**GIT BASH**

**EXERCISE 1**

**Commands:**

parva@CharviSankar MINGW64 ~

$ git --version

git version 2.50.1.windows.1

parva@CharviSankar MINGW64 ~

$ git config --global user.name "CharviSankar"

parva@CharviSankar MINGW64 ~

$ git config --global user.email "cs.parvathareddycharvi123@gmail.com"

parva@CharviSankar MINGW64 ~

$ git config --global --list

user.name=CharviSankar

user.email=cs.parvathareddycharvi123@gmail.com

core.editor="C:/Program Files/Notepad++/notepad++.exe" -multiInst -nosession

parva@CharviSankar MINGW64 ~

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

parva@CharviSankar MINGW64 ~

$ git config --global -e

parva@CharviSankar MINGW64 ~

$ mkdir GitDemo

parva@CharviSankar MINGW64 ~

$ cd GitDemo

parva@CharviSankar MINGW64 ~/GitDemo

$ git init

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

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ echo "Hello Git" > welcome.txt

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ ls

welcome.txt

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ cat welcome.txt

Hello Git

parva@CharviSankar 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)

parva@CharviSankar 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

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git commit

Aborting commit due to empty commit message.

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git commit

[master (root-commit) d532470] Added welcome.txt file

1 file changed, 1 insertion(+)

create mode 100644 welcome.txt

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git remote add origin https://gitlab.com/CharviSankar/gitdemo.git

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git push -u origin master

info: please complete authentication in your browser...

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git push -u origin master

info: please complete authentication in your browser...

Enumerating objects: 3, done.

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

Writing objects: 100% (3/3), 245 bytes | 245.00 KiB/s, done.

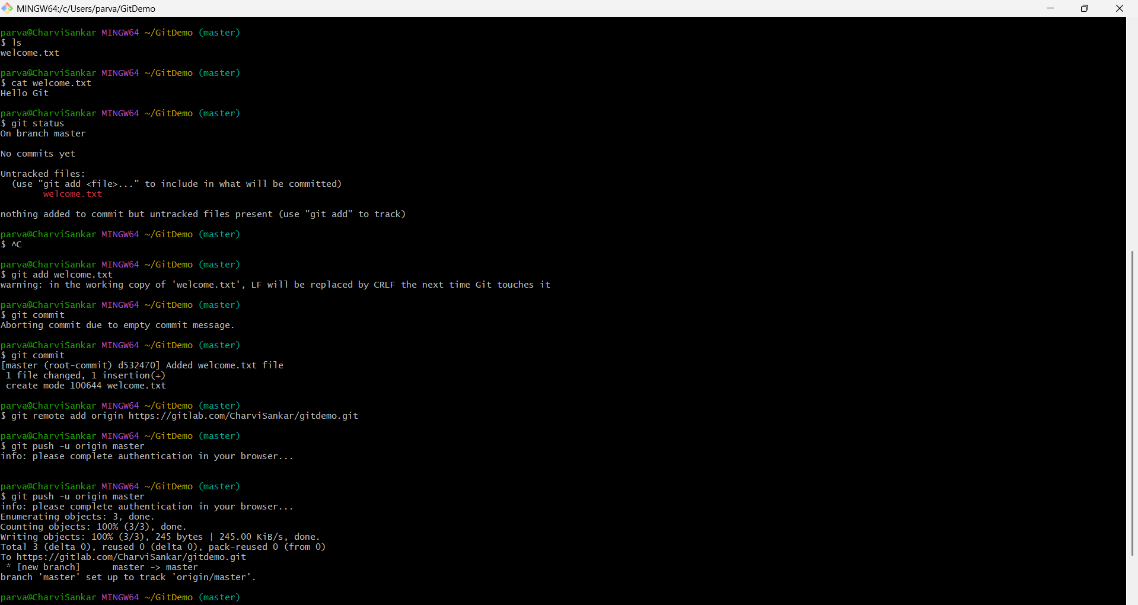
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

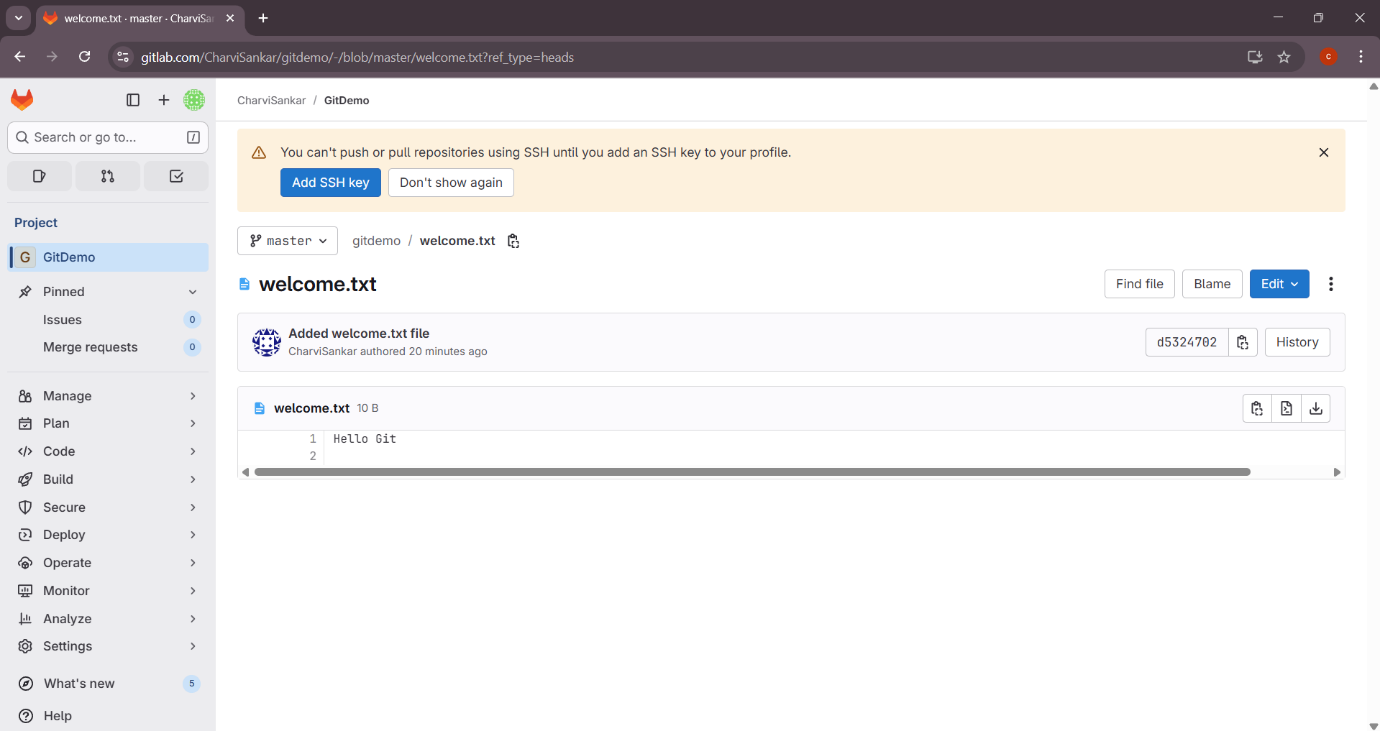
To https://gitlab.com/CharviSankar/gitdemo.git

\* [new branch] master -> master

branch 'master' set up to track 'origin/master'.

**OUTPUT:**





**EXERCISE 2**

Commands:

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ touch error.log

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ mkdir log

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ notepad++ .gitignore

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git add .gitignore

parva@CharviSankar MINGW64 ~/GitDemo (master)

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

[master 114bb35] Add .gitignore to ignore .log files and log folder

1 file changed, 2 insertions(+)

create mode 100644 .gitignore

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git status

On branch master

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

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

nothing to commit, working tree clean

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git status

On branch master

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

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

nothing to commit, working tree clean

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ cat .gitignore

\*.log

log/

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git rm --cached \*.log

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

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git rm --cached -r log/

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

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git commit -m "Ignore log files and log folder"

On branch master

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

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

nothing to commit, working tree clean

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git status

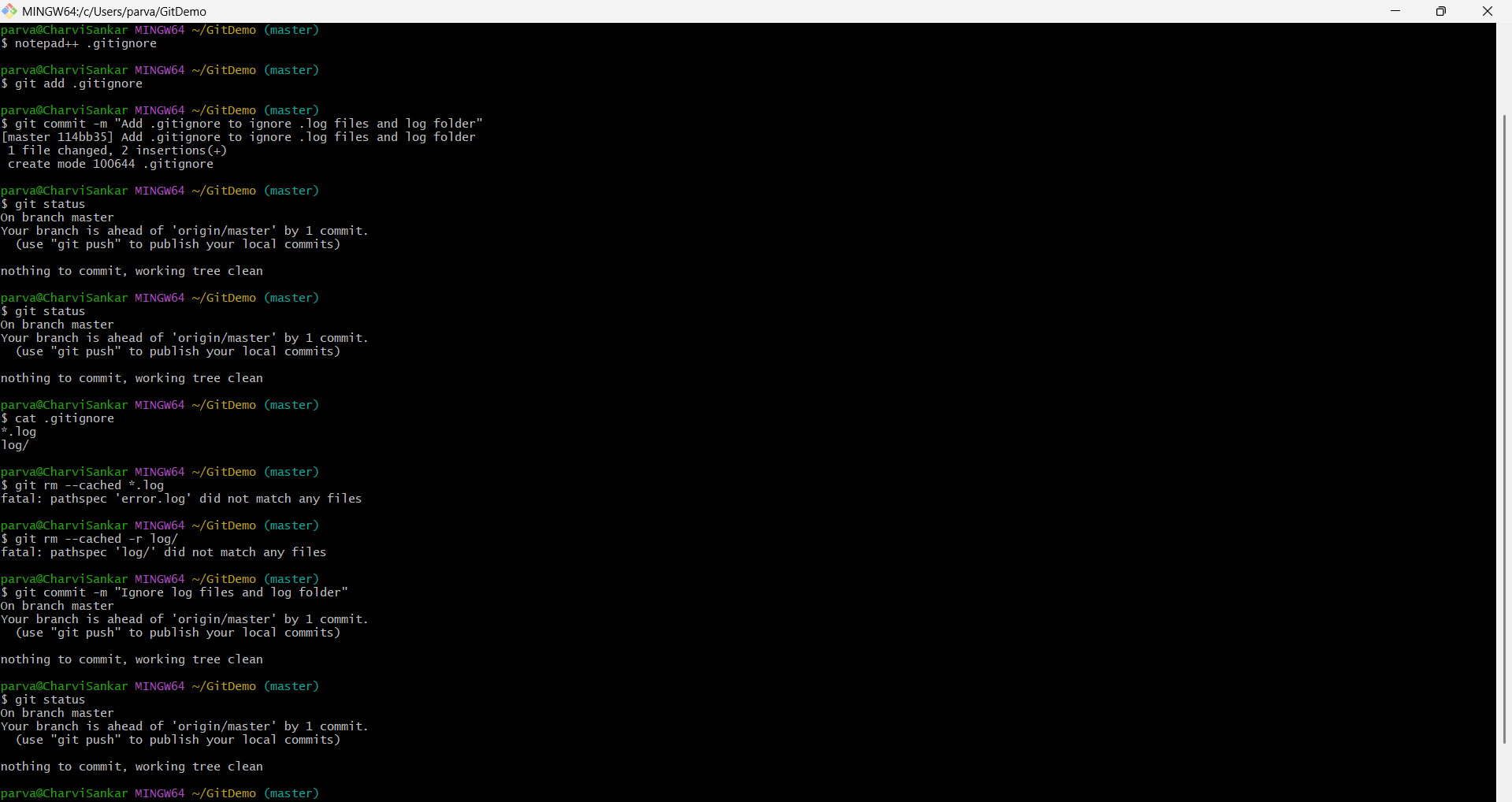
On branch master

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

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

nothing to commit, working tree clean

**OUTPUT:**



**EXERCISE 3:**

**Commands:**

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git branch GitNewBranch

fatal: a branch named 'GitNewBranch' already exists

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git branch -a

GitNewBranch

\* master

remotes/origin/master

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git checkout GitNewBranch

Switched to branch 'GitNewBranch'

parva@CharviSankar MINGW64 ~/GitDemo (GitNewBranch)

$ echo "This is a test file in GitNewBranch." > testfile.txt

parva@CharviSankar MINGW64 ~/GitDemo (GitNewBranch)

$ git add testfile.txt

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

parva@CharviSankar MINGW64 ~/GitDemo (GitNewBranch)

$ git commit -m "Added testfile.txt in GitNewBranch"

On branch GitNewBranch

nothing to commit, working tree clean

parva@CharviSankar MINGW64 ~/GitDemo (GitNewBranch)

$ git status

On branch GitNewBranch

nothing to commit, working tree clean

parva@CharviSankar MINGW64 ~/GitDemo (GitNewBranch)

$ git checkout master

Switched to branch 'master'

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

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

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git diff master..GitNewBranch

**diff --git a/testfile.txt b/testfile.txt**

**new file mode 100644**

**index 0000000..f401ec8**

**--- /dev/null**

**+++ b/testfile.txt**

@@ -0,0 +1 @@

+This is a test file in GitNewBranch.

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git difftool master GitNewBranch

This message is displayed because 'diff.tool' is not configured.

See 'git difftool --tool-help' or 'git help config' for more details.

'git difftool' will now attempt to use one of the following tools:

opendiff kdiff3 tkdiff xxdiff meld kompare gvimdiff diffuse diffmerge ecmerge p4merge araxis bc codecompare smerge emerge vimdiff nvimdiff

Viewing (1/1): 'testfile.txt'

Launch 'vimdiff' [Y/n]?

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ ^C

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ ^C

parva@CharviSankar MINGW64 ~/GitDemo (master)

$ git merge GitNewBranch

Updating 114bb35..2a36f43

Fast-forward

testfile.txt | 1 +

1 file changed, 1 insertion(+)

create mode 100644 testfile.txt

This message is displayed because 'diff.tool' is not configured.

See 'git difftool --tool-help' or 'git help config' for more details.

'git difftool' will now attempt to use one of the following tools:

opendiff kdiff3 tkdiff xxdiff meld kompare gvimdiff diffuse diffmerge ecmerge p4merge araxis bc codecompare smerge emerge vimdiff nvimdiff

Viewing (1/1): 'testfile.txt'

Launch 'vimdiff' [Y/n]? Updating 114bb35..2a36f43

Fast-forward

testfile.txt | 1 +

1 file changed, 1 insertion(+)

create mode 100644 testfile.txt

\* 2a36f43 (**HEAD** -> **master**, **GitNewBranch**) Added testfile.txt in GitNewBranch

\* 114bb35 Add .gitignore to ignore .log files and log folder

\* d532470 (**origin/master**) Added welcome.txt file

Deleted branch GitNewBranch (was 2a36f43).

On branch master

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

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

nothing to commit, working tree clean

**OUTPUT:**

