Exercise – 2

**Objectives**

* Explain .gitignore
* Explain how to ignore unwanted files using .gitignore

**In this lab, we will learn to:**

* Implement .gitignore to exclude specific files and folders from being tracked by Git

**Prerequisites**

To successfully complete this hands-on lab, ensure the following are done:

* Git environment is set up on your machine
* Notepad++ is configured as the default editor
* A Git repository is available locally and a remote repository is created in GitLab

**Estimated time to complete:** 20 minutes

**Understanding .gitignore**

* .gitignore is a special file used in Git to specify files or directories that should be **excluded** from version control.
* Common examples of files to ignore:
  + Temporary files (\*.tmp)
  + Log files (\*.log)
  + Build folders (/dist/, /log/)
  + IDE/project files (.vscode/, .idea/)

**Hands-On Instructions**

**Step 1: Setup Git Repository**

1. Open **Git Bash**.
2. Create or move to an existing project folder:

mkdir GitIgnoreDemo

cd GitIgnoreDemo

git init

**Step 2: Create Unwanted Files**

1. Create a .log file and a folder named log:

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

mkdir log

echo "Log folder file" > log/info.txt

**Step 3: Create and Update .gitignore**

1. Create the .gitignore file:

touch .gitignore

1. Open it using Notepad++:

notepad++ .gitignore

1. Add the following lines to ignore .log files and the log/ folder:

\*.log

log/

**Step 4: Verify Git Status**

1. Check status using:

git status

1. You should **not** see debug.log or the log/ folder in the output, confirming they are ignored.
2. Files like .gitignore **will** appear, since we want to commit it.

**Step 5: Stage and Commit Changes**

1. Add and commit only the .gitignore file:

git add .gitignore

git commit -m "Added .gitignore to exclude .log files and log folder"

1. Verify that debug.log and log/ remain untracked.

**Step 6: Push to Remote Repository**

1. Link remote repository (replace URL with your GitLab repo):

git remote add origin https://gitlab.com/yourusername/GitIgnoreDemo.git

1. Push changes:

git push -u origin master

**Conclusion**

* The .gitignore file successfully prevents unwanted files (\*.log, log/) from being added to Git.
* Verified via git status that only .gitignore was tracked and committed.
* The .gitignore mechanism is essential to maintain a clean repository by excluding temporary, build, and environment-specific files.