1. GitHub Repository Setup

Question:

Create a new GitHub repository for a personal project. Initialize the repository with a README.md file. Then, clone the repository to your local machine, make some changes to the README.md (e.g., add a project description), and push the changes back to GitHub.

Tasks:

Create a GitHub repository.

Clone the repository to your local machine.

Edit the README.md file.

Stage, commit, and push your changes to GitHub.

1. Cloning a Repository

Question:

Clone an existing GitHub repository to your local machine and make a small change in one of the files.

Tasks:

Choose an existing public GitHub repository (e.g., your own or a sample repository).

Clone the repository to your local machine using git clone <repository\_url>.

Navigate into the cloned repository directory and open one of the files.

Make a small edit, such as modifying a line of text or fixing a typo.

Save your changes.

3. Pushing Changes to GitHub

Question:

After modifying a file in a cloned repository, commit the changes locally and push them to GitHub.

Tasks:

Clone a repository (from question 1).

Modify a file (e.g., README.md).

Stage the changes with git add.

Commit the changes with git commit -m "Updated README".

Push the changes to GitHub using git push.

4. Pulling the Latest Changes

Question:

You cloned a repository earlier, and now another collaborator has made changes to it. You need to pull the latest changes from GitHub to keep your local copy up to date.

Tasks:

Clone a repository (from previous questions).

Ask a collaborator to make a change and push it to the GitHub repository.

Use git pull origin main to pull the latest changes.

Verify that the changes made by your collaborator appear in your local repository.

5. Branching and Merging

Question:

In your GitHub repository, create a new branch called feature-login. Implement a simple login function in a login.py file. After completing the function, create a pull request to merge feature-login into the main branch.

Tasks:

Create a new branch feature-login.

Add a login.py file with a simple login function.

Commit the changes and push them to GitHub.

Create a pull request from feature-login to main and merge it.

6. Collaborating on GitHub

Question:

You and your team are working on a project. One team member adds a new feature to the project. You need to fetch the latest changes, create a new branch, and implement your feature without affecting the existing code.

Tasks:

Fork a repository (you can use an existing repository or a partner’s repository).

Clone the repository to your local machine.

Create a new branch for your feature.

Implement your changes and push them to your forked repository.

Submit a pull request to the original repository.