

4. Git-HOL

Steps to be followed:

1. Verify if master is in clean state.
2. Create a branch "GitWork". Add a file "hello.xml".
3. Update the content of "hello.xml" and observe the status
4. Commit the changes to reflect in the branch
5. Switch to master.
6. Add a file "hello.xml" to the master and add some different content than previous.
7. Commit the changes to the master
8. Observe the log by executing "git log --oneline --graph --decorate --all"
9. Check the differences with Git diff tool
10. For better visualization, use P4Merge tool to list out all the differences between master and branch
11. Merge the bran to the master
12. Observe the git mark up.
13. Use 3-way merge tool to resolve the conflict
14. Commit the changes to the master, once done with conflict
15. Observe the git status and add backup file to the .gitignore file.
16. Commit the changes to the .gitignore
17. List out all the available branches
18. Delete the branch, which merge to master.
19. Observe the log by executing "git log --oneline --graph --decorate"

```

ADMIN@DESKTOP-0UB4RGS MINGW64 ~/GitDemo (main)
$ git checkout main
Already on 'main'
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)

ADMIN@DESKTOP-0UB4RGS MINGW64 ~/GitDemo (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)

nothing to commit, working tree clean

ADMIN@DESKTOP-0UB4RGS MINGW64 ~/GitDemo (main)
$ git switch -c GitWork
Switched to a new branch 'GitWork'

ADMIN@DESKTOP-0UB4RGS MINGW64 ~/GitDemo (GitWork)
$ printf "<?xml version='1.0'?'>\n<message>Hello from GitWork branch</message>\n" > hello.xml

ADMIN@DESKTOP-0UB4RGS MINGW64 ~/GitDemo (GitWork)
$ git status
On branch GitWork
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        hello.xml

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

ADMIN@DESKTOP-0UB4RGS 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

ADMIN@DESKTOP-0UB4RGS MINGW64 ~/GitDemo (GitWork)
$ git commit -m "Add hello.xml on GitWork (branch version)"
[GitWork 21be39d] Add hello.xml on GitWork (branch version)
 1 file changed, 2 insertions(+)
 create mode 100644 hello.xml

ADMIN@DESKTOP-0UB4RGS MINGW64 ~/GitDemo (GitWork)
$ git status
On branch GitWork
nothing to commit, working tree clean

ADMIN@DESKTOP-0UB4RGS MINGW64 ~/GitDemo (GitWork)
$ git log --oneline --graph --decorate -n 5
* 21be39d (HEAD -> GitWork) Add hello.xml on GitWork (branch version)
* 66b5f5c (main) Add branchfile.txt in GitNewBranch
* bc79df2 (origin/main) Add .gitignore to ignore *.log files and log folder

```

```

ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (GitWork)
$ git switch main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (main)
$ printf "<?xml version=\"1.0\"?>\n<message>Hello from main branch</message>\n"
> hello.xml

ADMIN@DESKTOP-OUB4RGS 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

ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (main)
$ git commit -m "Add/modify hello.xml on main (master version) to cause conflict"
[main f29db13] Add/modify hello.xml on main (master version) to cause conflict
1 file changed, 2 insertions(+)
create mode 100644 hello.xml

ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (main)
$ git log --oneline --graph --decorate --all
* f29db13 (HEAD -> main) Add/modify hello.xml on main (master version) to cause conflict
| * 21be39d (GitWork) Add hello.xml on GitWork (branch version)
|/
* 66b5f5c Add branchfile.txt in GitNewBranch
* bc79df2 (origin/main) Add .gitignore to ignore *.log files and log folder

```

```

ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (main)
$ git diff --name-only main..GitWork
hello.xml

ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (main)
$ git diff main..GitWork
diff --git a/hello.xml b/hello.xml
index 26637e0..78a8252 100644
--- a/hello.xml
+++ b/hello.xml
@@ -1,2 +1,2 @@
<?xml version="1.0"?>
-<message>Hello from main branch</message>
+<message>Hello from GitWork branch</message>

ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (main)
$ git difftool main..GitWork

Viewing (1/1): 'hello.xml'
Launch 'p4merge' [Y/n]? Y

```

MINGW64:/c/Users/ADMIN/GitDemo

ADMIN@DESKTOP-OUB4RG5 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.

ADMIN@DESKTOP-OUB4RG5 MINGW64 ~/GitDemo (main|MERGING)

\$ git status

On branch main

Your branch is ahead of 'origin/main' by 2 commits.

(use "git push" to publish your local commits)

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")

ADMIN@DESKTOP-OUB4RG5 MINGW64 ~/GitDemo (main|MERGING)

\$ cat hello.xml

<?xml version="1.0"?>

<message>< HEAD

<message>Hello from main branch</message>

=====

<message>Hello from GitWork branch</message>

>>>>>>> GitWork

ADMIN@DESKTOP-OUB4RG5 MINGW64 ~/GitDemo (main|MERGING)

\$ git config --global merge.tool p4merge

ADMIN@DESKTOP-OUB4RG5 MINGW64 ~/GitDemo (main|MERGING)

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

ADMIN@DESKTOP-OUB4RG5 MINGW64 ~/GitDemo (main|MERGING)

\$ git config --global mergetool.p4merge.trustExitCode true

ADMIN@DESKTOP-OUB4RG5 MINGW64 ~/GitDemo (main|MERGING)

\$ git mergetool

Merging:

hello.xml

Normal merge conflict for 'hello.xml':

{local}: created file

{remote}: created file

ADMIN@DESKTOP-OUB4RG5 MINGW64 ~/GitDemo (main|MERGING)

\$ git add hello.xml

ADMIN@DESKTOP-OUB4RG5 MINGW64 ~/GitDemo (main|MERGING)

\$ git commit -m "Resolve merge conflict for hello.xml (merged GitWork into main)"

"

[main 06f0058] Resolve merge conflict for hello.xml (merged GitWork into main)

MINGW64:/c/Users/ADMIN/GitDemo

```
ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 4 commits.
  (use "git push" to publish your local commits)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        hello.xml.orig

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

ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (main)
$ git log --oneline --graph --decorate -n 10
* 06f0058 (HEAD -> main) Resolve merge conflict for hello.xml (merged Gitwork
into main)
|
| * 21be39d (Gitwork) Add hello.xml on Gitwork (branch version)
| | f29db13 Add/modify hello.xml on main (master version) to cause conflict
| |
|/
* 66b5f5c Add branchfile.txt in GitNewBranch
* bc79df2 (origin/main) Add .gitignore to ignore *.log files and log folder
```

```
ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (main)
$ printf "%.orig\n%.bak\n" >> .gitignore

ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (main)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the ne
xt time Git touches it

ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (main)
$ git commit -m "Ignore merge backup files (*.orig, *.bak)"
[main 258d086] Ignore merge backup files (*.orig, *.bak)
 1 file changed, 2 insertions(+)

ADMIN@DESKTOP-OUB4RGS MINGW64 ~/GitDemo (main)
$ git branch -a
  Gitwork
* main
remotes/origin/main
```



```
ADMIN@DESKTOP-OUB4RG5 MINGW64 ~/GitDemo (main)
$ git branch -d GitWork
Deleted branch GitWork (was 21be39d).

ADMIN@DESKTOP-OUB4RG5 MINGW64 ~/GitDemo (main)
$ git log --oneline --graph --decorate -n 20
* 258d086 (HEAD -> main) Ignore merge backup files (*.orig, *.bak)
* 06f0058 Resolve merge conflict for hello.xml (merged GitWork into main)
| \
| * 21be39d Add hello.xml on GitWork (branch version)
* | f29db13 Add/modify hello.xml on main (master version) to cause conflict
|/
* 66b5f5c Add branchfile.txt in GitNewBranch
* bc79df2 (origin/main) Add .gitignore to ignore *.log files and log folder
| \
| * 4907001 Initial commit
* 7b4f7fe Initial commit: add welcome.txt

ADMIN@DESKTOP-OUB4RG5 MINGW64 ~/GitDemo (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 5 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```