Activity: Rate-a-Restaurant

The owner of a fast food restaurant is trying to understand how people feel about the service they're receiving. The owner decides to implement a rating system. As customers finish interacting with their cashier, and pick up their food, they are asked to rate their cashier and over-all experience. The owner knows that customers don't have a lot of time, so they want this to be as easy as possible for the customers. Therefore, all a customer has to do for each rating is hit one of four buttons; -2, -1, +1, +2. This should let the owner get an average rating for their cashiers and for their overall service!

1) Creating the Rating System

In order for the owner's rating scheme to work, they need an interface that customers can use. Luckily, they know that you have some software development skills, and they've called you up to help them out. They want you to design a website for them that will accept, store, and track the ratings that come in. For now, all you need to generate is a proof-of-concept.

Essentially, the owner wants you to create a website with two similar sections and a third results section. The top section should have a picture of the employee (cashier), and should ask the customer to rate their experience with the cashier. Under the picture and prompt should be the four rating buttons. The second section (below) should be very similar, but has no picture. The prompt should ask the customer to rate their overall experience. Finally, there should be a "Display Ratings" button at the bottom of the page. When clicked, it should display the numerical ratings for the cashier and for the restaurant overall. It should also change its text to "Hide Ratings", so that clicking it repeatedly toggles the numbers appearing and disappearing.

Step 1: Create the Website

Our first task is to create the website files. There is no starter code for this project, so you'll need to create a new folder that contains the `index.html`, `style.css`, and `script.js` files.

Once you have those 3 files in place, go ahead and run `npm install` and `npm start`. This should open up a blank webpage in your browser. Let's start filling it up!

Step 2: Build the Website

To build the website, we'll be first working in the `index.html` file. The first thing we need to do is to give our HTML file the basic structure we always work with. Feel free to copy the following into your HTML file to start:

```
<!DOCTYPE html>
<html>
  <head>
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width">
 <title>Restaurant Rating</title>
 <link href="style.css" rel="stylesheet" type="text/css" />
  </head>
  <body>

 <script src="script.js"></script>
  </body>
</html>
```

Click here to copy

Let's start with the top section of our website. We first need an image of our cashier. Use this link to get the image that we'll be using. We are going to add this image using an `img` tag. It will be placed at the top of the `body` section. If your image is too big (it probably will be), feel free to add an id to the image, and use your CSS file to change the size. If the id you gave the image was "cashier", the CSS would look like the following:

```
#cashier{
 height:200px
  }
```

Click here to copy

Next we need a text prompt. Go ahead and add a `p` tag with the text, "Please rate your experience with your cashier today".

Finally, we need 4 buttons to make our ratings (we can do this using `button` tags). They should be 1, 2, 3, 4. Be sure to give each button a unique id, such as "cashier-1", "cashier-2", etc.

At this point, our website code should look like the following:

```
<!DOCTYPE html>
<html>
  <head>
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width">
 <title>Restaurant Rating</title>
 <link href="style.css" rel="stylesheet" type="text/css" />
  </head>
  <body>
 <img src="_assets/cashier.jpg" id="cashier">
 <p>Please rate your experience with your cashier today</p>
 <button type="button" id="cashier-1">1</button>
 <button type="button" id="cashier-2">2</button>
 <button type="button" id="cashier-3">3</button>
 <button type="button" id="cashier-4">4</button>
 <script src="script.js"></script>
  </body>
</html>
```

Click here to copy

Now we need to add the second part of our website. This will basically be copying and pasting several lines of code from the first section, then making a few edits. Copy the `p` tag and all of the buttons, and paste them below your current code. Now change the second message to say, "Please rate your overall experience today". Each of the button id's should also change to be "overall-#" (remember, each id must be unique).

Lastly, we need to add the "Display Ratings" button to the website. We want this to be on its own line below the rating butons, so we'll add a break (`br`) tag right before it.

When you've finished building the website, it should look like this:

```
<!DOCTYPE html>
<html>
  <head>
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width">
 <title>Restaurant Rating</title>
 <link href="style.css" rel="stylesheet" type="text/css" />
  </head>
  <body>
 <img src="_assets/cashier.jpg" id="cashier">
 <p>Please rate your experience with your cashier today</p>
 <button type="button" id="cashier-1">1</button>
 <button type="button" id="cashier-2">2</button>
 <button type="button" id="cashier-3">3</button>
 <button type="button" id="cashier-4">4</button>

 <p>Please rate your overall experience today</p>
 <button type="button" id="overall-1">1</button>
 <button type="button" id="overall-2">2</button>
 <button type="button" id="overall-3">3</button>
 <button type="button" id="overall-4">4</button>

 <br/>
 <button type="button" id="show-hide">Display Ratings</button>
 <script src="script.js"></script>
  </body>
</html>
```

Click here to copy

Step 3: Make Rating Buttons Functional

Your HTML is nearly complete but doesn't do anything yet and there might be a little more to add. We want to make sure that each time a rating button is clicked, it affects either the cashier rating or the overall rating. To do this, we'll create an event listener that handles click events. It should add a specific value to the cashier or overall score depending on what is clicked.

The click listeners for each button are going to look very similar. We will walk you through how to do the first two buttons, and then you will be responsible for making the other 6 rating buttons functional.

First, use the `document.getElementById()` method to query for the button we want - in this case, the button with the "cashier-1" id. Store the result of the query in a variable called `cashierScoreBtn1`. To do this, pass in the argument "cashier-1".

Next, we're going to add an event listener to this button. We'll use the following code to do this.

```
cashierScoreBtn1 = document.getElementById("cashier-1")
cashierScoreBtn1.addEventListener("click", function() {
  //code
})
```

Click here to copy

Note that each cashier button click listener should do almost the same thing to update the cashier's score. Before we move on, we need to create three variables outside of the function we just created, to keep track of this score. We will call these:

- `cashier` - This variable will keep track of the average score for our cashier.
- `c_tot` - This variable will keep track of the total score for our cashier.
- `c_num` - This variable will keep track of the number of ratings our cashier has received.

Make sure to set each of these variables to 0.

Now, inside of our click listener, we need to update the score for our cashier each time someone clicks the button. The button we're currently working on is the 1-point button, so each time it is clicked, the `c_tot` variable should be updated by adding the value 1. Then, the number of reviews received should be incremented, and finally the average should be calculated. To make sure that we're doing it right, also include a `console.log()` of the running average for our cashier inside your click listener.

Your JavaScript code should look like this:

```
//create variables to hold the cashier scores
let cashier = 0
let c_tot = 0
let c_num = 0

//cashier button 1
cashierScoreBtn1 = document.getElementById("cashier-1")
cashierScoreBtn1.addEventListener("click", function() {
 c_tot = c_tot + 1 //adding 1 to score
 c_num = c_num + 1
 cashier = c_tot / c_num
 console.log(cashier)
})
```

Click here to copy

Make sure to test your code before moving forward. Each time you click the "1" rating button below the prompt for the cashier, a new average score should be outputted to the console. Because we are only adding a rating score of 1 each time, the average score will always be 1 for now.

Next, we're going to add another click listener, this time to the cashier's "2" rating button. The steps are very similar to the "1" button, with the small change that we're adding 2 tot he rating instead of 1, and we need to query for a different Id. At this point, your JavaScript code should look like this:

```
//create variables to hold the cashier scores
let cashier = 0
let c_tot = 0
let c_num = 0

//cashier button 1
cashierScoreBtn1 = document.getElementById("cashier-1")
cashierScoreBtn1.addEventListener("click", function() {
 c_tot = c_tot + 1 //adding 1 to score
 c_num = c_num + 1
 cashier = c_tot / c_num
 console.log(cashier)
}

//cashier button 2
cashierScoreBtn2 = document.getElementById("cashier-2")
cashierScoreBtn2.addEventListener("click", function() {
 c_tot = c_tot + 2 //adding 2 to score
 c_num = c_num + 1
 cashier = c_tot / c_num
 console.log(cashier)
})
```

Click here to copy

Again, make sure that you test your code before moving forward. At this point, if you click the "1" button, a rating of "1" should be logged. If you then click the "2" button, a "1.5" rating should be logged. As you click the "1" and "2" buttons in different sequences, check that the averages are what you expect.

Step 4: Add the Rest of the Event Listeners

At this point, we have shown you how to add 2 of the event listeners for the cashier's rating. Adding the "3" and "4" rating score buttons will be very similar, so do that now.

Once you have your 4 rating buttons working for the cashier, making the four rating buttons work for the overall score should be very straightforward. You will copy and paste all of the JS code that you created for your cashier rating buttons, including the "cashier", "c_tot", and "c_num" variables, and you will edit it to reflect the overall score instead of the cashier score. To do this, you will need to first change each of the button variables from "cashierScoreBtn#" to "overallScoreBtn#". Next, change each of the Ids that you query for from "cashier-#" to "overall-#". Lastly, change each of the "cashier", "c_tot", and "c_num" variables to "overall", "o_tot", and "o_num" everywhere that they appear.
When you are finished, you should have 8 click event listeners, and the ratings for each category should be logged when a rating button for that category is pressed.

Step 5: Show and Hide the Average Scores

The last piece of the proof-of-concept website that the restaurant owner has asked you for is the ability to see the average cashier and overall score. They don't want the score to be outputted or logged every time a rating button is pressed, but they do want to be able to check on the score. Therefore, they have asked you to implement a "show/hide" button. We've already put this button onto our webpage, but it doesn't do anything yet.
The first thing we need to do is add a click listener to our "Display Ratings" button. To do this, we're going to go add a unique ID to the button, and use the `document.selectElementById()` method to select it in our JavaScript, just like we've done for each of the other buttons.

Once we have our button selected, we need to make it do something. What it does is going to depend on what it displays. First, we need to add a `div` with two paragraphs to the bottom of our webpage in our HTML, one each for the cashier and for the overall section.

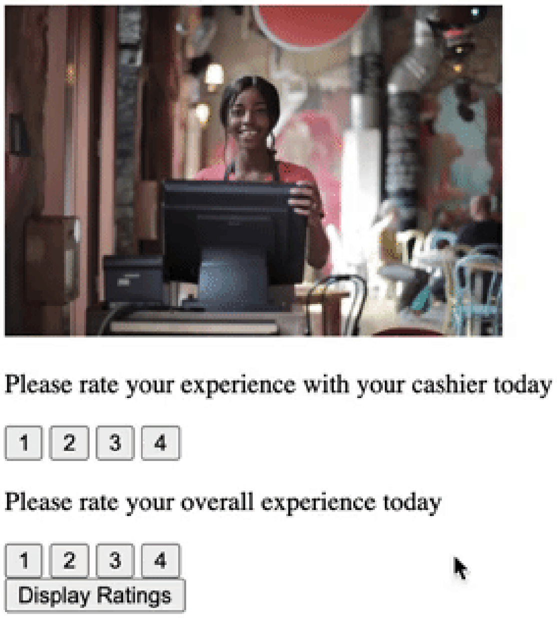
This is the trickiest part of the whole activity. See how far you can get with just the skills you already have, and try not to look at the solution code below until you've tried your best.

After adding the new `div` to our page, our HTML should look like this:

```
<!DOCTYPE html>
<html>
  <head>
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width">
 <title>Restaurant Rating</title>
 <link href="style.css" rel="stylesheet" type="text/css" />
  </head>
  <body>
 <img src="_assets/cashier.jpg" id="cashier">
 <p>Please rate your experience with your cashier today</p>
 <button type="button" id="cashier-1">1</button>
 <button type="button" id="cashier-2">2</button>
 <button type="button" id="cashier-3">3</button>
 <button type="button" id="cashier-4">4</button>

 <p>Please rate your overall experience today</p>
 <button type="button" id="overall-1">1</button>
 <button type="button" id="overall-2">2</button>
 <button type="button" id="overall-3">3</button>
 <button type="button" id="overall-4">4</button>

 <br/>
 <button type="button" id="show-hide">Display Ratings</button>
 <div id="show-scores" style="display:none">
   <p>The average cashier rating is <span id="cashierScore">0</span></p>
   <p>The average overall rating is <span id="overallScore">0</span>.</p>
 </div>

 <script src="script.js"></script>
  </body>
</html>
```

Click here to copy

Make sure that the score for each section is placed in its own `span` tag with a unique ID, so that you can change the contents later. When our button is clicked, it will first update the two paragraphs to contain the current average score for the cashier and the average overall score. Then, if the button says "Display Ratings", we're going to change the `style.display` attribute of the div to be 'block', and the button should say "Hide Ratings". Otherwise, if the button says "Hide Ratings", we'll change the `style.display` attribute of the div to be 'none'. The following gif may be helpful in understanding the desired behavior.



Please rate your experience with your cashier today

Please rate your overall experience today

Source: Giphy

One thing that will need to change in each of the event listeners that you've created so far is the `console.log` command. Becuase the manager doesn't want the rating logged with each button click, but he *does* want the ratings in the bottom div to be updated, each `console.log` will need to be changed to update the appropriate span tag instead. This can be done by updating the `textContent` of the appropriate span tag in each rating click listener. For example, if the span tag for the cashier has an Id of "cashierScore", and we stored the span in a variable called `cashierScore` (as shown below), we could use the command `cashierScore.textContent = cashier` to update the span with our new score.

The Jasvascript should look like this:

```
//create variables to hold the cashier and overall scores
let cashier = 0
let c_tot = 0
let c_num = 0
let overall = 0
let o_tot = 0
let o_num = 0

let cashierScore = document.getElementById("cashierScore")
let overallScore = document.getElementById("overallScore")

//cashier button 1
cashierScoreBtn1 = document.getElementById("cashier-1")
cashierScoreBtn1.addEventListener("click", function() {
 c_tot = c_tot + 1 //adding 1 to score
 c_num = c_num + 1
 cashier = c_tot / c_num
 cashierScore.textContent = cashier
})

//cashier button 2
cashierScoreBtn2 = document.getElementById("cashier-2")
cashierScoreBtn2.addEventListener("click", function() {
 c_tot = c_tot + 2 //adding 2 to score
 c_num = c_num + 1
 cashier = c_tot / c_num
 cashierScore.textContent = cashier
})

//cashier button 3

//cashier button 4

//overall button 1

//overall button 2

//overall button 3

//overall button 4

let cashierScore = document.getElementById("cashierScore")
let overallScore = document.getElementById("overallScore")

let showHide = document.getElementById("show-hide")
showHide.addEventListener('click',function(){
  let scoresDiv = document.getElementById("show-scores")

  cashierScore.textContent = cashier
  overallScore.textContent = overall

  let btnText = showHide.textContent
  if(btnText === "Display Ratings"){
 scoresDiv.style.display = "block"
 showHide.innerHTML = "Hide Ratings"
  }
  else{
 scoresDiv.style.display = "none"
 showHide.innerHTML = "Display Ratings"
  }
})
```

Click here to copy

And we're done! The proof-of-concept website is complete and can be handed to our restaurant manager.

Bonus Content

If you're ready to make this website more interesting, try the following steps:

- Make the styling of the page more professional by adjusting the CSS of the page.
- Include multiple cashier's, each with their own rating score.
- (Advanced) Make a loop to go over the buttons instead of having individual event listeners.