Experiment 3

Khozaima
T12-34

Aim: To Perform various GIT operations on local and Remote repositories

Theory:
The command mkdir git creates a new directory (folder) named "git" in the current working directory. This command is used to make a new directory in a Unix-like operating system.
The command cd git is used to change the current working directory to the directory named "git." After executing this command, any subsequent commands or file operations will occur within the "git" directory. "cd" stands for "change
directory."
The git config --global user.name and git config --global user.email commands are used to set your global Git username and email address, respectively. They are part of the configuration settings in Git and are associated with the commits you make.
If you want to check your configuration settings, you can use the git config --list command to list all the settings Git can find at that point

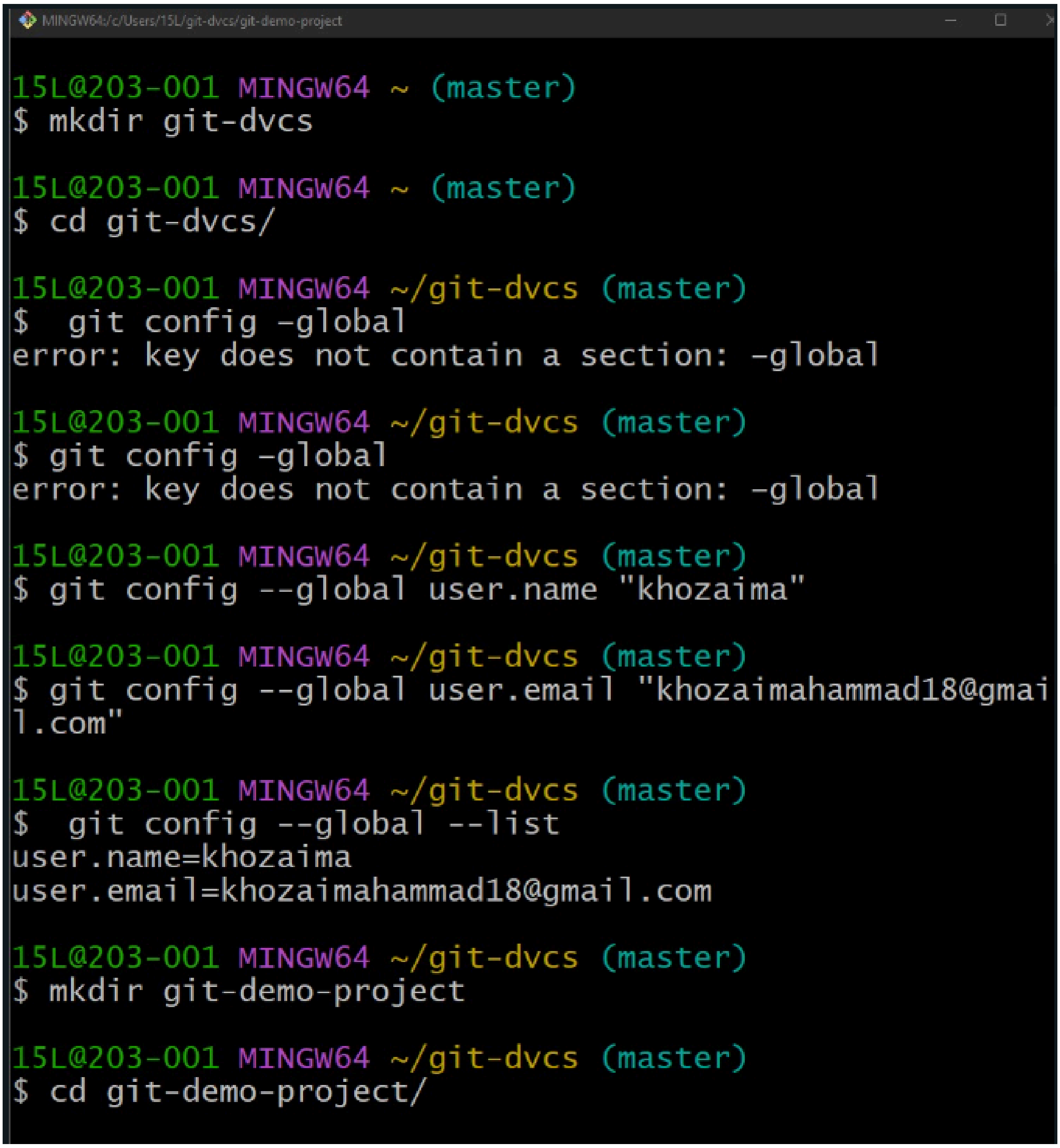
git commit -am "commit message"` stages and commits all changes in tracked files with a commit message in a single command.
The command nano index.html opens the Nano text editor for the file named "index.html." Nano is a simple command-line text editor that allows you to view and edit files directly in the terminal.

The command touch teststatus creates an empty file named "teststatus" in the current directory. The touch command is commonly used to update the timestamps of a file or create an empty file if it doesn't exist.
git checkout -- teststatus: Discards changes to the file "teststatus" in the working directory.This reverts the file to the state it has in the last commit.
The git add command is used to stage changes in the working directory for the next commit in Git. It prepares modifications, additions, or deletions to be included in the upcoming commit.

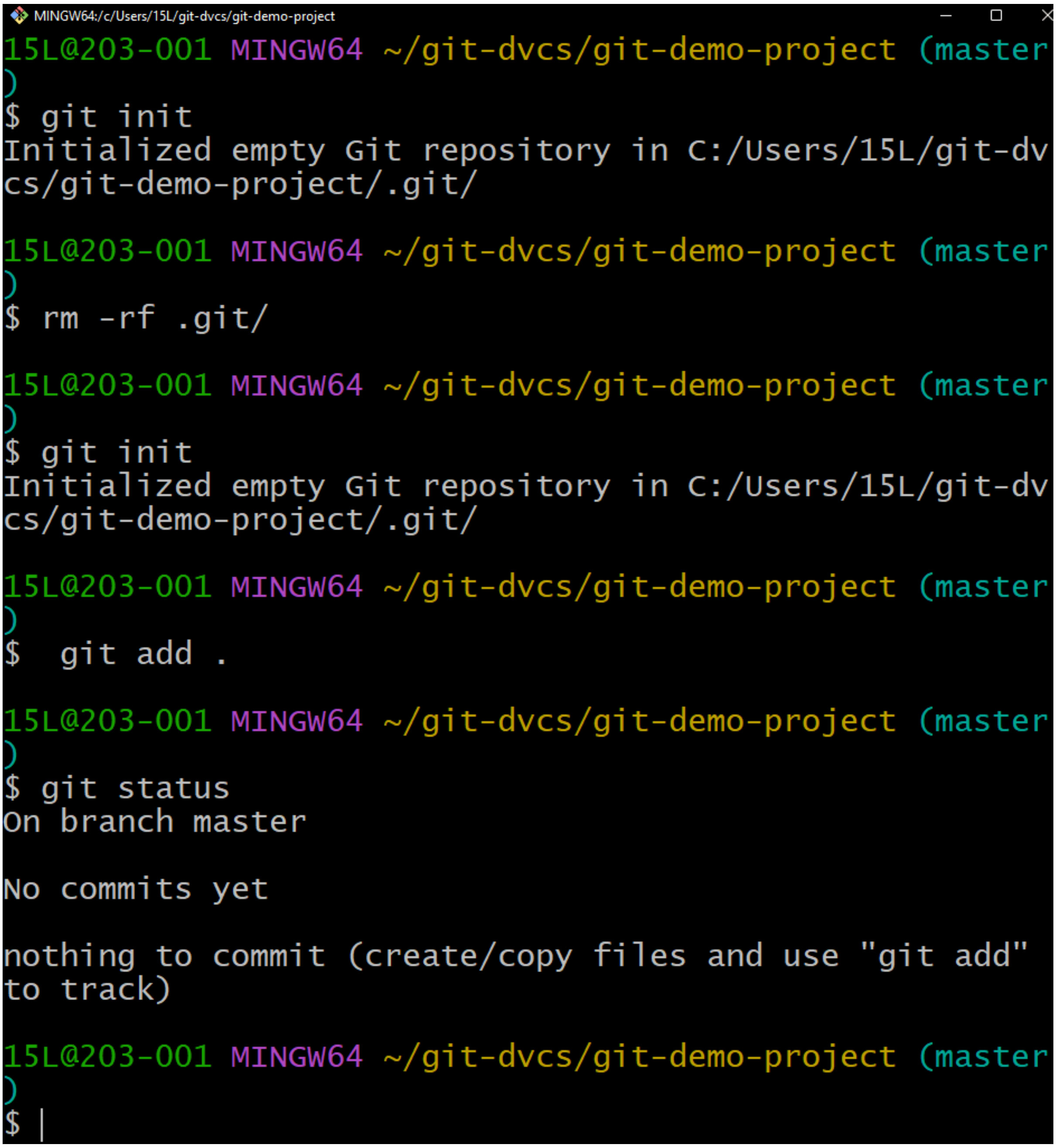
The git log command is used to display the commit history of a Git repository. It shows a chronological list of commits, including commit hashes, author information, timestamps, and commit messages.
The command git log --oneline displays a simplified and concise one-line representation of the commit history in a Git repository, showing only the commit SHA-1 hash and the commit message.
The git clone command is used to create a copy of a Git repository. When you run this command, it duplicates the entire repository, including its files, commit history, and branches, and downloads it to your local machine. This is often the initial step when you want to work with a project hosted on a remote Git repository.

The git pull command is used to fetch and integrate changes from a remote repository into the current branch of your local repository. It combines two actions: it fetches the changes from the remote repository, and then it automatically merges those changes into your local branch. This is a convenient way to update your local repository with the latest changes from the remote repository.
The git push command is used to upload or push the local changes in your Git repository to a remote repository. It updates the remote repository with the latest changes made in your local branch, making them accessible to others who share the same remote repository.
The git fetch command is used to retrieve changes from a remote repository. It fetches any new branches or changes made in the remote repository since your last interaction. However, it does not automatically merge these changes into your local branches. After using git fetch, you can inspect the changes and decide whether to integrate them using git merge or git rebase.

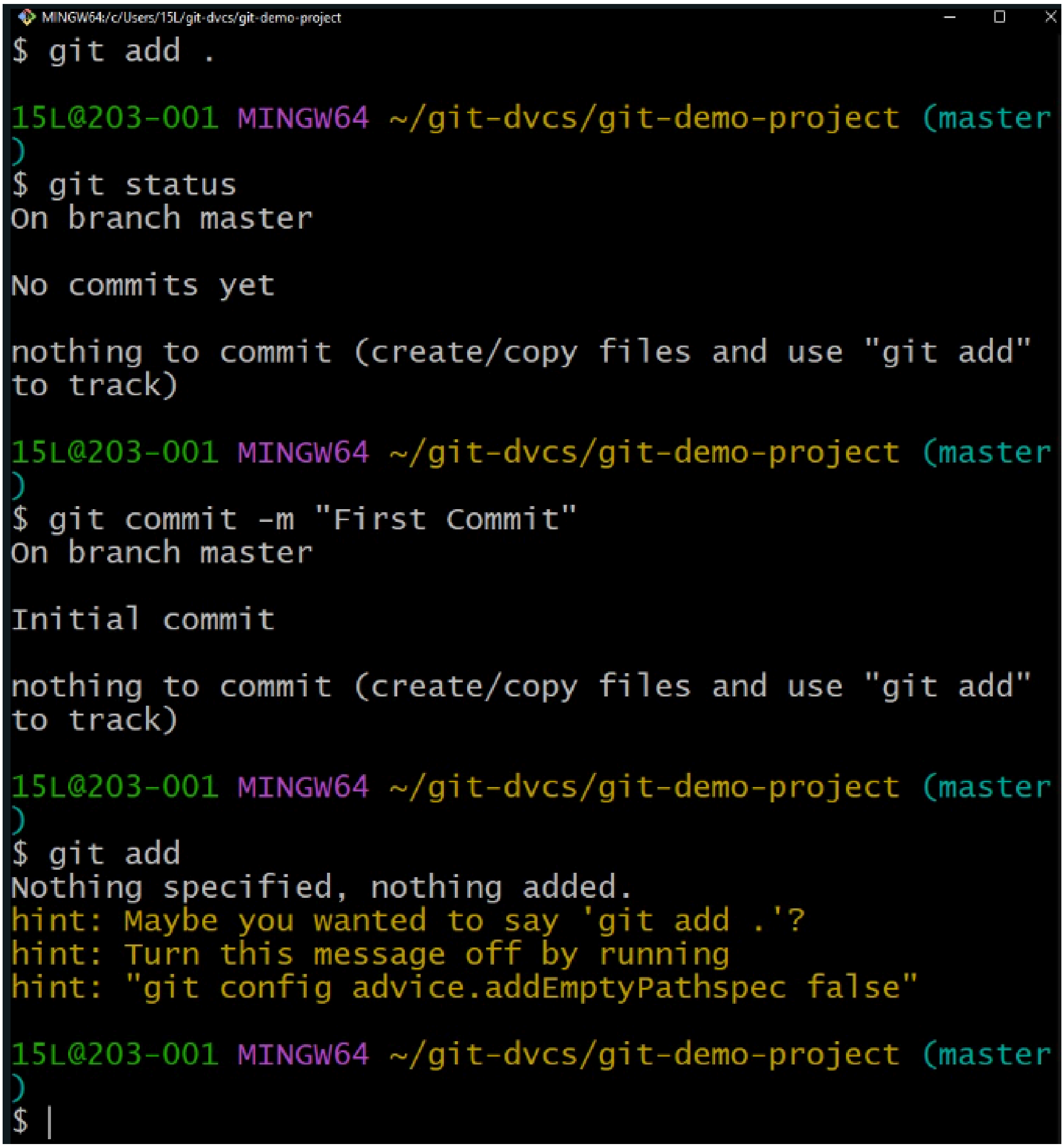
SS:



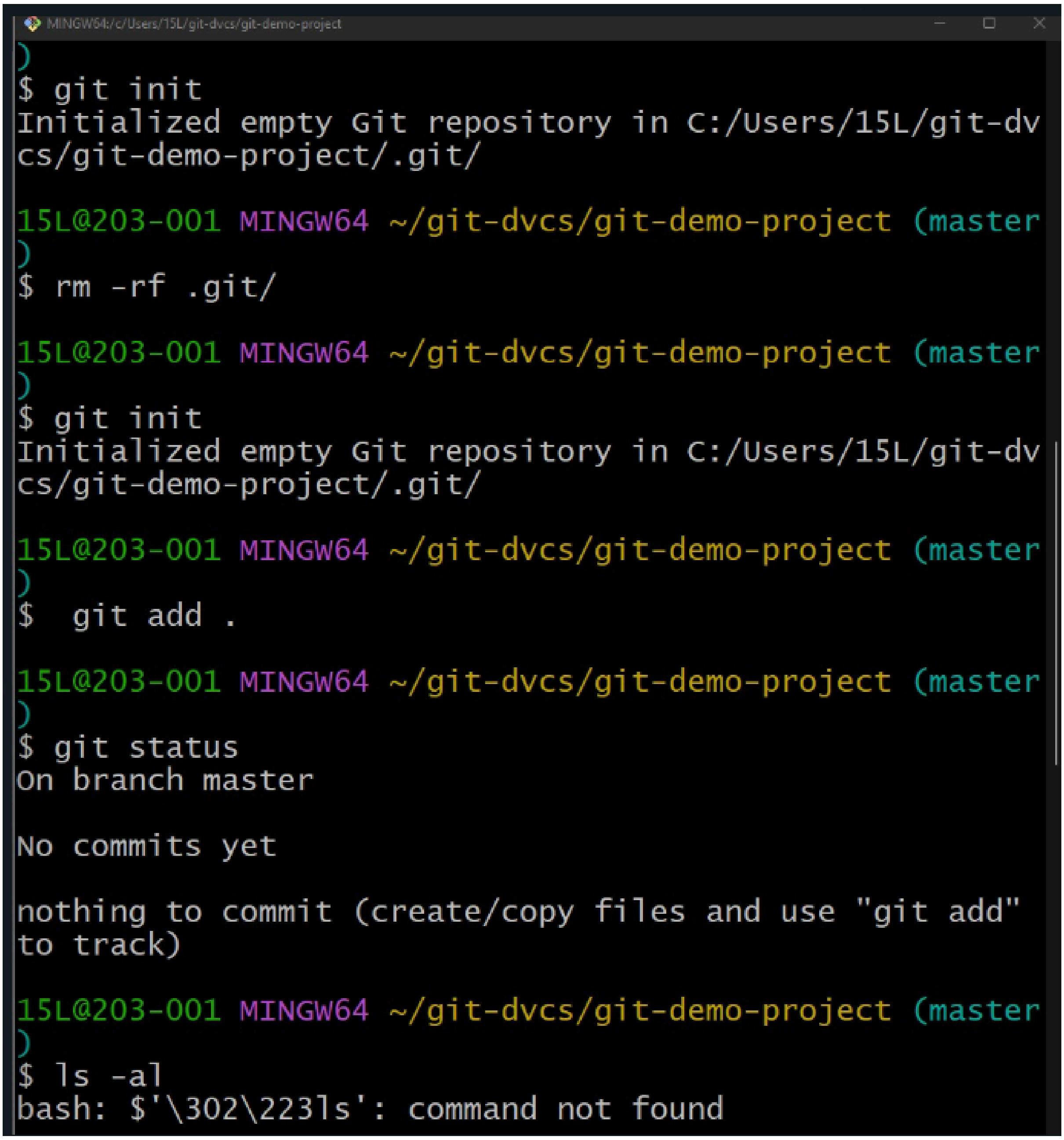
```
15L@203-001 MINGW64 ~ (master)
$ mkdir git-dvcs

15L@203-001 MINGW64 ~ (master)
$ cd git-dvcs/

15L@203-001 MINGW64 ~/git-dvcs (master)
$  git config -global
error: key does not contain a section: -global

15L@203-001 MINGW64 ~/git-dvcs (master)
$ git config -global
error: key does not contain a section: -global

15L@203-001 MINGW64 ~/git-dvcs (master)
$ git config --global user.name "khozaima"

15L@203-001 MINGW64 ~/git-dvcs (master)
$ git config --global user.email "khozaimahammad18@gmai
l.com"

15L@203-001 MINGW64 ~/git-dvcs (master)
$  git config --global --list
user.name=khozaima
user.email=khozaimahammad18@gmail.com

15L@203-001 MINGW64 ~/git-dvcs (master)
$ mkdir git-demo-project

15L@203-001 MINGW64 ~/git-dvcs (master)
$ cd git-demo-project/
```



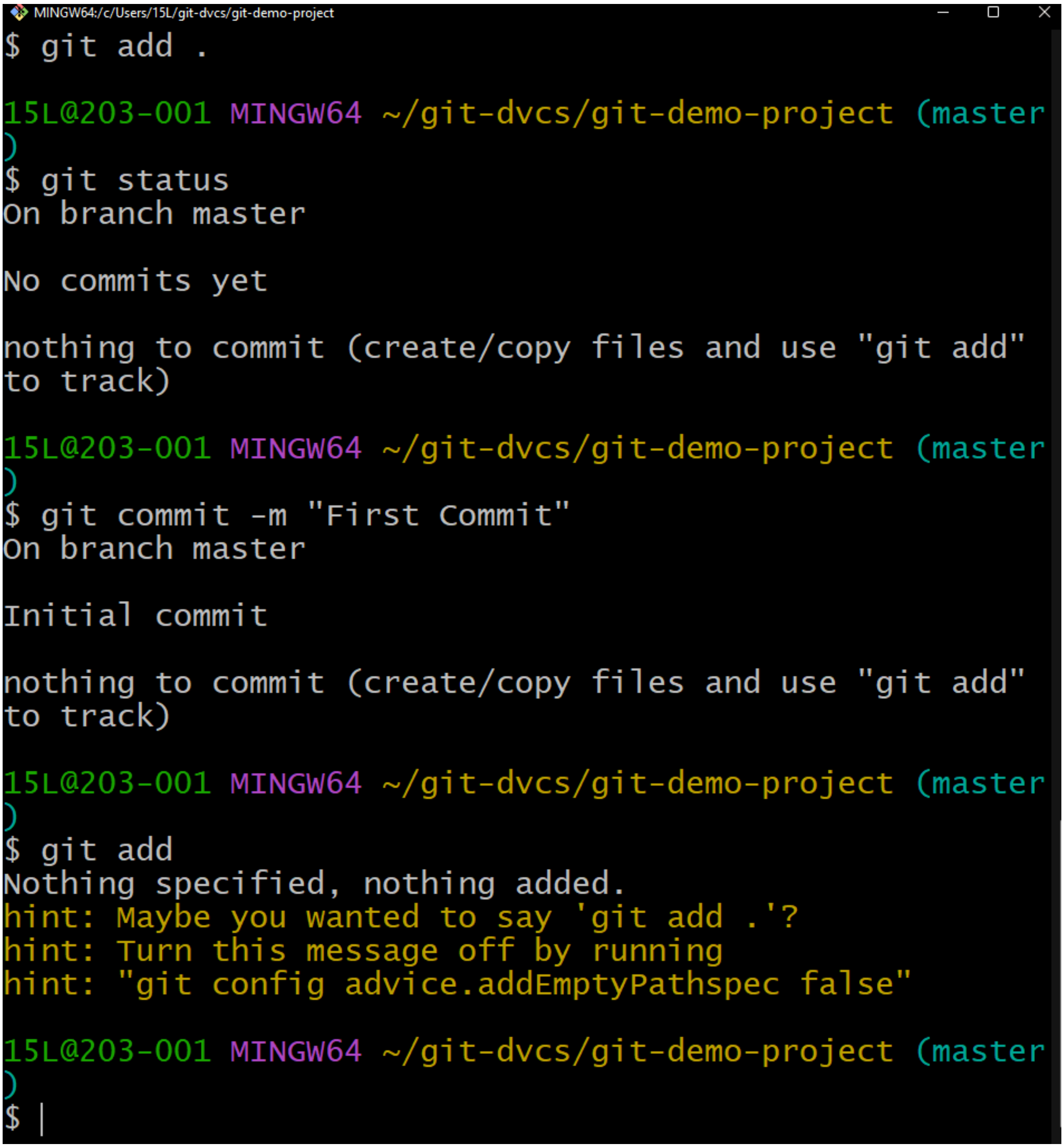
```
15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ git init
Initialized empty Git repository in C:/Users/15L/git-dv
cs/git-demo-project/.git/

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ rm -rf .git/

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ git init
Initialized empty Git repository in C:/Users/15L/git-dv
cs/git-demo-project/.git/

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$  git add .

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add"
to track)

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ 
```



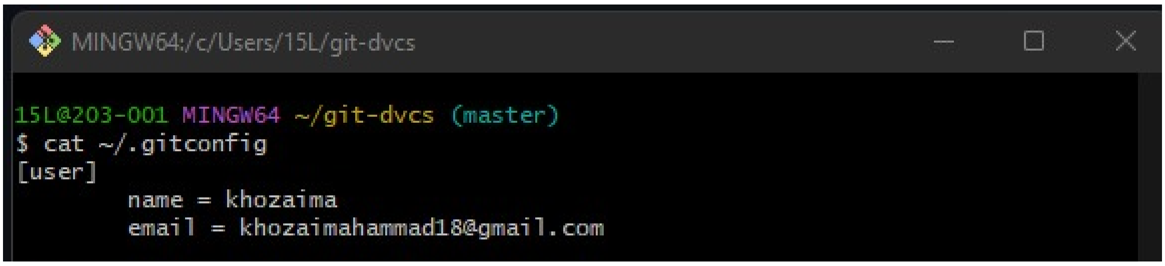
```
$ git add .

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add"
to track)

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ git commit -m "First Commit"
On branch master

Initial commit

nothing to commit (create/copy files and use "git add"
to track)

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ git add
Nothing specified, nothing added.
hint: Maybe you wanted to say 'git add .'?
hint: Turn this message off by running
hint: "git config advice.addEmptyPathspec false"

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ |
```



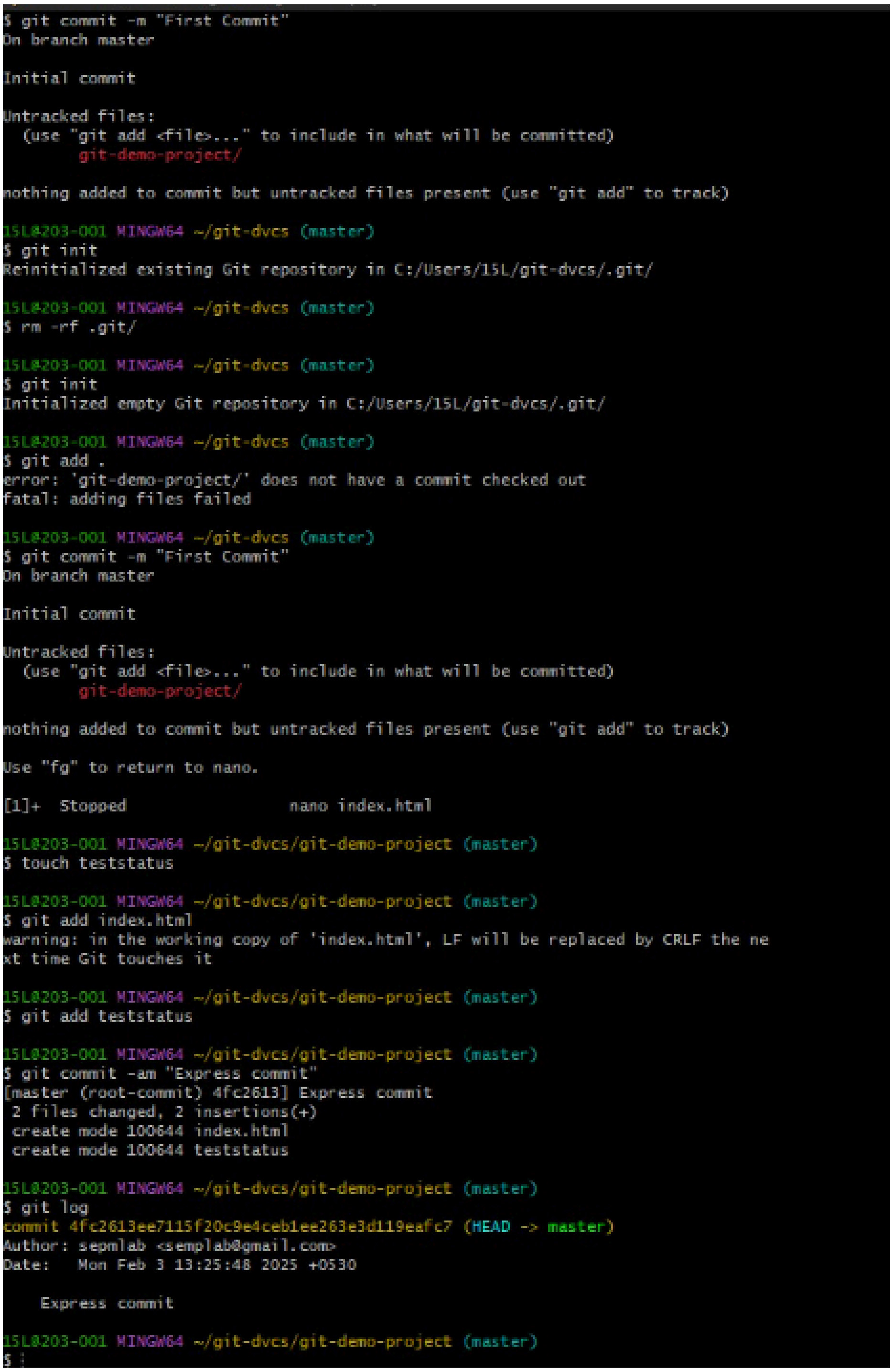
```
)
$ git init
Initialized empty Git repository in C:/Users/15L/git-dv
cs/git-demo-project/.git/

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ rm -rf .git/

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ git init
Initialized empty Git repository in C:/Users/15L/git-dv
cs/git-demo-project/.git/

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$  git add .

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add"
to track)

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ ls -al
bash: $'\302\223ls': command not found
```



```
$ git add .

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add"
to track)

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ git commit -m "First Commit"
On branch master

Initial commit

nothing to commit (create/copy files and use "git add"
to track)

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ git add
Nothing specified, nothing added.
hint: Maybe you wanted to say 'git add .'?
hint: Turn this message off by running
hint: "git config advice.addEmptyPathspec false"

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master
)
$ |
```



```
15L@203-001 MINGW64 ~/git-dvcs (master)
$ cat ~/.gitconfig
[user]
        name = khozaima
        email = khozaimahammad18@gmail.com
```



```
$ git commit -m "First Commit"
On branch master

Initial commit

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        git-demo-project/

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

15L@203-001 MINGW64 ~/git-dvcs (master)
$ git init
Reinitialized existing Git repository in C:/Users/15L/git-dvcs/.git/

15L@203-001 MINGW64 ~/git-dvcs (master)
$ rm -rf .git/

15L@203-001 MINGW64 ~/git-dvcs (master)
$ git init
Initialized empty Git repository in C:/Users/15L/git-dvcs/.git/

15L@203-001 MINGW64 ~/git-dvcs (master)
$ git add .
error: 'git-demo-project/' does not have a commit checked out
fatal: adding files failed

15L@203-001 MINGW64 ~/git-dvcs (master)
$ git commit -m "First Commit"
On branch master

Initial commit

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        git-demo-project/

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

Use "fg" to return to nano.

[1]+  Stopped                 nano index.html

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master)
$ touch teststatus

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git add index.html
warning: in the working copy of 'index.html', LF will be replaced by CRLF the ne
xt time Git touches it

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git add teststatus

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git commit -am "Express commit"
[master (root-commit) 4fc2613] Express commit
 2 files changed, 2 insertions(+)
 create mode 100644 index.html
 create mode 100644 teststatus

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git log
commit 4fc2613ee7115f20c9e4ceb1ee263e3d119eafc7 (HEAD -> master)
Author: sepmlab <semplab@gmail.com>
Date:   Mon Feb 3 13:25:48 2025 +0530

    Express commit

15L@203-001 MINGW64 ~/git-dvcs/git-demo-project (master)
$
```

Conclusion: We understand the git commands