**JavaScript Homework**

Let’s keep applying the Git collaboration skills we’re learning.

In this project, we’ll be using Git to write comments on your student’s JavaScript homework. Don’t worry! You don’t need to know JavaScript to do the project.

If the code below piques your interest, check out Codecademy’s [JavaScript](https://www.codecademy.com/learn/javascript) course here.

If you get stuck during this project or would like to see an experienced developer work through it, click “**Get Help**“ to see a **project walkthrough video**.

**Tasks**

**0/9Complete**

Mark the tasks as complete by checking them off

**JavaScript Homework**

**1.**

There is a remote called **maryrose-hw** which contains your student’s JavaScript homework.

Clone the remote, giving it a name of your choice.

**2.**

Use the terminal command cd to go into your cloned repository.

cd my\_clone

Don’t forget to replace my\_clone with the name you gave your clone in the previous step.

**3.**

Use the file navigator to open **homework.js** in the code editor.

Be sure to open **homework.js** in your clone, *not* in the remote.

**4.**

From the terminal, create a new branch with Git. Your branch will be commenting on Mary Rose’s JavaScript homework, so name the branch appropriately.

Next, switch over to your new branch.

**5.**

Time to start commenting on Mary Rose’s JavaScript homework.

Comments are used to leave notes for other programmers or yourself. In JavaScript, you can make a single-line comment by typing // at the beginning of any line. Then type your comment. For example:

// I'm a comment!

There are a number of comments already in **homework.js** to guide you.

Make whatever comments you’d like to Mary Rose’s JavaScript homework. Then click Save.

**6.**

Add your changes to the Git staging area.

**7.**

Now, make a commit.

**8.**

Push your branch up to the remote.

**9.**

Use:

cd ../maryrose-hw

To change directories back to the remote. Use your Git knowledge to see the branch you just pushed.