Why Do We Need This

You might be like me where I start work on the next feature to my project at night, but then go to bed before I actually finish. Which means that, when I start working the next day, there are uncommitted changes. This is fine because I haven't finished the new feature, but I can't remember exactly what I've done since my last commit. `git status` will tell us what files have been changed, but not what those changes actually were.

The `git diff` command is used to find out this information!

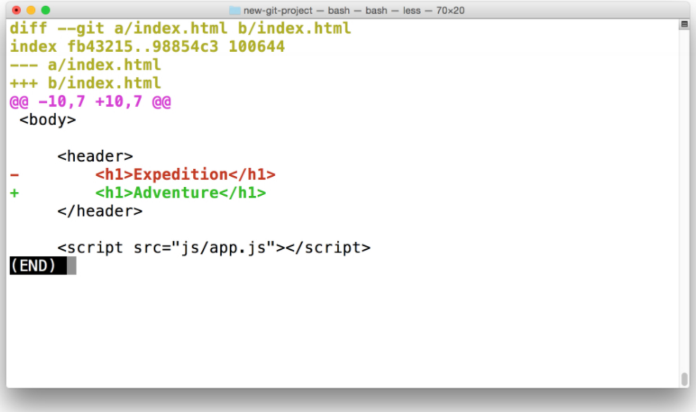
`git diff`

The `git diff` command can be used to see changes that have been made but haven't been committed, yet.

```
$ git diff
```

To see `git diff` in action, we need some uncommitted changes! In `index.html`, let's reword the heading. Change the heading from "Expedition" to "Adventure". Save the file and run `git diff` on the Terminal.

You should see the following:



```
diff --git a/index.html b/index.html
index fb43215..98854c3 100644
--- a/index.html
+++ b/index.html
@@ -10,7 +10,7 @@
 <body>

     <header>
-        <h1>Expedition</h1>
+        <h1>Adventure</h1>
     </header>

     <script src="js/app.js"></script>
(END)
```

The Terminal application showing the output of the `git diff` command.

Wow, doesn't that look familiar! It's the same output that we say with `git log -p`! Wanna know a secret? `git log -p` uses `git diff` under the hood. So you've actually already learned how to read the output of `git diff`!

If you don't remember what the different sections are, check out the Annotated "git log -p" Output from the previous lesson.

> 💡 *These Changes Were Not Committed* 💡

> *The changes in this section were used to demo the output of `git diff`. They were not committed to the repository. If you'd like, you can definitely commit the changes to the repository, just know that your `git log` will look slightly different from mine because it includes this extra commit.*

Git Diff Recap

To recap, the `git diff` command is used to see changes that have been made but haven't been committed, yet:

```
$ git diff
```

This command displays:

- the files that have been modified
- the location of the lines that have been added/removed
- the actual changes that have been made

Further Research

- git diff from the Git Docs

NEXT