# HW4

## QTM 350 - Data Science Computing

### Assignment 04: Advanced Git

#### Due 02 October 2024

In this assignment, you will explore Git in more depth, focusing on commands we have not covered in class. You should research how to solve each task using specific Git commands. Feel free to use any tools you prefer to complete the assignment. Explain each command in simple terms. Of course, you are welcome to test them as well! If you have any questions, please let me know.

#### Questions

1. What command would you use to display all the commits that have not yet been pushed to the remote repository?

`git log origin/main..HEAD` displays all the commits that have not yet been pushed to the remote repository by showing the log of the commits that are on the most recent version of the local repository but not yet on the remote repository.

2. Explain the purpose of the `git tag` command.

The `git tag` command is used to create, list, delete, and verify tags in Git repositories. Tags are references that point to specific points in Git history. These are often used to keep track of release versions.

3. How do you remove a file from the repository history?

`git filter-branch --force --index-filter 'git rm --cached --ignore-unmatch <filepath>' --prune-empty --tag-name-filter cat -- --all`

This command deletes the the file from the staging area, incex, and repository history, –prune-empty causes all empty commits to disappear from commit history. –tag-name-filter cat updates tags to the newly updated commmits, – –all specifies that this should apply to all tages and all branches.

4. How do you temporarily save your changes without committing them?

git stash. This saves all the changes and reverts the working directory back to the most recent commit.

5. How do you amend the most recent commit message?

`git commit --amend` opens a text editor that allows the user to change the commit message

6. How can you split a commit into multiple commits?

`git rebase -i HEAD~n` start an interactive rebase `git reset HEAD^` undo recent commit but keep changes in repository `git add <first group of files>` recommit separated into 2 gorups `git commit -m <commit message> git add <first second group of files> git commit -m <commit message> git rebase --continue` finish rebase and push `git push origin <branch-name> --force`

7. How do you display the changes between two branches?

`git diff <branch1>..<branch2>` input the two branches to see the differences between the branches

8. What command would you use to see which remote repository a local repository is connected to?

`git remote -v` shows the remote repository and upstream repositories

9. Explain the difference between `git merge` and `git rebase`.

`git merge` combines changes from different branches into one, preserving the commit history, while `git rebase` moves, combines, or splits a sequence of commits to a new base commit.

10. How can you search for a specific commit message?

`git log --grep="<commit-message>"` this searches the git log for the message inserted. It shows all the possible results.