**How might using version control make you more confident to make changes that could break something?**

Being able to revert back to a previous version at any point using git checkout (commit\_id)

However its best to always maintain the most recent commit id, as checking out to a previous version will result in loss of logs

Explore significant changes in code base

Withouct VCS, - you make very small iterative changes – making program incrementally better

With VCS, you have a large space of things to try, makes you better programmer, explore deep architecture concepts

LESSON 2

What happens when you initialize a repository? Why do you need to do it?

How is the staging area different from the working directory and the repository?

What value do you think it offers?

How can you use the staging area to make sure you have one commit per logical

change?

What are some situations when branches would be helpful in keeping your history

organized? How would branches help?

How do the diagrams help you visualize the branch structure?

What is the result of merging two branches together? Why do we represent it in

the diagram the way we do?

What are the pros and cons of Gitâ€™s automatic merging vs. always doing merges

manually?