Git Exercise

Exercise 1: Initializing a Git Repository

1. Create a new directory called "my_repo".
2. Navigate into the "my_repo" directory.
3. Initialize a Git repository in the "my_repo" directory.

Exercise 2: Adding and Committing Changes

1. Create a new file called "index.html" in the "my_repo" directory.
2. Add some content to the "index.html" file.
3. Add the "index.html" file to the staging area.
4. Commit the changes with a meaningful commit message.

Exercise 3: Checking Status and Viewing Commit History

1. Check the status of your Git repository.
2. View the commit history of your Git repository.
3. Identify the commit that corresponds to the last exercise.

Exercise 4: Creating and Switching Between Branches

1. Create a new branch called "feature_branch".
2. Switch to the "feature_branch" branch.
3. Create a new file called "style.css" in the "my_repo" directory.
4. Add the "style.css" file to the staging area.
5. Commit the changes to the "feature_branch" branch.

Exercise 5: Merging Branches

1. Switch back to the main branch.
2. Merge the "feature_branch" into the main branch.
3. Resolve any merge conflicts that may arise.

Exercise 6: Working with Remote Repositories

1. Create a new repository on a remote hosting service (e.g., GitHub).
2. Connect your local repository to the remote repository.
3. Push the changes from your local repository to the remote repository.
4. Clone the remote repository to a new directory.

Exercise 7: Fetching and Pulling Changes

1. Make changes to the "index.html" file in the remote repository.
2. Fetch the changes from the remote repository to your local repository.
3. Merge the changes from the remote repository into your local repository using the "git pull" command.

Next page ……

Exercise 8: all command exercise

1. Create a new repository on GitHub. Name it "my-first-repo".
2. Clone the repository to your local machine using the command line.
3. Create a new file called "README.md" in the root directory of the cloned repository.
4. Add some content to the README file. For example, you could write a brief description of the repository or instructions for how to use it.
5. Use Git to add the README file to the repository, commit the changes, and push the changes to GitHub.
6. Check that the changes have been pushed to the GitHub repository by visiting the repository's page on GitHub.
7. Create a new branch in the repository called "feature-branch".
8. Switch to the new branch using Git.
9. Create a new file in the repository called "feature-file.txt".
10. Add some content to the file. For example, you could write a list of features that you plan to add to the repository.
11. Use Git to add the new file to the repository, commit the changes, and push the changes to GitHub.
12. Check that the changes have been pushed to the "feature-branch" by visiting the branch's page on GitHub.
13. Merge the changes from the "feature-branch" into the "master" branch using Git.
14. Push the merged changes to GitHub.
15. Check that the changes have been merged into the "master" branch by visiting the branch's page on GitHub.