Q1) Code

KIIT@BT-22053575 MINGW64 ~

$ git config --list

diff.astextplain.textconv=astextplain

filter.lfs.clean=git-lfs clean -- %f

filter.lfs.smudge=git-lfs smudge -- %f

filter.lfs.process=git-lfs filter-process

filter.lfs.required=true

http.sslbackend=openssl

http.sslcainfo=C:/Program Files/Git/mingw64/etc/ssl/certs/ca-bundle.crt

core.autocrlf=true

core.fscache=true

core.symlinks=true

pull.rebase=false

credential.helper=manager

credential.https://dev.azure.com.usehttppath=true

init.defaultbranch=master

user.name=Anand

user.email=22053575@kiit.ac.in

filter.lfs.clean=git-lfs clean -- %f

filter.lfs.smudge=git-lfs smudge -- %f

filter.lfs.process=git-lfs filter-process

filter.lfs.required=true

KIIT@BT-22053575 MINGW64 ~

$ git config --global core.editor "notepad++ -multiInst -notabbar -nosession -noPlugin"

KIIT@BT-22053575 MINGW64 ~

$ git config --global -e

KIIT@BT-22053575 MINGW64 ~

$ mkdir GitDemo

KIIT@BT-22053575 MINGW64 ~

$ cd GitDemo

KIIT@BT-22053575 MINGW64 ~/GitDemo

$ git init

Initialized empty Git repository in C:/Users/KIIT/GitDemo/.git/

KIIT@BT-22053575 MINGW64 ~/GitDemo (master)

$ echo "Welcome to GitDemo" > welcome.txt

KIIT@BT-22053575 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)

KIIT@BT-22053575 MINGW64 ~/GitDemo (master)

$ ^C

KIIT@BT-22053575 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

KIIT@BT-22053575 MINGW64 ~/GitDemo (master)

$ ^C

KIIT@BT-22053575 MINGW64 ~/GitDemo (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: welcome.txt

KIIT@BT-22053575 MINGW64 ~/GitDemo (master)

$ git commit

[master (root-commit) 76c6a85] Initial commit

1 file changed, 1 insertion(+)

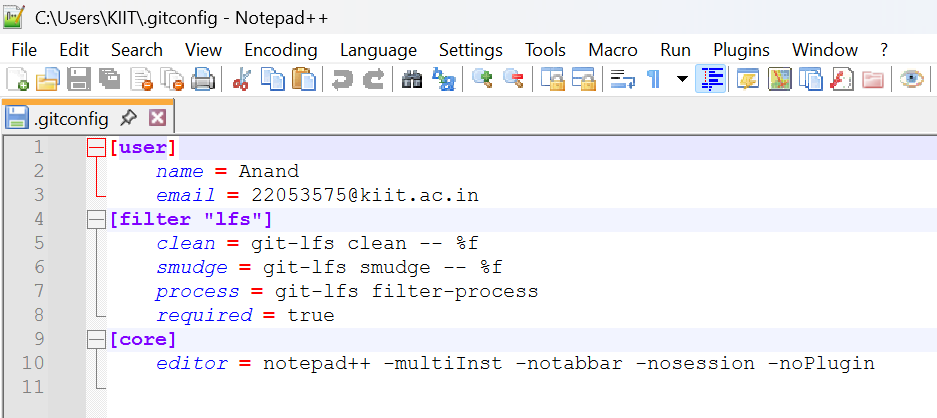
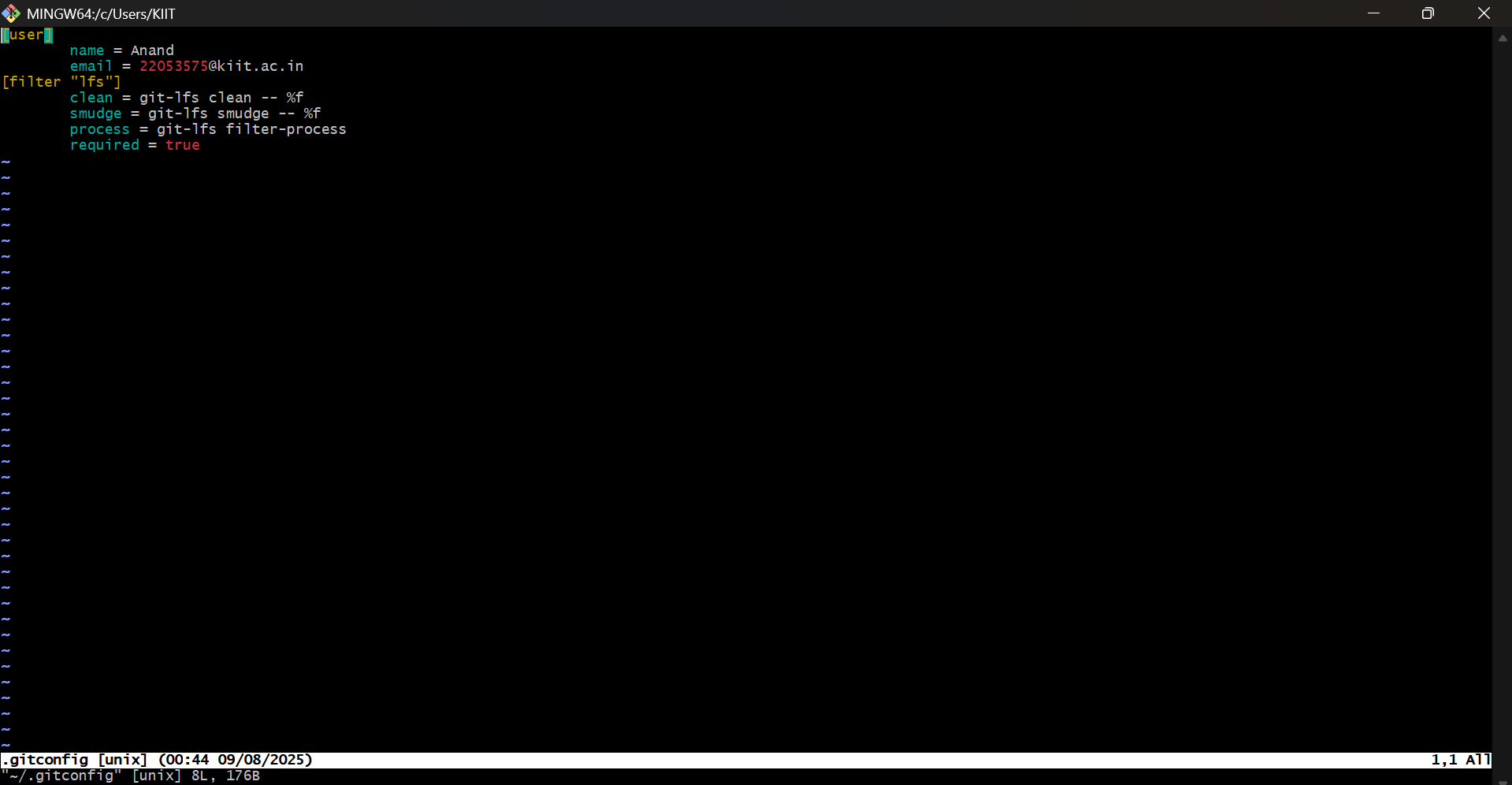
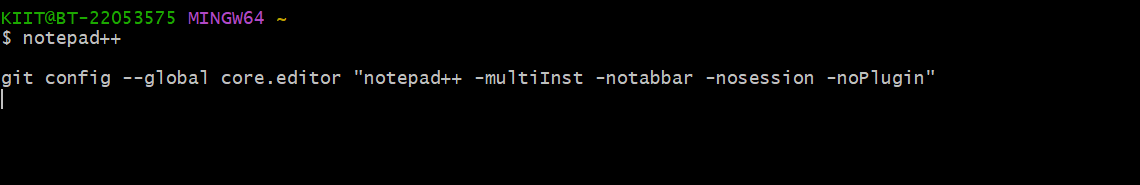
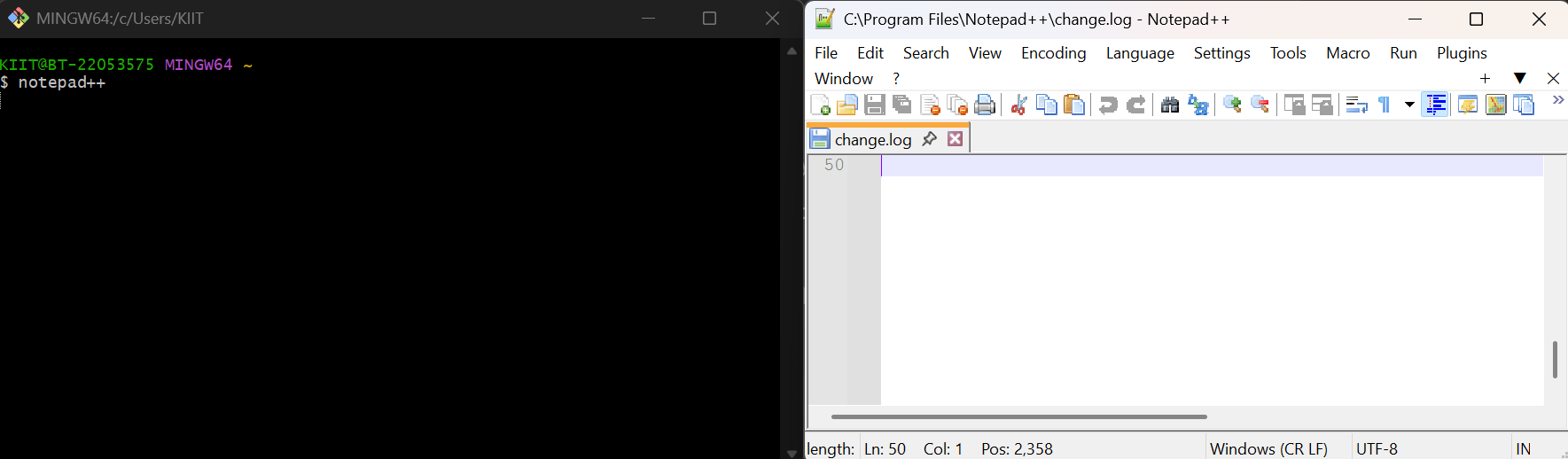
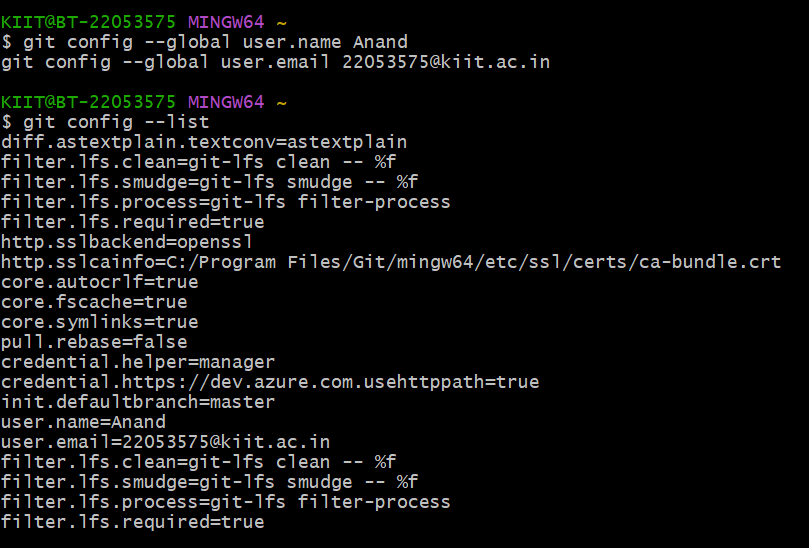
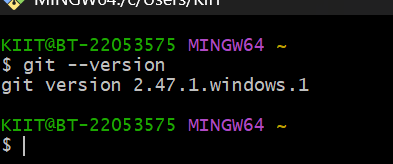
create mode 100644 welcome.txt

KIIT@BT-22053575 MINGW64 ~/GitDemo (master)

$ git status

On branch master

nothing to commit, working tree clean

Outputs  
  


Q2)

KIIT@BT-22053575 MINGW64 ~

$ cd ~/Desktop

KIIT@BT-22053575 MINGW64 ~/Desktop

$ ^[[200~git clone https://github.com/AAnand11111/6364216\_Weekwise\_HandsOn.git~

bash: $'\E[200~git': command not found

KIIT@BT-22053575 MINGW64 ~/Desktop

$ git clone https://github.com/AAnand11111/6364216\_Weekwise\_HandsOn.git

Cloning into '6364216\_Weekwise\_HandsOn'...

remote: Enumerating objects: 339, done.

remote: Counting objects: 100% (339/339), done.

remote: Compressing objects: 100% (327/327), done.

remote: Total 339 (delta 54), reused 5 (delta 1), pack-reused 0 (from 0)

Receiving objects: 100% (339/339), 16.00 MiB | 3.46 MiB/s, done.

Resolving deltas: 100% (54/54), done.

KIIT@BT-22053575 MINGW64 ~/Desktop

$ cd week8

bash: cd: week8: No such file or directory

KIIT@BT-22053575 MINGW64 ~/Desktop

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop

$ cd 6364216\_Weekwise\_HandsOn

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ cd week8

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ touch a\_test\_file.log

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ mkdir log

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ git status

On branch main

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

Untracked files:

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

a\_test\_file.log

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

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ git add .gitignore

fatal: pathspec '.gitignore' did not match any files

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ echo "\*.log" > .gitignore

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ echo "log/" >> .gitignore

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ ls -a

./ ../ .gitignore a\_test\_file.log log/ welcome.txt

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ cat .gitignore

\*.log

log/

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ git add .gitignore

warning: in the working copy of 'week8/.gitignore', LF will be replaced by CRLF the next time Git touches it

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ git commit -m "feat(week8): Add .gitignore to ignore log files and folder"

[main ca1234e] feat(week8): Add .gitignore to ignore log files and folder

1 file changed, 2 insertions(+)

create mode 100644 week8/.gitignore

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ git push

Enumerating objects: 6, done.

Counting objects: 100% (6/6), done.

Delta compression using up to 8 threads

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

Writing objects: 100% (4/4), 379 bytes | 379.00 KiB/s, done.

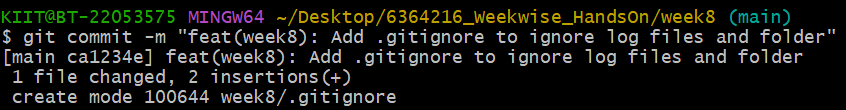
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)

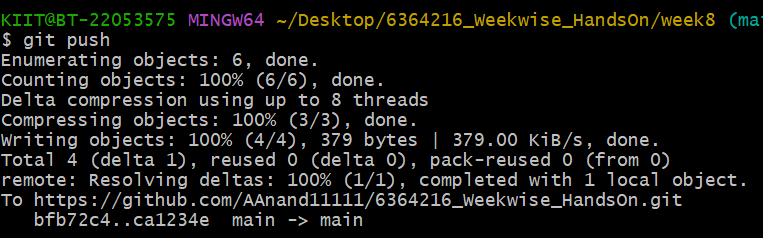
remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/AAnand11111/6364216\_Weekwise\_HandsOn.git

bfb72c4..ca1234e main -> main

**Output**





Q3)

Codes

# --- SETUP ---

# Navigate to your local repository and into the correct week's folder

cd ~/Desktop/6364216\_Weekwise\_HandsOn/week8

# --- ASSIGNMENT STEPS ---

# 1. Create the file and folder to be ignored [cite: 14]

touch a\_test\_file.log

mkdir log

# 2. Check the status to see the new untracked items

git status

# 3. Create the .gitignore file and add rules to ignore the new items [cite: 15]

echo "\*.log" > .gitignore

echo "log/" >> .gitignore

# 4. Verify that Git is now ignoring the file and folder [cite: 16]

git status

# 5. Add, commit, and push the .gitignore file to your repository

git add .gitignore

git commit -m "feat(week8): Add .gitignore to ignore log files and folder"

git push

# --- SETUP ---

# Ensure you are in the correct repository folder

cd ~/Desktop/6364216\_Weekwise\_HandsOn/week8

# --- BRANCHING ---

# 1. Create a new branch named "GitNewBranch" [cite: 31]

git branch GitNewBranch

# 2. List all local and remote branches to see the new one [cite: 32]

git branch -a

# 3. Switch to the new branch to start working in it [cite: 34]

git switch GitNewBranch

# 4. Add a new file with content to the branch [cite: 34]

echo "This is a new feature developed in a branch." > feature.txt

# 5. Commit the new file to the current branch ("GitNewBranch") [cite: 35]

git add feature.txt

git commit -m "feat: Add new feature file in branch"

# --- MERGING ---

# 6. Switch back to your main branch [cite: 38]

# (Use "master" if "main" does not work for you)

git switch main

# 7. Check for differences between the main branch and the new branch [cite: 39, 40]

# Command-line difference view

git diff GitNewBranch

# Visual difference tool (optional, launches a new window)

git difftool GitNewBranch

# 8. Merge the work from "GitNewBranch" into your main branch [cite: 41]

git merge GitNewBranch

# 9. View the project history as a graph to see the merge commit [cite: 42]

git log --oneline --graph --decorate

# 10. Delete the now-unnecessary branch as part of cleanup [cite: 43]

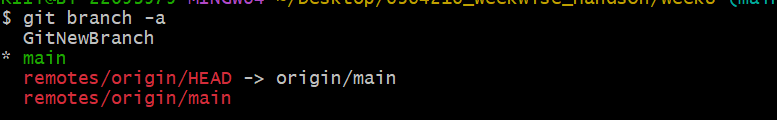
git branch -d GitNewBranch

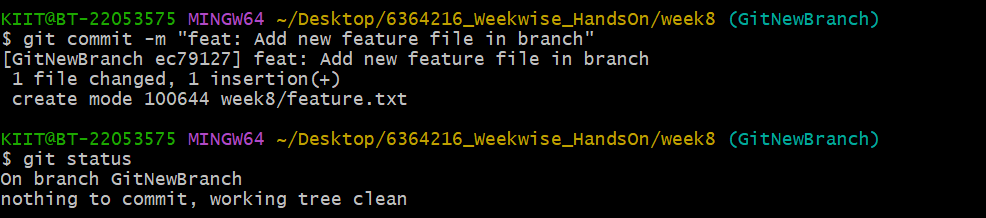
# 11. Verify the branch has been deleted

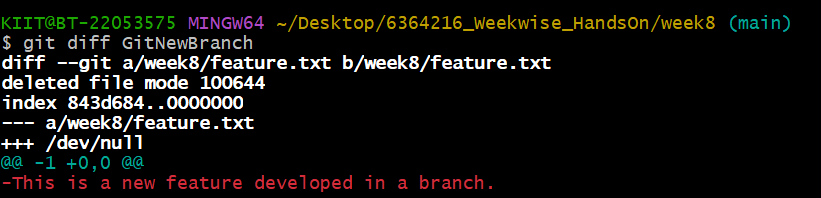
git branch

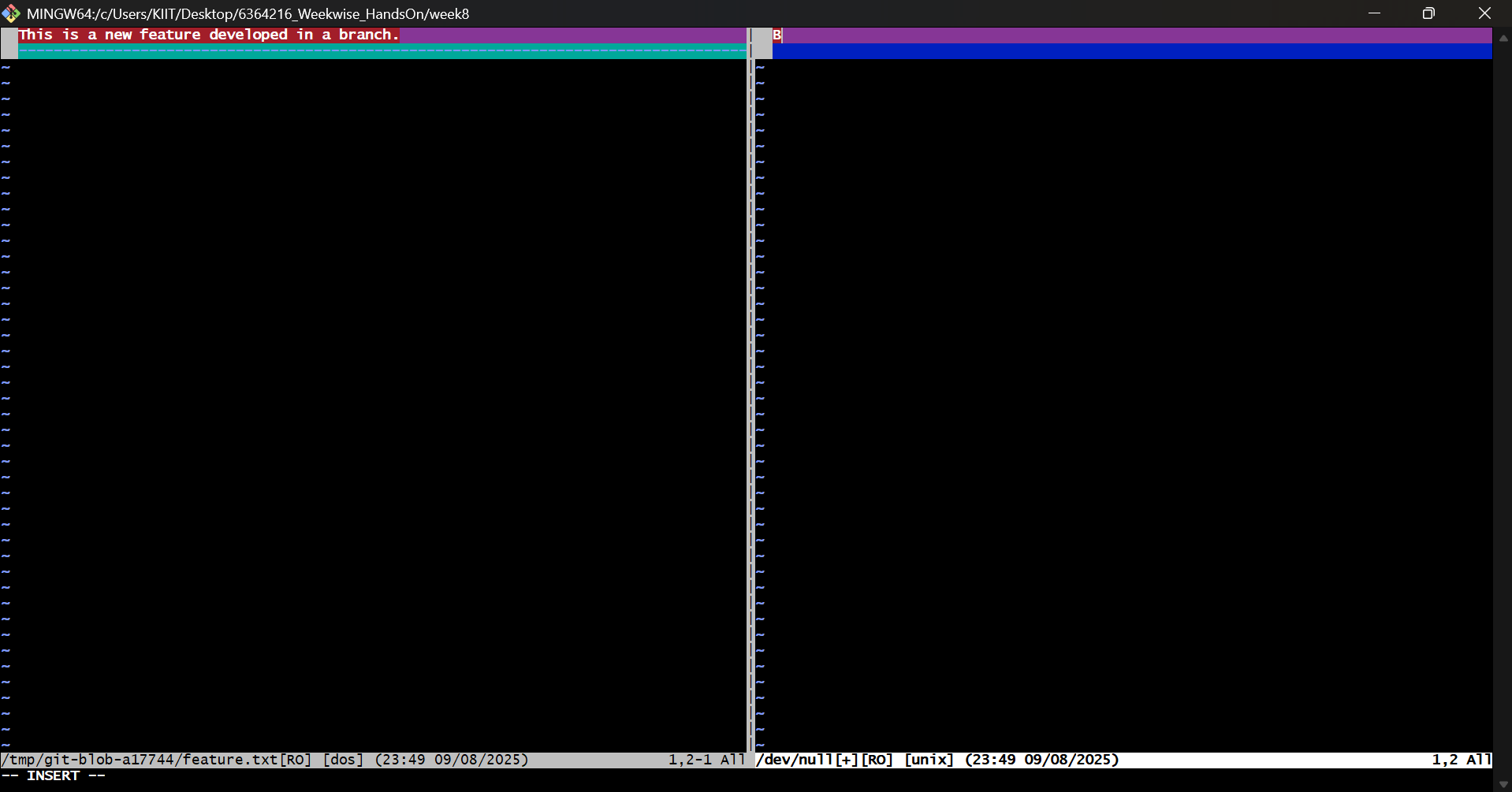
# 12. Push the newly merged changes (including feature.txt) to GitHub

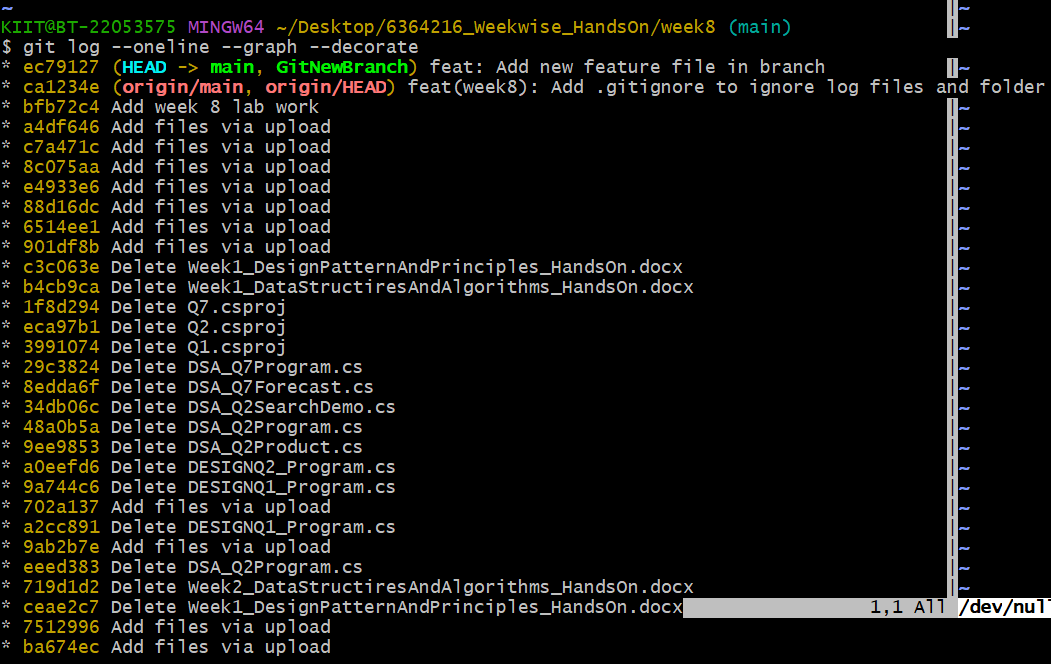
git push

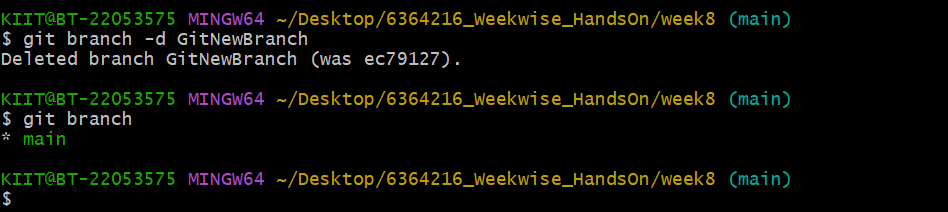












Q4)

Code

KIIT@BT-22053575 MINGW64 ~

$ cd Desktop

KIIT@BT-22053575 MINGW64 ~/Desktop

$ git clone https://github.com/AAnand11111/6364216\_Weekwise\_HandsOn.git

fatal: destination path '6364216\_Weekwise\_HandsOn' already exists and is not an empty directory.

KIIT@BT-22053575 MINGW64 ~/Desktop

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop

$ git clone https://github.com/AAnand11111/6364216\_Weekwise\_HandsOn.git Week8-Project-Fixed

Cloning into 'Week8-Project-Fixed'...

remote: Enumerating objects: 343, done.

remote: Counting objects: 100% (343/343), done.

remote: Compressing objects: 100% (329/329), done.

remote: Total 343 (delta 55), reused 9 (delta 2), pack-reused 0 (from 0)

Receiving objects: 100% (343/343), 16.00 MiB | 3.35 MiB/s, done.

Resolving deltas: 100% (55/55), done.

KIIT@BT-22053575 MINGW64 ~/Desktop

$ cd Week8-Project-Fixed

KIIT@BT-22053575 MINGW64 ~/Desktop/Week8-Project-Fixed (main)

$ git push

Everything up-to-date

KIIT@BT-22053575 MINGW64 ~/Desktop/Week8-Project-Fixed (main)

$ cd ~/Desktop/6364216\_Weekwise\_HandsOn

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ git push

Enumerating objects: 6, done.

Counting objects: 100% (6/6), done.

Delta compression using up to 8 threads

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

Writing objects: 100% (4/4), 422 bytes | 422.00 KiB/s, done.

Total 4 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/AAnand11111/6364216\_Weekwise\_HandsOn.git

ca1234e..ec79127 main -> main

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ git status

On branch main

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

nothing to commit, working tree clean

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ git branch GitWork

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ git switch GitWork

Switched to branch 'GitWork'

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ echo "<message>Hello from the GitWork branch!</message>" > hello.xml

bash: !: event not found

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ echo '<message>Hello from the GitWork branch!</message>' > hello.xml

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (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

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ ^[[200~git commit -m "feat(main): Add hello.xml in main branch"~

bash: $'\E[200~git': command not found

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ git commit -m "feat(main): Add hello.xml in main branch"

[GitWork e8c0595] feat(main): Add hello.xml in main branch

1 file changed, 1 insertion(+)

create mode 100644 hello.xml

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

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

\* e8c0595 (HEAD -> GitWork) feat(main): Add hello.xml in main branch

\* ec79127 (origin/main, origin/HEAD, main) feat: Add new feature file in branch

\* ca1234e feat(week8): Add .gitignore to ignore log files and folder

\* bfb72c4 Add week 8 lab work

\* a4df646 Add files via upload

\* c7a471c Add files via upload

\* 8c075aa Add files via upload

\* e4933e6 Add files via upload

\* 88d16dc Add files via upload

\* 6514ee1 Add files via upload

\* 901df8b Add files via upload

\* c3c063e Delete Week1\_DesignPatternAndPrinciples\_HandsOn.docx

\* b4cb9ca Delete Week1\_DataStructiresAndAlgorithms\_HandsOn.docx

\* 1f8d294 Delete Q7.csproj

\* eca97b1 Delete Q2.csproj

\* 3991074 Delete Q1.csproj

\* 29c3824 Delete DSA\_Q7Program.cs

\* 8edda6f Delete DSA\_Q7Forecast.cs

\* 34db06c Delete DSA\_Q2SearchDemo.cs

\* 48a0b5a Delete DSA\_Q2Program.cs

\* 9ee9853 Delete DSA\_Q2Product.cs

\* a0eefd6 Delete DESIGNQ2\_Program.cs

\* 9a744c6 Delete DESIGNQ1\_Program.cs

\* 702a137 Add files via upload

\* a2cc891 Delete DESIGNQ1\_Program.cs

\* 9ab2b7e Add files via upload

\* eeed383 Delete DSA\_Q2Program.cs

\* 719d1d2 Delete Week2\_DataStructiresAndAlgorithms\_HandsOn.docx

\* ceae2c7 Delete Week1\_DesignPatternAndPrinciples\_HandsOn.docx

\* 7512996 Add files via upload

\* ba674ec Add files via upload

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ git merge GitWork

Already up to date.

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ git switch main

Switched to branch 'main'

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

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ git reset --hard ec79127

HEAD is now at ec79127 feat: Add new feature file in branch

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ git switch GitWork

Switched to branch 'GitWork'

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ git reset --hard ec79127

HEAD is now at ec79127 feat: Add new feature file in branch

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ echo '<message>Hello from the GitWork branch!</message>' > hello.xml

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (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

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ git commit -m "feat(work): Add hello.xml in GitWork branch"

[GitWork fd29deb] feat(work): Add hello.xml in GitWork branch

1 file changed, 1 insertion(+)

create mode 100644 hello.xml

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (GitWork)

$ git switch main

Switched to branch 'main'

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

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ echo '<message>Hello from the main branch!</message>' > hello.xml

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (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

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ ^[[200~git commit -m "feat(main): Add hello.xml in main branch"~

bash: $'\E[200~git': command not found

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ git commit -m "feat(main): Add hello.xml in main branch"

[main 8899c45] feat(main): Add hello.xml in main branch

1 file changed, 1 insertion(+)

create mode 100644 hello.xml

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (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.

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main|MERGING)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main|MERGING)

$ ls -a

./ ../ .git/ Q6ConsoleApp1.sln WEEK1/ Week\_6/ hello.xml week2/ week3/ week4/ week5/ week7/ week8/

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main|MERGING)

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

\* 8899c45 (HEAD -> main) feat(main): Add hello.xml in main branch

| \* fd29deb (GitWork) feat(work): Add hello.xml in GitWork branch

|/

\* ec79127 (origin/main, origin/HEAD) feat: Add new feature file in branch

\* ca1234e feat(week8): Add .gitignore to ignore log files and folder

\* bfb72c4 Add week 8 lab work

\* a4df646 Add files via upload

\* c7a471c Add files via upload

\* 8c075aa Add files via upload

\* e4933e6 Add files via upload

\* 88d16dc Add files via upload

\* 6514ee1 Add files via upload

\* 901df8b Add files via upload

\* c3c063e Delete Week1\_DesignPatternAndPrinciples\_HandsOn.docx

\* b4cb9ca Delete Week1\_DataStructiresAndAlgorithms\_HandsOn.docx

\* 1f8d294 Delete Q7.csproj

\* eca97b1 Delete Q2.csproj

\* 3991074 Delete Q1.csproj

\* 29c3824 Delete DSA\_Q7Program.cs

\* 8edda6f Delete DSA\_Q7Forecast.cs

\* 34db06c Delete DSA\_Q2SearchDemo.cs

\* 48a0b5a Delete DSA\_Q2Program.cs

\* 9ee9853 Delete DSA\_Q2Product.cs

\* a0eefd6 Delete DESIGNQ2\_Program.cs

\* 9a744c6 Delete DESIGNQ1\_Program.cs

\* 702a137 Add files via upload

\* a2cc891 Delete DESIGNQ1\_Program.cs

\* 9ab2b7e Add files via upload

\* eeed383 Delete DSA\_Q2Program.cs

\* 719d1d2 Delete Week2\_DataStructiresAndAlgorithms\_HandsOn.docx

\* ceae2c7 Delete Week1\_DesignPatternAndPrinciples\_HandsOn.docx

\* 7512996 Add files via upload

\* ba674ec Add files via upload

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main|MERGING)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main|MERGING)

$ git commit

U hello.xml

error: Committing is not possible because you have unmerged files.

hint: Fix them up in the work tree, and then use 'git add/rm <file>'

hint: as appropriate to mark resolution and make a commit.

fatal: Exiting because of an unresolved conflict.

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main|MERGING)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main|MERGING)

$ git add hello.xml

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main|MERGING)

$

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main|MERGING)

$ git commit

[main efb8131] Merge branch 'GitWork'

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ git branch -d GitWork

Deleted branch GitWork (was fd29deb).

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ git push

Enumerating objects: 10, done.

Counting objects: 100% (10/10), done.

Delta compression using up to 8 threads

Compressing objects: 100% (8/8), done.

Writing objects: 100% (9/9), 987 bytes | 329.00 KiB/s, done.

Total 9 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)

remote: Resolving deltas: 100% (3/3), completed with 1 local object.

To https://github.com/AAnand11111/6364216\_Weekwise\_HandsOn.git

ec79127..efb8131 main -> main

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ ^C

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ git mv hello.xml week8/

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ git commit -m "chore: Move hello.xml into week8 folder"

[main 1989448] chore: Move hello.xml into week8 folder

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

rename hello.xml => week8/hello.xml (100%)

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn (main)

$ git push

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Delta compression using up to 8 threads

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

Writing objects: 100% (3/3), 335 bytes | 335.00 KiB/s, done.

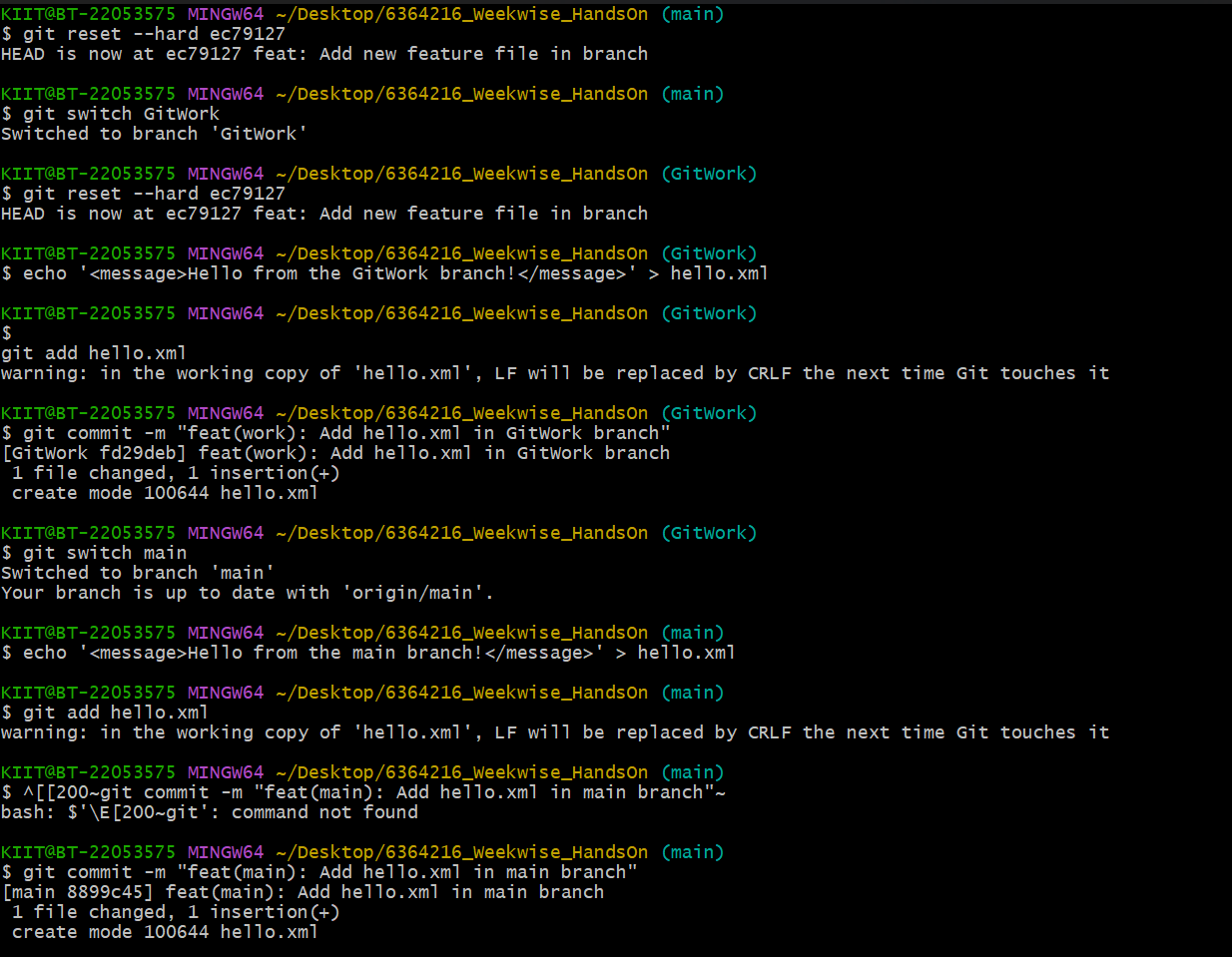
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)

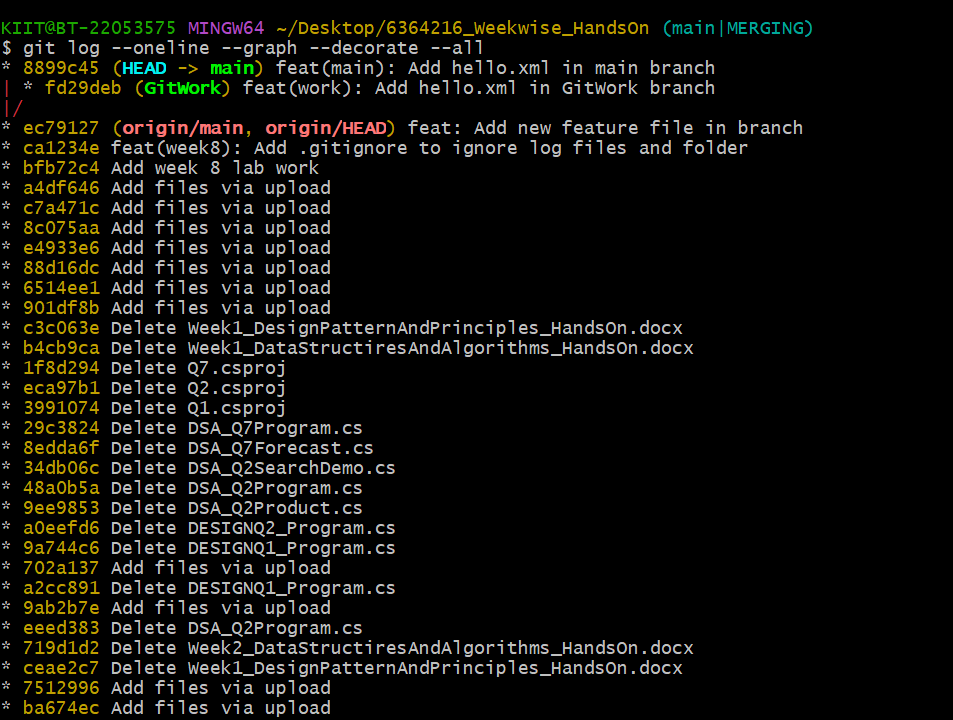
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.

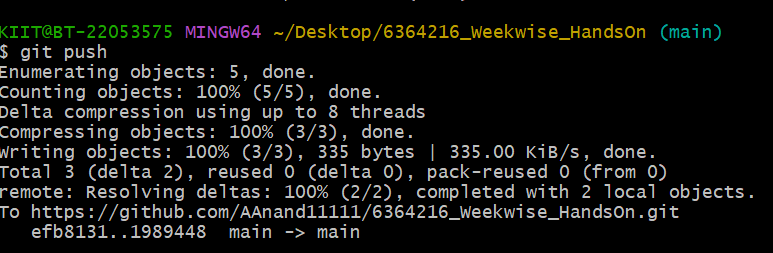
To https://github.com/AAnand11111/6364216\_Weekwise\_HandsOn.git

efb8131..1989448 main -> main

Outputs







Q5)

Codes

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ git status

On branch main

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

nothing to commit, working tree clean

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ git branch -a

\* main

remotes/origin/HEAD -> origin/main

remotes/origin/main

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ git pull origin main

From https://github.com/AAnand11111/6364216\_Weekwise\_HandsOn

\* branch main -> FETCH\_HEAD

Already up to date.

KIIT@BT-22053575 MINGW64 ~/Desktop/6364216\_Weekwise\_HandsOn/week8 (main)

$ git push

Everything up-to-date

Outputs



