

WEEK 8

GIT

EXERCISE 1:

Setup your machine with Git Configuration and create the project in gitlab and push the changes to the repository.

```
MINGW64~/c:/Users/Lenovo/GitDemo
- f, --[no-]fetch          fetch the remote branches
- --[no-]tags              import all tags and associated objects when fetching
                           or do not fetch any tag at all (--no-tags)
- t, --[no-]track <branch> branch(es) to track
- m, --[no-]master <branch> master branch
- --[no-]mirror[=(push|fetch)] set up remote as a mirror to push to or fetch from

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git remote -v

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git remote add origin https://gitlab.com/dhaarini2104-group/GitDemo.git

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git remote -v
origin  https://gitlab.com/dhaarini2104-group/GitDemo.git (fetch)
origin  https://gitlab.com/dhaarini2104-group/GitDemo.git (push)

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git push -u origin master
info: please complete authentication in your browser...
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 290 bytes | 290.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: To create a merge request for master, visit:
remote:   https://gitlab.com/dhaarini2104-group/GitDemo/-/merge_requests/new?merge_request%5Bsource_branch%5D=master
remote:
To https://gitlab.com/dhaarini2104-group/GitDemo.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ |
```

```
MINGW64~/c:/Users/Lenovo/GitDemo
Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ echo "welcome to the version control"> welcome.txt

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ ls
welcome.txt

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ cat welcome.txt
welcome to the version control

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git status
On branch master

No commits yet

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

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

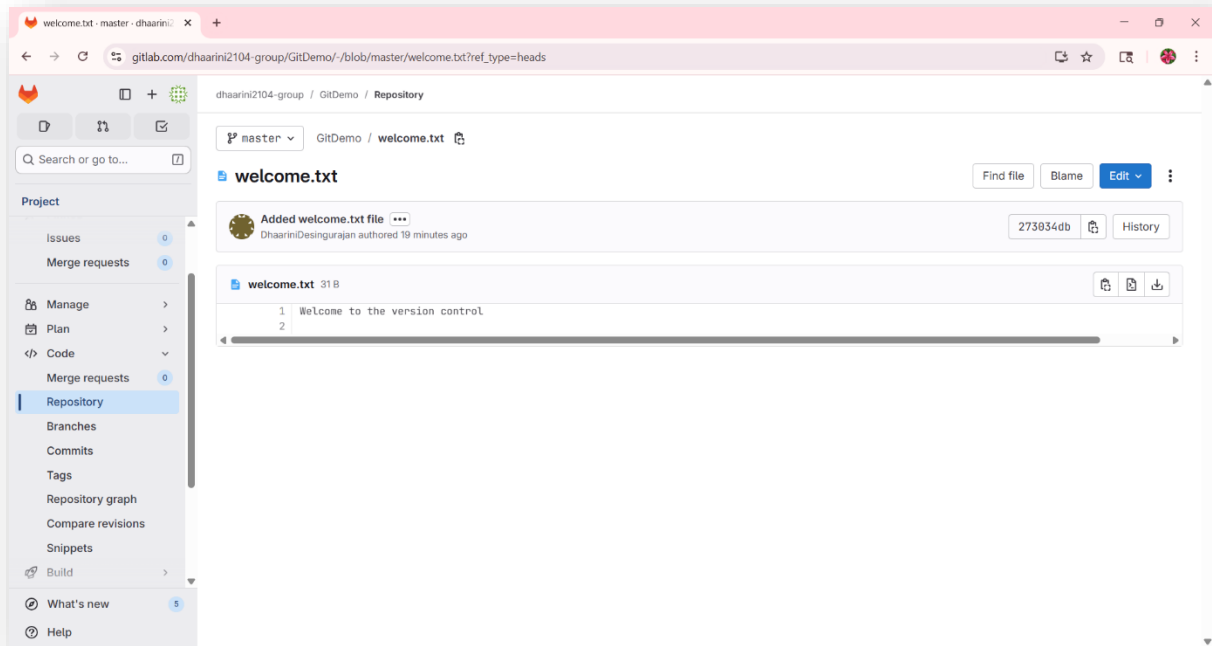
Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git add welcome.txt
warning: in the working copy of 'welcome.txt', LF will be replaced by CRLF the next time Git touches it

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git commit
[master (root-commit) 273034d] Added welcome.txt file This file is used for Git hands-on lab
1 file changed, 1 insertion(+)
create mode 100644 welcome.txt

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git status
On branch master
nothing to commit, working tree clean

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git remote add https://gitlab.com/dhaarini2104-group/GitDemo
usage: git remote add [<options>] <name> <url>

- f, --[no-]fetch          fetch the remote branches
- --[no-]tags              import all tags and associated objects when fetching
                           or do not fetch any tag at all (--no-tags)
- t, --[no-]track <branch> branch(es) to track
- m, --[no-]master <branch>
```



Exercise 2:

Create a **“.log”** file and a **log folder** in the working directory of Git. Update the **.gitignore** file in such a way that on committing, these files (.log extensions and log folders) are ignored.

```
MINGW64/c/Users/Lenovo/GitDemo
Lenovo@LAPTOP-APTNITHN MINGW64 ~/GitDemo (master)
$ touch error.log
Lenovo@LAPTOP-APTNITHN MINGW64 ~/GitDemo (master)
$ mkdir log
Lenovo@LAPTOP-APTNITHN MINGW64 ~/GitDemo (master)
$ echo "dummy values for the file">log/sample.txt
Lenovo@LAPTOP-APTNITHN MINGW64 ~/GitDemo (master)
$ touch .gitignore
Lenovo@LAPTOP-APTNITHN MINGW64 ~/GitDemo (master)
$ notepad++ .gitignore
Lenovo@LAPTOP-APTNITHN MINGW64 ~/GitDemo (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

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

nothing added to commit but untracked files present (use "git add" to track)
Lenovo@LAPTOP-APTNITHN MINGW64 ~/GitDemo (master)
$ git add .gitignore
Lenovo@LAPTOP-APTNITHN MINGW64 ~/GitDemo (master)
$ git commit -m"Added .gitignore .log files and log folder"
git: 'commit' is not a git command. See 'git --help'.

The most similar command is
    commit
Lenovo@LAPTOP-APTNITHN MINGW64 ~/GitDemo (master)
$ git commit -m"Added .gitignore .log files and log folder"
git: 'commit' is not a git command. See 'git --help'.

The most similar command is
    commit
Lenovo@LAPTOP-APTNITHN MINGW64 ~/GitDemo (master)
$ git commit -m"Added .gitignore .log file and log folder"
[master 9c45123] Added .gitignore .log file and log folder
1 file changed, 2 insertions(+)
create mode 100644 .gitignore
```

Added .gitignore .log file and log folder

Commit 9c451232 authored 7 minutes ago by DhaariniDesingurajan

parent 273834db

Branches master

No related tags found

No related merge requests found

Changes 1

Showing 1 changed file with 2 additions and 0 deletions

Hide whitespace changes Inline Side-by-side

View file @ 9c451232

```
1 + *.log
2 + log/
\ No newline at end of file
```

Preview Write a comment or drag your files here..

dhaarini2104-group / GitDemo / Repository

You pushed to master 9 minutes ago

Create merge request

master GitDemo

GitDemo

Added .gitignore .log file and log folder

DhaariniDesingurajan authored 11 minutes ago

Name	Last commit	Last update
.gitignore	Added .gitignore .log file and log folder	11 minutes ago
welcome.txt	Added welcome.txt file	55 minutes ago

Exercise 3:

Branching and Merging

```
MINGW64~/Users/Lenovo/GitDemo
Lenovo@LAPTOP-AFTNITHN MINGW64 ~/GitDemo (master)
$ git checkout GitNewBranch
Switched to branch 'GitNewBranch'

Lenovo@LAPTOP-AFTNITHN MINGW64 ~/GitDemo (GitNewBranch)
$ git branch
* GitNewBranch
  master

Lenovo@LAPTOP-AFTNITHN MINGW64 ~/GitDemo (GitNewBranch)
$ echo "This is a new file in GitNewBranch">newfile.txt

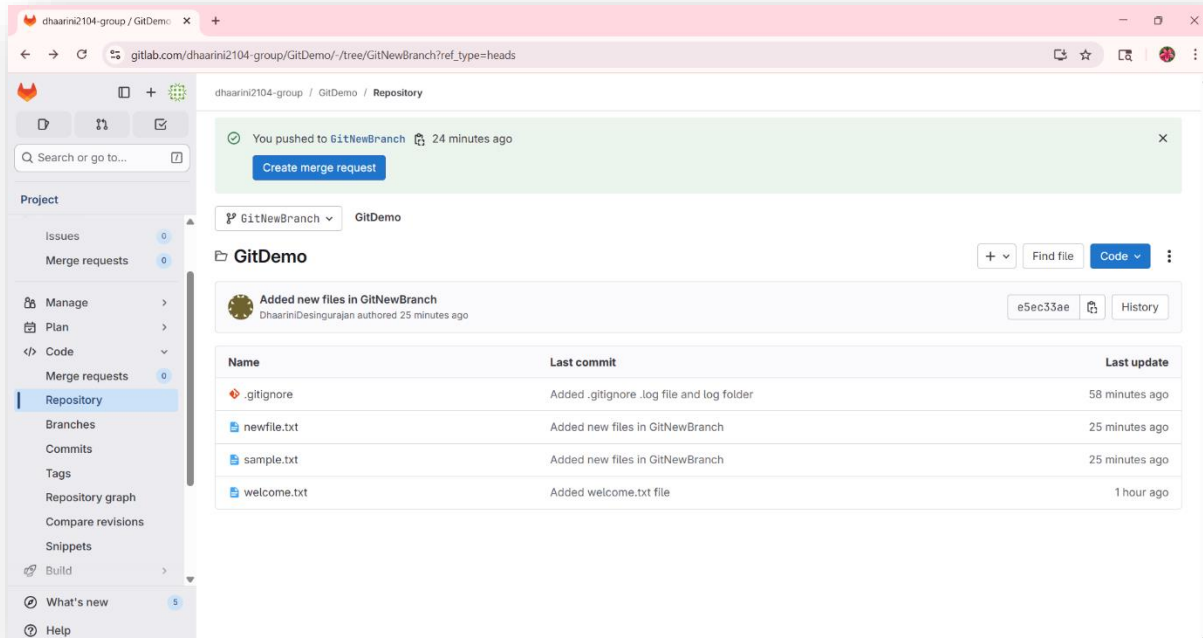
Lenovo@LAPTOP-AFTNITHN MINGW64 ~/GitDemo (GitNewBranch)
$ touch sample.txt

Lenovo@LAPTOP-AFTNITHN MINGW64 ~/GitDemo (GitNewBranch)
$ echo "Another file in branch">sample.txt

Lenovo@LAPTOP-AFTNITHN MINGW64 ~/GitDemo (GitNewBranch)
$ git add .
warning: in the working copy of 'newfile.txt', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'sample.txt', LF will be replaced by CRLF the next time Git touches it

Lenovo@LAPTOP-AFTNITHN MINGW64 ~/GitDemo (GitNewBranch)
$ git commit -m"Added new files in GitNewBranch"
[GitNewBranch e5ec33a] Added new files in GitNewBranch
 2 files changed, 2 insertions(+)
 create mode 100644 newfile.txt
 create mode 100644 sample.txt

Lenovo@LAPTOP-AFTNITHN MINGW64 ~/GitDemo (GitNewBranch)
$ git push origin GitNewBranch
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 432 bytes | 432.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: To create a merge request for GitNewBranch, visit:
remote:   https://gitlab.com/dhaarin2104-group/GitDemo/-/merge_requests/new?merge_request%5Bsource_branch%5D=GitNewBranch
remote:
To https://gitlab.com/dhaarin2104-group/GitDemo.git
 * [new branch]   GitNewBranch -> GitNewBranch
Lenovo@LAPTOP-AFTNITHN MINGW64 ~/GitDemo (GitNewBranch)
```



```

MINGW64/c/Users/Lenovo/GitDemo
$ cd GitDemo

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git diff GitNewBranch
diff --git a/newfile.txt b/newfile.txt
deleted file mode 100644
index 2129c56..0000000
--- a/newfile.txt
+++ /dev/null
@@ -1,0,0 @@
-This is a new file in GitNewBranch
diff --git a/sample.txt b/sample.txt
deleted file mode 100644
index 6d684d6..0000000
--- a/sample.txt
+++ /dev/null
@@ -1,0,0 @@
-Another file in branch

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git config --global merge.tool p4merge
git config --global mergetool.p4merge.path "C:/Program Files/Perforce/p4merge.exe"

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git difftool master GitNewBranch

Viewing (1/2): 'newfile.txt'
Launch 'p4merge' [Y/n]? y
The diff tool p4merge is not available as 'C:/Program Files/Perforce/p4merge.exe'
fatal: external diff died, stopping at newfile.txt

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ ^C

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git diff master GitNewBranch
diff --git a/newfile.txt b/newfile.txt
new file mode 100644
index 0000000..2129c56
--- /dev/null
+++ b/newfile.txt
@@ -0,0 +1 @@

```

```

MINGW64/c/Users/Lenovo/GitDemo
Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git diff master GitNewBranch
diff --git a/newfile.txt b/newfile.txt
new file mode 100644
index 0000000..2129c56
--- /dev/null
+++ b/newfile.txt
@@ -0,0 +1 @@
-This is a new file in GitNewBranch
diff --git a/sample.txt b/sample.txt
new file mode 100644
index 0000000..6d684d6
--- /dev/null
+++ b/sample.txt
@@ -0,0 +1 @@
-Another file in branch

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git merge GitNewBranch
Updating 9c45123..e5ec33a
Fast-forward
 newfile.txt | 1 +
 sample.txt | 1 +
 2 files changed, 2 insertions(+)
 create mode 100644 newfile.txt
 create mode 100644 sample.txt

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git log --oneline --graph --decorate
* e5ec33a (HEAD -> master, origin/GitNewBranch, GitNewBranch) Added new files in GitNewBranch
* 9c45123 (origin/master) Added .gitignore .log file and log folder
* 273034d Added welcome.txt file This file is used for Git hands-on lab

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git branch -d GitNewBranch
Deleted branch GitNewBranch (was e5ec33a).

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

```

Exercise 4:

- 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.

```
MINGW64/c/Users/Lenovo/GitDemo
Lenovo@LAPTOP-APTNIITHN MINGW64 ~
$ cd GitDemo

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (GitNewBranch)
$ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

nothing to commit, working tree clean

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git branch Gitwork

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

nothing to commit, working tree clean

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git branch
  GitNewBranch
* master

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git checkout Gitwork
Switched to branch 'Gitwork'
```

```
MINGW64/c/Users/Lenovo/GitDemo
Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (Gitwork)
$ echo "<message>Hello from Gitwork</message>">hello.xml

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (Gitwork)
$ echo "<message>Updated in Gitwork branch</message>">hello.xml

Lenovo@LAPTOP-APTNIITHN 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)

Lenovo@LAPTOP-APTNIITHN 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

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (Gitwork)
$ git commit -m"Added and updated hello.xml in Gitwork branch"
[Gitwork 4cf2150] Added and updated hello.xml in Gitwork branch
 1 file changed, 1 insertion(+)
 create mode 100644 hello.xml

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (Gitwork)
$ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ echo "<message>Different content from master</message>">hello.xml

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (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

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
```

```

MINGW64~/Users/Lenovo/GitDemo
Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git commit -m"Added and updated hello.xml in master branch"
[master a9a195e] Added and updated hello.xml in master branch
1 file changed, 1 insertion(+)
create mode 100644 hello.xml

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git log --oneline --graph --decorate --all
* a9a195e (HEAD -> master) Added and updated hello.xml in master branch
| * 4cf2150 (Gitwork) Added and updated hello.xml in Gitwork branch
|/
* e5ec33a (origin/GitNewBranch, GitNewBranch) Added new files in GitNewBranch
* 9c45123 (origin/master) Added .gitignore .log file and log folder
* 273034d Added welcome.txt file This file is used for Git hands-on lab

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git diff Gitwork
diff --git a/hello.xml b/hello.xml
index 3d4e617..d1d7e82 100644
--- a/hello.xml
+++ b/hello.xml
@@ -1,1 @@
-~<message>Updated in Gitwork branch</message>
+~<message>Different content from master</message>

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (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.

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master|MERGING)
$ cat hello.xml
<<<<<<< HEAD
<message>Different content from master</message>
=====
<message>Updated in Gitwork branch</message>
>>>>>>> Gitwork

```

```

MINGW64~/Users/Lenovo/GitDemo
Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master|MERGING)
$ git mergetool
Merging:
hello.xml

Normal merge conflict for 'hello.xml':
{local}: created file
{remote}: created file
The merge tool p4merge is not available as 'C:/Program Files/Perforce/p4merge.exe'

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master|MERGING)
$ git add hello.xml
git commit -m "Resolved conflict and merged Gitwork into master"
[master cf471a3] Resolved conflict and merged Gitwork into master

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git status
echo "*.xml" >> .gitignore
On branch master
Your branch is ahead of 'origin/master' 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_BACKUP_1555.xml
        hello_BASE_1555.xml
        hello_LOCAL_1555.xml
        hello_REMOTE_1555.xml

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

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git add .gitignore
git commit -m "Ignored backup files in .gitignore"
warning: in the working copy of ".gitignore", LF will be replaced by CRLF the next time Git touches it
[master 9435c98] Ignored backup files in .gitignore
1 file changed, 1 insertion(+), 1 deletion(-)

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)

```

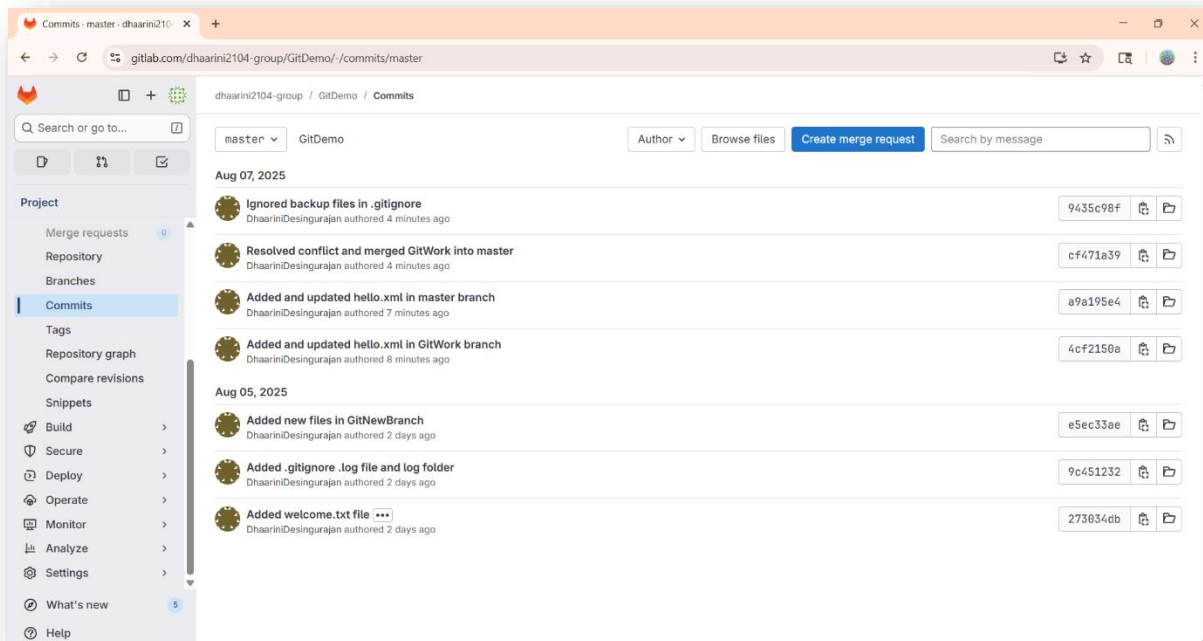
```
MINGW64~/Users/Lenovo/GitDemo
Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git branch
  GitNewBranch
  Gitwork
* master

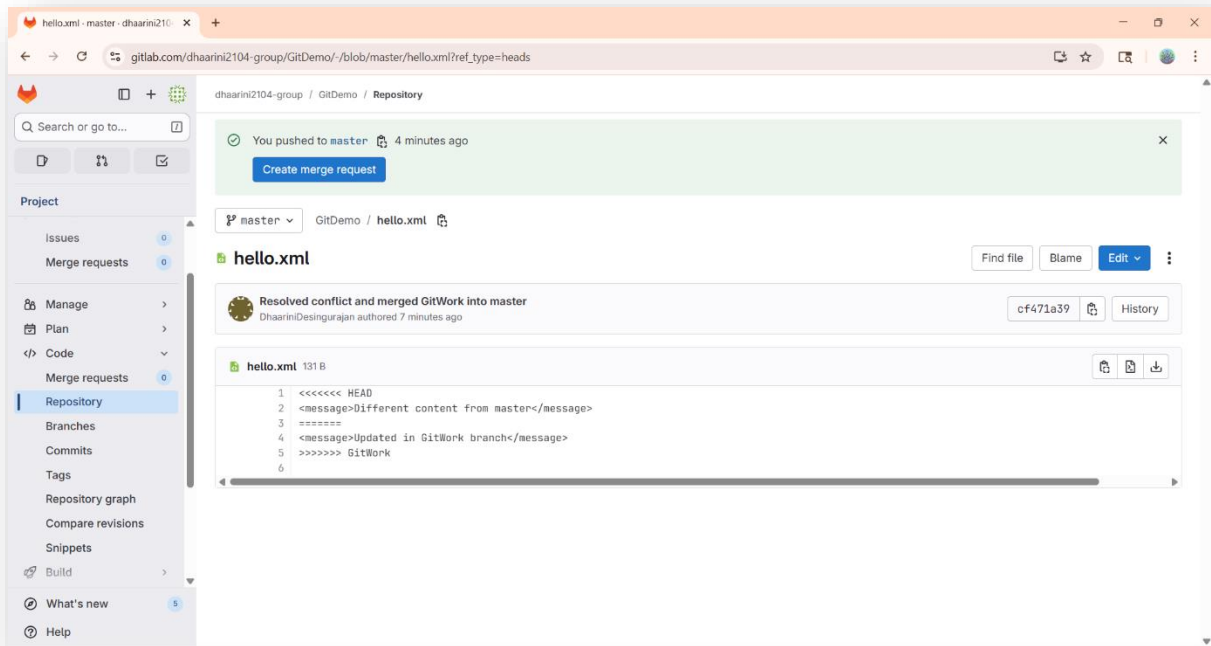
Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git remote -v
origin  https://gitlab.com/dhaarini2104-group/GitDemo.git (fetch)
origin  https://gitlab.com/dhaarini2104-group/GitDemo.git (push)

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git push origin master
Enumerating objects: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 8 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (12/12), 1.21 KiB | 621.00 KiB/s, done.
Total 12 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: To create a merge request for master, visit:
remote:   https://gitlab.com/dhaarini2104-group/GitDemo/-/merge_requests/new?merge_request%5Bsource_branch%5D=master
remote:
To https://gitlab.com/dhaarini2104-group/GitDemo.git
9c45123..9435c98  master -> master

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git branch -d Gitwork
Deleted branch Gitwork (was 4cf2150).

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/GitDemo (master)
$ git log --oneline --graph --decorate
* 9435c98 (HEAD -> master, origin/master) Ignored backup files in .gitignore
*   cf471a3 Resolved conflict and merged Gitwork into master
|
| * 4cf2150 Added and updated hello.xml in Gitwork branch
| | a9a195e Added and updated hello.xml in master branch
|_|
```





Exercise 5:

- Execute steps involving clean up and push back to remote Git.

```
MINGW64/c:/Users/Lenovo/Git-T03-HOL_002

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (main)
$ git checkout master
Switched to branch 'master'

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ git status
On branch master
nothing to commit, working tree clean

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ git branch -a
  main
* master
  remotes/origin/HEAD -> origin/main
  remotes/origin/main

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ git pull origin master
fatal: couldn't find remote ref master

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ git checkout master
git pull origin master
Already on 'master'
fatal: couldn't find remote ref master

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ ^C

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ git ls-remote --heads origin
98358014ac817662611b132d4d4a8a3ccd3f945c      refs/heads/main

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ git switch main
```

```

MINGW64/c/Users/Lenovo/Git-T03-HOL_002
Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (main)
$ git branch -m master main
fatal: a branch named 'main' already exists

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (main)
$ git branch
* main
  master

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (main)
$ git push origin Git-T03-HOL_002
error: src refspec Git-T03-HOL_002 does not match any
error: failed to push some refs to 'https://gitlab.com/dhaarini2104-group/git-t03-hol_002.git'

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (main)
$ ^C

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (main)
$ git push origin main:Git-T03-HOL_002
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: To create a merge request for Git-T03-HOL_002, visit:
remote:   https://gitlab.com/dhaarini2104-group/git-t03-hol_002/-/merge_requests/new?merge_request%5Bsource_branch%5D=Git-T03-HOL_002
remote:
To https://gitlab.com/dhaarini2104-group/git-t03-hol_002.git
 * [new branch]      main -> Git-T03-HOL_002

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (main)
$ git checkout master
Switched to branch 'master'

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ git merge Git-T03-HOL_002
merge: Git-T03-HOL_002 - not something we can merge

```

```

MINGW64/c/Users/Lenovo/Git-T03-HOL_002
urce_branch%5D=Git-T03-HOL_002
remote:
To https://gitlab.com/dhaarini2104-group/git-t03-hol_002.git
 * [new branch]      main -> Git-T03-HOL_002

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (main)
$ git checkout master
Switched to branch 'master'

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ git merge Git-T03-HOL_002
merge: Git-T03-HOL_002 - not something we can merge

Did you mean this?
    origin/Git-T03-HOL_002

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ ^C

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ git merge origin/Git-T03-HOL_002
Already up to date.

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ git push origin master
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: To create a merge request for master, visit:
remote:   https://gitlab.com/dhaarini2104-group/git-t03-hol_002/-/merge_requests/new?merge_request%5Bsource_branch%5D=master
remote:
To https://gitlab.com/dhaarini2104-group/git-t03-hol_002.git
 * [new branch]      master -> master

Lenovo@LAPTOP-APTNIITHN MINGW64 ~/Git-T03-HOL_002 (master)
$ |

```

