**Git Reflection**

1. **Version control adds additional steps like creating a repo, commit, etc., and it slows me down from completing my work. Is that overhead worthwhile?**

The Overhead is always worthwhile because Version Control keeps a track of each and every commit you do. So, by any chance if there’s any issue you can revert back and finish.

1. **What is a good logical way to structure the commits when creating web pages?**

The good logical way to structure the commit is to commit the file whenever a necessary improvement is done.

1. **Is it good to include assets like images and videos in the git repo or is it better to keep them outside the repo? What if there are videos on the web page? How do we ignore them from staging and committing to the repo?**

If the Images and Videos are their own work, then I suggest to keep them in repo. If not, you can always recover them.

Usage of .gitignore helps in ignoring the files that you feel should be ignored from staging and committing.

1. **It looks like staging is not required before making a commit to the repository. Do you agree?**

Before we make a Commit we must tell what files we want to commit, This is called Staging. If you want to commit without staging, then each and every file will be committed.

1. **Imagine a scenario where the old project files, that are previously in the version history, are deleted from the project folder and commits are done to the same repo. Will we lose the old files forever? How does this work?**
2. **I’m working on a data science project with the Google collab. Collab already maintains version control. How is this different from using git? Which one is better? Is it possible to maintain collab files on git?**

The main difference between Google collab and Git is mainly you can create branches as working directories and merge them later on Git which is not possible on Collab. The other difference between them is Git compares differences between two changes whereas Collab only allows you to know that the changes have happened.