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.

1.Open your terminal (or Git Bash on Windows).

Navigate to the directory where you want to create the repository. Use the cd command to change directories. For example:

cd /path/to/your/project

2.Initialize the Git repository. Use the git init command:

git init

3.Navigate to your project directory:

cd /path/to/your/project

Initialize the Git repository:

git init

After running this command, you'll see a message like:

Initialized empty Git repository in /path/to/your/project/.git/

4.Optionally, add your files to the repository:

git add .

This command stages all files in the directory for the initial commit.

5.Commit the files to the repository:

git commit -m "Initial commit"

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

1. **Open your terminal**.

2. **Navigate to your project directory** (if you're not already there):

   cd /path/to/your/project

3. **Create a new branch named 'feature'**:

   git branch feature

   4.**Switch to the 'feature' branch**:

   git checkout feature

   Alternatively, you can create and switch to the 'feature' branch in one command:

   git checkout -b feature

   5.**Make changes in the 'feature' branch**. For example, let's create a new file named **feature.txt** and add some content to it:

   echo "This is a feature branch" > feature.txt

   6.**Add the changes to the staging area**:

   git add feature.txt

   7.**Commit the changes**:

   git commit -m "Add feature.txt with initial content"

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

1.Open your terminal.

2.Navigate to your project directory (if you're not already there):

    cd /path/to/your/project

3.Create a new branch named 'feature':

    git branch feature

4.Switch to the 'feature' branch:

    git checkout feature

   Alternatively, you can create and switch to the 'feature' branch in one command:

    git checkout -b feature

5.Make changes in the 'feature' branch. For example, let's create a new file named feature.txt and add some content to it:

    echo "This is a feature branch" > feature.txt

6.Add the changes to the staging area:

    git add feature.txt

7.Commit the changes:

    git commit -m "Add feature.txt with initial content"