Assignment 1:Initialize a new Git repository in a directory of your choice. Add a simple text file to the repository and make the first commit.

These are the following steps:

1. We will create a directory where you want to initialize the Git repository.
2. Then we will open the command prompt and navigate to the directory which we created. **cd path/to/your/directory**
3. Then we will use the **git init** command to initialize a new Git repository to the chosen directory.
4. Create a simple text file in the directory and create the file.
5. Then **git add** command to stage the text file for commit.
6. Commit the changes to the repository using the **git commit** command. This command will open a text editor where you can enter a commit message.
7. Alternatively, you can directly add a commit message using the **git commit -m.**
8. After committing, you can use **git log** to view.

Assignment 2: Create a new branch named 'feature' and switch to it. Make changes in the 'feature' branch and commit them.

These are the following steps:

1.We will create a new branch using the **git branch** command followed by the branch name to create a new branch.
2. Switch to the newly created branch, use the **git checkout** command.
3.Alternatively, you can combine branch creation and switching into one command using the **git checkout -b command.**
4.We will make changes to our files.we can use any text editor.
5.After making changes, stage them for commit using the **git add** command.
6.Commit the changes to the branch using the **git commit** command.

Assignment 3: Create a 'hotfix' branch to fix an issue in the main code. Merge the 'hotfix' branch into 'main' ensuring that the issue is resolved.

These are the following steps:

1.Create a 'hotfix' branch using the **git checkout** command with the -b flag to create and switch to the 'hotfix' branch.
2.Make the necessary changes in the code.
3.Once the issue was fixed commit the changes in the hotfix branch by using **git add.** And **git commit -m.**

4.Before merging the 'hotfix' branch into 'main', switch back to the 'main' branch by using **git checkout main.**
5.Merge the changes from the 'hotfix' branch into the 'main' branch using the **git merge** command.
6.If there are any merge conflicts, resolve them manually. Git will indicate which files have conflicts. After resolving conflicts, stage the changes and continue the merge process by using the **git add. And git merge** commands.
7.Once the merge is complete, verify the changes.
8.Finally, push the changes to your remote repository the 'main' branch by using **git push origin.**