**Git**

HandsOn 4

**1.Explain how to resolve the conflict during merge.**

**1. Identify the conflict**

After running:

git merge <branch>

You might see:

CONFLICT (content): Merge conflict in hello.xml

Automatic merge failed; fix conflicts and then commit the result.

Check the status:

git status

**2. Open the conflicted file**

**Inside, you’ll see conflict markers inserted by Git:**

<<<<<<< HEAD

<greeting>Master version</greeting>

=======

<greeting>Branch updated</greeting>

>>>>>>> GitWork

**3. Decide the final content**

You have three options:

1. Keep your version (remove the other branch’s changes).
2. Keep their version (replace yours with theirs).
3. Combine both changes in a way that makes sense.

Example combined resolution:

<greeting>Master and Branch merged together</greeting>

**4. Remove conflict markers**

Delete all lines with:

<<<<<<<

=======

>>>>>>>

so that the file contains only valid content.

**5. Mark the conflict as resolved**

git add hello.xml

**6. Commit the merge**

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

HP@HP MINGW64 ~ (master)

$ mkdir Git-handson4

HP@HP MINGW64 ~ (master)

$ cd Git-handson4

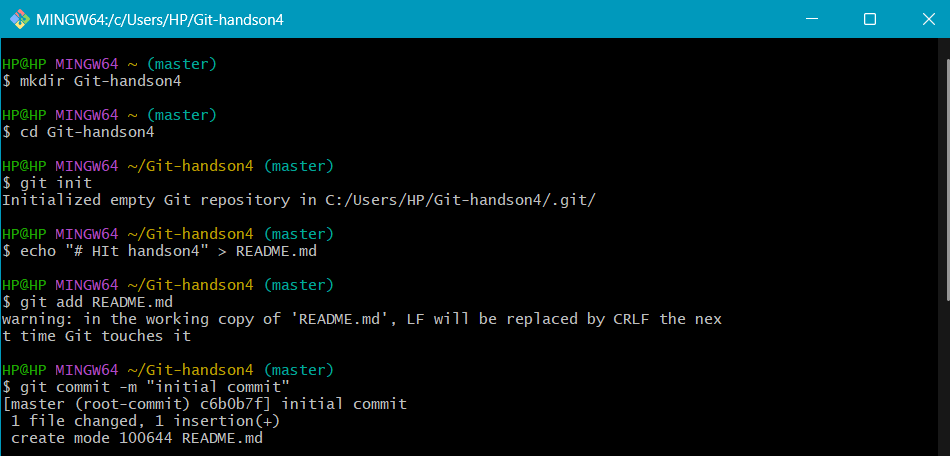
HP@HP MINGW64 ~/Git-handson4 (master)

$ git init

Initialized empty Git repository in C:/Users/HP/Git-handson4/.git/

HP@HP MINGW64 ~/Git-handson4 (master)

$ echo "# HIt handson4" > README.md



HP@HP MINGW64 ~/Git-handson4 (master)

$ git add README.md

warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

HP@HP MINGW64 ~/Git-handson4 (master)

$ git commit -m "initial commit"

[master (root-commit) c6b0b7f] initial commit

1 file changed, 1 insertion(+)

create mode 100644 README.md

HP@HP MINGW64 ~/Git-handson4 (master)

$ git status

On branch master

nothing to commit, working tree clean

HP@HP MINGW64 ~/Git-handson4 (master)

$ git checkout -b GitWork

Switched to a new branch 'GitWork'

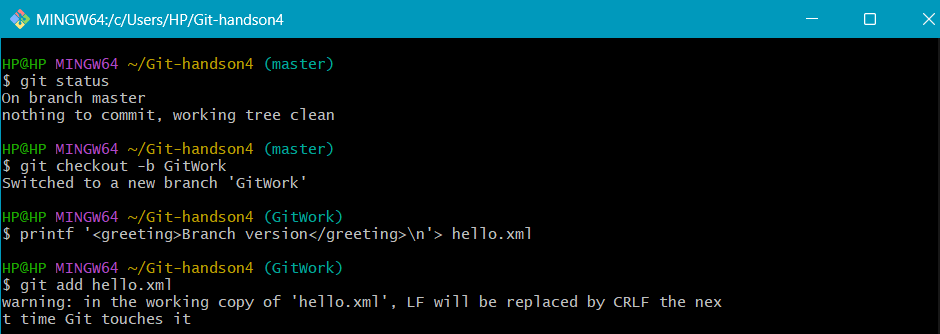
HP@HP MINGW64 ~/Git-handson4 (GitWork)

$ printf '<greeting>Branch version</greeting>\n'> hello.xml

HP@HP MINGW64 ~/Git-handson4 (GitWork)

$ git add hello.xml

warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git touches it



HP@HP MINGW64 ~/Git-handson4 (GitWork)

$ git commit -m "Add hello.xml on GitWork Branch"

[GitWork ae2bd50] Add hello.xml on GitWork Branch

1 file changed, 1 insertion(+)

create mode 100644 hello.xml

HP@HP MINGW64 ~/Git-handson4 (GitWork)

$ printf '<greeting>Branch updated</greeting>\n'>hello.xml

HP@HP MINGW64 ~/Git-handson4 (GitWork)

$ git status

On branch GitWork

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

modified: hello.xml

no changes added to commit (use "git add" and/or "git commit -a")

HP@HP MINGW64 ~/Git-handson4 (GitWork)

$ git add hello.xml

warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git touches it

HP@HP MINGW64 ~/Git-handson4 (GitWork)

$ git commit -m "Update hello.xml on GitWork branch"

[GitWork f54644c] Update hello.xml on GitWork branch

1 file changed, 1 insertion(+), 1 deletion(-)

HP@HP MINGW64 ~/Git-handson4 (GitWork)

$ git checkout master

Switched to branch 'master'

HP@HP MINGW64 ~/Git-handson4 (master)

$ printf '<greeting>Master version</greeting>\n' > hello.xml

HP@HP MINGW64 ~/Git-handson4 (master)

$ git add hello.xml

warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git touches it

HP@HP MINGW64 ~/Git-handson4 (master)

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

[master 5f57485] Add hello.xml on master with different content

1 file changed, 1 insertion(+)

create mode 100644 hello.xml

HP@HP MINGW64 ~/Git-handson4 (master)

$ git log --oneline --graph --decorate --all

\* 5f57485 (**HEAD** -> **master**) Add hello.xml on master with different content

| \* f54644c (**GitWork**) Update hello.xml on GitWork branch

| \* ae2bd50 Add hello.xml on GitWork Branch

|/

\* c6b0b7f initial commit

HP@HP MINGW64 ~/Git-handson4 (master)

$ git diff master..GitWork -- hello.xml

**diff --git a/hello.xml b/hello.xml**

**index 9d27880..73f4bd5 100644**

**--- a/hello.xml**

**+++ b/hello.xml**

@@ -1 +1 @@

-<greeting>Master version</greeting>

+<greeting>Branch updated</greeting>

HP@HP MINGW64 ~/Git-handson4 (master)

$ git config --global diff.tool p4merge

HP@HP MINGW64 ~/Git-handson4 (master)

$ git config --global difftool.p4merge.cmd "\"C:/Program Files/Perforce/p4merge.exe\" \"$LOCAL\" \"$REMOTE\""

HP@HP MINGW64 ~/Git-handson4 (master)

$ git config --global merge.tool p4merge

HP@HP MINGW64 ~/Git-handson4 (master)

$ git config --global mergetool.p4merge.cmd "\"C:/Program Files/Perforce/p4merge.exe\" \"$BASE\" \"$LOCAL\" \"$REMOTE\" \"$MERGED\""

HP@HP MINGW64 ~/Git-handson4 (master)

$ git config --global mergetool.prompt false

HP@HP MINGW64 ~/Git-handson4 (master)

$ git checkout master

Already on 'master'

HP@HP MINGW64 ~/Git-handson4 (master)

$ git merge GitWork

Auto-merging hello.xml

CONFLICT (add/add): Merge conflict in hello.xml

Automatic merge failed; fix conflicts and then commit the result.

HP@HP MINGW64 ~/Git-handson4 (master|MERGING)

$ ^C

HP@HP MINGW64 ~/Git-handson4 (master|MERGING)

$ git status

On branch master

You have unmerged paths.

(fix conflicts and run "git commit")

(use "git merge --abort" to abort the merge)

Unmerged paths:

(use "git add <file>..." to mark resolution)

both added: hello.xml

no changes added to commit (use "git add" and/or "git commit -a")

HP@HP MINGW64 ~/Git-handson4 (master|MERGING)

$ git add hello.xml

HP@HP MINGW64 ~/Git-handson4 (master|MERGING)

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

[master 9dff145] Resolved merge conflict in hello.xml

HP@HP MINGW64 ~/Git-handson4 (master)

$ echo "\*.orig" >> .gitignore

HP@HP MINGW64 ~/Git-handson4 (master)

$ git add .gitignore

warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it

HP@HP MINGW64 ~/Git-handson4 (master)

$ git commit -m "Ignore merge backup files"

[master 43976d5] Ignore merge backup files

1 file changed, 1 insertion(+)

create mode 100644 .gitignore

HP@HP MINGW64 ~/Git-handson4 (master)

$ git branch

GitWork

\* master

HP@HP MINGW64 ~/Git-handson4 (master)

$ git branch -d GitWork

Deleted branch GitWork (was f54644c).

HP@HP MINGW64 ~/Git-handson4 (master)

$ git log --oneline --graph --decorate

\* 43976d5 (**HEAD** -> **master**) Ignore merge backup files

\* 9dff145 Resolved merge conflict in hello.xml

|\

| \* f54644c Update hello.xml on GitWork branch

| \* ae2bd50 Add hello.xml on GitWork Branch

\* | 5f57485 Add hello.xml on master with different content

|/

\* c6b0b7f initial commit

HP@HP MINGW64 ~/Git-handson4 (master)

$ git remote add origin https://gitlab.com/dhatchayini/Git-handson4.git

HP@HP MINGW64 ~/Git-handson4 (master)

$ git push -u origin master

Enumerating objects: 18, done.

Counting objects: 100% (18/18), done.

Delta compression using up to 4 threads

Compressing objects: 100% (12/12), done.

Writing objects: 100% (18/18), 1.70 KiB | 45.00 KiB/s, done.

Total 18 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)

remote:

remote: To create a merge request for master, visit:

remote: https://gitlab.com/dhatchayini/git-handson4/-/merge\_requests/new?merge\_request%5Bsource\_branch%5D=master

remote:

To https://gitlab.com/dhatchayini/Git-handson4.git

\* [new branch] master -> master

branch 'master' set up to track 'origin/master'.

HP@HP MINGW64 ~/Git-handson4 (master)

$ git push

Everything up-to-date

