WEEK 8 GIT EXERCISE 1:

1. Check if Git is installed:

```
git –version
```

2. Set up Git global user config:

3. Confirm configuration:

```
git config –list
```

1. Create project folder and initialize Git:

```
mkdir GitDemo
cd GitDemo
git init
```

2. Verify Git repo is initialized:

```
ls -a
```

3. Create and edit a file:

```
notepad++ welcome.txt
```

4. Check the file exists:

```
ls
```

5. View the content:

```
cat welcome.txt
```

6. Check Git status:

```
git status
```

7. Stage the file:

```
git add welcome.txt
```

8. Commit with multiline message (opens Notepad++):

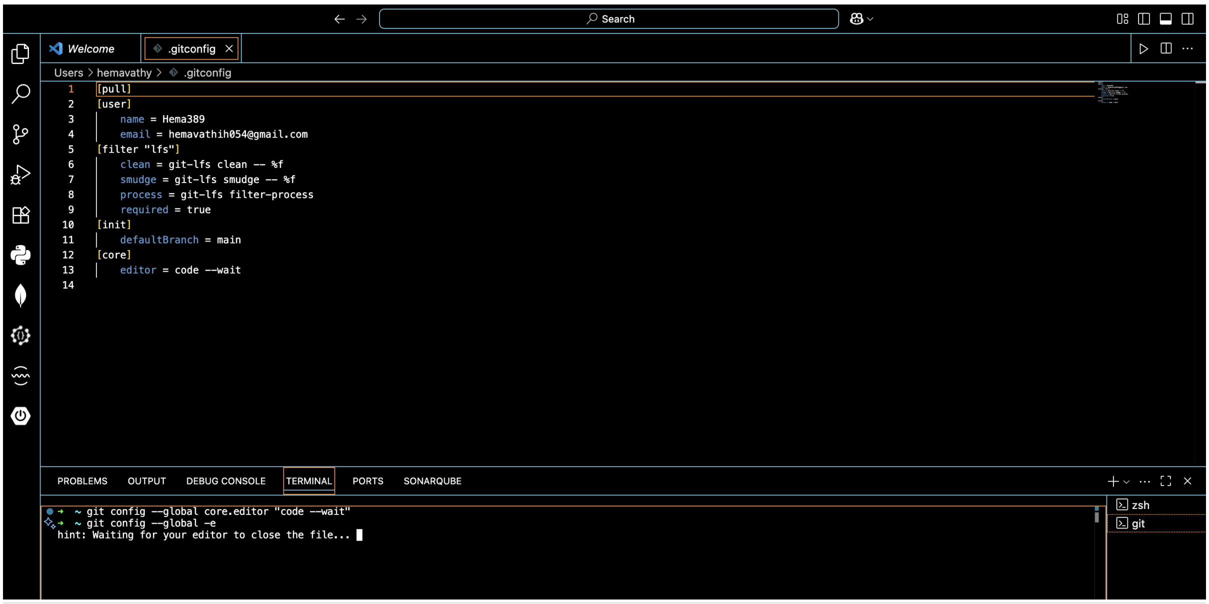
- ➢ git commit

9. Confirm file is committed:

- ➢ git status

Connect to Remote Repository:

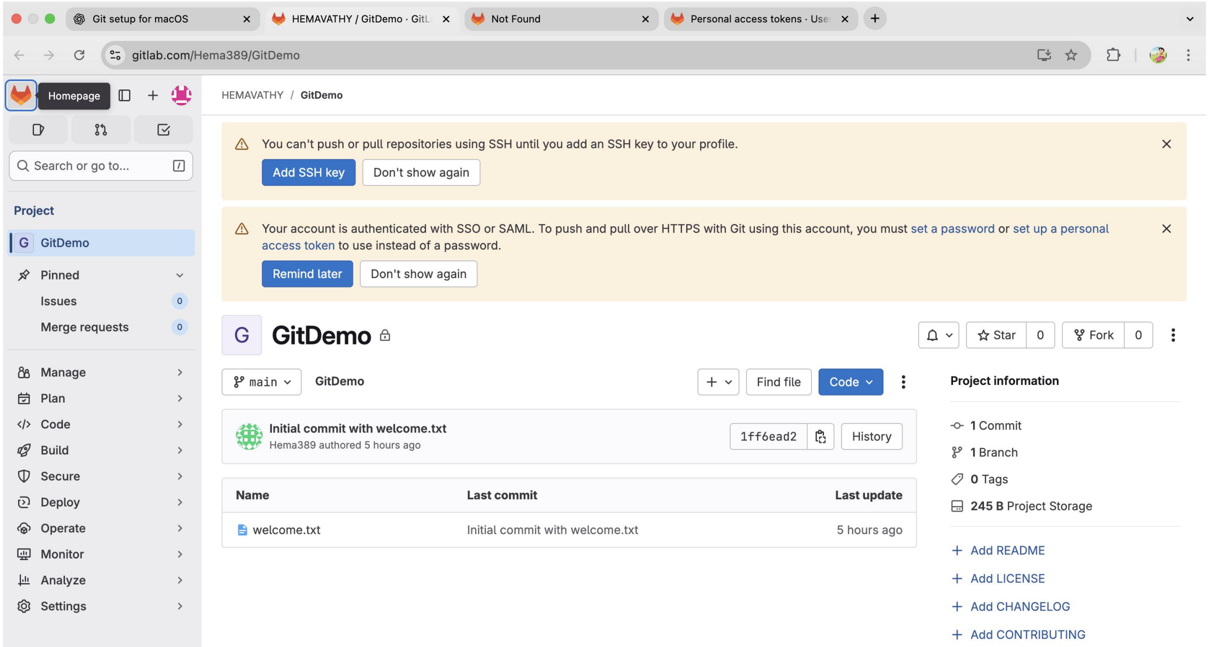
Output:



Users > hemavathy > .gitconfig

```
[pull]
[user]
	name = Hema389
	email = hemavathih054@gmail.com
[filter "lfs"]
	clean = git-lfs clean -- %f
	smudge = git-lfs smudge -- %f
	process = git-lfs filter-process
	required = true
[init]
	defaultBranch = main
[core]
	editor = code --wait
```

```
→  ~ git config --global core.editor "code --wait"
→  ~ git config --global -e
hint: Waiting for your editor to close the file...
```



gitlab.com/Hema389/GitDemo

HEMAVATHY / GitDemo

You can't push or pull repositories using SSH until you add an SSH key to your profile.

Your account is authenticated with SSO or SAML. To push and pull over HTTPS with Git using this account, you must set a password or set up a personal access token to use instead of a password.

GitDemo

Initial commit with welcome.txt
Hema389 authored 5 hours ago
1ff6ead2

Name	Last commit	Last update
welcome.txt	Initial commit with welcome.txt	5 hours ago

Project information

1 Commit
1 Branch
0 Tags
245 B Project Storage

EXERCISE 2:

```
cd ~/GitDemo
```

Create test files and folder to ignore

- echo "error log" > error.log

- mkdir log
- echo "log content" > log/system.log

Create a .gitignore file

- touch .gitignore
- echo "*.log" >> .gitignore
- echo "log/" >> .gitignore
- *.log
- log/

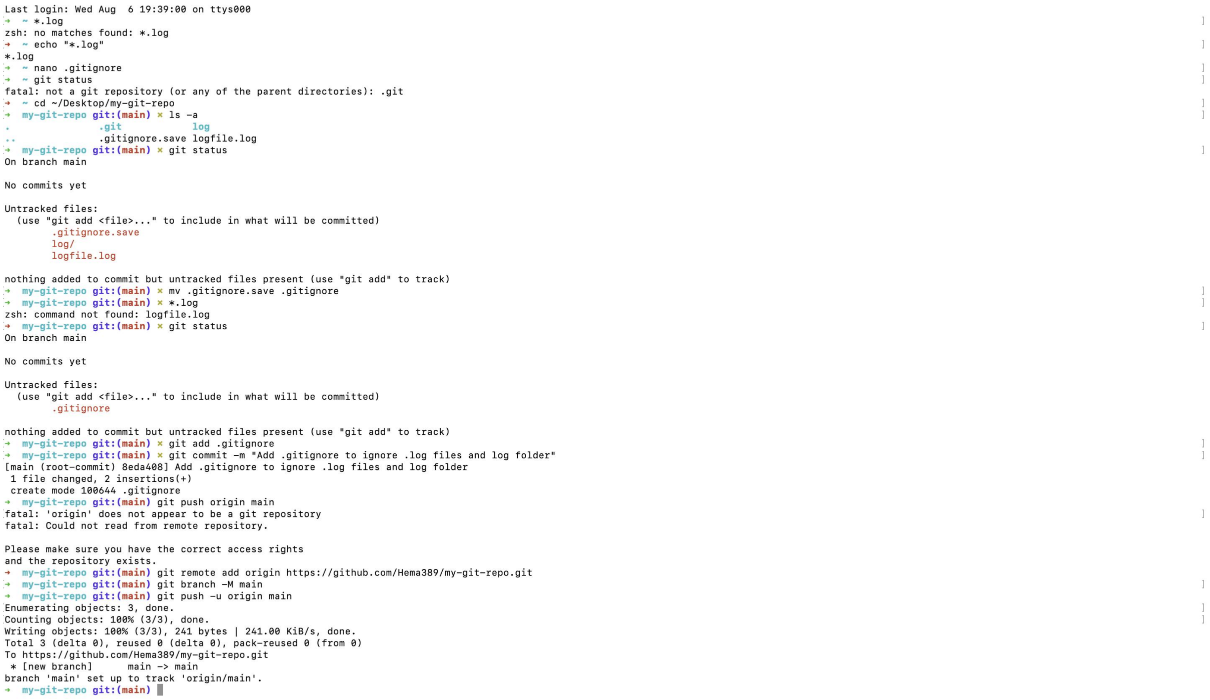
Check Git status

- git status

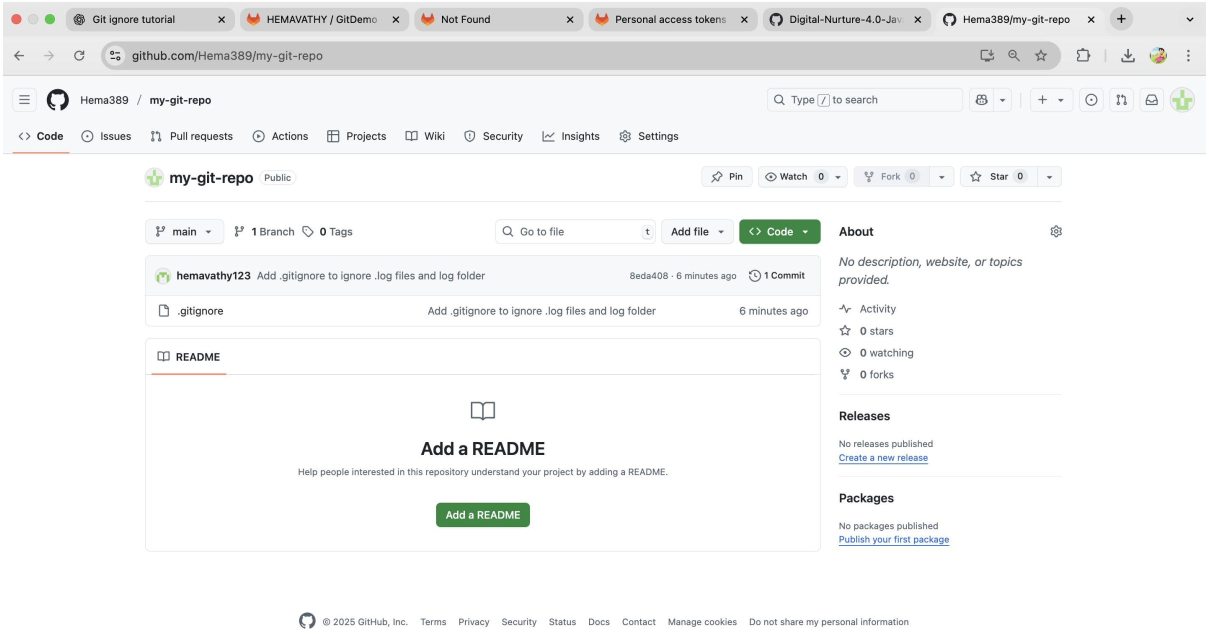
Add and commit only .gitignore

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

Output:



```
Last login: Wed Aug  6 19:39:00 on ttys000
➜  ~ *.log
zsh: no matches found: *.log
➜  ~ echo "*.log"
*.log
➜  ~ nano .gitignore
➜  ~ git status
fatal: not a git repository (or any of the parent directories): .git
➜  ~ cd ~/Desktop/my-git-repo
➜  my-git-repo git:(main) ✗ ls -a
.               .git            log
..              .gitignore.save logfile.log
➜  my-git-repo git:(main) ✗ git status
On branch main

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore.save
        log/
        logfile.log

nothing added to commit but untracked files present (use "git add" to track)
➜  my-git-repo git:(main) ✗ mv .gitignore.save .gitignore
➜  my-git-repo git:(main) ✗ *.log
zsh: command not found: logfile.log
➜  my-git-repo git:(main) ✗ git status
On branch main

No commits yet

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

nothing added to commit but untracked files present (use "git add" to track)
➜  my-git-repo git:(main) ✗ git add .gitignore
➜  my-git-repo git:(main) ✗ git commit -m "Add .gitignore to ignore .log files and log folder"
[main (root-commit) 8eda408] Add .gitignore to ignore .log files and log folder
 1 file changed, 2 insertions(+)
 create mode 100644 .gitignore
➜  my-git-repo git:(main) git push origin main
fatal: 'origin' does not appear to be a git repository
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.
➜  my-git-repo git:(main) git remote add origin https://github.com/Hema389/my-git-repo.git
➜  my-git-repo git:(main) git branch -M main
➜  my-git-repo git:(main) git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 241 bytes | 241.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Hema389/my-git-repo.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
➜  my-git-repo git:(main) 
```



my-git-repo Public

1 Branch 0 Tags

hemavathy123 Add .gitignore to ignore .log files and log folder 8eda408 · 6 minutes ago 1 Commit

.gitignore Add .gitignore to ignore .log files and log folder 6 minutes ago

README

Add a README

Help people interested in this repository understand your project by adding a README.

About

No description, website, or topics provided.

Activity

0 stars

0 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

© 2025 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

EXERCISE 3:

Create a new branch called GitNewBranch

git checkout -b GitNewBranch

List all branches

git branch

Add a new file in this branch

echo "This is a new feature file" > feature.txt

Stage and commit the file

git add feature.txt

git commit -m "Add feature.txt in GitNewBranch"

Check branch status

git status

MERGING

Switch back to the master branch

git checkout master

Check differences between master and GitNewBranch

git diff master..GitNewBranch

Merge GitNewBranch into master

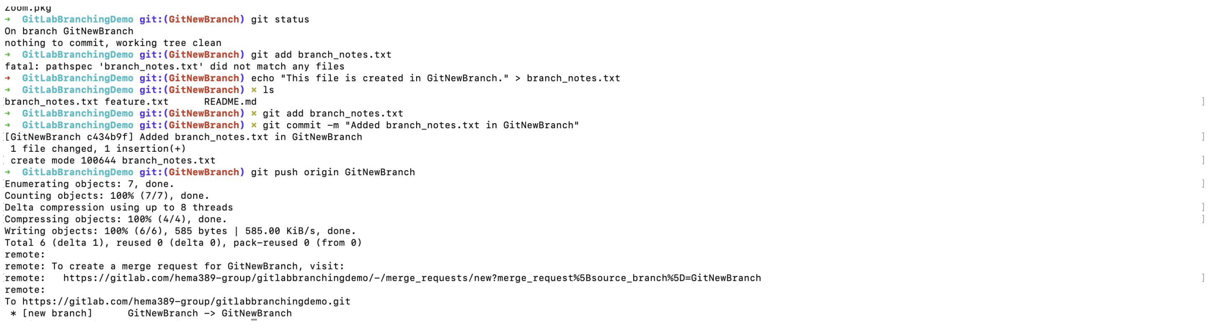
git merge GitNewBranch

Delete the merged branch

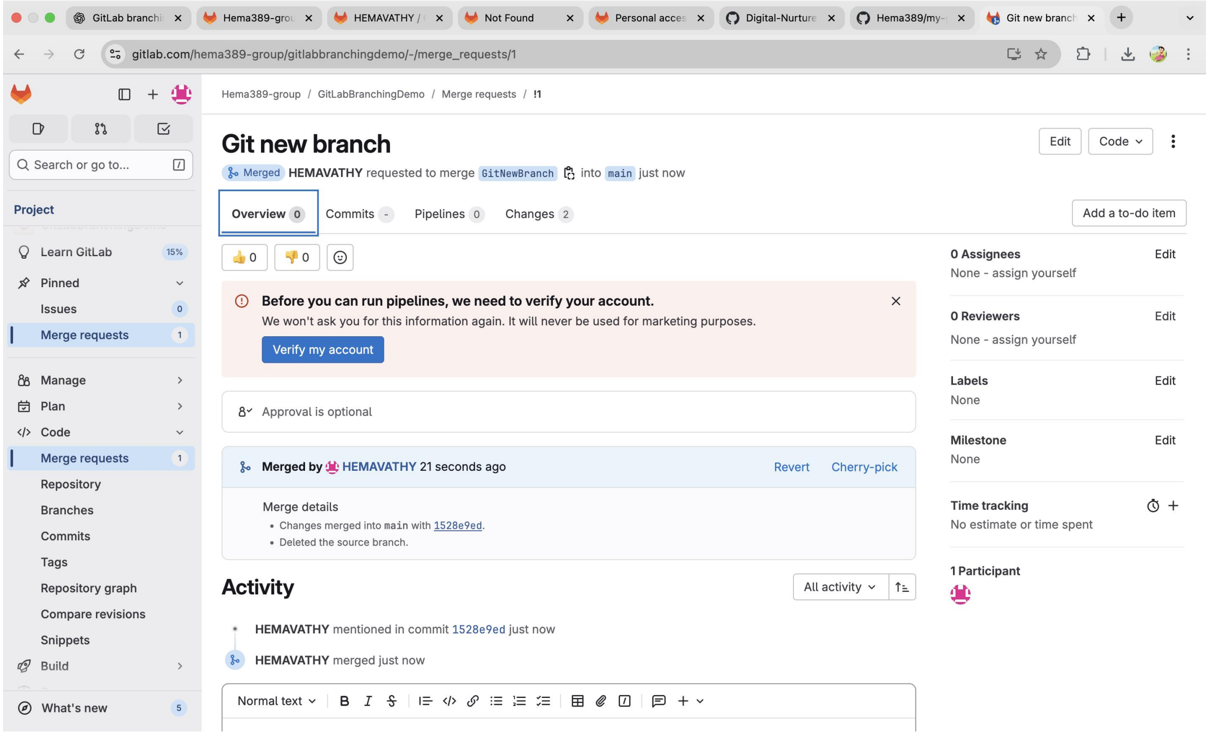
git branch -d GitNewBranch

git status

Output:



```
zoom.pkg
➜  GitLabBranchingDemo git:(GitNewBranch) git status
On branch GitNewBranch
nothing to commit, working tree clean
➜  GitLabBranchingDemo git:(GitNewBranch) git add branch_notes.txt
fatal: pathspec 'branch_notes.txt' did not match any files
➜  GitLabBranchingDemo git:(GitNewBranch) echo "This file is created in GitNewBranch." > branch_notes.txt
➜  GitLabBranchingDemo git:(GitNewBranch) ✗ ls
branch_notes.txt feature.txt      README.md
➜  GitLabBranchingDemo git:(GitNewBranch) ✗ git add branch_notes.txt
➜  GitLabBranchingDemo git:(GitNewBranch) ✗ git commit -m "Added branch_notes.txt in GitNewBranch"
[GitNewBranch c434b9f] Added branch_notes.txt in GitNewBranch
 1 file changed, 1 insertion(+)
 create mode 100644 branch_notes.txt
➜  GitLabBranchingDemo git:(GitNewBranch) git push origin GitNewBranch
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 585 bytes | 585.00 KiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: To create a merge request for GitNewBranch, visit:
remote:   https://gitlab.com/hema389-group/gitlabbranchingdemo/-/merge_requests/new?merge_request%5Bsource_branch%5D=GitNewBranch
remote:
To https://gitlab.com/hema389-group/gitlabbranchingdemo.git
 * [new branch]      GitNewBranch -> GitNewBranch
```



Git new branch

Merged HEMAVATHY requested to merge GitNewBranch into main just now

Before you can run pipelines, we need to verify your account.
We won't ask you for this information again. It will never be used for marketing purposes.

Approval is optional

Merged by HEMAVATHY 21 seconds ago

Merge details
- Changes merged into main with 1528e9ed.
- Deleted the source branch.

Activity

HEMAVATHY mentioned in commit 1528e9ed just now

HEMAVATHY merged just now

0 Assignees Edit
None - assign yourself

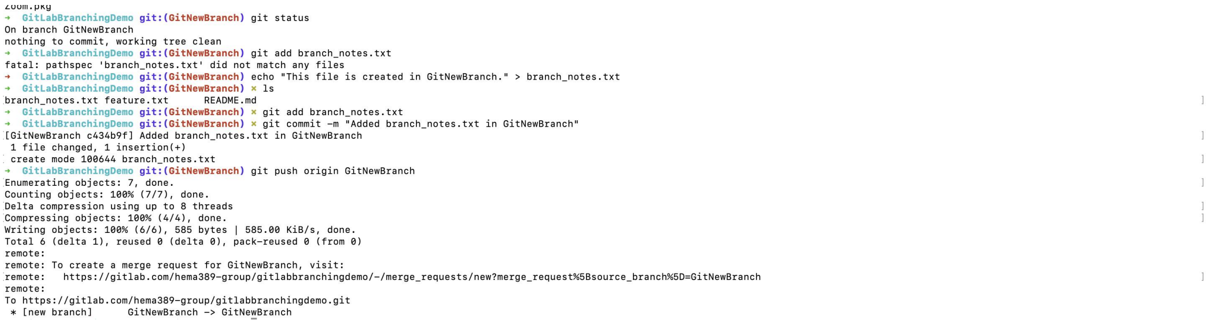
0 Reviewers Edit
None - assign yourself

Labels Edit
None

Milestone Edit
None

Time tracking
No estimate or time spent

1 Participant



```
zoom.pkg
➜  GitLabBranchingDemo git:(GitNewBranch) git status
On branch GitNewBranch
nothing to commit, working tree clean
➜  GitLabBranchingDemo git:(GitNewBranch) git add branch_notes.txt
fatal: pathspec 'branch_notes.txt' did not match any files
➜  GitLabBranchingDemo git:(GitNewBranch) echo "This file is created in GitNewBranch." > branch_notes.txt
➜  GitLabBranchingDemo git:(GitNewBranch) ✗ ls
branch_notes.txt feature.txt      README.md
➜  GitLabBranchingDemo git:(GitNewBranch) ✗ git add branch_notes.txt
➜  GitLabBranchingDemo git:(GitNewBranch) ✗ git commit -m "Added branch_notes.txt in GitNewBranch"
[GitNewBranch c434b9f] Added branch_notes.txt in GitNewBranch
 1 file changed, 1 insertion(+)
 create mode 100644 branch_notes.txt
➜  GitLabBranchingDemo git:(GitNewBranch) git push origin GitNewBranch
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 585 bytes | 585.00 KiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: To create a merge request for GitNewBranch, visit:
remote:   https://gitlab.com/hema389-group/gitlabbranchingdemo/-/merge_requests/new?merge_request%5Bsource_branch%5D=GitNewBranch
remote:
To https://gitlab.com/hema389-group/gitlabbranchingdemo.git
 * [new branch]      GitNewBranch -> GitNewBranch
```

EXERCISE 4:

Create a new branch GitWork and switch to it

git checkout -b GitWork

Add a file hello.xml and update its content

echo "<message>Hello from GitWork branch</message>" > hello.xml

Stage and commit the file

git add hello.xml

```
git commit -m "Added hello.xml from GitWork branch"
```

Switch back to master

```
git checkout master
```

Add a different version of hello.xml in master

```
echo "<message>Hello from master branch</message>" > hello.xml
```

Commit changes in master

```
git add hello.xml
```

```
git commit -m "Added hello.xml from master branch"
```

View logs in graph format

```
git log --oneline --graph --decorate –all
```

Check differences using CLI tool

```
git diff master GitWork
```

Merge GitWork into master

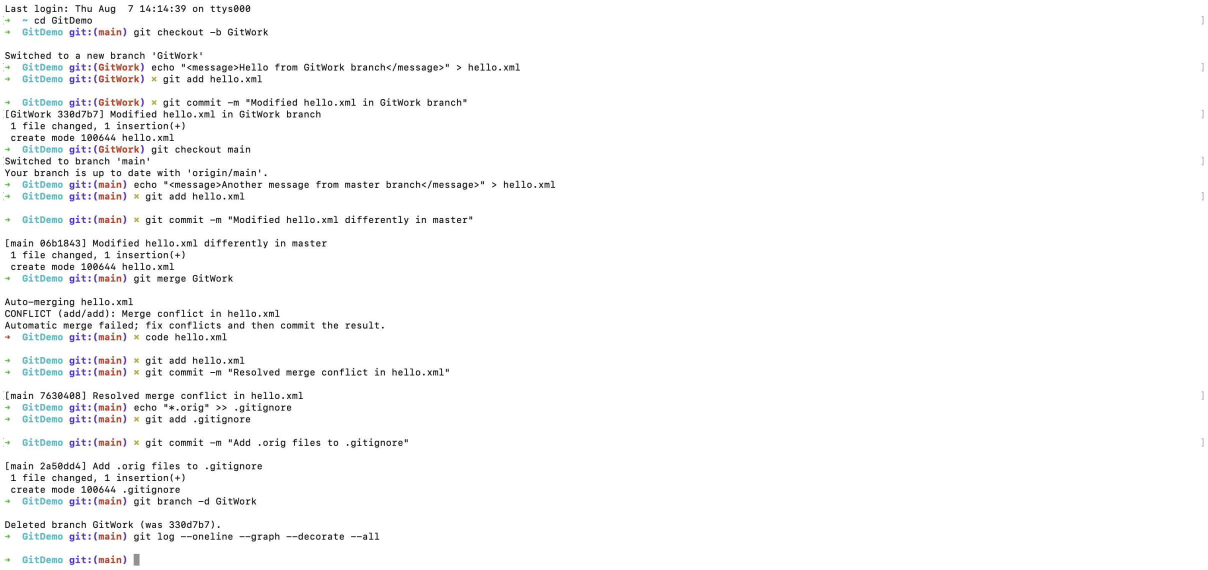
```
git merge GitWork
```

Commit after resolving conflict

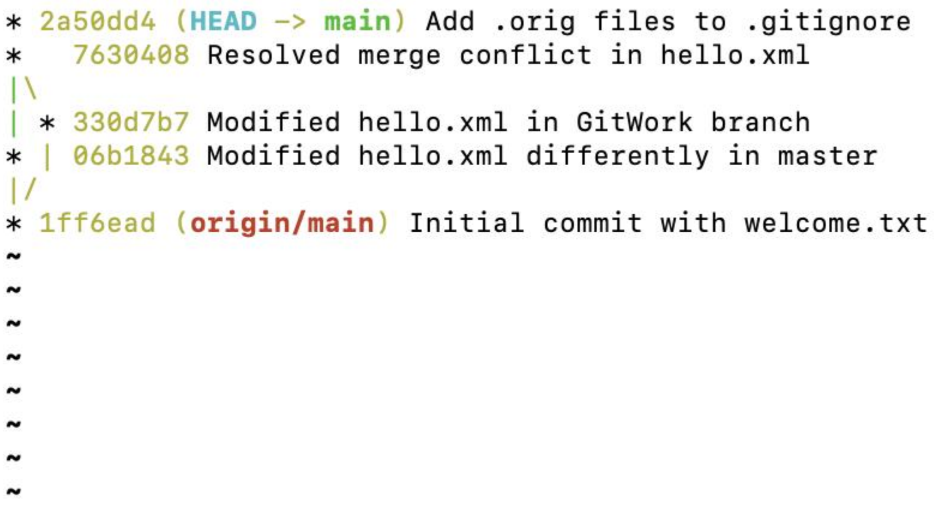
```
git add hello.xml
```

```
git commit -m "Resolved merge conflict in hello.xml"
```

Output:



```
Last login: Thu Aug  7 14:14:39 on ttys000
➜  ~ cd GitDemo
➜  GitDemo git:(main) git checkout -b GitWork

Switched to a new branch 'GitWork'
➜  GitDemo git:(GitWork) echo "<message>Hello from GitWork branch</message>" > hello.xml
➜  GitDemo git:(GitWork) ✗ git add hello.xml

➜  GitDemo git:(GitWork) ✗ git commit -m "Modified hello.xml in GitWork branch"
[GitWork 330d7b7] Modified hello.xml in GitWork branch
 1 file changed, 1 insertion(+)
 create mode 100644 hello.xml
➜  GitDemo git:(GitWork) git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
➜  GitDemo git:(main) echo "<message>Another message from master branch</message>" > hello.xml
➜  GitDemo git:(main) ✗ git add hello.xml

➜  GitDemo git:(main) ✗ git commit -m "Modified hello.xml differently in master"

[main 06b1843] Modified hello.xml differently in master
 1 file changed, 1 insertion(+)
 create mode 100644 hello.xml
➜  GitDemo git:(main) git merge GitWork

Auto-merging hello.xml
CONFLICT (add/add): Merge conflict in hello.xml
Automatic merge failed; fix conflicts and then commit the result.
➜  GitDemo git:(main) ✗ code hello.xml

➜  GitDemo git:(main) ✗ git add hello.xml
➜  GitDemo git:(main) ✗ git commit -m "Resolved merge conflict in hello.xml"

[main 7630408] Resolved merge conflict in hello.xml
➜  GitDemo git:(main) echo "*.orig" >> .gitignore
➜  GitDemo git:(main) ✗ git add .gitignore

➜  GitDemo git:(main) ✗ git commit -m "Add .orig files to .gitignore"

[main 2a50dd4] Add .orig files to .gitignore
 1 file changed, 1 insertion(+)
 create mode 100644 .gitignore
➜  GitDemo git:(main) git branch -d GitWork

Deleted branch GitWork (was 330d7b7).
➜  GitDemo git:(main) git log --oneline --graph --decorate --all

➜  GitDemo git:(main)
```



```
* 2a50dd4 (HEAD -> main) Add .orig files to .gitignore
*   7630408 Resolved merge conflict in hello.xml
|\
| * 330d7b7 Modified hello.xml in GitWork branch
* | 06b1843 Modified hello.xml differently in master
|/
* 1ff6ead (origin/main) Initial commit with welcome.txt
~
~
~
~
~
~
~
~
```

EXERCISE 5:

Verify if master is in clean state

git checkout master

git status

List all available branches

git branch -a

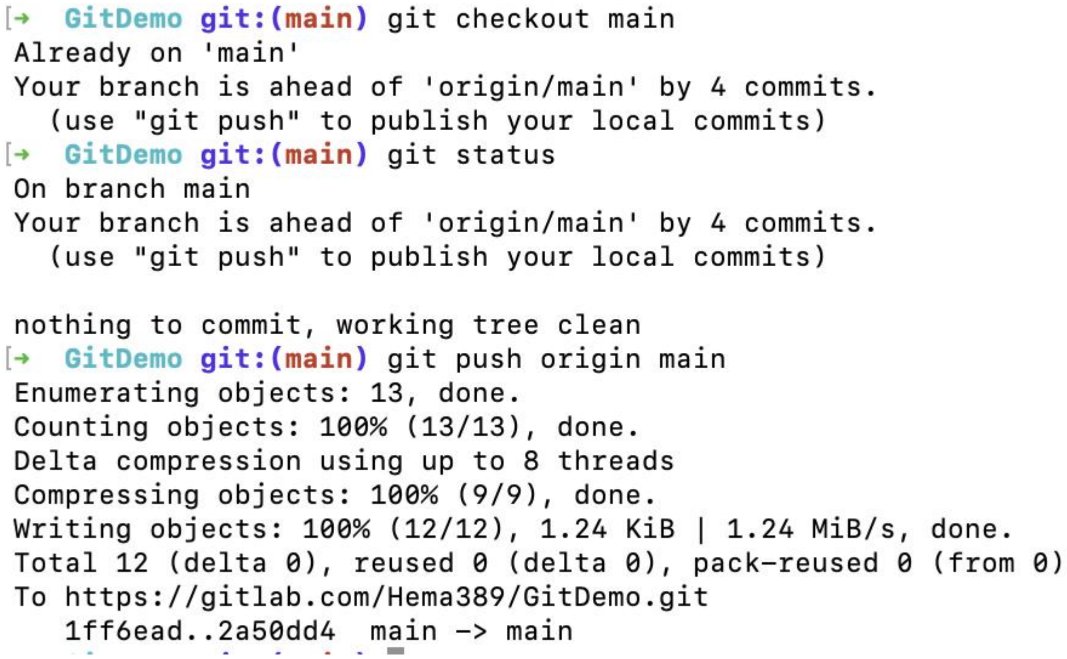
Pull latest changes from remote

git pull origin master

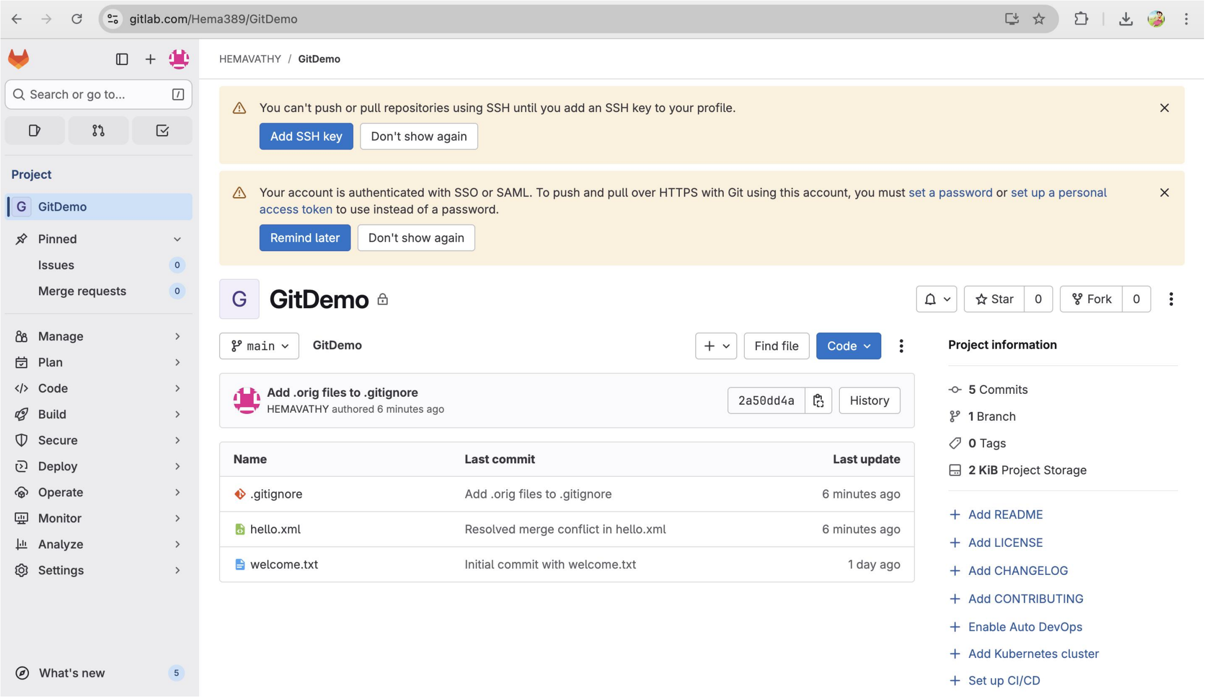
Push your local commits to remote

git push origin master

Output:



```
[→  GitDemo git:(main) git checkout main
 Already on 'main'
 Your branch is ahead of 'origin/main' by 4 commits.
   (use "git push" to publish your local commits)
[→  GitDemo git:(main) git status
 On branch main
 Your branch is ahead of 'origin/main' by 4 commits.
   (use "git push" to publish your local commits)

 nothing to commit, working tree clean
[→  GitDemo git:(main) git push origin main
 Enumerating objects: 13, done.
 Counting objects: 100% (13/13), done.
 Delta compression using up to 8 threads
 Compressing objects: 100% (9/9), done.
 Writing objects: 100% (12/12), 1.24 KiB | 1.24 MiB/s, done.
 Total 12 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
 To https://gitlab.com/Hema389/GitDemo.git
    1ff6ead..2a50dd4  main -> main
```



You can't push or pull repositories using SSH until you add an SSH key to your profile.

Add SSH key | Don't show again

Your account is authenticated with SSO or SAML. To push and pull over HTTPS with Git using this account, you must set a password or set up a personal access token to use instead of a password.

Remind later | Don't show again

GitDemo

Add .orig files to .gitignore
HEMAVATHY authored 6 minutes ago

2a50dd4a | History

Name	Last commit	Last update
.gitignore	Add .orig files to .gitignore	6 minutes ago
hello.xml	Resolved merge conflict in hello.xml	6 minutes ago
welcome.txt	Initial commit with welcome.txt	1 day ago

Project information

- 5 Commits
- 1 Branch
- 0 Tags
- 2 KiB Project Storage