard" />

<img src="sb\_master.png" alt="master card" />

</label>

<label class="cardLabel">

<input type="radio" name="cCard" />

<img src="sb\_visa.png" alt="visa" />

</label>

</fieldset>

<!-- CARD NUMBER FIELD -->

<label for="cardNumber">Credit Card Number</label>

<input

name="cardNumber"

id="cardBox"

required="required"

pattern="^(?:4[0-9]{12}(?:[0-9]{3})?|5[1-5][0-9]{14}|6(?:011|5[0-9][0-9])[0-9]{12}|3[47][0-9]{13}|3(?:0[0-5]|[68][0-9])[0-9]{11}|(?:2131|1800|35\d{3})\d{11})$"

/>

<!-- EXPIRATION DATE FIELD -->

<label>Expiration Date</label>

<select id="monthList" required="required">

<option value="--Month--">--Month--</option>

<option value="January (01)">January (01)</option>

<option value="February (02)">February (02)</option>

<option value="March (03)">March (03)</option>

<option value="April (04)">April (04)</option>

<option value="May (05)">May (05)</option>

<option value="June (06)">June (06)</option>

<option value="July (06)">July (07)</option>

<option value="August (08)">August (08)</option>

<option value="September (09)">September (09)</option>

<option value="October (10)">October (10)</option>

<option value="November (11)">November (11)</option>

<option value="December (12)">December (12)</option>

</select>

<select id="yearList" required="required">

<option value="-Year-">-Year-</option>

<option value="2020">2020</option>

<option value="2021">2021</option>

<option value="2022">2022</option>

<option value="2023">2023</option>

<option value="2024">2024</option>

</select>

<!-- CSC FIELD -->

<label for="csc">CSC</label>

<input

id="cscBox"

pattern="^\d{3}$"

placeholder="nnn"

required="required"

/>

</fieldset>

<button alt="next" type="submit">

<img src="sb\_button.png" alt="next" />

</button>

</form>

<aside>

<h1>Menu Ideas</h1>

<h2>Salt-Roasted Pecans</h2>

<ul>

<li>2 cups pecans</li>

<li>3 Tbs. butter, melted</li>

<li>1 1/4 tsp. fine sea salt</li>

</ul>

<p>

Preheat oven to 325&deg;. Toss pecans and butter together; add salt

and toss again. Spread in a single layer on a baking sheet. Bake about

15 minutes. Cook on baking sheet.

</p>

<h2>Dried Beef Sticks</h2>

<ul>

<li>5 lbs. ground chuck</li>

<li>5 Tbs. quick salt</li>

<li>3 tsp. mustard seed</li>

<li>2 tsp. granulated garlic</li>

<li>2 tsp. cracked pepper</li>

<li>1 Tbs. hickory smoked salt</li>

<li>1 Tbs. liquid smoke</li>

</ul>

<p>

Mix beef and season. Place in a large covered container for three

days. On third day, shape beef into sticks, and bake at 150&deg; for 8

hours (turn half-way through).

</p>

</aside>

</section>

<footer>

<nav>

<ul>

<li class="newgroup"><a href="#">Home</a></li>

<li><a href="#">Keyword Search</a></li>

<li class="newgroup"><a href="#">Spices</a></li>

<li><a href="#">Seasonings</a></li>

<li><a href="#">Blends</a></li>

<li><a href="#">Salts &amp; Peppers</a></li>

<li><a href="#">Popcorn Seasonings</a></li>

<li><a href="#">Dip Mixes</a></li>

<li><a href="#">Bulk Spices</a></li>

<li><a href="#">Extracts</a></li>

<li><a href="#">Flavorings</a></li>

<li class="newgroup"><a href="#">Spice Jars</a></li>

<li><a href="#">Spice Jar Labels</a></li>

<li><a href="#">Spice Sets</a></li>

<li><a href="#">Mortar &amp; Pestles</a></li>

<li class="newgroup"><a href="#">Cookbooks</a></li>

<li><a href="#">Online Recipes</a></li>

<li><a href="#">Forums</a></li>

<li class="newgroup"><a href="#">My Account</a></li>

<li><a href="#">Checkout</a></li>

<li><a href="#">Order History</a></li>

<li><a href="#">Shipping Info</a></li>

<li class="newgroup"><a href="#">Tech Support</a></li>

<li><a href="#">Hours</a></li>

<li><a href="#">Contact Us</a></li>

</ul>

</nav>

</footer>

</body>

</html>

/\*

New Perspectives on HTML5 and CSS3, 8th Edition

Tutorial 7

Case Problem 2

Spice Bowl Payment Form Validation Style Sheet

Author:

Date: 5/12/2023

Filename: sb\_validate.css

\*/

/\* Validation Styles \*/

input:not([type="radio"]):focus {

background-color: rgb(255,218,165)

}

input:not([type='radio']):focus:valid {

background: rgb(215, 255, 215) url(sb\_valid.png) no-repeat right;

}

input:not([type='radio']):focus:invalid{

background: rgb(255, 245, 215) url(sb\_invalid.png) no-repeat right;

}

**Summary**

*The Spice Bowl* - Rita Sato is the manager of the web development team for The Spice Bowl, an online grocery store specializing in gourmet spices. She has asked you to create web forms for the site. You will start your work by developing a payment form used to collect billing and credit data from the store’s customers. The form should include validation tests for credit card numbers to ensure that the card numbers match the correct credit card number patterns. The page should also include a form in which users can log into their Spice Bowl account. *Figure 7–67* shows a preview of the page you will create for Rita.

[](https://cdn.filestackcontent.com/GVvn3MvwRwOdvdI6huRO)*Figure 7-67*

## Document Setup

Open the sb\_payment.html and sb\_validate.css files and enter your **name** and the **date** in the comment section of each file. Return to the sb\_payment.html file and add links to the sb\_forms.css and sb\_validate.css style sheet files to the document head. Add a script element to load the sb\_script.js file.

## Create the Login Form

The page will contain a form in which customers can log into their account. Directly after the sb\_logo.png image, insert a form element with the ID login. Have the form use the action at http://www.example.com/sb/login via the post method. Within the login form, insert the following fields and controls:

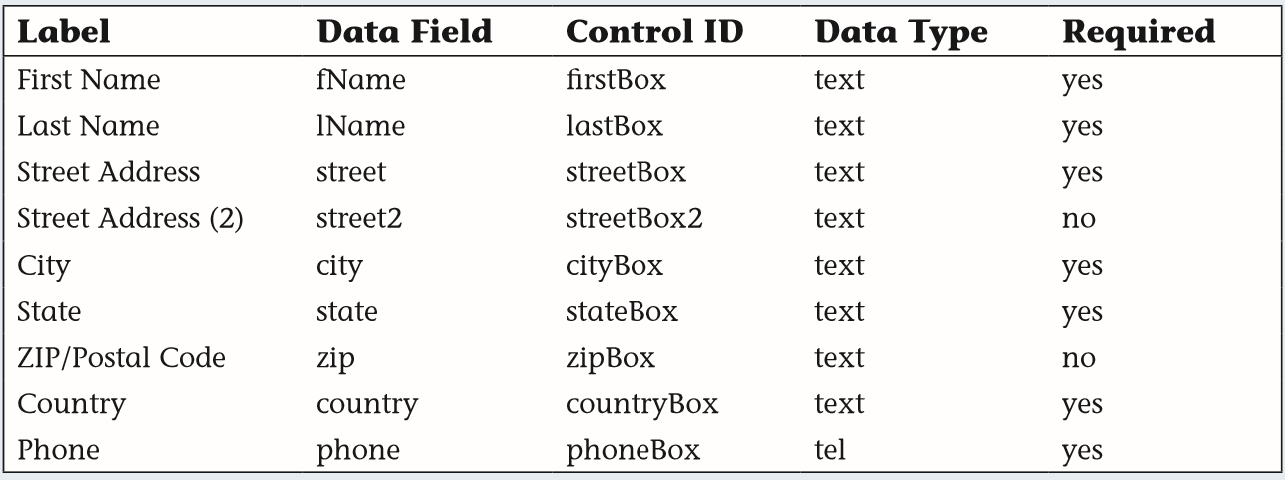
1. A text input box with the ID userBox for the username field. Add the placeholder text **username**.
2. A text input box with the ID pwdBox for the password field. Add the placeholder text **password**.

## Create the Payment Form

Next, insert a payment form. Directly below the **Payment Form** h1 header, insert a form element that employs action at http://www.example.com/sb/payment via the post method. Assign the web form the id payment.

Insert a field set with the id billing to the payment form. Add the legend **Billing Information (required)** to the field set.

Within the billing field set, add the labels and input boxes specified in Figure 7–68. Note that none of the input boxes contain placeholder text. You do not need to enclose the label or input elements within div elements.

[](https://cdn.filestackcontent.com/le2NdknRIKbbuB3O89Su)Figure 7-68

Create a data list with the ID stateList containing the two-letter abbreviations of all the states. (You can use the list of abbreviations in the sb\_state.txt file.) Apply the data list to the stateBox input text box.

The text of the zip field should follow the ^\d{5}(-\d{4})?$ regular expression pattern. The text of the phone field should follow the ^\d{10}$|^(\(\d{3}\)\s\*)?\d{3}[\s-]?\d{4}$ pattern. Note that both regular expression patterns can be found in the sb\_regex.txt file.

3/5

## Create the Payment Form Continued...

Set the default value of the country field to **United States**.

Create a field set with the ID creditCard, which you use to insert credit card fields. Add the legend **Credit Card (required)**.

Within the creditCard field set, insert another field set containing four label elements, with each label element belonging to the cardLabel class. Within each of the four label elements, insert an option button from the cCard field with the value **amex**, **discover**, **master**, and **visa**. Make cCard a required field. Follow each option button with an image element containing the image of its corresponding credit card image using the sb\_amex.png, sb\_discover.png, sb\_master.png, and sb\_visa.png files.

After the creditCard field set, insert a label containing the text **Credit Card Number** associated with the cardBox input control. Create an input text box with the ID cardBox for the cardNumber field. Make the field cardNumber required and have the field value follow the regular expression pattern for credit card numbers (using the regular expression in the sb\_regex.txt file). Create the **Expiration Date** label associated with the monthList control. Add a selection list with the ID monthList for the cardMonth field. Make the cardMonth field required. Populate the selection list with the option text **--Month--**, **January (01)**, **February (02)** and so forth up to **December (12)**. For the **--Month--**, set the field value to an empty text string. For the month options, insert the month value from **01** up to **12**.

Add a selection list with the ID yearList for the cardYear field. Make the cardYear field required. Populate the selection list with the option text **--Year--**, **2020**, **2021**, **2022**, **2023**, and **2024**. For the **--Year--**, set the field value to an empty text string and, for the year options, insert the 4-digit year value.

Create a label with the text **CSC** associated with the cscBox. Add a text input box with the ID cscBox for the required csc field. Have the csc field value follow the regular expression pattern ^\d{3}$. Set the maximum length of the field value to **3** characters and display the placeholder text **nnn**.

After the creditCard field set, insert a button element of the submit type. Within the button, insert the sb\_button.png inline image with the alternate text **next**.

The text for the button is part of the button image, so the value attribute is not needed.

**Validation Styles**

Open the *sb\_validate.css* file to design the validation styles for the web form. Within the "Validation Styles" section, insert the following style rules to perform inline validation:

1. For every input element that is not a radio type and that has the focus, change the background color to **rgb(255, 218, 165)**.

Use the attribute selector input:not([type='radio']) to select input elements that are not radio types.

1. For every input element that is not a radio type and that has the focus with a valid value, change the background color to **rgb(215, 255, 215)** and display the image file *sb\_valid.png* with no tiling in the right edge of the input box.
2. For every input element that is not a radio type and that has the focus with an invalid value, change the background color to **rgb(255, 245, 215)** and display the image file *sb\_invalid.png* with no tiling in the right edge of the input box.

Lastly, open *sb\_payment.html* in the browser preview. Verify that you cannot submit the form without all required fields entered in the proper format. Verify the validation checks for the credit card number by confirming that the form rejects the following credit card number 6012123456789019 (which does not follow a valid card number pattern). Further verify that the form accepts the following credit card number 6011123456789019 (which follows a valid card number pattern).