**WEEK 8**

**Module 11 – Version control – GIT**

**4.Git-HOL**

**1. Go to your repository directory**

*cd D:\CTS\WEEK-8\Ex4*

**2. Verify if master is in clean state**

*git checkout master*

*git status*

**3. Create a branch GitWork and switch to it**

*git checkout -b GitWork*

**4. Add a file hello.xml**

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

**5. Check status**

*git status*

**6. Stage and commit changes**

*git add hello.xml*

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

**7. Switch to master**

*git checkout master*

**8. Add a file hello.xml in master with different content**

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

**9. Stage and commit changes**

*git add hello.xml*

*git commit -m "Add hello.xml in Master branch"*

**10. View commit history (all branches)**

*git log --oneline --graph --decorate --all*

**11. View CLI differences**

*git diff master GitWork*

**12. View differences using P4Merge**

*git difftool master GitWork*

**13. Merge branch into master (this will cause conflict)**

*git merge GitWork*

**14. Resolve the conflict**

* Open hello.xml in your editor (e.g., Notepad++).
* You’ll see conflict markers like:
* Edit the file to keep the correct final version.
* Save the file.

**15. Mark the conflict as resolved**

*git add hello.xml*

**16. Commit the merge**

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

**17. Add backup file to .gitignore**

*echo "\*.bak" >> .gitignore*

**18. Stage and commit .gitignore**

*git add .gitignore*

*git commit -m "Add .bak files to .gitignore"*

**19. List all branches**

*git branch*

**20. Delete the merged branch**

*git branch -d GitWork*

**21. View final commit graph**

*git log --oneline --graph –decorate*