CPSC1520 – JavaScript 1 Exercise: Developer Console and Intro to JavaScript

Introduction

The console is a great tool to see what's going on on the page and will also be how we debug our JavaScript throughout the course. It allows us to execute statements on the page (like we've seen in class) in real time!
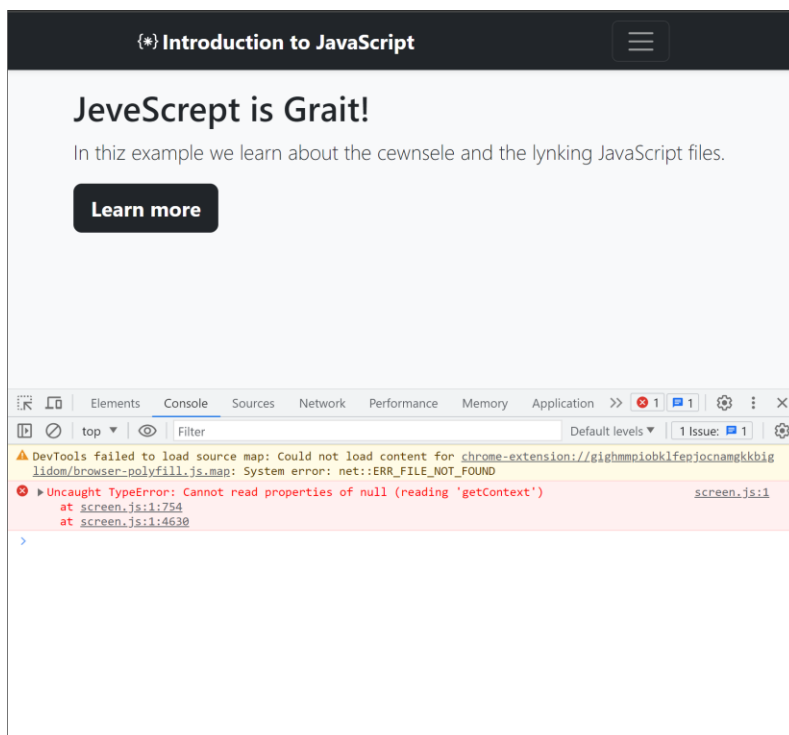
In this first lab assessment, you will try fixing typos on our page in the console first; second, you will put all of these changes in your main.js; and third, you will push it up to GitHub.

How to access the console.

There are a few ways to access the console:

- Right-click on an element in the page and select inspect element from the context menu
- Press ctrl+shift+j (Windows) or cmd+option+j (Mac)
- Press F12
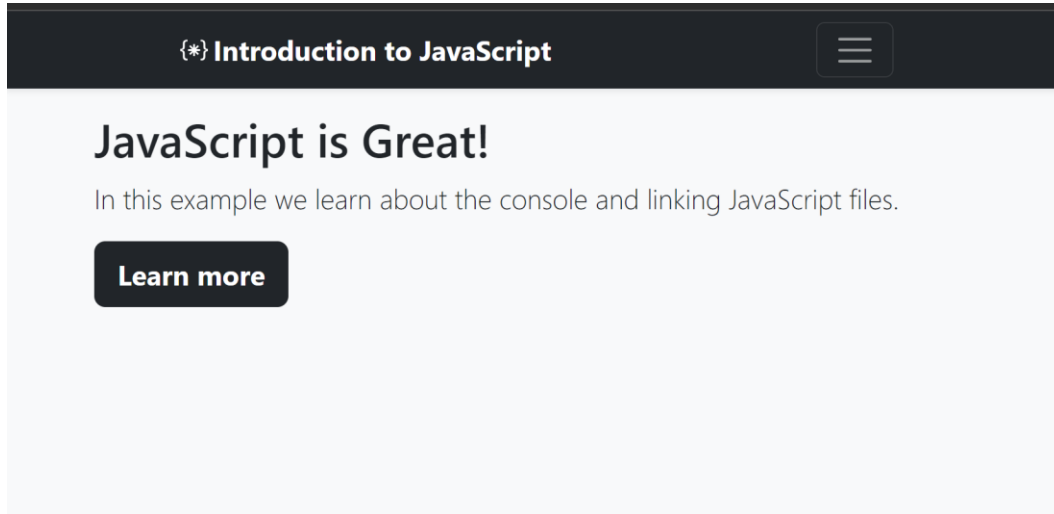- From the customize menu, select More tools and then Developer tools

When you open the accompanying folder with the files, you should see something like this

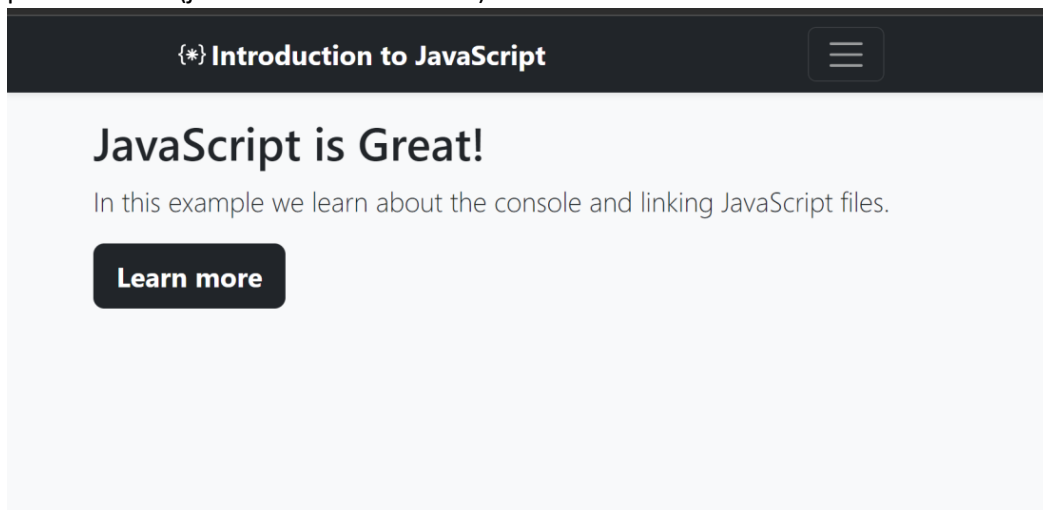
Exercise Step 1 – fix typos in the console

1. Open the chrome web browser

2. Select the h1 with the id of "app-title" and the paragraph with the class "app-description"s using the querySelector and assign it to a variable using an appropriate name. ". If you need help with this, please refer to your notes.
3. Modify the "innerText" or the "innerHTML" of the variables selected. To fix the typos so that it looks like the picture below.

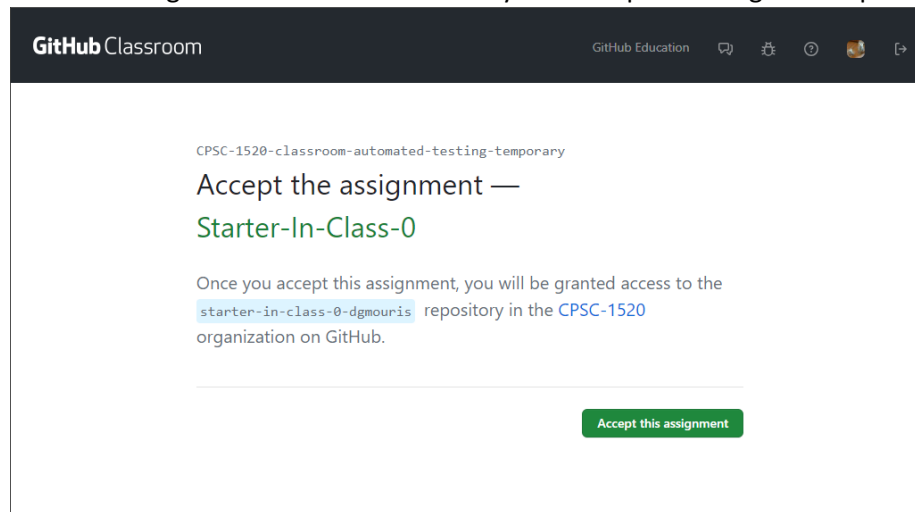## Exercise Step 2 – use a JavaScript file in your html.
1. In your "main.js" file in the accompanying files, add a "console.log" statement that says "js file successfully linked."
2. Link your JavaScript file to your HTML using the script tag we learned in class (refer to notes if you can't).
3. Use the work from the exercise step above (selecting the about element and fixing the typos) and add it to the JavaScript file so that when you refresh the page, the html should look like the picture below (just like the last exercise).
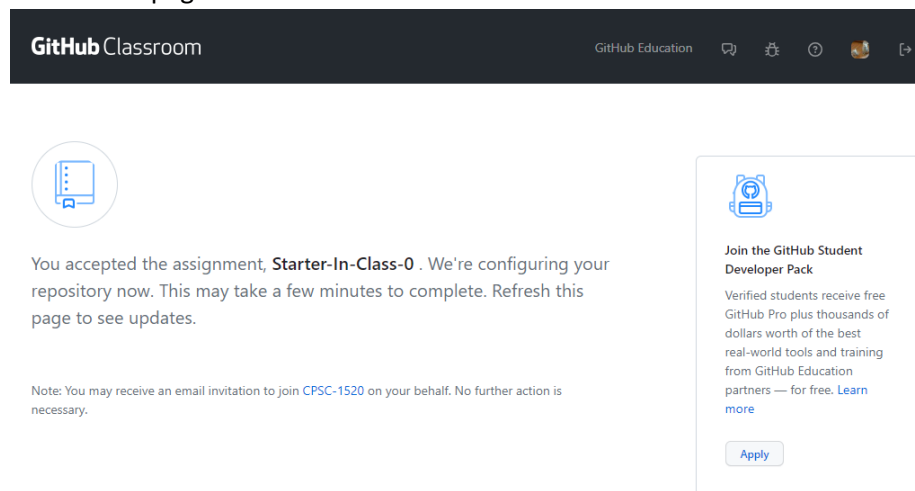

Exercise Step 3 – Push up your code to github (accepting this assignment)
1. Open the link given and accept the assignment. Your link should look something like this.
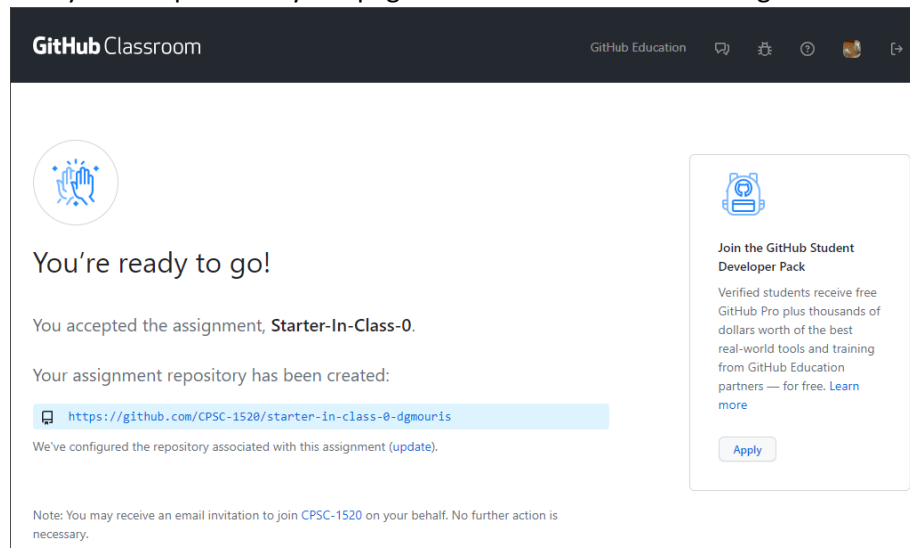
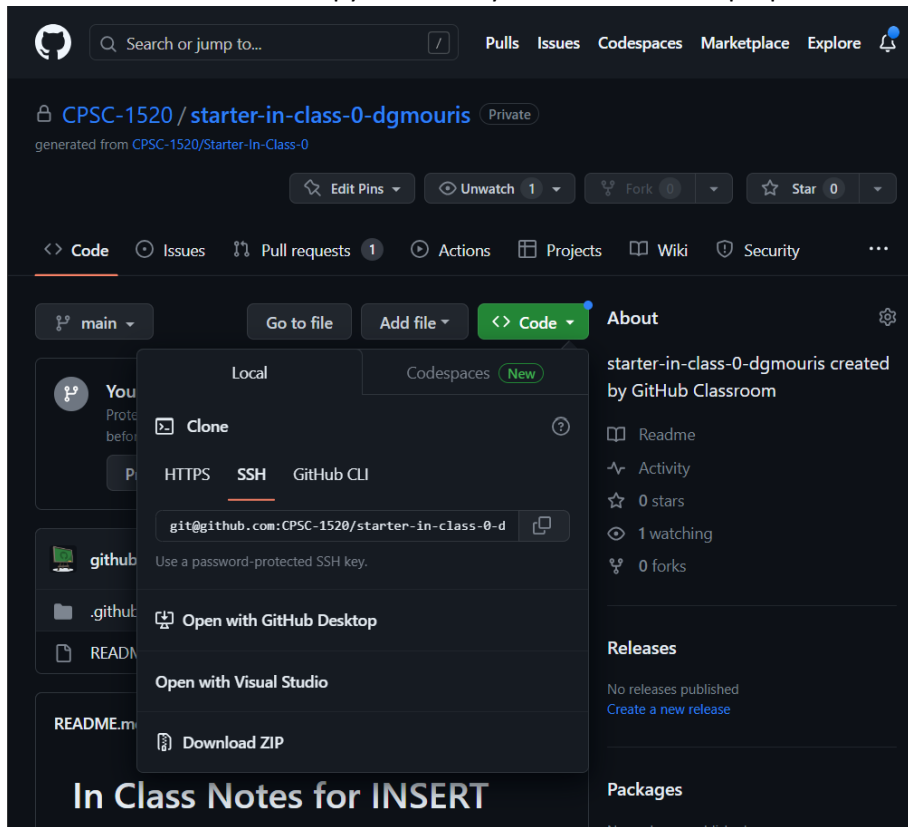Note the image will be different because you'll accept the assignment specified.

You'll see a page like this.

One you're repo is ready the page should look like the following.

2. You should see the page below once you click on the link highlighted in blue. Click the button that says "Code." You'll need to select "HTTPS" unless you've set up "SSH" (you can also set up GitHub CLI". Click on the copy icon once you've selected the proper icon.

3. Clone the repository in your console (or if you're using GitHub Desktop) using the "git clone REPO_URL" command.

```
DMOURIS@W309-DMORR2 C:\Users\dmouris\temp
$ git clone git@github.com:CPSC-1520/starter-in-class-0-dgmouris.git
Cloning into 'starter-in-class-0-dgmouris'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 8 (delta 0), reused 2 (delta 0), pack-reused 0
Receiving objects: 100% (8/8), done.

DMOURIS@W309-DMORR2 C:\Users\dmouris\temp
$
```

And go into this folder.

4. Make your changes, then add them to staging (using "git add .") and commit them (using "git commit -m "CHANGE THIS MESSAGE"). Once committed, push them up to GitHup (using "git

push") it should look like below.

```
DMOURIS@W309-DMORR2 C:\Users\dmouris\temp\starter-in-class-0-dgmouris
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

DMOURIS@W309-DMORR2 C:\Users\dmouris\temp\starter-in-class-0-dgmouris
$ git add README.md

DMOURIS@W309-DMORR2 C:\Users\dmouris\temp\starter-in-class-0-dgmouris
$ git commit -m "Made changes"
[main 9532c1b] Made changes
 1 file changed, 1 insertion(+), 3 deletions(-)

DMOURIS@W309-DMORR2 C:\Users\dmouris\temp\starter-in-class-0-dgmouris
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 20 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 373 bytes | 373.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:CPSC-1520/starter-in-class-0-dgmouris.git
   b6ef88e..9532c1b  main -> main

DMOURIS@W309-DMORR2 C:\Users\dmouris\temp\starter-in-class-0-dgmouris
$
```
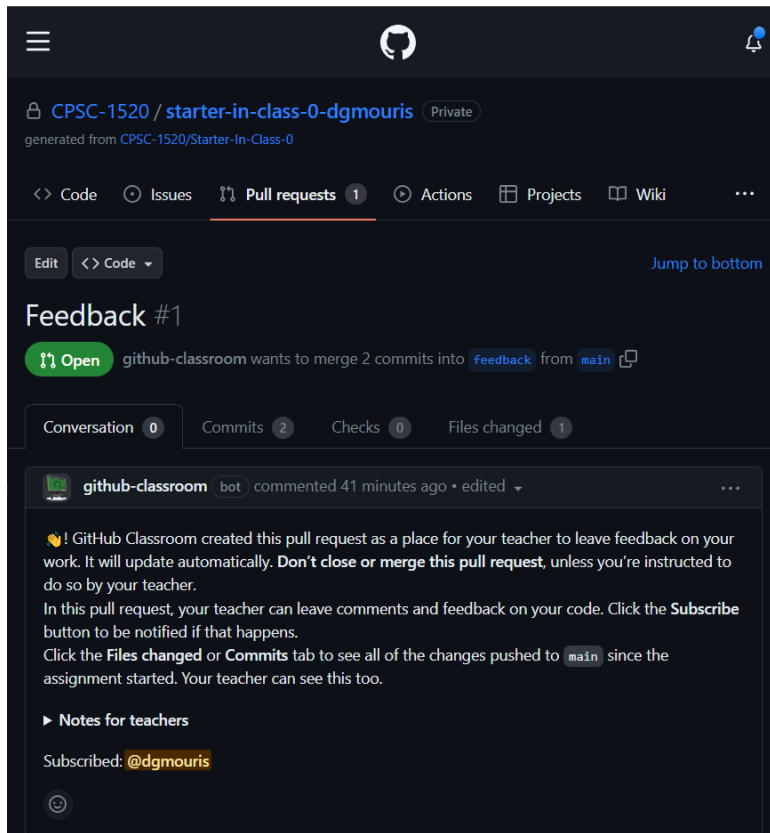
5. If you click "Pull Requests" and then the first item called "Feedback" you should see your commit (seen at the bottom).
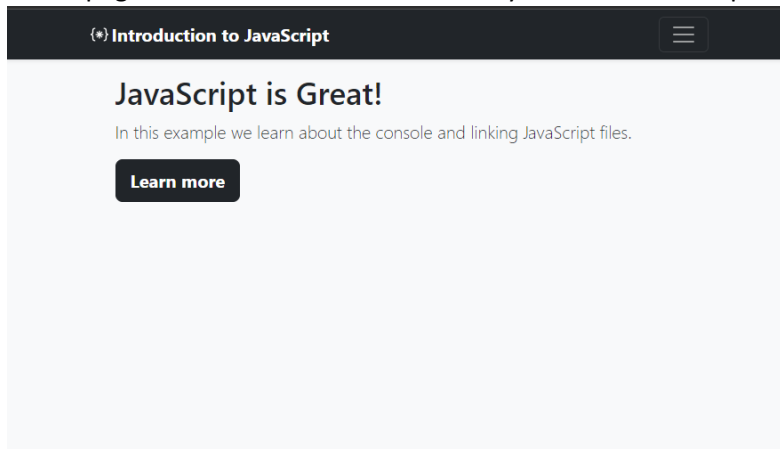
6. The link of your repository to Moodle.

Grading

I'll give full marks if:

- When I load the page there are no errors in the console.
- When page is loaded this should be exactly what I see in the picture below.

Note: If you don't follow the instructions, you'll get a zero. There are no marks in between.