1. Create folder SampleProject inside SampleDemo
2. Open git bash here
3. Commands :
   1. git init (*for making SampleProject a git repo)*

*(after this created TestGitProject in Eclipse)*

* 1. git add . *( for adding all files to staging area ..all java project files got added to staging area)*
  2. git status

(*after this added steps word doc in project dir)*

* 1. git status

* 1. git add steps.docx
  2. git commit (*opens a text editor ..type commit message there……follow https://www.javatpoint.com/git-commit )*

*Then created remote repo: “git-collaboration”*

* 1. git remote add origin <https://github.com/joshisangeeta/git-collaboration.git>

*(this is for adding remote to local repo)*

* 1. git branch -M main *(-M is a flag (shortcut) for --move --force per the docs page on git branch . It renames the branch main (since the default branch name for repositories created using the command line is master , while those created in GitHub)*
  2. git push -u origin main

BRANCHING

1. git checkout main (*For our particular example, we only have our main branch at this point, so this command isn't necessary. But in general, it's important to ensure you are on the main branch before creating a new branch.)*
2. git branch login *(This branch command creates a branch with the specified name from the current branch. In this case, the branch is named login.)*
3. git checkout login ( *The checkout command checks out the branch so you can work with it.)*

or

git checkout -b login *(The -b option creates a new branch with the provided name and checks out that branch.)*

1. git branch *(The branch command lists all existing branches in the repository.)*

*(add login model class from eclipse.)*

1. git add . *(to stage the changes )*
2. git rm steps\_forBranching.docx --cached *(for removing file only from staging area…to remove from working directory---use rm without git)*
3. git commit -m “new branch for adding login feature”
4. git checkout main  *( we want to create another branch for another urgent task…register…for that first switch to main branch)*
5. git checkout -b register  *(create a new branch & check out that branch)*
6. git branch *(check all branches and the current branch)*

*(then added register model class in project)*

1. git add . *(to stage the changes:- registermodel)*
2. git commit -m “fixing register issues”
3. git log *( to verify)*