**CSC 5750 – Principles of Web Technology**

**Homework 3 – [your name]**

**25 points – Due October 24, 6pm**

**Late deadline is October 26, 11:59pm, but 20% off**

**a)** Save this document with your name and the homework number somewhere in the file name.

**b)** Type/paste your answers into the document.

**c)** Gather the following files into a ZIP file:

● This document

● Main page (.HTML) and associated images

● Admin page (.HTML)

● Login page (.HTML)

● Cashier page (.HTML)

● CSS file (.CSS)

● Any other optional files specific to your implementation

**d)** Submit the ZIP file to the Blackboard item where you downloaded this document.

You've been hired again by the local restaurant that you created a web site for in Homework 1. Enhance that web site per the following:

**Main page**

Edit the **Main page** from Homework 1 to connect it to the Styles page (described below). Move any inline styles from the main page to the styles page. Add a link in the menu to the Login page (described below).

**Admin page**

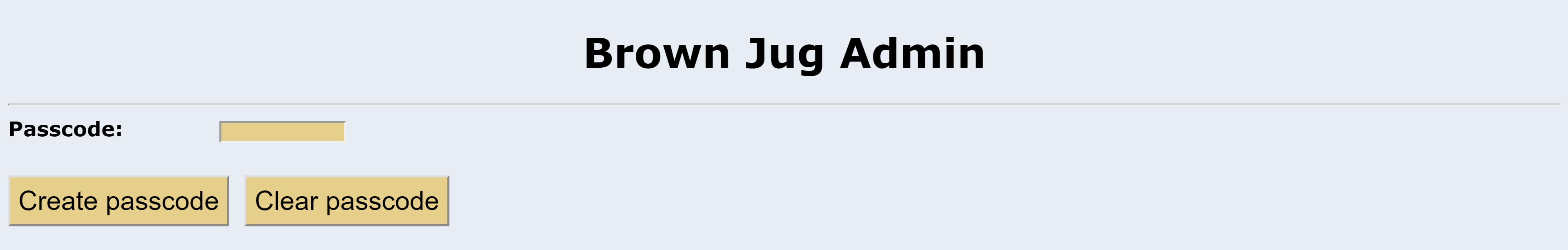
Add an **Admin page** to enable restaurant staff to set and clear a passcode. Use local storage to store the passcode. The page has the following controls:

● Passcode label and password-style text box

● Create passcode button – triggers code to check the passcode. If blank, it shows an error message. If not blank, it saves the passcode in local storage.

● Clear passcode button – triggers code to remove the passcode from local storage.

The page is accessed separately from all other pages.



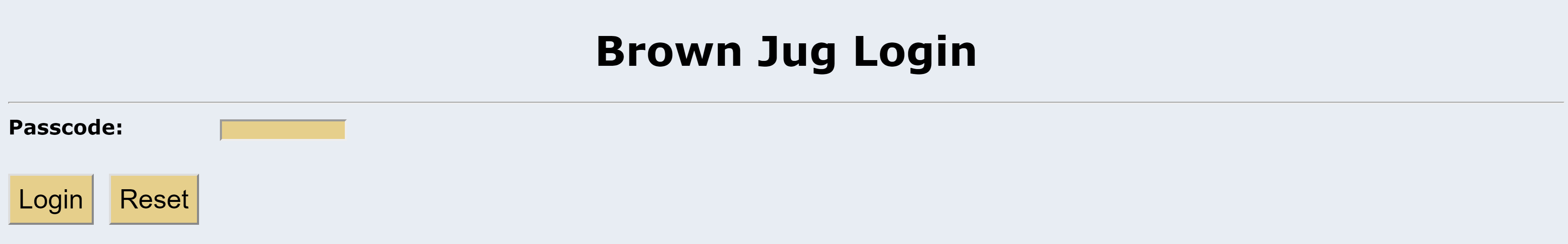
**Login page**

Add a **Login page** to enable restaurant staff to log in the Cashier page (described below). The page has the following controls:

● Passcode label and password-style text box

● Login button – triggers code to check the passcode. If blank, it shows an error message. If not blank, it checks if it is valid. If invalid, it shows and error message. If valid, it opens the Casher page.

● Reset button – triggers code to clear the passcode text box.



**Cashier page**

Add a **Cashier page** to enable restaurant staff to enter a customer sale and calculate its totals. The page has the following controls:

● Entrée label and combo box – data for the combo box comes from a global 7x2 array containing the name and cost for six entrée items (the first item is “Select entree ...", 0). An onload body event reads the array and builds the combo box.

● Drink label and combo box – data for the combo box comes from a global 4x2 array containing the name and cost for three drink items (the first item is “Select drink ...", 0). An onload body event reads the array and builds the combo box.

● Tip label and number-style text box – the text box ranges from 0 to 100 in increments of 5. The default value is 15.

● Subtotal label and read-only text box – contains the sale subtotal (entrée + drink).

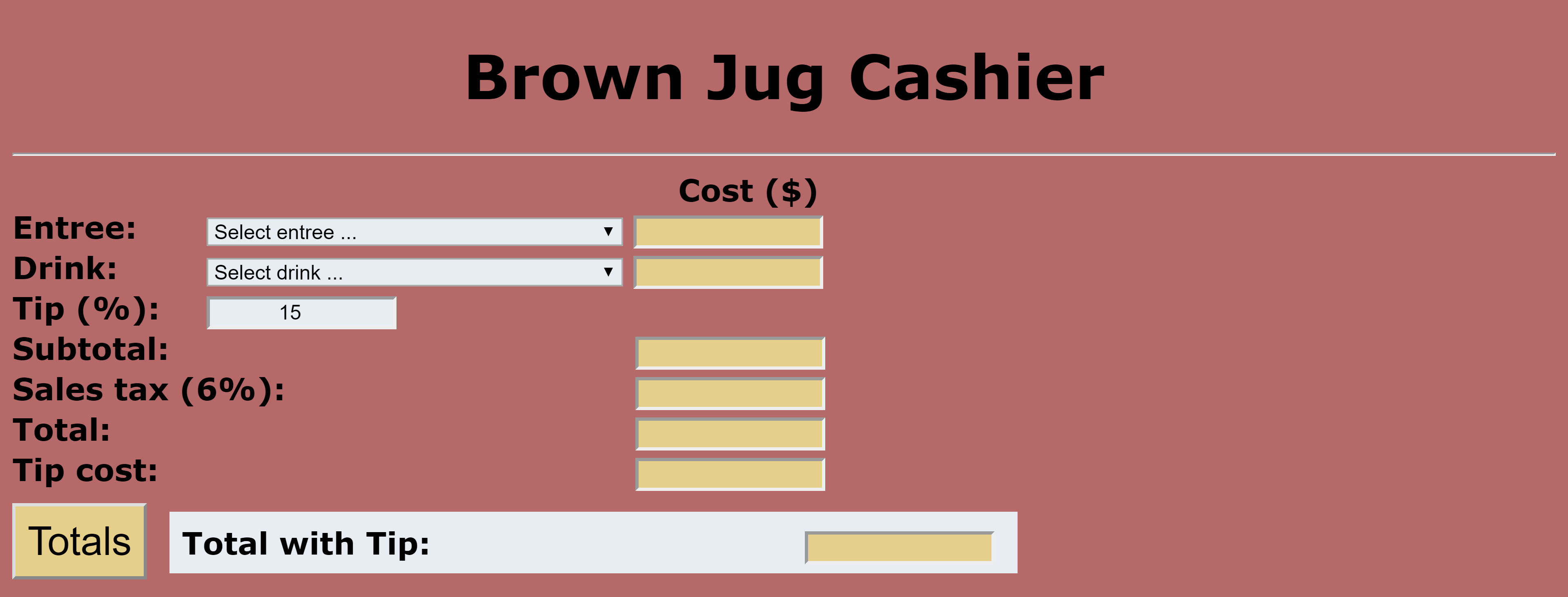
● Sales tax label and read-only text box – contains the sale tax (subtotal \* 6%).

● Total label and read-only text box – contains the sale total (subtotal + sales tax).

● Tip cost label and read-only text box – contains the tip cost (total \* tip%).

● Total with Tip cost label and read-only text box – contains the total with tip cost (total + tip cost).

● Totals button – triggers code to get staff inputs. If an entrée wasn’t selected, it shows an error message. If a drink wasn’t selected, it shows an error message. If both items were selected, it performs a lookup in the entrées array to get and show the entrée cost. It also performs a lookup in the drinks array to get and show the drink cost. It then calculates and shows sale totals. Show all numbers to two decimal places.



**Styles page**

Add a **Styles page** to hold external styles. Define styles for at least the following controls:

● Different Body styles for each page.

● Button style.

● Header styles.

● Different Input (text box) styles.

● Different Label styles.

● Select style.

● Span style.

● Table style (for Main page).

**Other code**

For sale totals, add a roundNumber function that rounds off all calculations to two decimal places.

**Screenshots**

Include the following screenshots in this document:

● Admin page – no passcode specified.

● Login page – no passcode specified.

● Login page – incorrect passcode.

● Cashier page – no entrée specified.

● Cashier page – no drink specified.

● Cashier page – entrée-drink combination 1 with 15% tip.

● Cashier page – entrée -drink combination 2 with 20% tip.

● Cashier page – entrée -drink combination 3 with 25% tip.

**Admin page**

*[your Admin page HTML code here]\**

**If possible, format your code like this:**

**Font “Courier New”**

**Font size “9”**

**Bold**

*[your Admin page screenshots here]\*\**

**Login page**

*[your Login page HTML code here]\**

**If possible, format your code like this:**

**Font “Courier New”**

**Font size “9”**

**Bold**

*[your Login page screenshots here]\*\**

**Cashier page**

*[your Cashier page HTML code here]\**

**If possible, format your code like this:**

**Font “Courier New”**

**Font size “9”**

**Bold**

*[your Cashier page screenshots here]\*\**

**CSS file**

*[your CSS code here]\**

**Main page**

*[your Main page HTML code here]\**

**If possible, format your code like this:**

**Font “Courier New”**

**Font size “9”**

**Bold**

*[your Main page screenshots here]\*\**

**If possible, format your code like this:**

**Font “Courier New”**

**Font size “9”**

**Bold**

\* **Copying-and-pasting web page code to a Word document**

1) From the HTML editor window, press **CTRL-A** and press **CTRL-C**.

2) From within the Word document, press **CTRL-V**.

\*\* **Copying-and-pasting application output to a Word document**

1) From the web page, maximize the browser window.

2) From the browser window, press **ALT-PrintScreen**.

3) From within the Word document, press **CTRL-V**.