Front End Web Lab Project 1

Using **no external JavaScript libraries or frameworks**, Build a simple tip calculator.

Narrative:

Provide a simple app that allows one to calculate an amount of a tip to apply to a bill.

Enter the total amount of the bill.

User selects 10%, 15%, 20% and it displays the amount that should be left as a tip, and a total to be paid (bill amount plus tip).

Optional Iteration 1:

Store the selected preferred tip amount in web storage and retrieve its value at application start.

Optional Iteration 2:

Add an option to add a custom tip amount.

Optional Iteration 3:

Make an option to split the bill *n* ways, showing the amount due for each person.

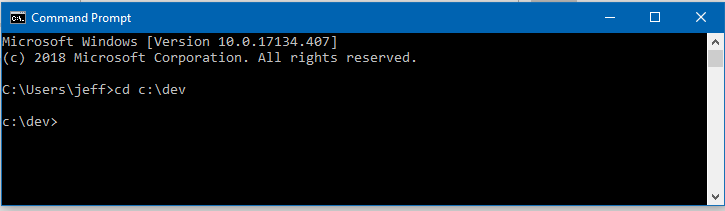
Optional Iteration 4:

Unit test all the application behavior.

# Setting up our Development Environment

Steps:

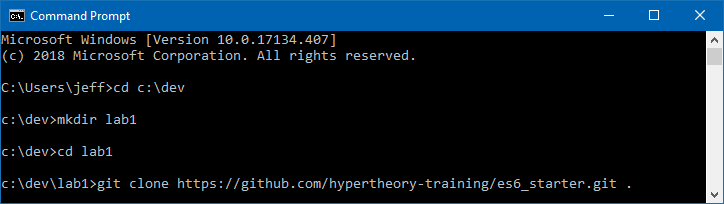
* Open a windows command prompt.
  + Hit the windows key, then type “cmd”
* In the command prompt window, change directory into c:\dev



* Create a directory in that folder called “lab1”, and then enter that directory and issue the following command:

git clone https://github.com/hypertheory-training/front-end-web-100-starter.git .

Don’t forget the period at the end. That means “put the files you pull right here”.



* After the files are retrieved, type:

code .

Which means “open Visual Studio Code in this directory”.

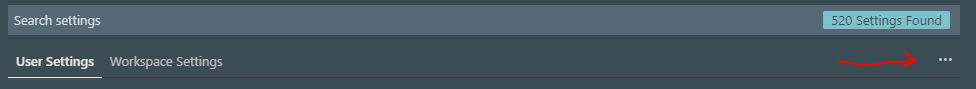
* You can then close your console window.

Next, set up your Visual Studio Code settings. You can use mine at <https://gist.githubusercontent.com/JeffryGonzalez/e3252ff3e476339811d8b51f73df9121/raw/7dfca22852b5ab0b0026717285a424a742caf27e/settings.json> as a starting point.

Some of the settings might be underlined in green, because they refer to extensions we haven’t installed for this project. You can either remove them, or ignore them – they won’t cause a problem.

To get to your settings in Visual Studio Code, hit the F1 function key and type “user settings”. Look for the option in the dropdown called “Preferences: Open User Settings”

A user interface is displayed showing all the settings you can configure. You can use the interface to find the settings in my file, or you can click the ellipses (…) underneath the search bar at the top of the UI.

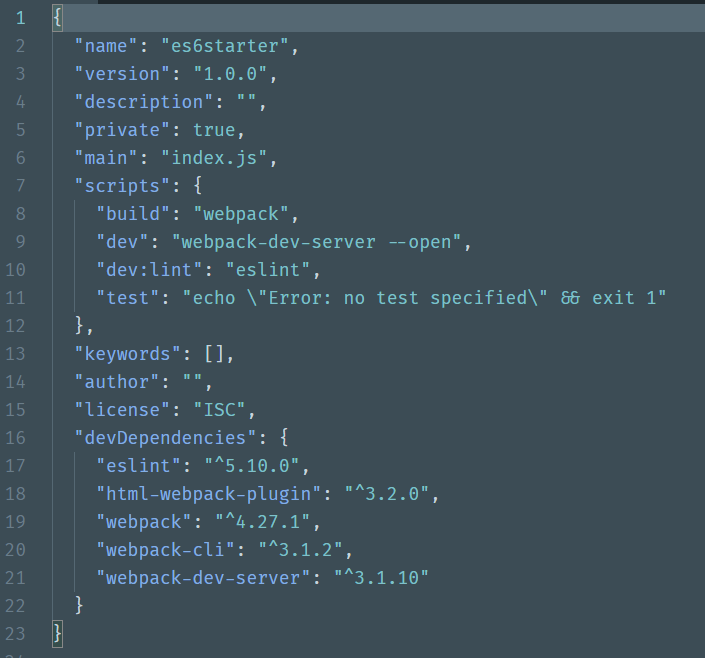


Select “Open Settings.Json”. On the right hand side are your settings (the left shows all the settings available for customization. Check them out!). Replace the JSON object on the right with the settings in the link I provided earlier. Save (Ctrl+S) and close (Ctrl+W) the file.

Open the Visual Studio Code integrated terminal (Ctrl+`) and issue following command:

npm install

This will take a while. The Node Package Manager will read the package.json file (and package\_lock.json) and install all the dependencies we need. While it installs, look over the package.json file to see the dependences:



We are using Webpack and the Webpack-Dev-Server to be our module bundler and development server, as we did in the class.

For more information on both of these, see: <https://webpack.js.org/guides/>

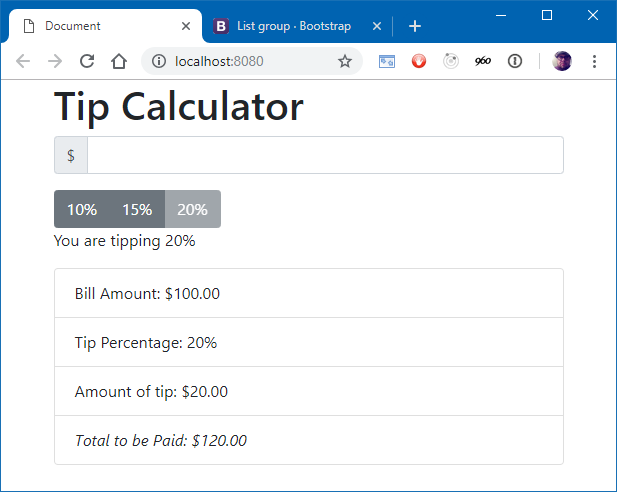
After the dependencies install, in the command prompt, we will ask npm to run our dev script.

npm run dev

This will compile our code and start a development server at <http://localhost:8080>. It will also open the default browser on your machine. You may want to change your default browser to Google Chrome, so you use the developer tools. Here’s a guide: <https://www.laptopmag.com/articles/make-chrome-firefox-default-browser-windows-10>

# Creating the Project

In the index.html file in the root of your project, create the markup to create a hard-coded user interface similar to this one:



I have included a link to the bootstrap library in the index.html for styling. You can refer to the documentation at <http://getbootstrap.com>. (hint: I used a page-header component, an input group, and a list group for my example. Yours doesn’t have to be *exactly* like mine, but it is pretty rad.)

# Make it Work

Using the narrative at the beginning of this document, create the functionality of the tip calculator.

You must complete at least the basic functionality as described. There are also optional iterations if time allows.

Some tips:

* The button which indicates the selected tip amount should be disabled. The others should not.
* The input should be limited to numbers. If a number less than zero is indicated, make the border of the amount input red and clear out the calculated values in the list.
* Make sure you are formatting the output correctly.
* The calculated amounts in the list should update automatically as the user enters a value. No waiting for “enter”, no “do it” button.
* You can use the elements id property to identify meaningful elements. Please, no hooking to events in the HTML source.

# Rubric for Evaluating Code

As you build the application, try to apply and extend the concepts you learned in class. You will, most likely, have to do some extreme googling and research. Not everything you need to accomplish this lab was covered in the class. Your code reviewer will work with you to make sure your application is structured properly.