1) In this video the author explains several version control systems that are non source code related.

**Microsoft Word: Track Changes and Ctrl + Z to undo.**

2) What does it mean when we say a source code versioning system is concurrent ?

**Allowing the user to track multiple files and multiple users to work on a code repo at the same time.**

3) The author identifies several advantages of a distributed version control system. Name 2.

**No need to communicate with a central server. Encourages participation in forking changes.**

Git Installation

4) What is the command necessary to set Git's default text editor to Atom?

**git config core.editor “atom --wait”**

*GITting Started*

5) Does it matter where I can the GIT command init ? Why?

**Yes this will tell git what directory to track. If you type git init while outside your project directory your files won’t be tracked.**

6) How would we remove the repo from being tracked by Git?

**You would remove the .git directory.**

7) What git command does the author use to add all of the files in the folder to the current project?

**git add .**

8) [Critical Thinking] Why are commit messages best written in the present tense?

For readability purposes, the commit message is a label for the changes. This would make git log much more readable because it would clearly show what each commit (or set of changes) is doing.

*Git Concepts and Architecture*

9) [Critical Thinking] What is the advantage of a three tree architecture over a two tree architecture?

**You can group a set of changes into the main repository by the staging index and commits.**

10) [Critical Thinking] What is the difference between the git add command and the git commit command?

**git add will add files to the staging index and then git commit will add to the repo the commit in the staging area.**

11) What is the purpose of the checksum ?

**Create a unique identifier for individual commits.**

12) To what does the HEAD pointer point?

Points to the tip of the current branch in the repo.

*Making changes to Files*

13) What command does the author use to see whether or not there are changes that need to be committed?

**git status**

14) When using the git add command, is there any difference between adding a new file or editing an existing file?

**Not in the use of git add, the command we call is the same but git will be able to distinguish between a new file which would show up initially as an untracked file and a modified file.**

*Using Git with a real project*

15) What command does the author use to stage every file in the explore\_california directory to the repository?

**git add .**

16) What command does the author use to commit every file in the explore\_california directory and add a comment?

**git commit -am “<commit-message>”**

17) What command does the author use to add and commit the files at one time?

**git add <name-of-specific-file>**

**git commit -m “<commit-message>”**

18) [Critical Thinking] The author says there is a caveat for using the shortcut you identified in the previous question. In which 2 circumstances should you not use the shortcut?

**When you have new files in the repo that are untracked.**

**If you’ve made changes in files that you don’t want to be part of the next commit.**

19) What is the caveat, or issue we need to remember when using the stage and commit shortcut?

**The shortcut does not include untracked files.**

20) What can be done differently if we group our commits together topically as the author does in this video?

**You can collaborate more efficiently. When someone wants to pull just a certain set of changes if you have granular commits they can pull a commit without affecting code they don’t want changed.**

***Undoing Changes***

21) How does the process of reverting a commit go hand in hand with making atomic commits?

**When you make atomic commits you can use git revert <SHA> confidently, without worrying that git will change code that doesn’t relate topically to the last commit.**