DN 4.0 JAVA FSE SOLUTIONS – WEEK 8

Skill: GIT

Hands-On 5

Objective

This document outlines the steps taken to clean up the local repository after a merge, pull the latest changes from the remote, and push all local commits to the remote Git repository.

Step 1: Verify and Pull Changes

Before pushing, it's important to ensure the local repository is in a clean state and up-to-date with the remote.

- **Verify master branch status:** I first verified that my master branch had a clean working tree. The output showed that the branch was ahead of the remote by several commits from the previous exercises.
    - **Command:**
    - git status

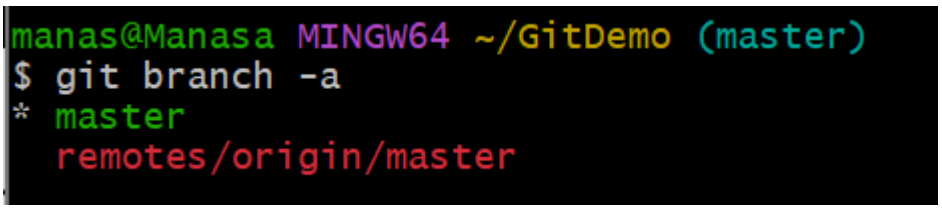
    Output:

```
manas@Manasa MINGW64 ~/GitDemo (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 5 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```

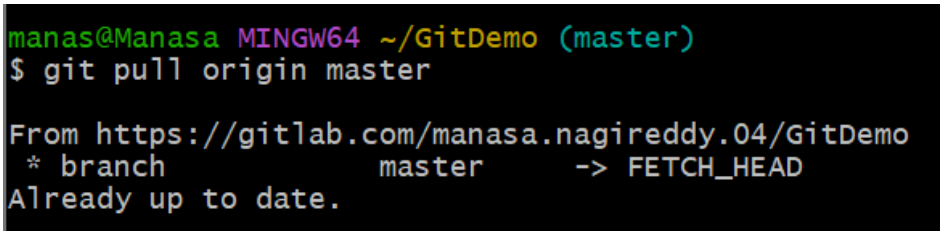
- **List all branches:** I used the -a flag to list all local and remote branches to confirm that the GitWork branch from the previous exercise was successfully deleted locally.
    - **Command:**
    - git branch -a

Output:



- **Pull the remote repository:** I executed the git pull command to fetch and merge any potential changes from the remote repository. In this case, there were no new changes from the remote, so it was already up-to-date.
    - **Command:**
    - git pull origin master

Output:



Step 2: Push Changes to Remote

Now that the local repository is clean and up-to-date, I pushed all pending commits to the remote GitLab repository.

- **Push the changes:** I used the git push command to upload all local commits from the previous exercises to the remote repository.
    - **Command:**
    - git push origin master

Output:

```
manas@Manasa MINGW64 ~/GitDemo (master)
$ git push origin master
Enumerating objects: 17, done.
Counting objects: 100% (17/17), done.
Delta compression using up to 12 threads
Compressing objects: 100% (12/12), done.
Writing objects: 100% (15/15), 1.42 KiB | 1.42 MiB/s, done.
Total 15 (delta 6), reused 0 (delta 0), pack-reused 0 (from 0)
To https://gitlab.com/manasa.nagireddy.04/GitDemo.git
   580210e..102c3f7  master -> master
```

Step 3: Observe the Changes in the Remote Repository

I then verified that all changes, including the merge and conflict resolution commits, were successfully reflected in the remote GitLab repository.

- **Verify on GitLab:** I navigated to the GitLab project page in a web browser. The commit history and the updated hello.xml and .gitignore files were all present.

  Output: