DSI: Unix Shell, Git and GitHub
Assignment 2 & Quiz: Git and GitHub

Part 1

Part 1 of Assignment 2 is a quiz. Please complete to the best of your ability. Notes are permitted. Please email your responses to the Instructor.

1. Check all that are TRUE about version control:
   - Can revert files to a previous state
   - Can compare changes over time
   - Can see who modified something last
   - Can recover lost files

2. What is the difference between centralized version control systems and distributed version control systems?

3. What are the three states that files can reside in?
   - a) committed, changed, waiting
   - b) saved, changed, staged
   - c) committed, modified, staged
   - d) saved, modified, staged

4. What command initializes a new repository?
   - a) `git clone`
   - b) `git branch`
   - c) `git fork`
   - d) `git init`

5. What does `git diff` do?
   - a) compares the differences between the home directory and staging area
   - b) compares the differences between the working directory and staging area
   - c) compares the differences between the working directory and what's been committed
   - d) compares the differences between the staging area and what's been committed

6. How do you add a message to your commit? (select all that apply)
   - a) `git commit -m`
   - b) `git commit -messages`
   - c) `git commit`
   - d) `git commit -message`

7. How do you add a remote repo? (select all that apply)

- a) `git remote`
- b) `git add remote`
- c) `git clone`
- d) `git add clone`

8. What is the difference between `git pull` and `git fetch`?

9. How do you switch branches?
   - a) `git checkout`
   - b) `git checkout -b`
   - c) `git branch -c`
   - d) `git branch`

10. Why are messages important? What would make a good commit message?

11. Please correct the merge shown below (both codes are suitable, neither has errors):

```
<<<<<<< HEAD
df.loc[df['sex'] == 'f', 'age'].mean()
=======
df.loc[df['sex'] == 'm', 'age'].mean()
>>>>>>> branch_1
```

Part 2

1. `fork` and `clone` this class GitHub repo.
2. `push` your Assignment 1 to the folder labelled “assignment-2.” Your additions should include…
   - All components necessary to run Assignment 1
   - Proper folder structure (inputs, outputs, scripts)
   - A README.md file. The README should include components discussed in the workshop. Feel free to research good READMEs and add anything that you believe will add value to your README
3. Create a `pull request` to add your additions to the class repo.

Rubric:

Component	**Yes**	**No**
1. Repo contains all necessary components to run Shell script and has the correct folder structure		
2. README is comprehensive and includes components discussed in class plus at least one component learned from outside sources		
3. Pull request has been successfully requested without any merge errors		