Assignment 3 Solution

Introduction

The goal of this assignment is to consolidate skills in advanced Git commands and GitHub operations. It is assumed that you are up to date with concepts and commands taught in previous modules (Modules 1 to Module 4). This assignment focuses on the following topics:

- Git Branching and Merging, and techniques
- Commit History rewrite operations such as
    - Amendment of Git commits
    - Rebase
    - Pull with Rebase
    - Reflog
- Git Tags

Solution statement with steps (Instructor Solution)

```
#################### STEP-1 (Set up repo)

# Fork the following public GitHub project - https://github.com/bibroy/insh-assignment-3
# Once you have forked project insh-assignment-3, simply clone the repository in your local repo

# Switch (cd) to project directory insh-assignment-3
$ cd insh-assignment-3

#################### STEP-2 (Working with Git Branches)

# Create a Git branch named branch1
$ git branch branch1

# Rename branch1 to feature-branch-1
$ git branch -m branch1 feature-branch-1

# Now display the list of all local branches
$ git branch

# Now display the list of all branches
$ git branch -a

#################### STEP-3 (Working with fast-forward merge)
```

Checkout the branch feature-branch-1
$ **git checkout feature-branch-1**

Edit robots.txt. Make a simple change in the file
$ **vi robots.txt**

commit your changes
$ **git commit -am "commit message 5"**

checkout master branch
$ **git checkout master**

Do a git diff between feature-branch-1 and master branch
$ **git diff master feature-branch-1**

Now do a FAST FORWARD MERGE of feature-branch-1 with master branch
$ **git merge feature-branch-1**

Run git log command
$ **git log --oneline**

#################### **STEP-4 (Dealing with no-fast-forward merge)**

Checkout the branch feature-branch-1
$ **git checkout feature-branch-1**

Edit robots.txt. Make a simple change in the file
$ **vi robots.txt**

commit your changes
$ **git commit -am "commit message 6"**

checkout master branch
$ **git checkout master**

Do a git diff between feature-branch-1 and master branch
$ **git diff master feature-branch-1**

Now do a NO FAST FORWARD MERGE of feature-branch-1 with the master branch
$ **git merge feature-branch-1 --no-ff**

Run git log command
$ **git log --oneline**

#################### **STEP-5 (Getting adept at 3-way merge)**

Checkout the branch feature-branch-1

$ **git checkout feature-branch-1**

Edit robots.txt. Make a simple change in the file
$ **vi robots.txt**

commit your changes
$ **git commit -am "commit message 7"**

checkout master branch
$ **git checkout master**

Edit humans.txt. Make a simple change in the file.
$ **vi humans.txt**

commit your changes
$ **git commit -am "commit message 8"**

Do a git diff between feature-branch-1 and master branch
$ **git diff master feature-branch-1**

Now do a 3-WAY MERGE of feature-branch-1 with master branch
$ **git merge feature-branch-1 -m "3-way merge commit message"**

Run git log command
$ **git log --oneline**

delete branch feature-branch-1
$ **git branch -d feature-branch-1**

Run git log command
$ **git log --oneline**

#################### **STEP-6 (Amend your Git Commits)**

Edit robots.txt. Make a simple change in the file. Let's call it simple-change-9
$ **vi robots.txt**

Add your changes to staging
$ **git add .**

Amend most recent commit
$ **git commit --amend --no-edit**

#################### **STEP-7 (Rebase, setting up a flat commit history)**

Create and checkout to a new branch called feature-branch-2

$ **git checkout -b feature-branch-2**

Edit robots.txt. Make a simple change in the file. Let's call it simple-change-10
$ **vi robots.txt**

commit the simple-change-10
$ **git commit -am "commit message 10"**

checkout master branch
$ **git checkout master**

Edit humans.txt. Make a simple change in the file. Let's call it simple-change-11
$ **vi humans.txt**

commit the simple-change-11
$ **git commit -am "commit message 11"**

checkout feature-branch-2
$ **git checkout feature-branch-2**

# Run command to rebase feature-branch-2 based on the master
$ git rebase master

checkout to master branch
$ **git checkout master**

Run git log command
$ **git log --oneline --decorate --graph --all**

Run git command to merge feature-branch-2
$ **git merge feature-branch-2**

Run git log command again to see the effect of rebasing
$ **git log --oneline --decorate --graph --all**

#################### **STEP-8 (Reflog, a log of different kind)**

Run appropriate command to display all the entries in Reflog
$ **git reflog**

Run command to see the fifth prior value of the HEAD of your repository
$ **git show HEAD@{5}**

Display reflog information for master branch formatted like the git log output
$ **git log -g master**

#################### **STEP-9 (Tag your Commits)**

```
# Create a lightweight tag named "v1.4-lwtag"
$ git tag v1.4-lwtag

# List out available tags in the repo
$ git tag --list

# display tag details for the tag named "v1.4-lwtag"
$ git show v1.4-lwtag
```