**GIT bash commands used:**

HP@DESKTOP-RD6TT5J MINGW64 ~

$ C:/Users/HP/GitDemo

bash: C:/Users/HP/GitDemo: Is a directory

HP@DESKTOP-RD6TT5J MINGW64 ~

$ pwd

/c/Users/HP

HP@DESKTOP-RD6TT5J MINGW64 ~

$ cd GitDemo

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (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)

errors.log

log/

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

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ cat > .gitignore <<'EOF' \*.log

> log/

> EOF

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (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)

.gitignore

errors.log

log/

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

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git status --ignored

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)

.gitignore

errors.log

log/

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

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git ls-files --others --ignored --exclude-standard

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git add .gitignore

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

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

[main 3ccdf6c] Add .gitignore to ignore .log files and log/ folder

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

create mode 100644 .gitignore

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git status

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

(use "git push" to publish your local commits)

Untracked files:

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

errors.log

log/

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

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git check-ignore -v errors.log

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git rm --cached errors.log

fatal: pathspec 'errors.log' did not match any files

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git rm -r --cached log/

fatal: pathspec 'log/' did not match any files

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git commit -m "Remove log files from repo, keep locally (ignored)"

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

(use "git push" to publish your local commits)

Untracked files:

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

errors.log

log/

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

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git status --ignored

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

(use "git push" to publish your local commits)

Untracked files:

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

errors.log

log/

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

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ \*.log

bash: errors.log: command not found

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ cat > .gitignore <<'EOF'

> \*.log

> log/

> EOF

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git status

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

(use "git push" to publish your local commits)

Changes not staged for commit:

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

(use "git restore <file>..." to discard changes in working directory)

modified: .gitignore

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

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git status --ignored

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

(use "git push" to publish your local commits)

Changes not staged for commit:

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

(use "git restore <file>..." to discard changes in working directory)

modified: .gitignore

Ignored files:

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

errors.log

log/

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

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git ls-files --others --ignored --exclude-standard

errors.log

log/app.log

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git add .gitignore

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

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

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

[main 2c49024] Add .gitignore to ignore .log files and log/ folder

1 file changed, 2 insertions(+)

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git status

On branch main

Your branch is ahead of 'origin/main' by 2 commits.

(use "git push" to publish your local commits)

nothing to commit, working tree clean

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git check-ignore -v errors.log

.gitignore:1:\*.log errors.log

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git push origin main

Enumerating objects: 7, done.

Counting objects: 100% (7/7), done.

Delta compression using up to 4 threads

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

Writing objects: 100% (6/6), 594 bytes | 297.00 KiB/s, done.

Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

To https://github.com/venky212/GitDemo

2032de0..2c49024 main -> main

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git rm --cached errors.log

fatal: pathspec 'errors.log' did not match any files

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git rm -r --cached log/

fatal: pathspec 'log/' did not match any files

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ cat > .gitignore <<'EOF'

> \*.log

> log/

> EOF

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git rm --cached errors.log

fatal: pathspec 'errors.log' did not match any files

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ nano .gitignore

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git status --ignored

On branch main

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

Changes not staged for commit:

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

(use "git restore <file>..." to discard changes in working directory)

modified: .gitignore

Ignored files:

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

errors.log

log/

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

HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

$ git add .gitignore

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

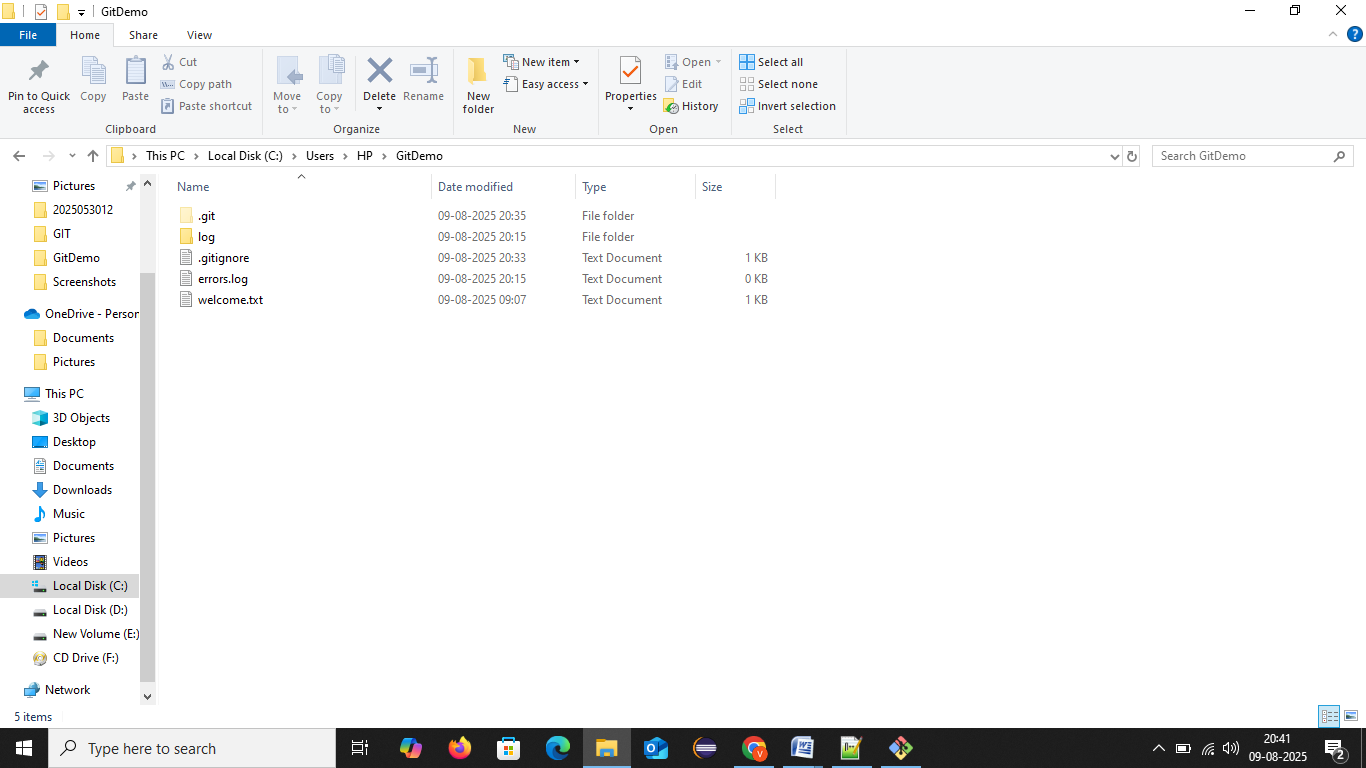
HP@DESKTOP-RD6TT5J MINGW64 ~/GitDemo (main)

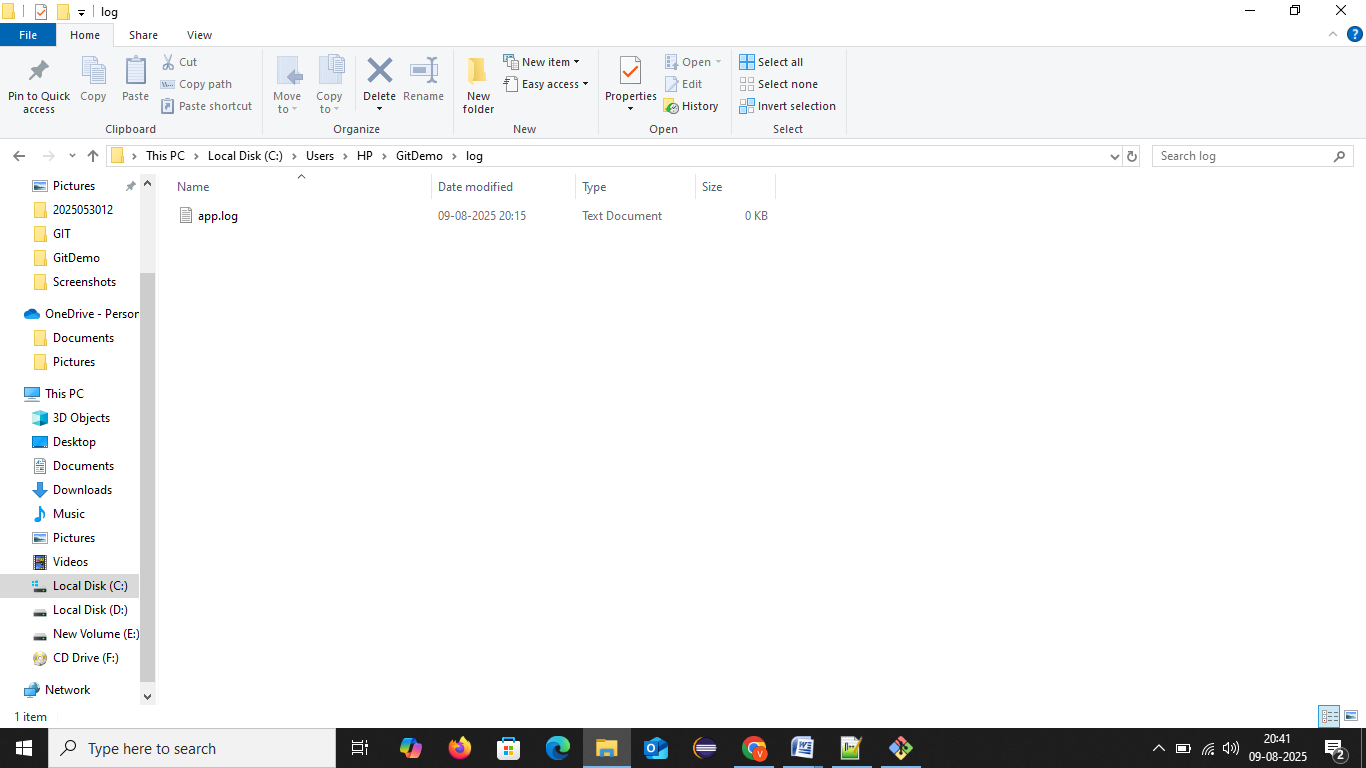
$ git commit -m "Ignore .log files and log directory"

[main aaf9d44] Ignore .log files and log directory

1 file changed, 1 insertion(+)

**log files created:**

****

****