Git Hands-on Lab: Conflict Resolution (Git-T03-HOL\_001)

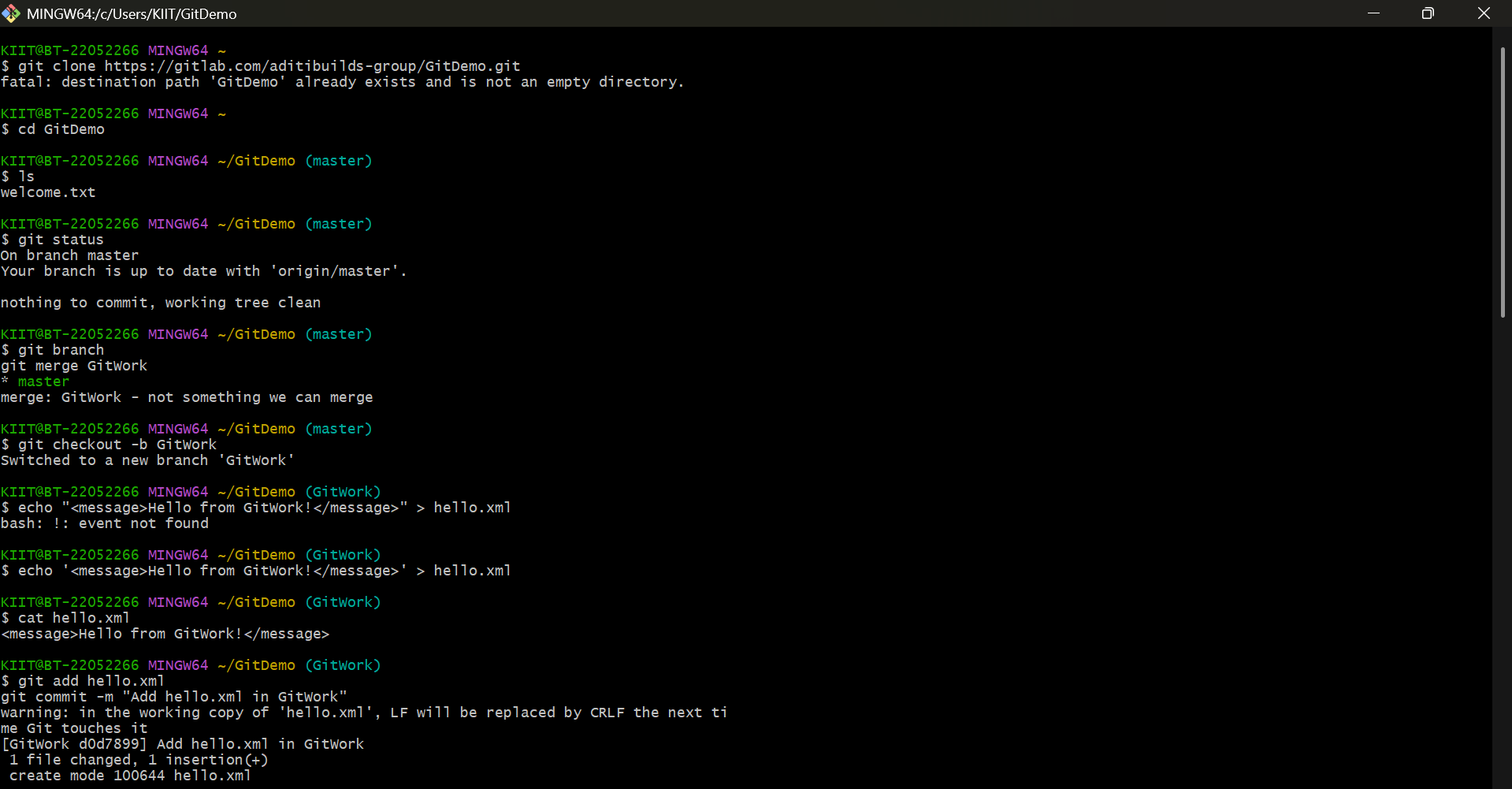
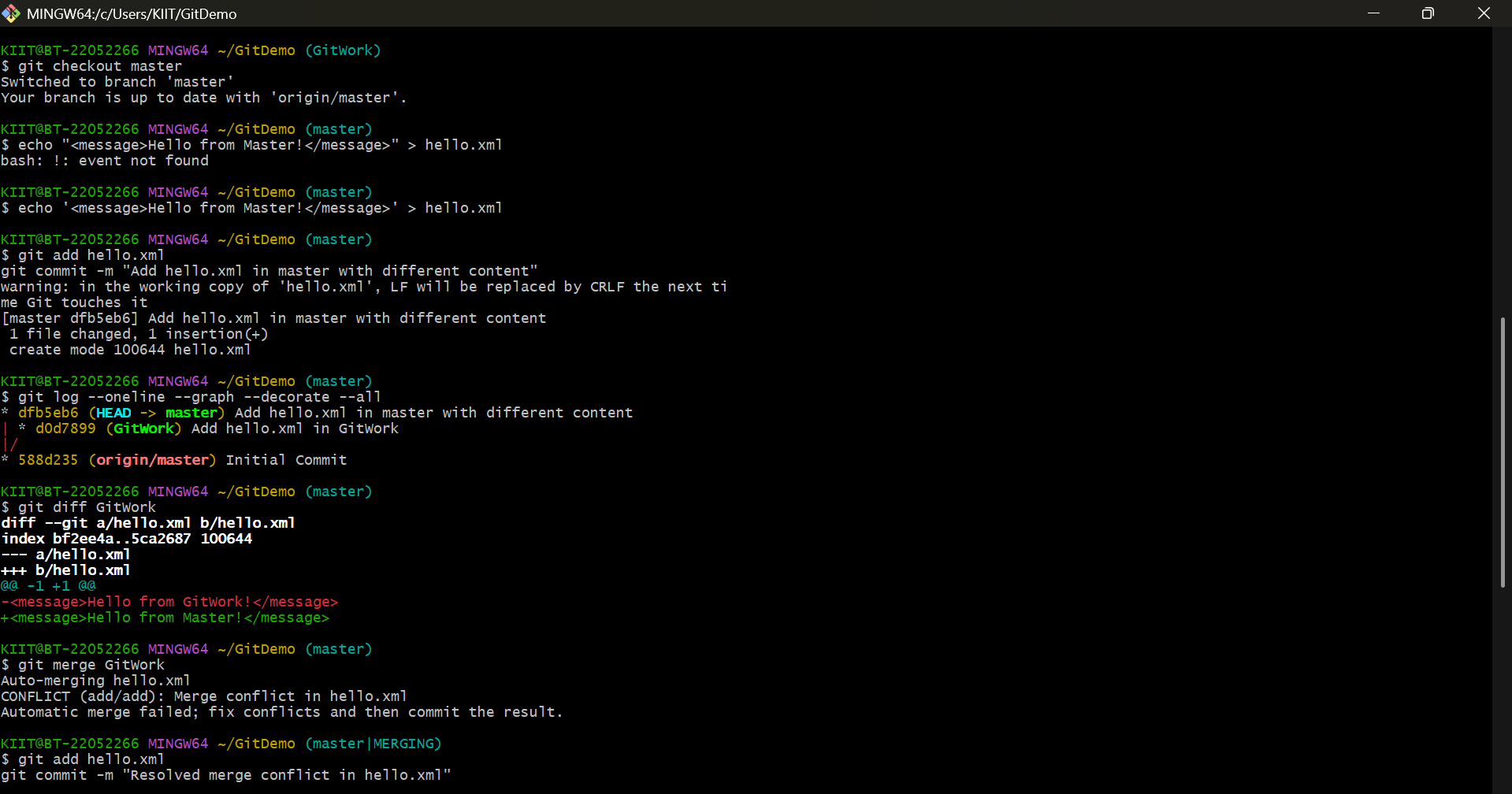
# Objectives

• Explain how to resolve the conflict during merge.  
• Implement conflict resolution when multiple users are updating the trunk (or master) in such a way that it results into a conflict with the branch’s modification.

# Prerequisites

• Hands-on ID: “Git-T03-HOL\_001”  
• Please follow the below steps for creating a free account in GitHub.  
• Do not use cognizant credentials to login to GitHub.  
• Estimated time to complete this lab: 30 minutes.

# Step-by-step Instructions

1. 1. Verify if master is in clean state:  
    git status
2. 2. Create a branch “GitWork”. Add a file “hello.xml”:  
    git checkout -b GitWork  
    echo '<message>Hello from GitWork!</message>' > hello.xml
3. 3. Update the content of “hello.xml” and observe the status:  
    echo '<message>Updated from GitWork</message>' > hello.xml  
    git status
4. 4. Commit the changes to reflect in the branch:  
    git add hello.xml  
    git commit -m 'Added hello.xml in GitWork'
5. 5. Switch to master:  
    git checkout master
6. 6. Add a file “hello.xml” to the master and add different content:  
    echo '<message>Hello from master!</message>' > hello.xml
7. 7. Commit the changes to the master:  
    git add hello.xml  
    git commit -m 'Added hello.xml in master'
8. 8. Observe the log:  
    git log --oneline --graph --decorate --all
9. 9. Check the differences:  
    git diff master GitWork
10. 10. (Optional) Use P4Merge or another visual diff tool to compare branches.
11. 11. Merge the branch into the master:  
     git merge GitWork
12. 12. Observe the git markup indicating conflicts.
13. 13. Use any editor or 3-way merge tool to resolve the conflict in hello.xml.
14. 14. After resolving the conflict:  
     git add hello.xml  
     git commit -m 'Resolved merge conflict in hello.xml'
15. 15. Observe the git status and ignore backup file:  
     echo '\*.orig' >> .gitignore
16. 16. Commit the .gitignore changes:  
     git add .gitignore  
     git commit -m 'Ignore backup files'
17. 17. List all branches:  
     git branch
18. 18. Delete the merged branch:  
     git branch -d GitWork
19. 19. Final log visualization:  
     git log --oneline --graph --decorate
20. 
21. 
22. 