WEEK 8 – GIT

EXERCISE 1

Step 1: Setting up machine with Git configuration

git –version

git config --global user.name "Retika"

git config --global user.email “[retikasundaram@gmail.com](mailto:retikasundaram@gmail.com)”

git config –list

Step 2: Integrate Notepad++

notepad++

alias np='notepad++'

git config --global core.editor "'C:/Program Files/Notepad++/notepad++.exe' -multiInst -notabbar -nosession -noPlugin"

git config --global –e

Step 3- Adding a file

mkdir GitDemo

cd GitDemo

git init

ls –a

echo "Welcome to Git Repository" > welcome.txt

ls

cat welcome.txt

git status

git add welcome.txt

git commit

Initial commit

- Added welcome.txt

- Contains a welcome message

git status

Step 4- Pushing local repo

git remote add origin [https://gitlab.com/Retika /GitDemo.git](https://gitlab.com/Retika%20/GitDemo.git)

git pull origin master

git push -u origin master

Exercise 2

cd GitDemo

echo "This is a log file" > debug.log

mkdir log

echo "Log folder file" > log/info.txt

notepad++ .gitignore

\*.log

log/

git status

git add .gitignore

git commit -m "Add .gitignore to exclude .log files and log/ folder"

git push origin master

Exercise 3

git branch GitNewBranch

git branch –a

git checkout GitNewBranch

echo "This is a new branch file" > newbranch.txt

git add newbranch.txt

git commit -m "Added newbranch.txt in GitNewBranch"

git status

git checkout master

git diff master GitNewBranch

git config --global diff.tool p4merge

git config --global merge.tool p4merge

git config --global mergetool.p4merge.path "C:/Program Files/Perforce/p4merge.exe"

git difftool master GitNewBranch

git merge GitNewBranch

git log --oneline --graph –decorate

git branch -d GitNewBranch

git status

git push origin GitNewBranch

Exercise 4

git checkout master

git status

git checkout -b GitWork

echo "<message>Hello from GitWork branch</message>" > hello.xml

git status

git add hello.xml

git commit -m "Add hello.xml in GitWork branch"

git checkout master

echo "<message>Different content from master branch</message>" > hello.xml

git add hello.xml

git commit -m "Add hello.xml in master with different content"

git log --oneline --graph --decorate –all

git diff GitWork

git difftool GitWork

git merge GitWork

<<<<<<< HEAD

<message>Different content from master branch</message>

=======

<message>Hello from GitWork branch</message>

>>>>>>> GitWork

git mergetool

git add hello.xml

git commit -m "Resolved merge conflict in hello.xml"

echo "\*.orig" >> .gitignore

git add .gitignore

git commit -m "Ignore merge backup files"

git branch

git branch -d GitWork

git log --oneline --graph –decorate

Exercise 5

git checkout master

git status

git branch –a

git pull origin master

git push origin master