

# HW# 1

Aaron Segura

Frederick Lee

September 3, 2018

# 1 Task 1: Use 'git diff' on 2 commit hash tags

## 1.1 What

We were asked to use the hash tags of the two most recent commits in our repository and apply them to the git diff command like so:

```
$ git diff hash1 hash2
```

## 1.2 How

To do this, we first typed in:

```
$ git log
```

We then copied the hashes for the 2 most recent commits. The following was the text output:

```
diff --git a/README.txt b/README.txt
index 2ea85b3..abf48a4 100644
--- a/README.txt
+++ b/README.txt
@@ -1,4 +1,4 @@
-README file for CS 471, Fall 2018, Frederick Lee
+README file for CS 471, Fall 2018, Aaron Segura & Frederick Lee
```

## 1.3 Why

The following command:

```
$ git diff hash1 hash2
```

shows the user or users what information has changed in each file, between the two commits. The lines in the output prefixed by a single minus sign (-) are from the commit whose hash is the first argument; the lines prefixed by a (+) are from the commit whose hash is the second argument.

## 2 Task 2: Use 'git diff' for a temporary change

### 2.1 What

We're asked to make a temporary change to test.txt (which exists at the root of our git repository), then use 'git diff', report the output, and interpret the output.

### 2.2 How

Applying the 'git diff' command after modifying test.txt outputs:

```
diff --git a/HW1/test.txt b/HW1/test.txt
index fd23994..5d2d2c9 100644
--- a/HW1/test.txt
+++ b/HW1/test.txt
@@ -1,3 +1,4 @@
  My first commit of text.txt
  I am Aaron Segura
  I am learning version control
+hello this is a temporary line
```

### 2.3 Why

The output shows the file differences between what one has on a local machine and what is contained in the most recent commit in the repository. After the change was made we then finished by reverting the temporary change.