MODULE -5

MANAGING SOURCE CODE – GIT

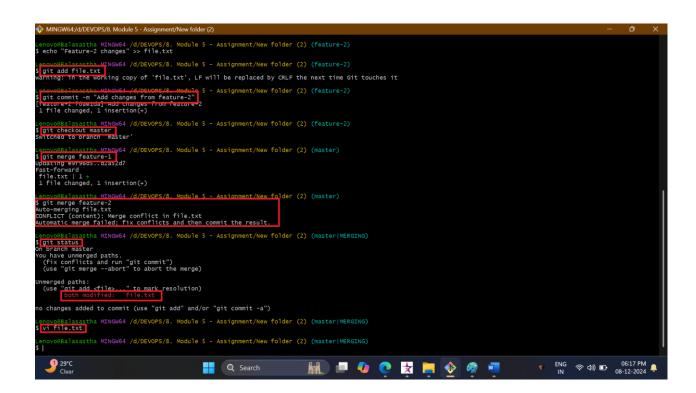# AND GITHUB

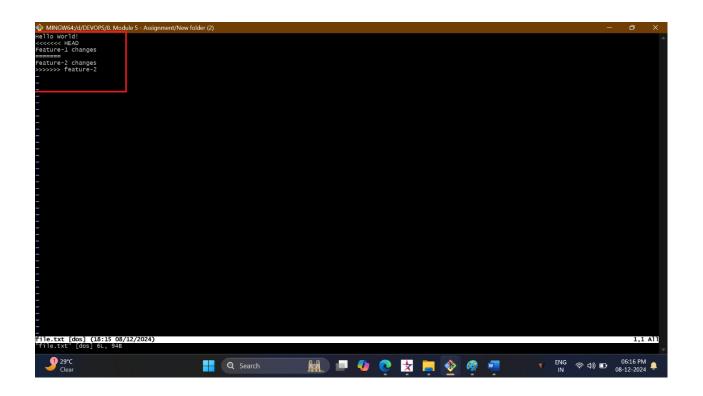## SUBMITTED BY – BALASASTHA P

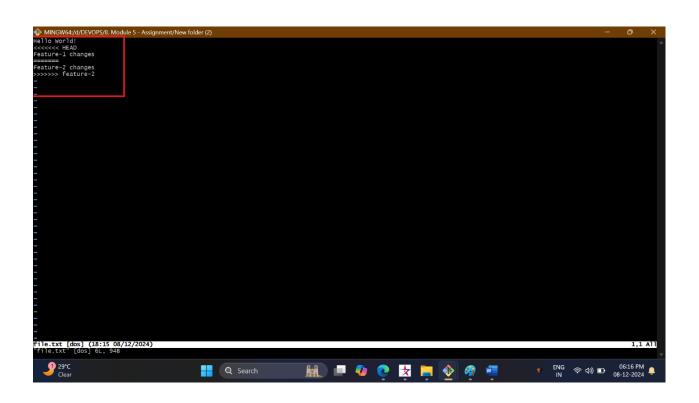## SUBMITTED TO – STARAGILE

DATE OF SUBMISSION – 08-12-2024

L2

L2 - Create Local git repository and demonstrate git merge and Merge Conflicts with the steps to resolve merge conflicts.

To work with git merge and resolve merge conflicts, first create a folder → Open Git Bash in the folder and initialize a Git repository using ***"git init"*** → Add a file using ***"echo 'Hello World!' > file.txt"***, commit the changes with ***"git add file.txt"*** and ***"git commit -m 'Initial commit'"*** → Create a new branch and enter into the branch with ***"git checkout -b feature-1"*** → Add a file using ***"echo 'Feature-1 changes' > file.txt"***, commit the changes with ***"git add file.txt"*** and ***"git commit -m 'Add changes from feature-1'"*** → Switch to the main branch with ***"git checkout main"*** and create and enter into another branch using ***"git checkout -b feature-2"*** → Modify file.txt differently with ***"echo 'Add changes from feature-2' >> file.txt"*** and commit the changes with ***"git add file.txt"*** and ***"git commit -m 'feature-2 first commit'"*** → Return to main with ***"git checkout main"*** and merge feature-1 using ***"git merge feature-1"*** (this will succeed) → Merge feature-2 using ***"git merge feature-2"***, which will cause a conflict → Open the conflicting file ***"vi file.txt"*** resolve it by combining changes (e.g., keeping both ***"feature-1 changes"*** and ***"feature-2 changes"***) → Commit the resolved file with ***"git add file.txt"*** and merge with ***"git commit"*** → Verify the result using ***"git log --oneline --graph"***.

```
MINGW64:/d/DEVOPS/8. Module 5 - Assignment/New folder (2)

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2)
$ git init
Initialized empty Git repository in D:/DEVOPS/8. Module 5 - Assignment/New folder (2)/.git/

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (master)
$ echo "Hello World!" > file.txt

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (master)
$ git add file.txt
warning: in the working copy of 'file.txt', LF will be replaced by CRLF the next time Git touches it

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (master)
$ git commit -m "Initial commit"
[master (root-commit) e9f96d5] Initial commit
 1 file changed, 1 insertion(+)
 create mode 100644 file.txt

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (master)
$ git checkout -b feature-1
Switched to a new branch 'feature-1'

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (feature-1)
$ echo "Feature-1 changes" >> file.txt

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (feature-1)
$ git add file.txt
warning: in the working copy of 'file.txt', LF will be replaced by CRLF the next time Git touches it

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (feature-1)
$ git commit -m "Add changes from feature-1"
[feature-1 dza52d7] Add changes from feature-1
 1 file changed, 1 insertion(+)

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (feature-1)
$ git checkout master
Switched to branch 'master'

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (master)
$ git checkout -b feature-2
Switched to a new branch 'feature-2'

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (feature-2)
$ echo "Feature-2 changes" >> file.txt
```



```
MINGW64:/d/DEVOPS/8. Module 5 - Assignment/New folder (2)

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (feature-2)
$ echo "Feature-2 changes" >> file.txt

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (feature-2)
$ git add file.txt
warning: in the working copy of 'file.txt', LF will be replaced by CRLF the next time Git touches it

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (feature-2)
$ git commit -m "Add changes from feature-2"
[feature-2 70ae1da] Add changes from feature-2
 1 file changed, 1 insertion(+)

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (feature-2)
$ git checkout master
Switched to branch 'master'

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (master)
$ git merge feature-1
Updating e9f96d5..dza52d7
Fast-forward
 file.txt | 1 +
 1 file changed, 1 insertion(+)

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (master)
$ git merge feature-2
Auto-merging file.txt
CONFLICT (content): Merge conflict in file.txt
Automatic merge failed; fix conflicts and then commit the result.

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (master|MERGING)
$ git status
On branch master
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 modified:   file.txt

no changes added to commit (use "git add" and/or "git commit -a")

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (master|MERGING)
$ vi file.txt

Lenovo@Balasastha MINGW64 /d/DEVOPS/8. Module 5 - Assignment/New folder (2) (master|MERGING)
$ |
```

```
Hello world!
<<<<<<< HEAD
Feature-1 changes
=======
Feature-2 changes
>>>>>>> feature-2
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
file.txt [dos] (18:15 08/12/2024)                                    1,1 All
"file.txt" [dos] 6L, 94B
```



```
Hello world!
<<<<<<< HEAD
Feature-1 changes
=======
Feature-2 changes
>>>>>>> feature-2
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
file.txt [dos] (18:15 08/12/2024)                                    1,1 All
"file.txt" [dos] 6L, 94B
```

Assignment completed
Thank You!!!