11.2.2 Organize Your Repository

Dana has started to get into a JavaScript coding rhythm. But before she begins to create code for real and then commit scripts to her repo, she needs to organize it.

Building a page that contains JavaScript will require Dana to link additional JavaScript files to the `index.html` file that she'll be working on later. This means keeping track of multiple things at once: an HTML file, JavaScript files, images (for customizing the webpage) and a CSS style sheet. Therefore, it's a good idea for Dana to establish a solid folder structure now instead of when she's elbow deep in creating her JavaScript functions.

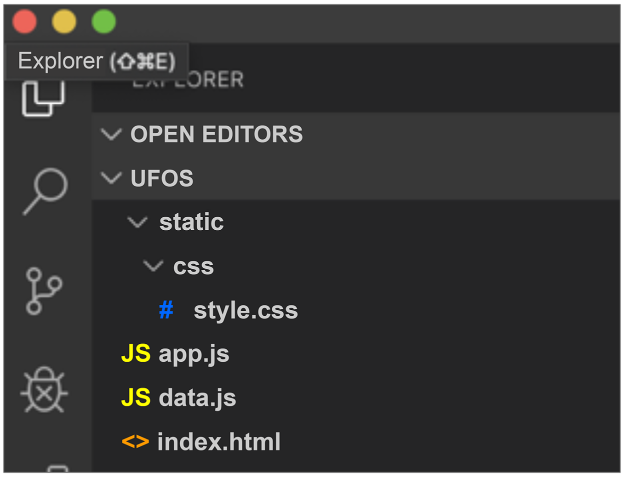
Before we get too far along with our coding, we need to set up a file organization system for our repo. The end result of this project will be an HTML page or application, so we need to establish the proper folder structure accordingly. At a high level, here's what we'll do:

- Create the `index.html` file.
- Create a subfolder to hold the CSS file (style.css).
- Create a subfolder for images.
- Create a subfolder to hold JavaScript.

First, in the repo folder we established earlier ("UFOs"), create the `index.html` file. This file is the window to our work: the table and Dana's article summary (along with titles and filters) will all be displayed through this file. We won't be coding it yet—that will come later—but we're creating it now so that it will be ready for us when it's time to build the page.

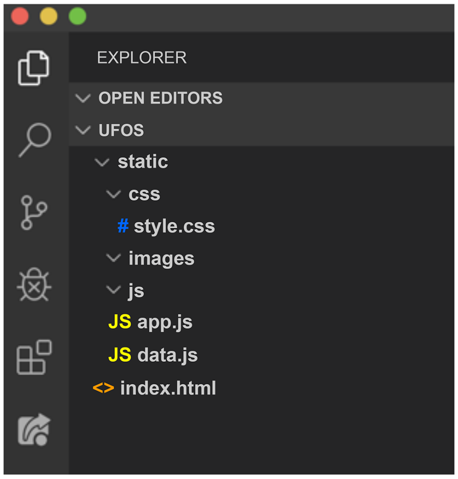
Next, create a subfolder in the repo folder named "static." This static folder will hold our static CSS file; this only means that it isn't being moved or altered externally. In VS code, right-click the menu and select "New Folder," and then name it "static."

Inside the static folder, create another subfolder named "css" to hold the `style.css` file. You can use the same right-click method to bring up the creation menu, but this time select "New File." We'll customize our webpage using the `style.css` sheet, but for now we can leave it blank. Here's what the folder structure should look like so far when viewing it in VS Code.



The next subfolder we'll create is our "static" folder to hold whatever images we want to add to our website when it's time to customize it. Create the folder now and name it "images." But for now, move on to the next step—we'll add images later.

The third and final subfolder we'll create is one to hold our JavaScript. Name the folder "js" and move the `data.js` and `app.js` files into it so that your folder structure looks like this:



```
EXPLORER
∨ OPEN EDITORS
∨ UFOS
  ∨ static
    ∨ css
       # style.css
    ∨ images
    ∨ js
      JS app.js
      JS data.js
  <> index.html
```

Establishing this folder structure is a best practice when creating webpages with JavaScript. It's important to keep things organized when creating a webpage using JavaScript components, as there are even more moving pieces than a static website. We'll be linking to images and a style sheet as well as JavaScript scripts. The organization presented here provides clearly designated spots to store the code we'll be working on, making it easier to locate them as we go.