**Git – In class Practice**

1. Install Git
2. Ubuntu : sudo apt-get install git
3. Windows: <https://gitforwindows.org/>
4. Mac : <https://git-scm.com/download/mac>
5. Open Git Bash
6. Check git version: git –version
7. Setup git configuration:

*git config --global user.name “your name”*

*git config --global user.email* [*yourname@example.com*](mailto:yourname@example.com)

1. Check the settings:

*git config – list*

1. Set default editor ( e.g. notepad):

*git config --global core.editor <“editor alias”>*

1. Try *git config help*, *git config –h* to see how help can be sought.
2. Create a folder – Gitexercises in your system.
3. Change the directory to Gitexercises in git bash
4. Initialize a local repository: *git init*
5. See the status of repository with *git status*
6. Create a new file called index.html : *touch index.html*
7. Open index.html: *notepad index.html*
8. Add a line - Hello World
9. Check status: *git status*
10. Add the untracked files : *git add .*
11. Commit the changes with appropriate message: *git commit –m “added index file”*
12. Check status: *git status*
13. Create an account in Git hub.
14. Create a repository in Git hub: TestGit
15. Map local repository with remote repository on github: *git remote add origin <Github Repo Url>*
16. View the remote repository: *git remote –v*
17. Push the changes to remote repository: *git push -u origin master*
18. Create a new file at local repository: “ignore.html”.: *touch ignore.html*

*Notepad ignore.html*

*Add a line: This file will be ignored*

1. Create a new file “.gitignore”
2. See what the current state . *git status*
3. Add ignore.css in ignore list inside .gitignore
4. See what the current state. ( You will see that ignore.html is not tracked).
5. Add your file . *git add .*
6. Commit your code with proper commit message. g*it commit –m “added .gitignore”*
7. Push the changes to remote repository: *git push -u origin master* (You will see that ignore.hml is not pushed)
8. Make some changes to index.html. Add your name.
9. Add and commit with appropriate message.
10. Check the difference between the changed local repository and the remote repository:

*git fetch origin master*

*git diff*

1. Cone your repository to a local folder.

**Self-Practice for Students:**

1. Go to the folder where the Java program of previous class is saved: JavaPrograms.
2. Initialize git repository.
3. Define .gitignore and include classes folder in it.
4. Make a repository named Java Programs
5. Push the Program of the previous class on git hub.
6. Verify that the code is appropriately pushed without classes folder.