Part 3:

GitHub is a website that allows user to upload and collaborate on codes. GitHub was founded by Chris Wanstrath, PJ Hyett, Tom Preston-Werner, and Scott Chacon in 2008. GitHub was created because they wanted to create a platform that was easier for developers to be developers which is working together even though they are from different parts of the world and solving problems together. The creators also wanted a collaborate space to build a community. Some platforms that are like GitHub are GitLab, BitBucket, SourceForge, and Launchpad. I would use a platform like GitHub because it is a more effective way of collaborating with other people. Instead of everybody working with one computer and on one part of a code, people can use GitHub to work on different parts of a code on their own computer and putting it together by putting it on GitHub.

Part 4:

Introduction Sequence-

1: Introduction to Git Commits

git commit

git commit

2: Branching in Git

git checkout -b bugFix

3: Merging in Git

git checkout -b bugfix

git commit

git checkout master

git commit

git merge bugfix

4: Rebase Introduction

git checkout -b bugfix

git commit

git checkout master

git commit

git checkout bugfix

git rebase master

Ramping Up-

1: Detach yo’ HEAD

git checkout C4

2: Relative Refs (^)

git checkout C4^

3: Relative Refs #2 (~)

git branch -f master C6

git branch -f bugfix C0

git checkout C1

4: Reversing Changes in Git

git reset local~1

git checkout pushed

git revert pushed

Moving Work Around-

1: Cherry-pick Intro

git cherry-pick C3 C4 C7

2: Interactive Rebase Intro

Git rebase -i master~4 --aboveAll

A Mixed Bag-

1: Grabbing Just 1 Commit

git checkout master

git cherry-pick C4

2: Juggling Commits

git rebase -i caption~2 --aboveAll

git commit --amend

git rebase -i caption~2 --aboveAll

git branch -f master caption

3: Juggling Commits #2

git checkout master

git cherry-pick C2

git commit --amend

git cherry-pick C3

4: Git Tags

git tag v0 C1

git tag v1 C2

git checkout C2

5: Git Describe

git commit

Advance Topics-

1: Rebasing over 9000 times

git rebase master bugFix

git rebase bugFix side

git rebase side another

git rebase another master

2: Multiple Parents

git branch bugWork master~^2~

3: Branch Spaghetti

git checkout one

git cherry-pick C4 C3 C2

git checkout two

git cherry-pick C5 C4 C3 C2

git branch -f three C2

Part 5:

Repository- A repository is a collection of files and folders that’s associated with the project.

Commit- Commit records the changes made to the repository. It can be named so it’s easier to keep track of what was done during those changes.

Push- Push updates the repository when changes is done to the files/ folders locally.

Branch- A branch is like a pointer to a commit where the default branch is the master branch.

Fork- Fork is a copy of a repository. Forking is helpful when you want to make changes to the project without messing up the original project.

Merge- Merge is when you combine two branches together.

Clone- Clone is creating a copy of a project that already exist in the project.

Pull- Pull is updating the local files/folders to match the one on GitHub.

Pull Request- Pull request tells other people what changes you pushed to the project.

Part7:

1. Fork the repository
2. Using command Line: git clone <https://github.com/MinyingHe/courses.git>
3. Open up the ReadMe.md with a document reader and edit it writing “ He, Minying February 10, 2019 4:15 PM”
4. Using command Line: git add .
5. Using Command Line: git commit -m “GitHub Training”
6. Using Command Line: git push origin master
7. Go on the forked repository and create Pull Request