Use of GitHub for Assignment

|  |
| --- |
| 1. **Accept link** to git-hub repository from sender. 2. Open git-bash on windows or bash on mac. **Move to directory** where you would like to store the git repository. E.g. cd Desktop 3. **Clone the repository**    1. git clone https://github.com/minogud2/softEngAssignment4    2. If problems encountered, make sure the SSH key is established before cloning repository. 4. Once established, **cd into the new directory**    1. cd softEngAssignment4 5. **Check out the existing branches**    1. git branch    2. A list of four branches should appear(see tree)- If you still see the previous branches I set up (base and working tree), git remote prune origin. If that doesn’t work then forcibly delete them by doing the following: git branch -D base and git branch -D WorkingTree 6. **Log into your branch**    1. git checkout Darragh    2. Bash will then display your change. See below.      1. **Begin working.**     1. git add .– adds the files and stations them for commit.    2. git commit -m “This is my first commit” - Ensure all commits are relative to what you are doing. Everyone shouldn’t be writing- version1 commit or initial commit. Instead- “Json file parsed into database. Committing database- version 1”.    3. git status - check that the files have been updated.    4. git push origin master- Pushes your completed work to the master for merging. 2. **Master will merge changes.**    1. As git administrator for the project, I will handle the merges for the time being. The changes will be merged by doing:    2. git checkout master    3. git merge Darragh |

Other useful information

|  |
| --- |
| **Adding branches**   * git branch NewBranchName   **Deleting Branches**   * git branch -d NewBranchName- deletes if there are no merge conflicts. * git branch -D NewBranchName- forcibly deletes irrespective of merge issues.   **Merging upstream changes into your local repository**   * git pull <remote>- does a git fetch and a git merge all in one**.** Brings the local branch up to date with the remote branches/master. It automatically merges changes without reviewing them first. May run into conflicts if branches not closely managed.   **Tutorials:**  For all aspects of Git:  <https://www.atlassian.com/git/tutorials> |

Deductive Process