**4. Git-HOL**

**Objective:** Create a branch file and master file with the same file name, cause a merge conflict, then resolve the merge conflict and push to the remote repo.

**Step-1:** Master is clean

**>git status**

**Step-2:** Create new branch

**>git branch GitWork  
>git checkout GitWork**

**Step-3:** Add file in branch

**>echo "<msg>Hello from GitWork branch</msg>" > hello.xml  
>git add hello.xml  
>git commit -m "Add hello.xml in GitWork"**

**Step-4:** Move to master

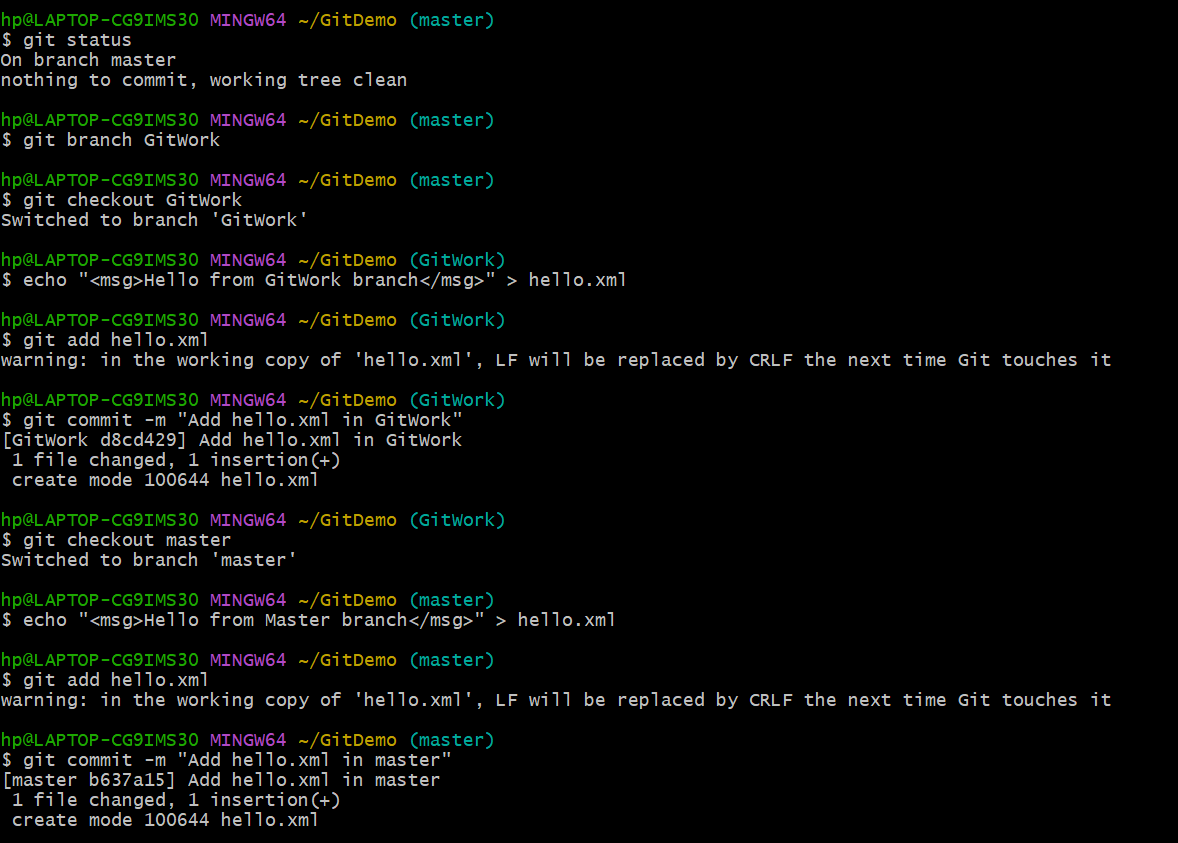
**>git checkout master**

**Step-5**: Add >

**>echo "<msg>Hello from Master branch</msg>" > hello.xml  
>git add hello.xml  
>git commit -m "Add hello.xml in master"**

**Step-6:** Merge branch into master

**>git merge GitWork**



**Step-7:** Resolve conflict

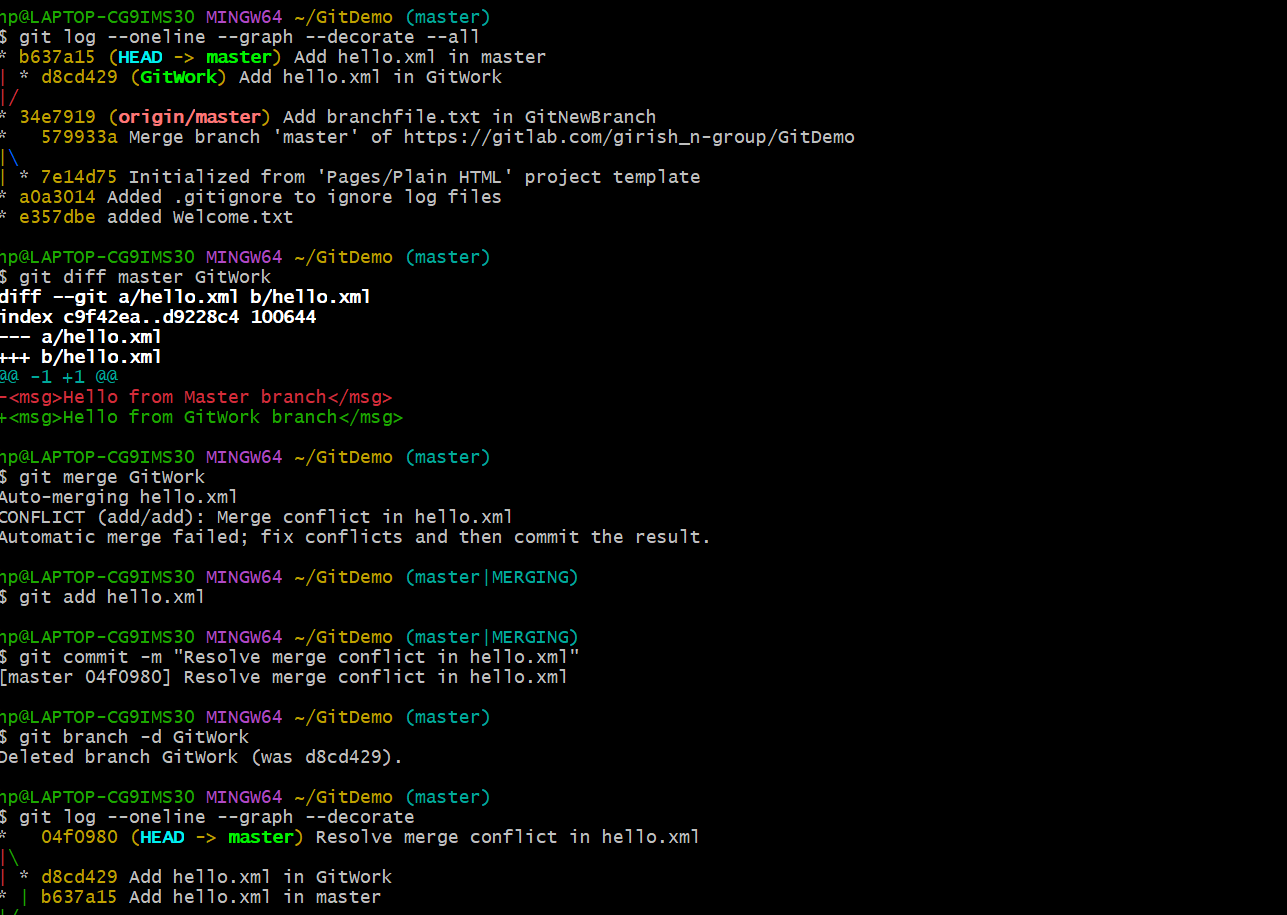
* Remove Git conflict markers (<<<<<<<, =======, >>>>>>>)
* Keep desired final content, for example:  
  <msg>Hello from both Master and GitWork branches</msg>

**Step-8:** Commit final file

**>git add hello.xml  
>git commit -m "Resolve merge conflict in hello.xml"**

**Step-9:** Delete merged branch

**>git branch -d GitWork**



**Step-10:** Push to remote

**>git push origin master**

**OUTPUT:**

