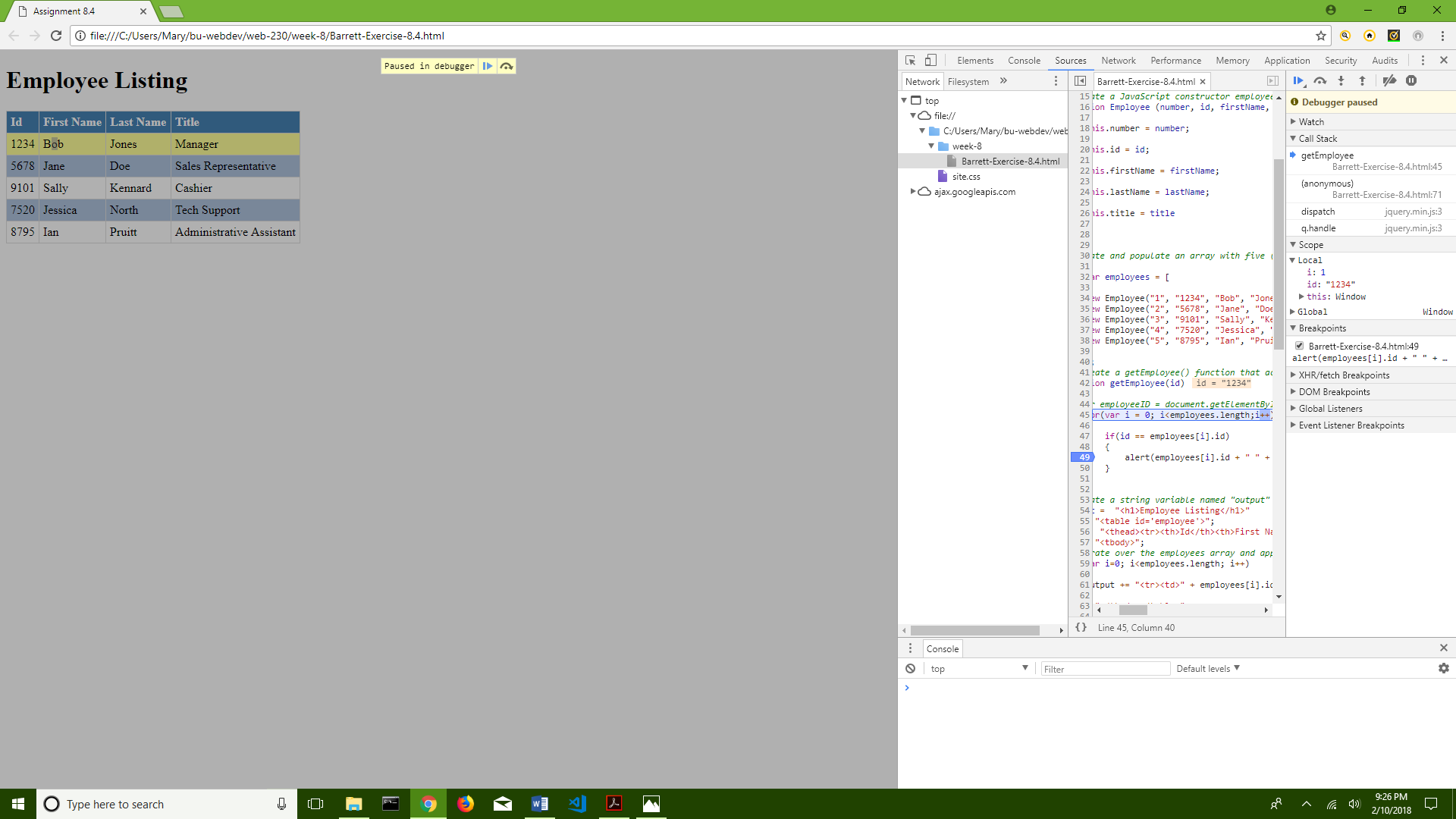
Mary Barrett

WEB 230

Assignment 9.4

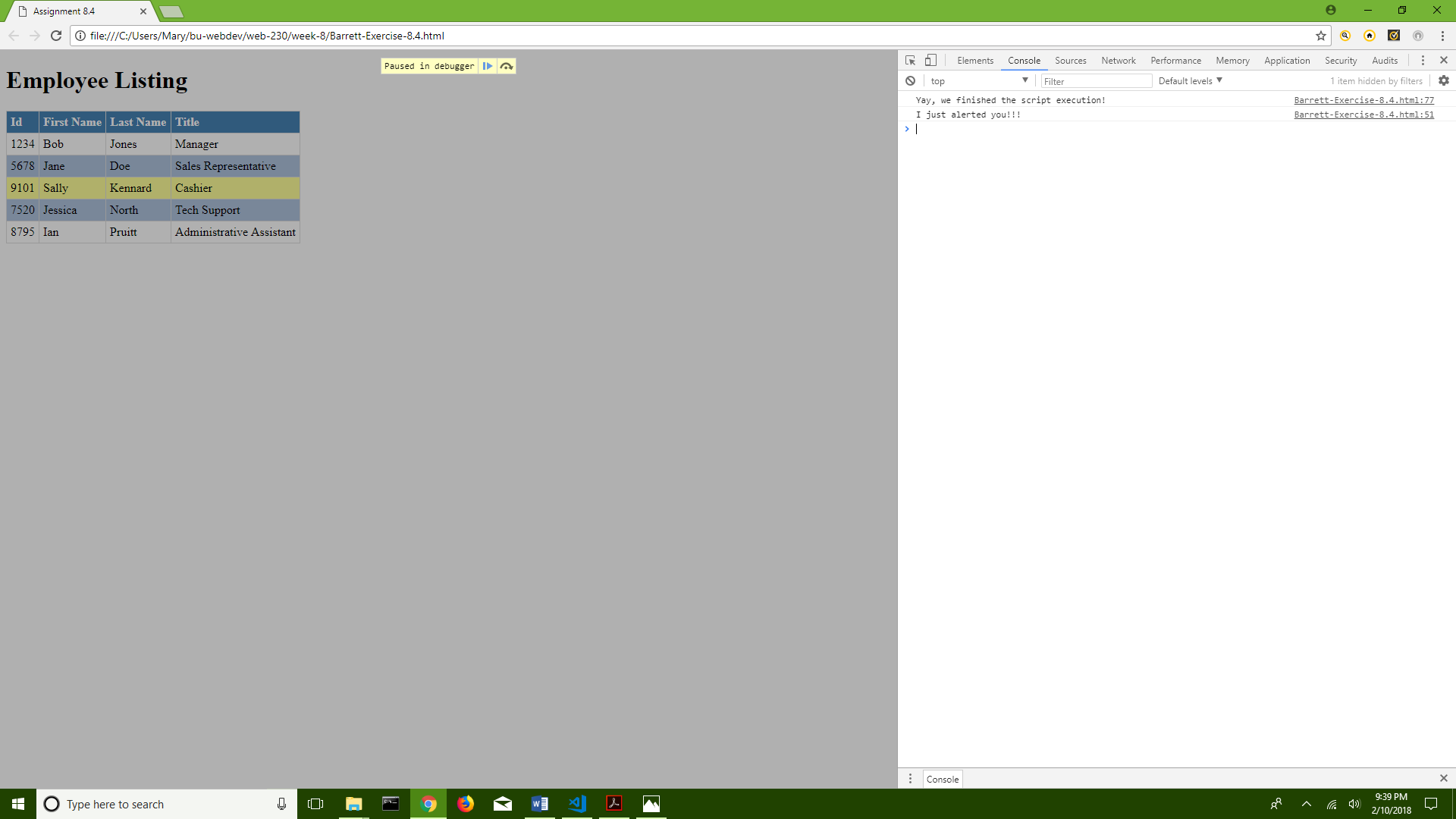
**How to set break points and "walk-through" a JavaScript application using Google Chrome:**

1. Open the website using Google Chrome and press F12 to open DevTools
2. Click on the ‘Sources’ tab near the top of the DevTools window
3. Navigate through the files in the left DevTools column until you reach the correct file and click on the file name
4. In the right DevTools column, select a line you wish to pause at and click the corresponding line number
5. You have just set a break point
6. Outside the DevTools, click on something on the screen that will activate the code, like a button
7. The code will stop at the designated break point
8. To ‘walk-through’ the JavaScript code, find the small tool that appears over the actual webpage and click on the button on the far right of the tool that looks like a curved arrow over a dot
9. Click the button as many times as needed to view the correct steps
10. Your screen should look like this during steps 8 and 9, though you can see evidence of all previous steps on this screenshot



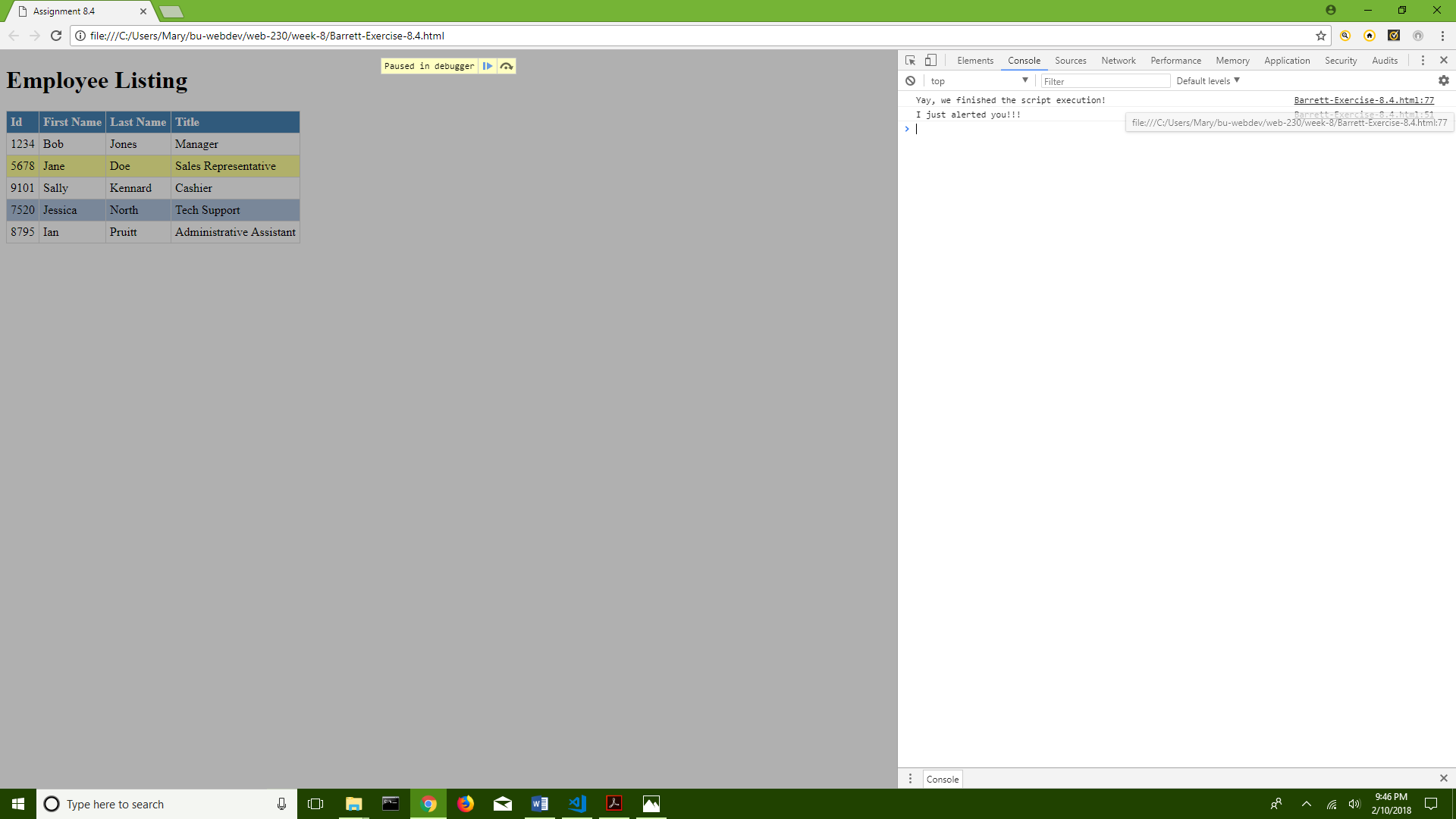
**How to** **log and read messages from the JavaScript console window on Google Chrome:**

1. Open the website using Google Chrome and press F12 to open DevTools
2. Click on the ‘Console’ tab near the top of the DevTools window
3. Open your coded file and enter a console.log() message
4. Save the file and go back to the Google Chrome DevTools
5. Refresh the page, and the message will come up in the ‘Console’ tab
6. You can add the console.log() entry anywhere in the code. If you add it after an alert, the console.log() will come up in the ‘Console’ tab when the alert is activated.
7. Your screen should look like this during step 6, though you can see evidence of all previous steps on this screenshot



**How to select HTML elements in the JavaScript console window in Google Chrome:**

1. Finish performing the steps for how to log and read messages from the JavaScript console window
2. Make sure you are on the ‘Console’ tab near the top of the DevTools window and can see the console.log() entries
3. On the far right of the ‘Console’ tab, you will see links to the html elements for those console.log() entries
4. Click on one of them. You have just selected a HTML element!
5. Your screen should look like this just before you click on the HTML element



**How to simulate CSS states (hover, focus) on live HTML elements in Google Chrome:**

1. Open the website using Google Chrome and press F12 to open DevTools
2. Click on the ‘Elements’ tab near the top of the DevTools window
3. In the right column of the DevTools, make sure the ‘Styles’ tab is open
4. Click on the button called ‘:hov’ located just underneath the right column tabs
5. You will several check-boxes that allow you to simulate various CSS states
6. In the left column, navigate through the code and select a portion you wish to simulate
7. In the right column, check-mark any state you wish to simulate
8. Your screen should look like this during step 7, though you can see evidence of all previous steps on this screenshot

