Git\_HOL\_2

**Objectives**

· Explain git ignore

· Explain how to ignore unwanted files using git ignore

In this hands-on lab, you will learn how to:

· Implement git ignore command to ignore unwanted files and folders

**STEPS**

# 1) Go to your project folder (or create one)

mkdir GitDemo

cd GitDemo

# 2) Initialize Git repository

git init

# 3) Configure your Git identity (only once on your system)

git config --global user.name "Your Name"

git config --global user.email "your@email.com"

# 4) Set Notepad++ as the default Git editor (adjust path if needed)

git config --global core.editor "'/c/Program Files/Notepad++/notepad++.exe' -multiInst -nosession -notabbar"

# 5) Create an initial file

echo "Welcome to GitDemo" > welcome.txt

# 6) Stage and commit the file

git add welcome.txt

git commit -m "Initial commit with welcome.txt"

# 7) Create a remote repo on GitHub named GitDemo (empty — no README)

# Then link it to your local repo (replace URL with yours)

git remote add origin https://github.com/YOUR-USERNAME/GitDemo.git

git remote -v

# 8) Rename local branch to main

git branch -M main

# 9) Push the commit to GitHub

git push -u origin main

# 10) Create a .log file and logs folder for the .gitignore demo

echo "This is a log file" > error.log

mkdir logs

echo "Log file inside folder" > logs/debug.log

# 11) Create .gitignore file to ignore .log files and logs folder

echo "\*.log" > .gitignore

echo "logs/" >> .gitignore

# 12) Check status to verify logs are ignored

git status

# 13) Stage and commit .gitignore

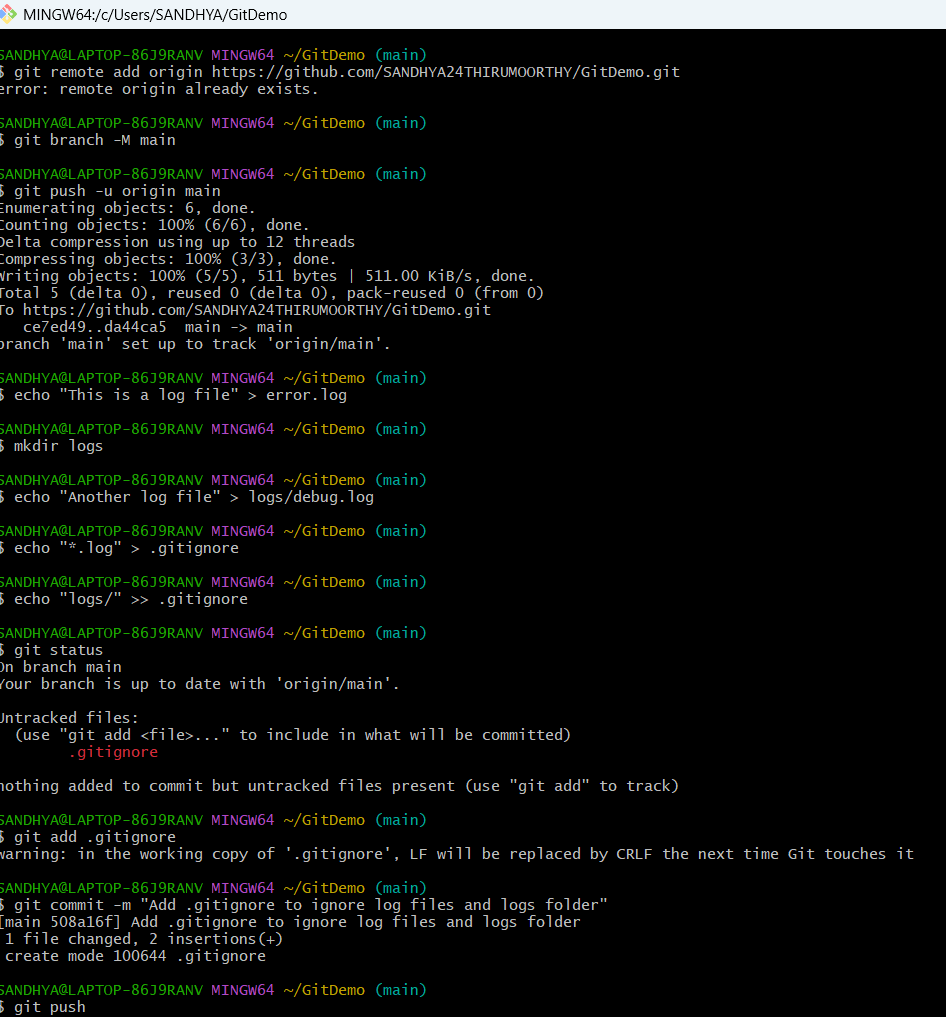
git add .gitignore

git commit -m "Add .gitignore to ignore log files and logs folder"

# 14) Push the changes to GitHub

git push

**OUTPUT**

****