| 5. Git-HOL |
| --- |

1. Introduction and Objectives

This lab walks you through the final step of the Git workflow: pushing your completed local work to a remote repository. At this stage, you’ve already created branches, made commits, merged changes, and resolved conflicts locally. Now, you will synchronize those changes with the remote repository so they are stored centrally and available to collaborators.

Objectives:

- Clean up any unused branches in your local repository.

- Push all local changes to the remote Git repository.

2. Prerequisites

- You have successfully completed the previous lab “Git-T03-HOL\_002”.

- You have a local Git repository with a configured remote named `origin`.

- You are on a system with Git installed and configured.

Step 1: Verify the Master Branch is in a Clean State

Before pushing, you must ensure that your `master` branch does not have uncommitted changes. This ensures that only finalized, saved work is pushed.

Command:

git status

Expected result:

- Output shows you are on the `master` branch.

- Working tree clean (no modified or untracked files).

Step 2: List All Available Branches

Check that all temporary feature branches (for example, `GitNewBranch` and `GitWork`) have been deleted after merging. This prevents clutter in your repository.

Command:

git branch

Expected result:

- Only the `master` branch should be listed.

- If extra branches are listed, delete them using:

git branch -d <branch\_name>

Step 3: Pull from the Remote Repository

Pulling before pushing ensures you have the latest updates from the remote repository. This prevents overwriting others’ changes and avoids unnecessary conflicts.

Command:

git pull origin master

Expected result:

- If no one else has made changes, Git should display “Already up to date.”

- If there are changes, you may need to merge them before proceeding.

Step 4: Push All Local Changes to the Remote Repository

Now push your local `master` branch to the remote repository (`origin`). This sends all committed changes to the server.

Command:

git push origin master

Notes:

- You may be prompted for your Git username and password or token.

- Once complete, your commits will be stored on the remote server.

Step 5: Verify the Changes in the Remote Repository

Open your remote repository’s URL (e.g., GitHub or GitLab) in a web browser to confirm your push was successful.

Check the following:

1. The new files you created (such as `index.html`) should now be listed.

2. In the “Commits” or “History” tab, you should see all commits made locally during the previous labs.

3. Any merge commits should also be visible, reflecting the resolution of previous conflicts.

By completing these steps, you have finalized your local Git work and synchronized it with your remote repository. This ensures a proper backup of your work and enables others to see and build upon your changes.