**GIT Hands On 4**

**Step 1:**

Create a new branch

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ git branch --show-current

main

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ git checkout main

Already on 'main'

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ git status

On branch main

nothing to commit, working tree clean

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ git checkout -b GitWork

Switched to a new branch 'GitWork'

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (GitWork)

$ git branch --show-current

GitWork

**Step 2:**

Add an xml file to it

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (GitWork)

$ cat > hello.xml <<'EOF'

<?xml version="1.0"?>

<message>This is hello.xml from GitWork branch (version A)</message>

EOF

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (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

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (GitWork)

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

[GitWork cfc9795] Add hello.xml in GitWork

1 file changed, 2 insertions(+)

create mode 100644 hello.xml

**Step 3:**

Check Status

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (GitWork)

$ git status

On branch GitWork

nothing to commit, working tree clean

**Step 4:**

Add the same file to main branch

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (GitWork)

$ git checkout main

Switched to branch 'main'

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ git status

On branch main

nothing to commit, working tree clean

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ cat > hello.xml <<'EOF'

<?xml version="1.0"?>

<message>This is hello.xml from master branch (version B)</message>

EOF

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ git add hello.xml

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

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ git commit -m "Add hello.xml in master"

[main 8f8dded] Add hello.xml in master

1 file changed, 2 insertions(+)

create mode 100644 hello.xml

**Step 5:**

Check the codes

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

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

\* 8f8dded (HEAD -> main) Add hello.xml in master

| \* cfc9795 (GitWork) Add hello.xml in GitWork

|/

\* 4694d2e Add branchfile.txt in GitNewBranch

\* 4ac0a01 Add .gitignore to ignore .log files and log folder

\* 8b4d33d Initial commit

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ git diff main GitWork -- hello.xml

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

**index 1378691..fb6efa5 100644**

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

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

@@ -1,2 +1,2 @@

<?xml version="1.0"?>

-<message>This is hello.xml from master branch (version B)</message>

+<message>This is hello.xml from GitWork branch (version A)</message>

**Step 6:**

Merge the files

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ 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.

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main|MERGING)

$ git status

On branch main

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")

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main|MERGING)

$ git mergetool hello.xml

Merging:

hello.xml

Normal merge conflict for 'hello.xml':

{local}: created file

{remote}: created file

The merge tool p4merge is not available as 'C:/Path/To/p4merge.exe'

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main|MERGING)

$ git add hello.xml

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main|MERGING)

$ git commit -m "Merge GitWork into main and resolve conflict"

[main c238b9d] Merge GitWork into main and resolve conflict

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ git status

On branch main

Untracked files:

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

hello\_BACKUP\_2072.xml

hello\_BASE\_2072.xml

hello\_LOCAL\_2072.xml

hello\_REMOTE\_2072.xml

nothing added to commit but untracked files present (use "git add" to track)

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

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

echo "\*~" >> .gitignore

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ git add .gitignore

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

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ git commit -m "Ignore merge/editor backup"

[main 13bac89] Ignore merge/editor backup

1 file changed, 2 insertions(+), 1 deletion(-)

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

$ git branch -a

GitWork

\* main

Alice Franny@DESKTOP-D0OM5S0 MINGW64 ~/GitDemo (main)

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

\* 13bac89 (HEAD -> main) Ignore merge/editor backup

\* c238b9d Merge GitWork into main and resolve conflict

|\

| \* cfc9795 (GitWork) Add hello.xml in GitWork

\* | 8f8dded Add hello.xml in master

|/

\* 4694d2e Add branchfile.txt in GitNewBranch

\* 4ac0a01 Add .gitignore to ignore .log files and log folder

\* 8b4d33d Initial commit

Also use .gitignore to ignore backups.