### Feature Branch Creation and Merging

**Scenario:**

You are contributing to a project called NewFeatureProject. The project manager has asked you to develop a new feature that involves creating a file named feature-details.txt.

This file will contain important information about the feature you're working on. To safely develop this new feature, you decide to create a new branch, add your changes, and then merge these changes back into the main branch.

**Task:**

Complete the commands below to successfully create the feature branch, add the new file, and merge your changes back into the main branch.

1. **Create a New Feature Branch and Switch to It:**

You decide to name your branch feature-info. To create and switch to this new branch, you use the following command:

**Type your answer here**

| git branch feature-info. |
| --- |

2. **Add the New Feature File:**

After creating feature-details.txt and adding the necessary feature information, you stage the file for commit with:

**Type your answer here**

| Git add . |
| --- |

3. **Commit Your Changes:**

You're ready to commit your staged changes. You decide on a commit message that clearly describes your addition. To commit, you use:

**Type your answer here**

| Git commit -m "changes" |
| --- |

4. **Switch Back to the Main Branch:**

Now that your feature is ready, you need to switch back to the main branch to prepare for merging your changes. The main branch is named main. To switch back, you use:

**Type your answer here**

| git branch --move main |
| --- |

5. **Merge Your Feature Branch into the Main Branch:**

With your feature complete and tested, it's time to merge feature-info into the main branch. To do this, you use the command:

**Type your answer here**

| git merge master |
| --- |