**GIT ASSIGNMENTS**

**=====================================**

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~

$ mkdir gitdemo

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~

$ cd gitdemo

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/gitdemo

$ cd gitdemo/

bash: cd: gitdemo/: No such file or directory

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/gitdemo

$ git init gitdemo

Initialized empty Git repository in C:/Users/SWEETY/gitdemo/gitdemo/.git/

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/gitdemo

$ ls -al

total 20

drwxr-xr-x 1 SWEETY 197121 0 Aug 18 19:59 ./

drwxr-xr-x 1 SWEETY 197121 0 Aug 18 19:59 ../

drwxr-xr-x 1 SWEETY 197121 0 Aug 18 19:59 gitdemo/

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/gitdemo

$ echo "welcome to the version control">>welcome.txt

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/gitdemo

$ ls -al

total 21

drwxr-xr-x 1 SWEETY 197121 0 Aug 18 20:01 ./

drwxr-xr-x 1 SWEETY 197121 0 Aug 18 19:59 ../

drwxr-xr-x 1 SWEETY 197121 0 Aug 18 19:59 gitdemo/

-rw-r--r-- 1 SWEETY 197121 31 Aug 18 20:01 welcome.txt

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/gitdemo

$ cat welcome.txt

welcome to the version control

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/projectdir/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)

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/projectdir/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

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/projectdir/gitdemo (master)

$ git commit

[master (root-commit) 06aea9c] first commit

1 file changed, 1 insertion(+)

create mode 100644 welcome.txt

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/projectdir/gitdemo (master)

$ git status

On branch master

nothing to commit, working tree clean

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/projectdir/gitdemo (master)

$ git remote add origin https://github.com/pravallikaadineni/gitdemo.git

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/projectdir/gitdemo (master)

$ git branch -M main

SWEETY@LAPTOP-SJD7JKLB MINGW64 ~/projectdir/gitdemo (main)

$ git push -u origin main

Enumerating objects: 3, done.

Counting objects: 100% (3/3), done.

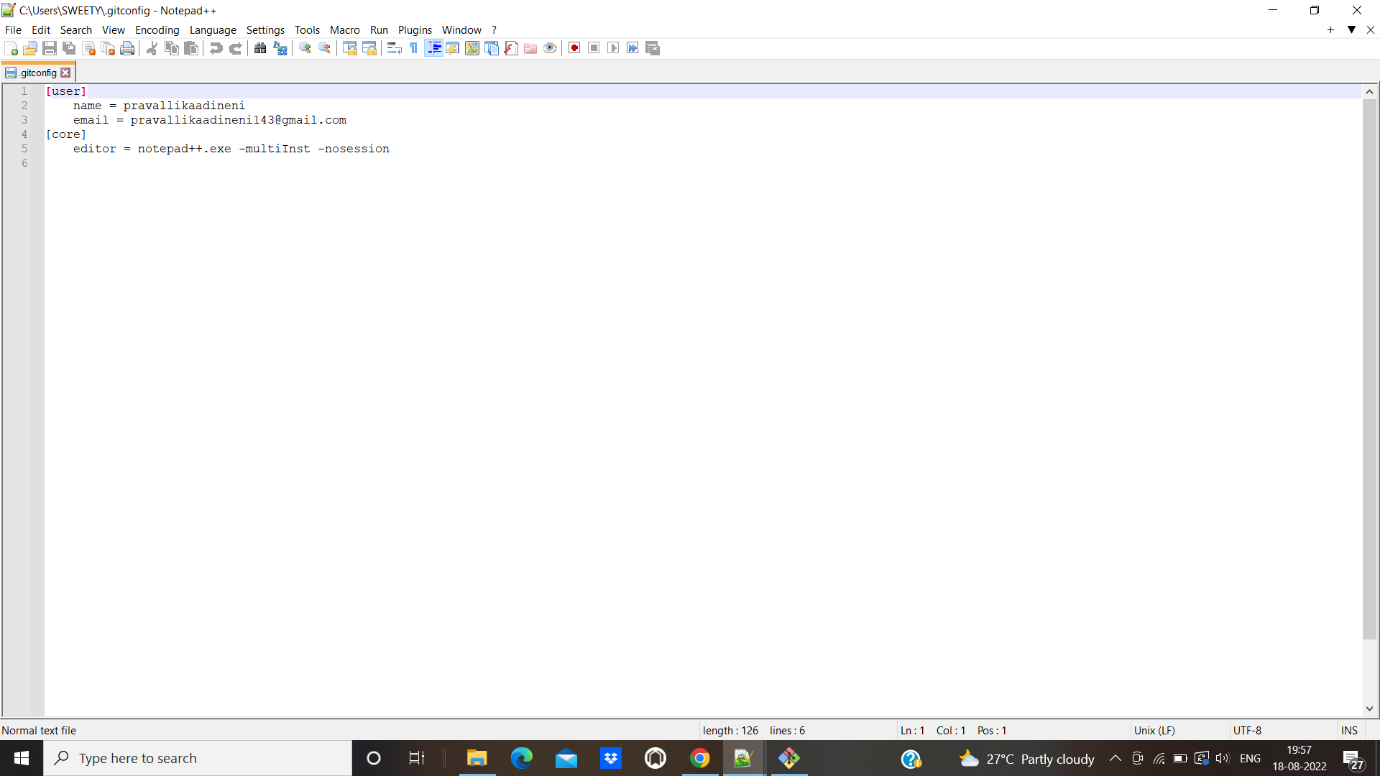
Writing objects: 100% (3/3), 248 bytes | 49.00 KiB/s, done.

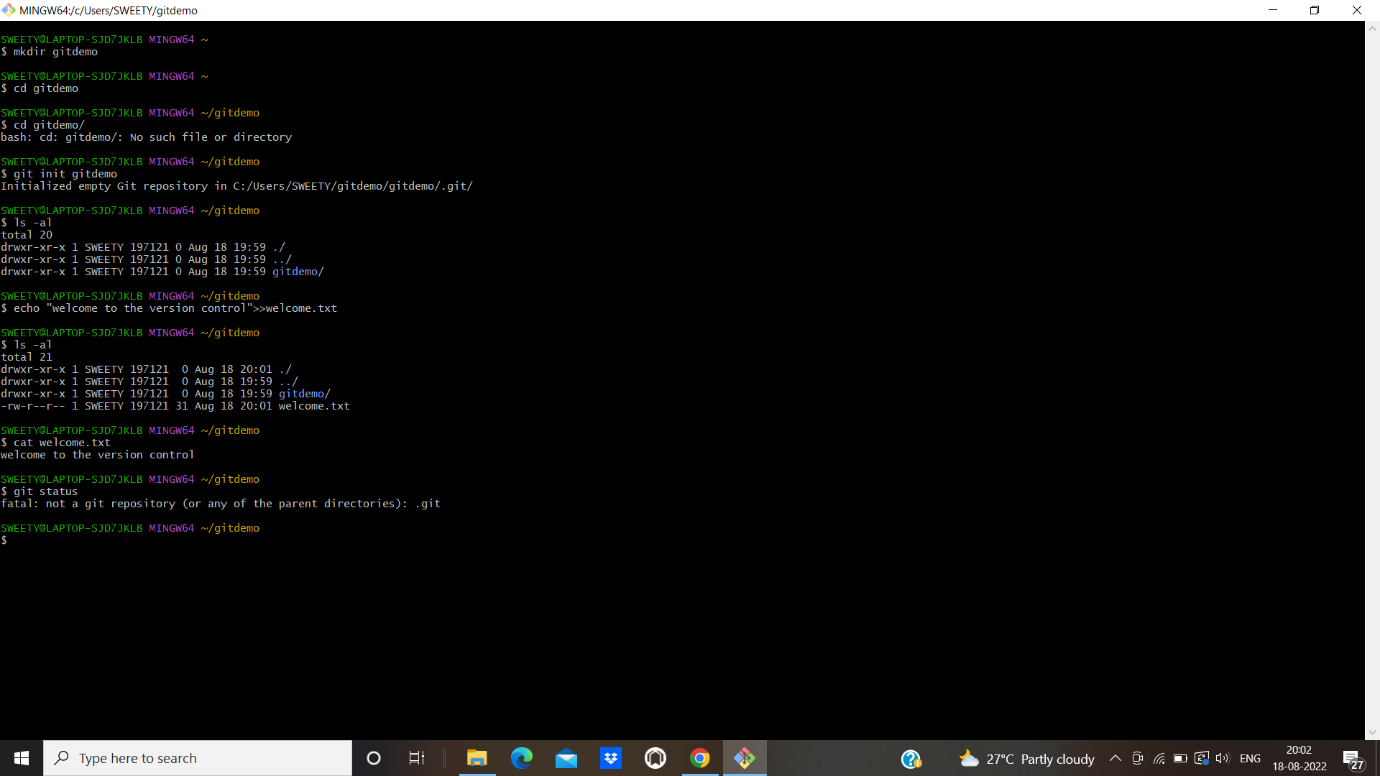
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

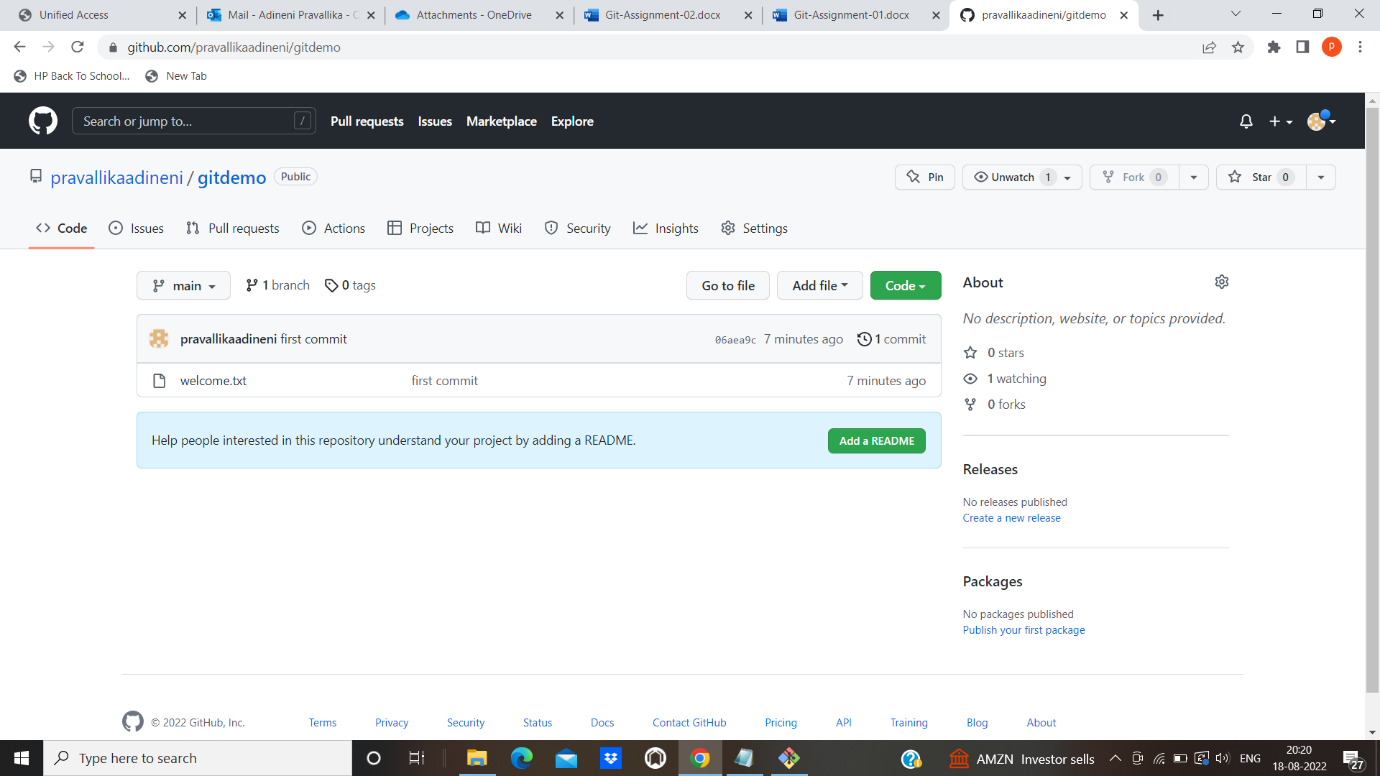
To https://github.com/pravallikaadineni/gitdemo.git

\* [new branch] main -> main

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







ASSIGNMENT 2

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git add .

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ cat ,gitignore

cat:, gitignore: No such file or directory

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ cat .gitignore

\*.log

/new folder/\*

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git add .

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git status

On branch main

Your branch is up to date with 'origin/main'.

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: .gitignore

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ touch hello.log

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ cat hello.log

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git add .

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ touch pravalli.txt

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git add .

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git commit -m "gitignore"

[main 79216e4] gitignore

2 files changed, 2 insertions(+)

create mode 100644 .gitignore

create mode 100644 pravalli.txt

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git push origin main

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), 340 bytes | 340.00 KiB/s, done.

Total 4 (delta 0), reused 0 (delta 0), pack-reused 0

To https://github.com/pravallikaadineni/gitdemo.git

06aea9c..79216e4 main -> main

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ echo "hello from log file">>hello.log

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ cat hello.log

hello from log file

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git branch

\* main

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git branch gitnewbranch

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git branch

gitnewbranch

\* main

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git checkout gitnewbranch

Switched to branch 'gitnewbranch'

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitnewbranch)

$ touch gitnew.txt

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitnewbranch)

$ echo "create new branch and add a file">>gitNew.txt

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitnewbranch)

$ git add .

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

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitnewbranch)

$ git commit -m "commit from new branch"

[gitnewbranch e4e7d5c] commit from new branch

1 file changed, 1 insertion(+)

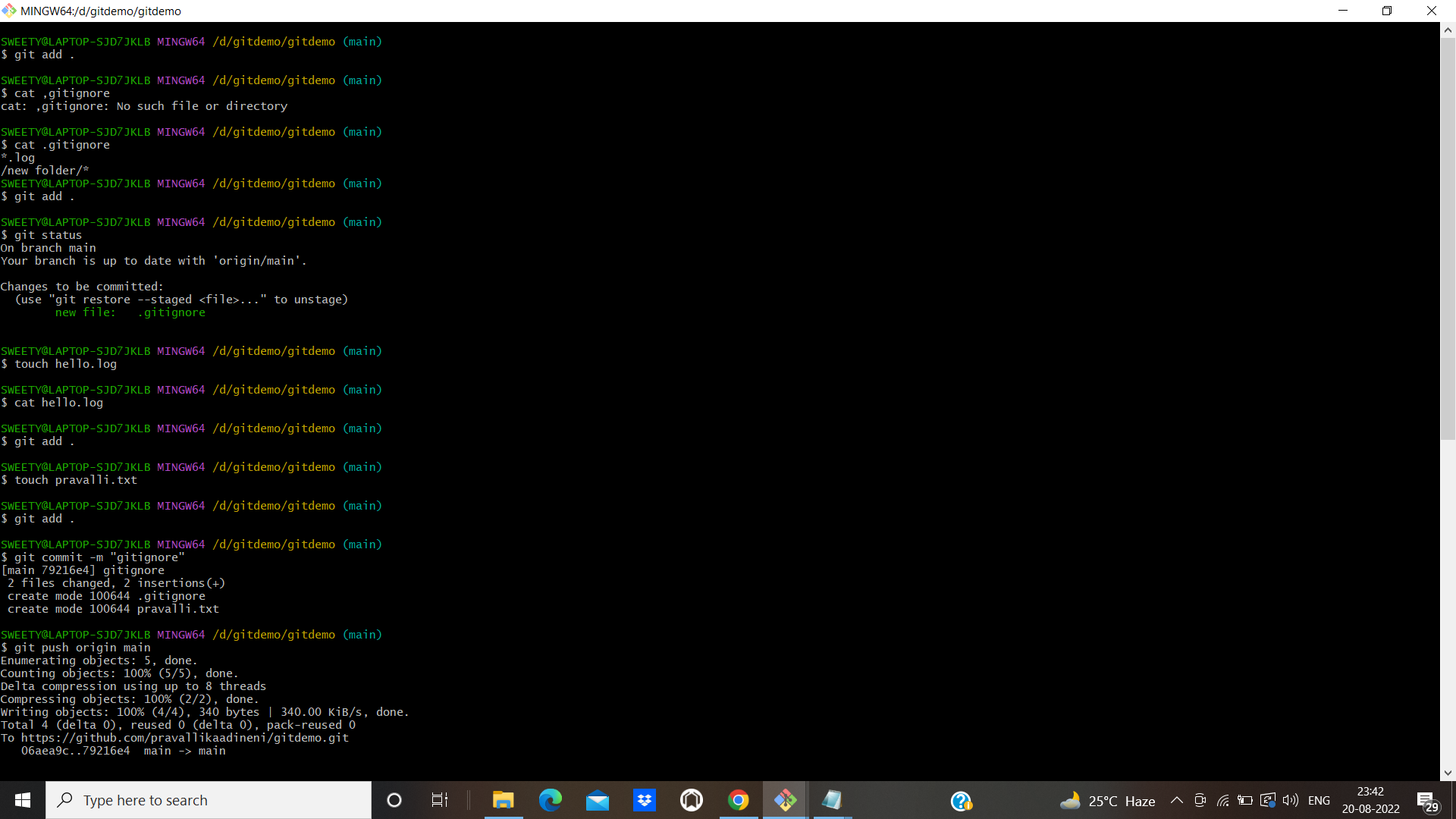
create mode 100644 gitnew.txt

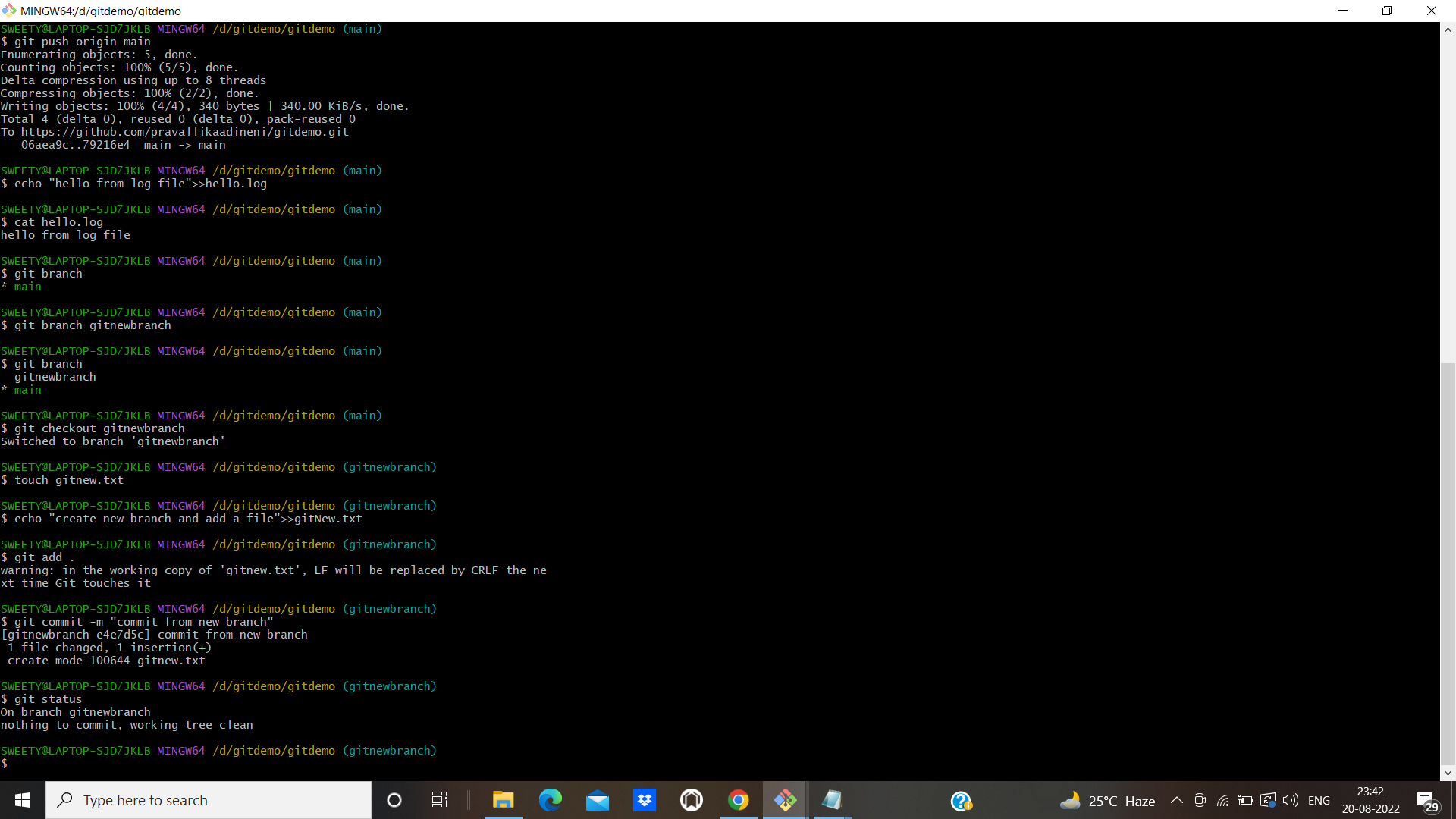
SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitnewbranch)

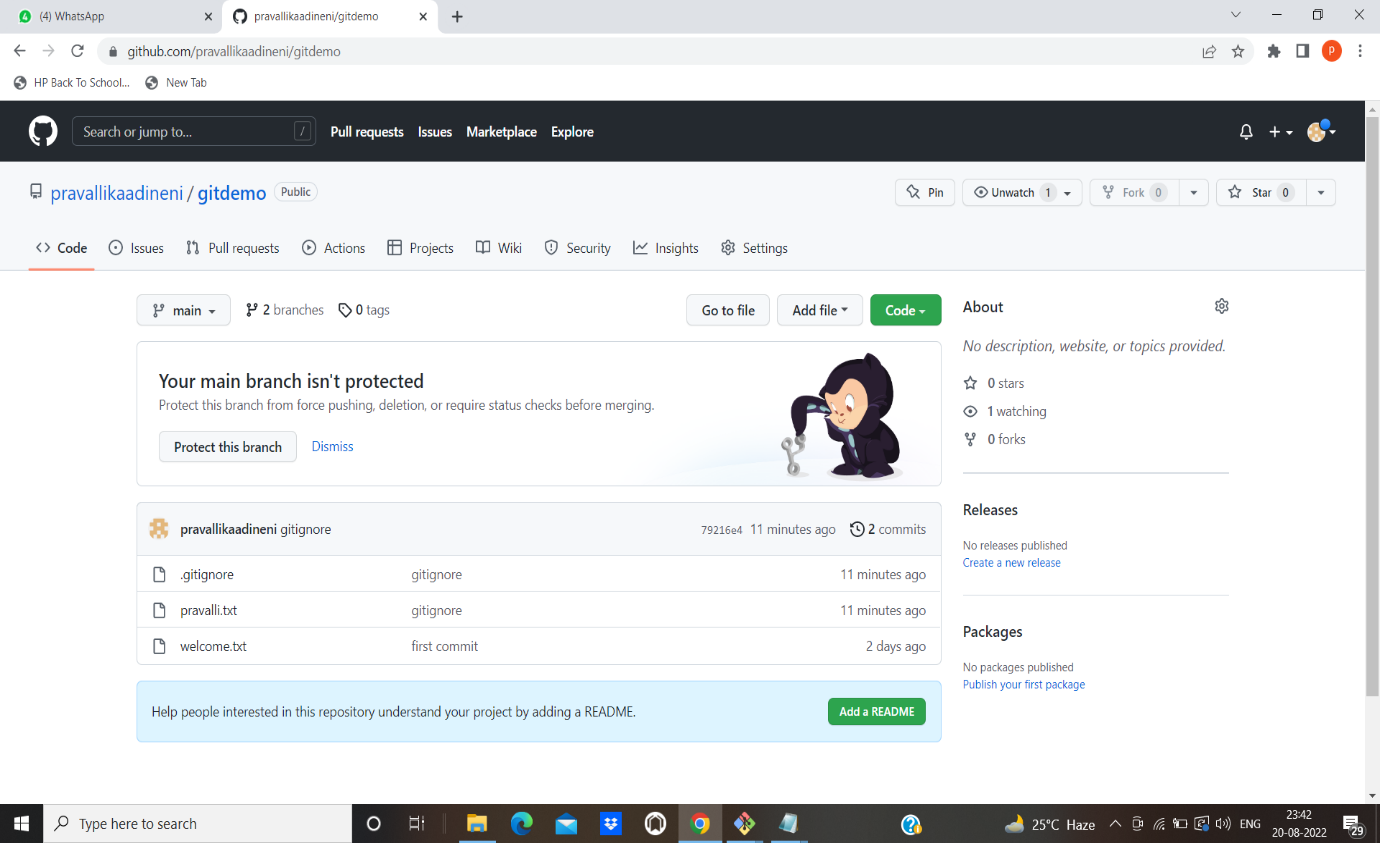
$ git status

On branch gitnewbranch

nothing to commit, working tree clean







Assignment 3&4

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

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

\* 79216e4 (HEAD -> main, origin/main, origin/HEAD) gitignore

\* 06aea9c (origin/neww) first commit

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git checkout gitnewbranch

Switched to branch 'gitnewbranch'

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitnewbranch)

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

\* e4e7d5c (HEAD -> gitnewbranch) commit from new branch

\* 79216e4 (origin/main, origin/HEAD, main) gitignore

\* 06aea9c (origin/neww) first commit

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitnewbranch)

$ git checkout main

Switched to branch 'main'

Your branch is up to date with 'origin/main'.

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git merge gitnewbranch

Updating 79216e4..e4e7d5c

Fast-forward

gitnew.txt | 1 +

1 file changed, 1 insertion(+)

create mode 100644 gitnew.txt

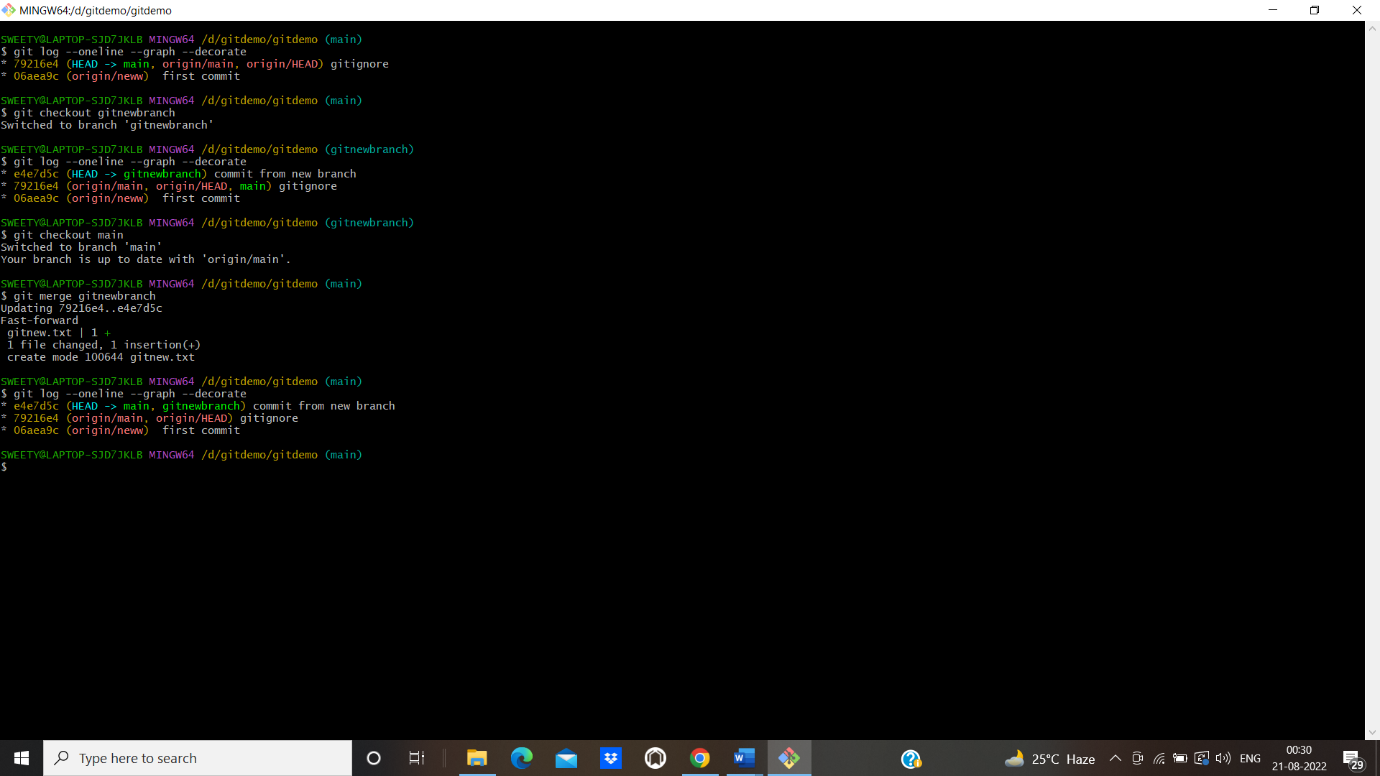
SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

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

\* e4e7d5c (HEAD -> main, gitnewbranch) commit from new branch

\* 79216e4 (origin/main, origin/HEAD) gitignore

\* 06aea9c (origin/neww) first commit



SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git branch -d gitnewbranch

Deleted branch gitnewbranch (was e4e7d5c).

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git push origin --delete gitnewbranch

error: unable to delete 'gitnewbranch': remote ref does not exist

error: failed to push some refs to 'https://github.com/pravallikaadineni/gitdemo.git'

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git add .

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git commit -m "upload"

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

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git push origin main

Enumerating objects: 4, done.

Counting objects: 100% (4/4), done.

Delta compression using up to 8 threads

Compressing objects: 100% (2/2), done.

Writing objects: 100% (3/3), 380 bytes | 380.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

To https://github.com/pravallikaadineni/gitdemo.git

79216e4..e4e7d5c main -> main

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git status

On branch main

Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git branch

\* main

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git status

On branch main

Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git branch gitwork

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git checkout gitwork

Switched to branch 'gitwork'

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitwork)

$ touch hello.xml

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitwork)

$ git add .

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitwork)

$ cat hello.xml

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitwork)

$ git status

On branch gitwork

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: hello.xml

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitwork)

$ git commit -m "commit from gitwork branch"

[gitwork 4d514d4] commit from gitwork branch

1 file changed, 0 insertions(+), 0 deletions(-)

create mode 100644 hello.xml

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitwork)

$ git checkout master

error: pathspec 'master' did not match any file(s) known to git

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitwork)

$ touch hello.xml

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitwork)

$ cat hello.xml

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitwork)

$ git commit -m "from main branch"

On branch gitwork

nothing to commit, working tree clean

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitwork)

$ git add .

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitwork)

$ git commit -m "from main branch"

On branch gitwork

nothing to commit, working tree clean

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (gitwork)

$ git checkout main

Switched to branch 'main'

Your branch is up to date with 'origin/main'.

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

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

\* e4e7d5c (HEAD -> main, origin/main, origin/HEAD) commit from new branch

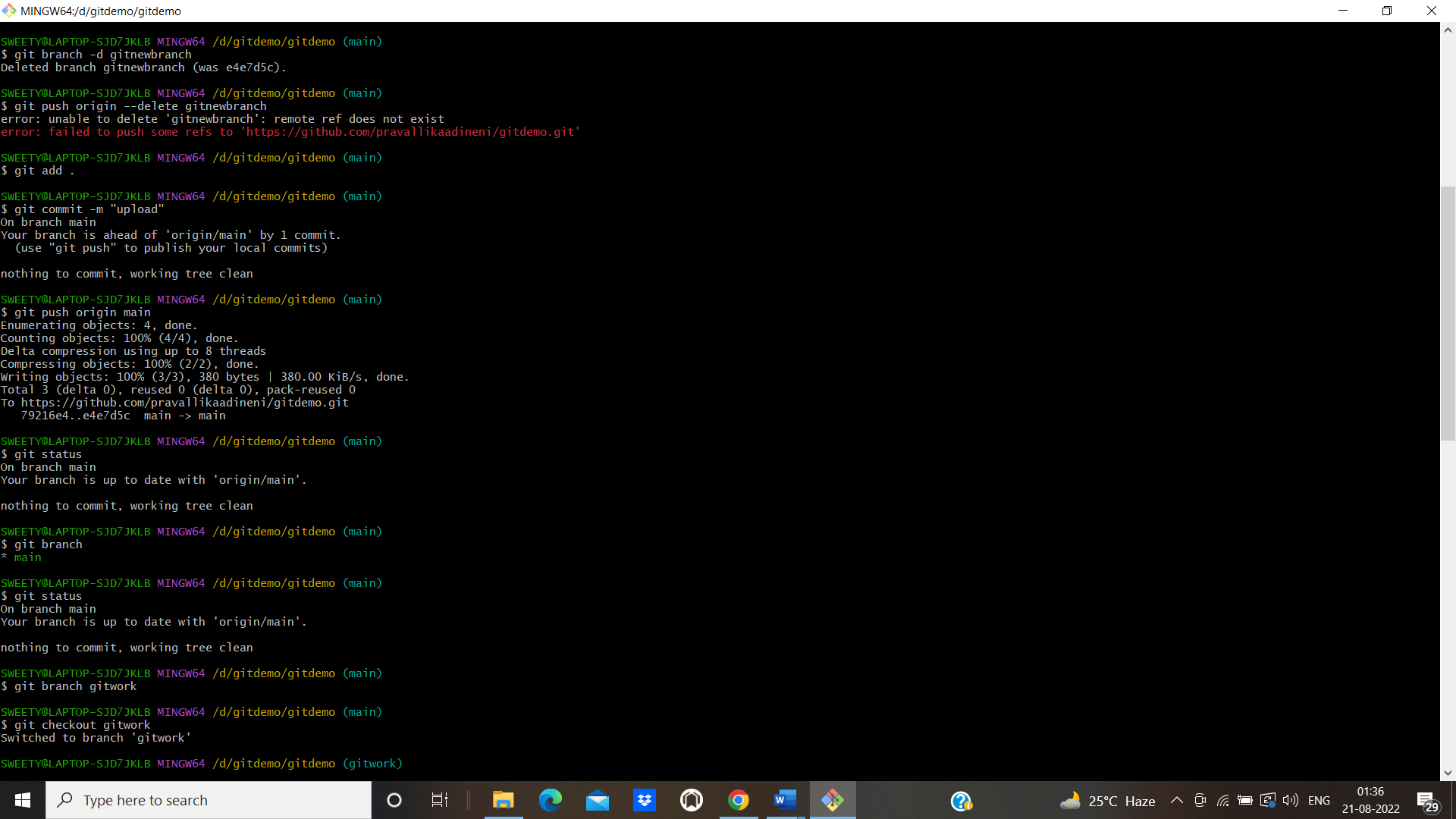
\* 79216e4 gitignore

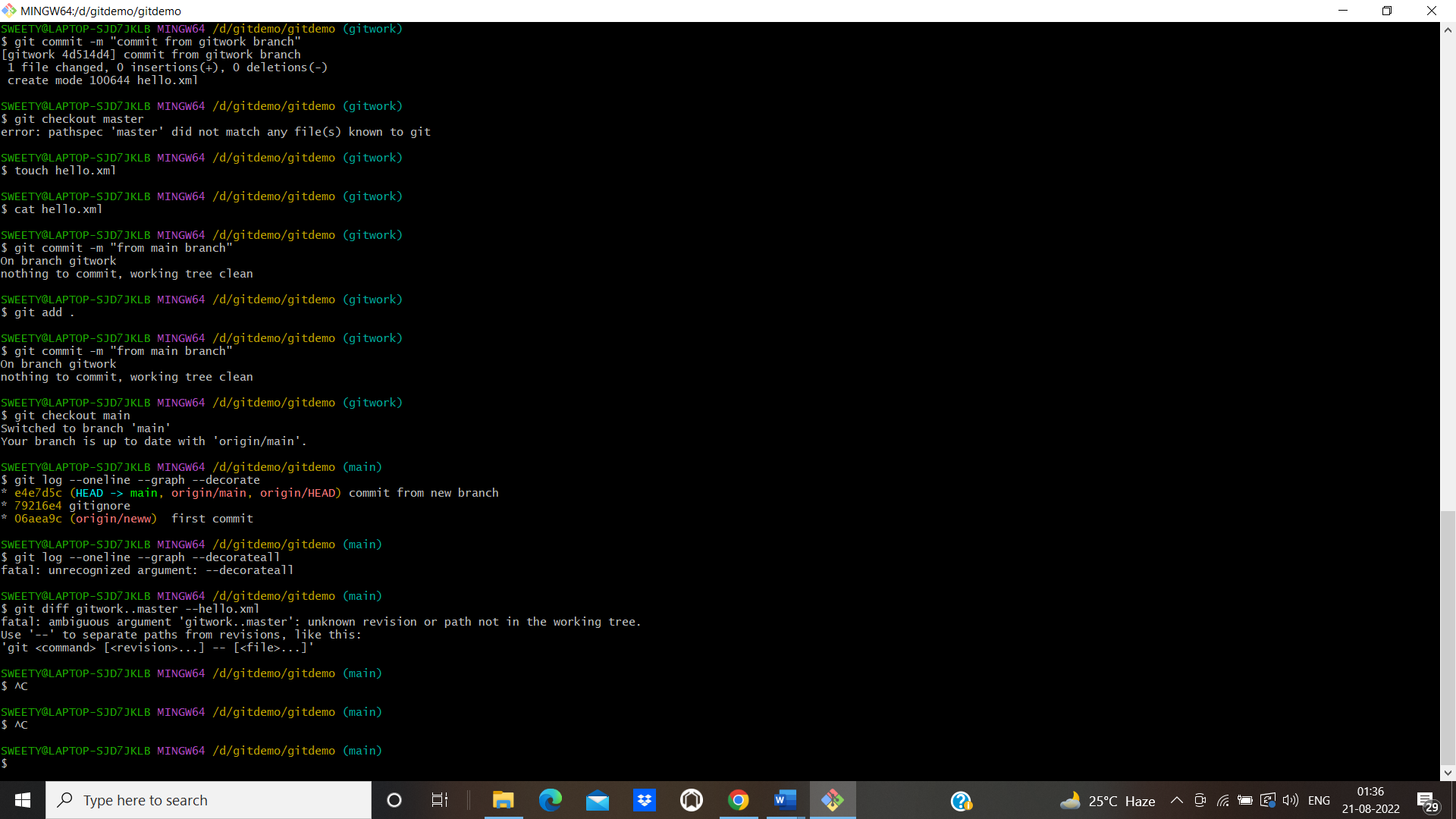
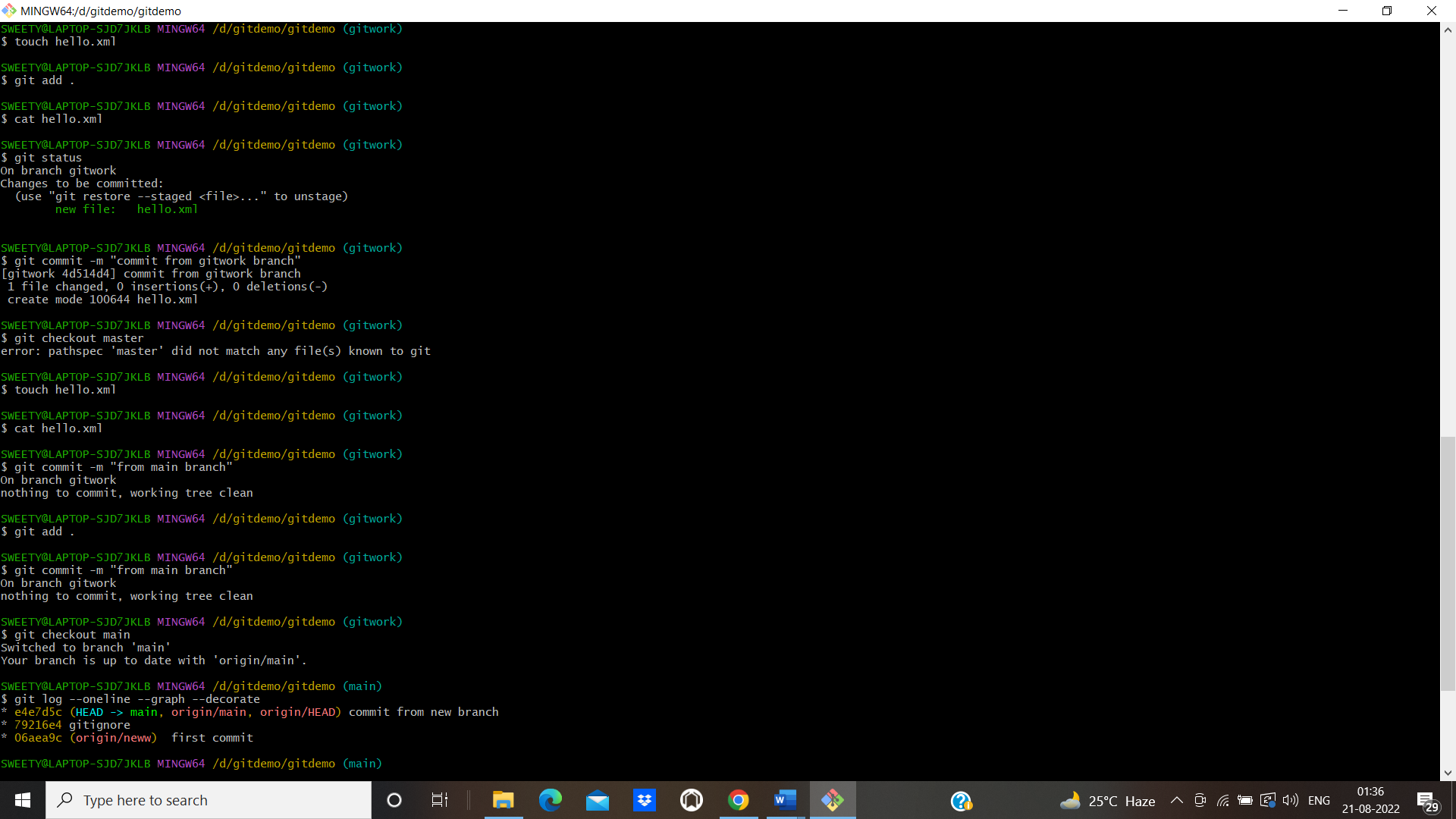
\* 06aea9c (origin/neww) first commit

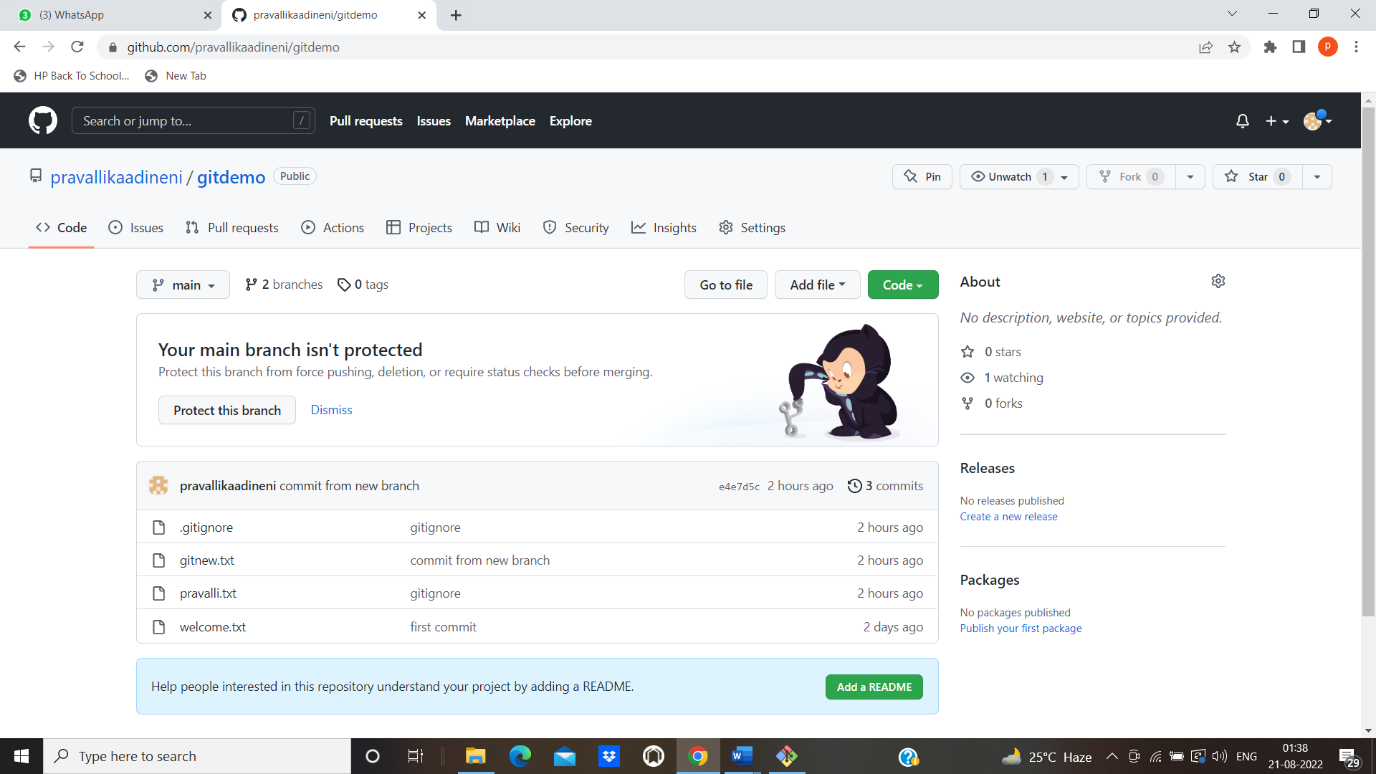
SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git log --oneline --graph --decorateall

fatal: unrecognized argument: --decorateall







Assignment 5

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git status

On branch main

Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git branch

gitwork

\* main

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git pull

Already up to date.

SWEETY@LAPTOP-SJD7JKLB MINGW64 /d/gitdemo/gitdemo (main)

$ git diff origin/main --gitnew.txt

usage: git diff [<options>] [<commit>] [--] [<path>...]

or: git diff [<options>] --cached [--merge-base] [<commit>] [--] [<path>...]

or: git diff [<options>] [--merge-base] <commit> [<commit>...] <commit> [--] [<path>...]

or: git diff [<options>] <commit>...<commit> [--] [<path>...]

or: git diff [<options>] <blob> <blob>

or: git diff [<options>] --no-index [--] <path> <path>

common diff options:

-z output diff-raw with lines terminated with NUL.

-p output patch format.

-u synonym for -p.

--patch-with-raw

output both a patch and the diff-raw format.

--stat show diffstat instead of patch.

--numstat show numeric diffstat instead of patch.

--patch-with-stat

output a patch and prepend its diffstat.

--name-only show only names of changed files.

--name-status show names and status of changed files.

--full-index show full object name on index lines.

--abbrev=<n> abbreviate object names in diff-tree header and diff-raw.

-R swap input file pairs.

-B detect complete rewrites.

-M detect renames.

-C detect copies.

--find-copies-harder

try unchanged files as candidate for copy detection.

-l<n> limit rename attempts up to <n> paths.

-O<file> reorder diffs according to the <file>.

-S<string> find filepair whose only one side contains the string.

--pickaxe-all

show all files diff when -S is used and hit is found.

-a --text treat all files as text.

