

# Week 8 : HandsOn Exercise

1) Familiar with Git commands like git init, git status, git add, git commit, git push, and git pull

## a. Git Configuration and Setup

```
91638@Shashi MINGW64 ~
$ notepad++

91638@Shashi MINGW64 ~
$ git version
git version 2.42.0.windows.2

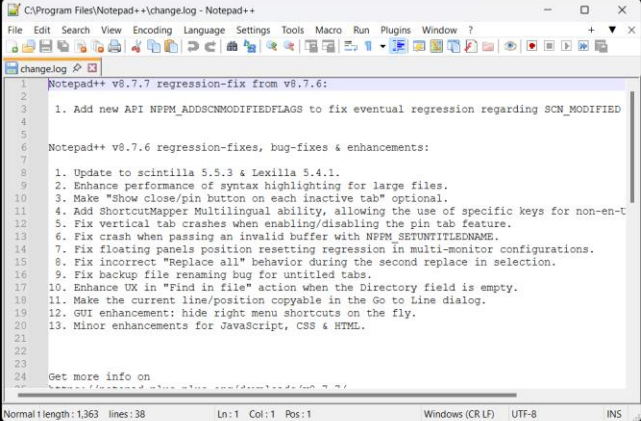
91638@Shashi MINGW64 ~
$ git config --global user.name "Shasmithabalamurugan"

91638@Shashi MINGW64 ~
$ git config --global user.email "Shasmithabalan@gmail.com"

91638@Shashi MINGW64 ~
$ git config --global --list
user.email=Shasmithabalan@gmail.com
user.name=Shasmithabalamurugan
user.password=Shasmitha123
credential.https://git-codecommit.ap-southeast-2.amazonaws.com.provider=generic

91638@Shashi MINGW64 ~
$ notepad++

91638@Shashi MINGW64 ~
$
```



The screenshot shows a Notepad++ window titled "C:\Program Files\Notepad++\change.log - Notepad++". The file content lists regression fixes and enhancements for Notepad++ v8.7.6, including updates to scintilla, syntax highlighting, and various bug fixes. The status bar at the bottom indicates a normal length of 1,363 lines and 38 lines of text.

## b. Notepad++ Integration

```
91638@Shashi MINGW64 ~
$ notepad++

91638@Shashi MINGW64 ~
$ git version
git version 2.42.0.windows.2

91638@Shashi MINGW64 ~
$ git config --global user.name "Shasmithabalamurugan"

91638@Shashi MINGW64 ~
$ git config --global user.email "Shasmithabalan@gmail.com"

91638@Shashi MINGW64 ~
$ git config --global --list
user.email=Shasmithabalan@gmail.com
user.name=Shasmithabalamurugan
user.password=Shasmitha123
credential.https://git-codecommit.ap-southeast-2.amazonaws.com.provider=generic

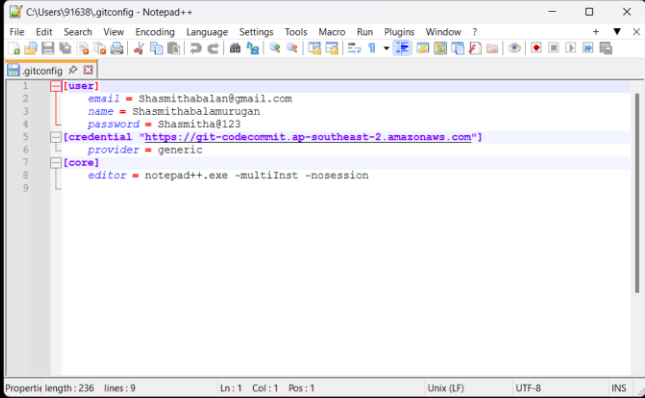
91638@Shashi MINGW64 ~
$ notepad++

91638@Shashi MINGW64 ~
$ notepad++.exe bash -profile

91638@Shashi MINGW64 ~
$ git config --global core.editor "notepad++.exe -multiInst -nosession"

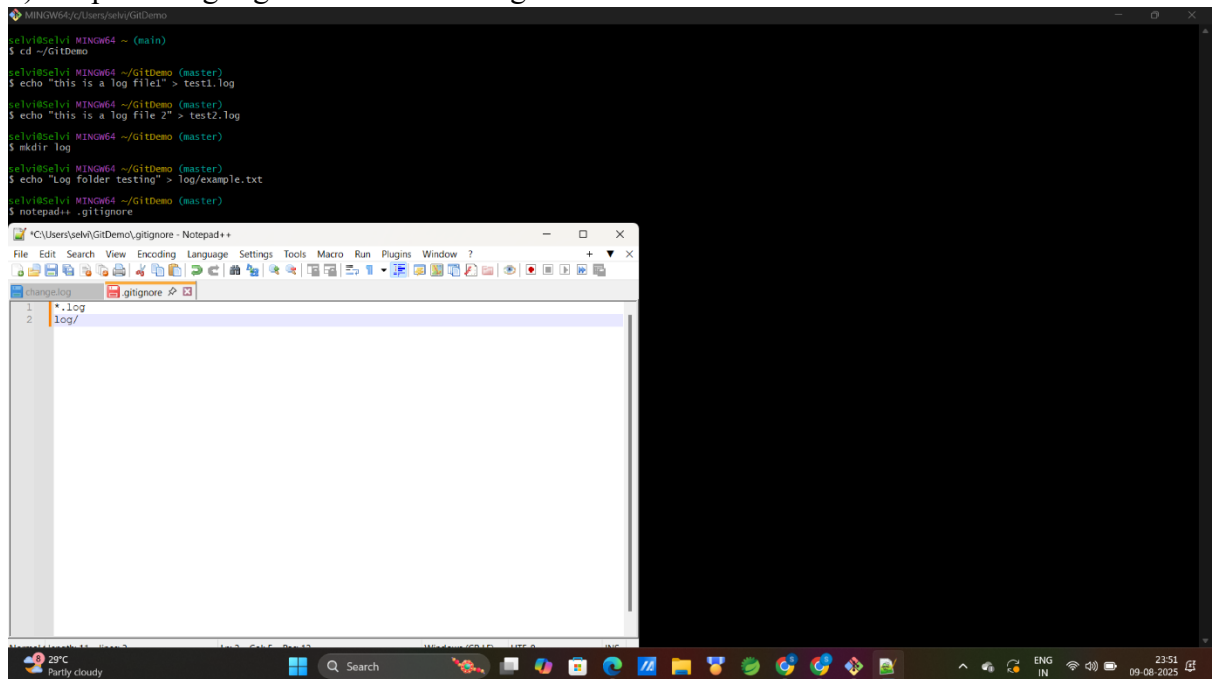
91638@Shashi MINGW64 ~
$ git config --global -e
hint: Waiting for your editor to close the file...

```



The screenshot shows a Notepad++ window titled "C:\Users\91638\gitconfig - Notepad++". The file content shows the configuration for the git user and the core.editor. The status bar at the bottom indicates a property length of 236 lines and 9 lines of text.

## 2) Implement git ignore command to ignore unwanted files and folders

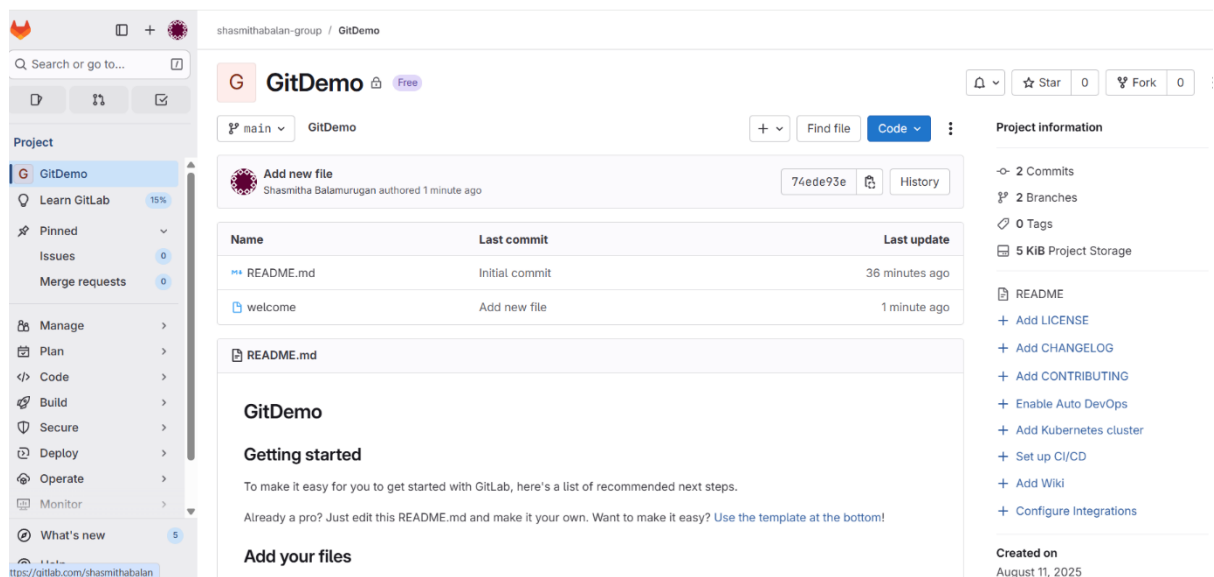


The screenshot shows a Windows terminal window with the following commands and output:

```
PS C:\Users\selvi\GitDemo> cd ~ (main)
PS C:\Users\selvi\GitDemo> cd ~\GitDemo
PS C:\Users\selvi\GitDemo> git init
PS C:\Users\selvi\GitDemo> echo "this is a log file1" > test1.log
PS C:\Users\selvi\GitDemo> echo "this is a log file 2" > test2.log
PS C:\Users\selvi\GitDemo> mkdir log
PS C:\Users\selvi\GitDemo> echo "Log folder testing" > log/example.txt
PS C:\Users\selvi\GitDemo> notepad .gitignore
```

The Notepad++ window shows the content of the .gitignore file:

```
1 *.log
2 log/
```



The screenshot shows the GitLab web interface for the 'GitDemo' project. The left sidebar contains navigation links: Project, Learn GitLab (15%), Pinned, Issues (0), Merge requests (0), Manage, Plan, Code, Build, Secure, Deploy, Operate, Monitor, and What's new (5). The main content area shows the project details for 'GitDemo' (Free). It includes a table of files:

Name	Last commit	Last update
README.md	Initial commit	36 minutes ago
welcome	Add new file	1 minute ago

Below the table, there is a section for 'README.md' with the following content:

### GitDemo

#### Getting started

To make it easy for you to get started with GitLab, here's a list of recommended next steps.

Already a pro? Just edit this README.md and make it your own. Want to make it easy? [Use the template at the bottom!](#)

#### Add your files

On the right side, there is a 'Project information' section with the following details:

- 2 Commits
- 2 Branches
- 0 Tags
- 5 KiB Project Storage

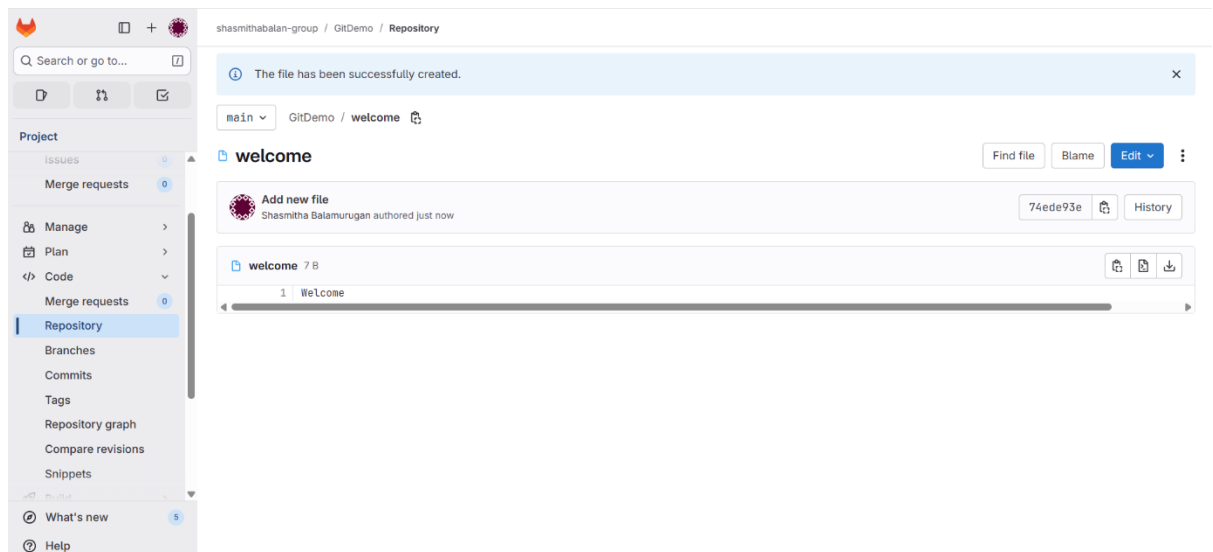
Below this, there are links to add various files and configurations:

- + README
- + Add LICENSE
- + Add CHANGELOG
- + Add CONTRIBUTING
- + Enable Auto DevOps
- + Add Kubernetes cluster
- + Set up CI/CD
- + Add Wiki
- + Configure Integrations

At the bottom, it shows 'Created on August 11, 2025'.

```
MINGW64 ~/GitDemo
$ git add
warning: in the working copy of 'test.txt', LF will be replaced by CRLF the next time Git touches it
$ git push
Everything up-to-date
$ git push origin master
Everything up-to-date
$ git push -u origin master
Everything up-to-date
branch 'master' set up to track 'origin/master'.
$ echo "test ignore" > newfile.log
$ mkdir log && echo "hidden file" > log/hidden.txt
mkdir: cannot create directory 'log': File exists
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   test.txt
$ git push
Everything up-to-date
$ git commit -m "Add test.txt"
[master efc90a6] Add test.txt
1 file changed, 1 insertion(+)
create mode 100644 test.txt
$ git push origin master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 342 bytes | 342.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://gitlab.com/Sel-vi-Murugan/GitDemo.git
390da3..efc90a6 master -> master
$
```



3)Construct a branch, do some changes in the branch, and merge it with master (or trunk)

```
MINGW64/c/Users/sehv/GitDemo
On branch Gitwork
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    hello.xml

nothing added to commit but untracked files present (use "git add" to track)
sehv@SEHV MINGW64 ~/GitDemo (Gitwork)
$ git add hello.xml
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git touches it
sehv@SEHV MINGW64 ~/GitDemo (Gitwork)
$ git commit -m "Add hello.xml in Gitwork Branch"
[Gitwork b47f46a] Add hello.xml in Gitwork Branch
1 file changed, 1 insertion(+)
create mode 100644 hello.xml
sehv@SEHV MINGW64 ~/GitDemo (Gitwork)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
sehv@SEHV MINGW64 ~/GitDemo (master)
$ echo "<greetings>Hello from master</greetings>" > hello.xml
sehv@SEHV MINGW64 ~/GitDemo (master)
$ git add hello.xml
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git touches it
sehv@SEHV MINGW64 ~/GitDemo (master)
$ git commit -m "Add hello.xml in master with different content"
[master 5a012ab] Add hello.xml in master with different content
sehv@SEHV MINGW64 ~/GitDemo (master)
$ git config --global diff.tool p4merge
sehv@SEHV MINGW64 ~/GitDemo (master)
$ git config --global merge.tool p4merge
sehv@SEHV MINGW64 ~/GitDemo (master)
$ 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.
sehv@SEHV MINGW64 ~/GitDemo (master/MERGING)
$ git mergetool
Merging:
hello.xml
Normal merge conflict for 'hello.xml':
  [local]: created file
  [remote]: created file
The merge tool p4merge is not available as 'p4merge'
sehv@SEHV MINGW64 ~/GitDemo (master/MERGING)
$

29°C
Partly cloudy
MINGW64/c/Users/sehv/GitDemo
sehv@SEHV MINGW64 ~/GitDemo (master)
$ git commit -m "Add test.txt"
[master efc90a6] Add test.txt
1 file changed, 1 insertion(+)
create mode 100644 test.txt
sehv@SEHV MINGW64 ~/GitDemo (master)
$ git push origin master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 342 bytes | 342.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
to https://gitlab.com/SeI-vi-Murugan/GitDemo.git
396daa3..efc90a6 master -> master
sehv@SEHV MINGW64 ~/GitDemo (master)
$ git branch GitNewBranch
sehv@SEHV MINGW64 ~/GitDemo (Git-T03-HOL_002)
$ git branch -a
* Git-T03-HOL_002
  GitNewBranch
  master
  remotes/origin/master
sehv@SEHV MINGW64 ~/GitDemo (Git-T03-HOL_002)
$ git checkout GitNewBranch
Switched to branch 'GitNewBranch'
sehv@SEHV MINGW64 ~/GitDemo (GitNewBranch)
$ echo "This is new branch file" > branchFile.txt
sehv@SEHV MINGW64 ~/GitDemo (GitNewBranch)
$ git add branchFile.txt
warning: in the working copy of 'branchFile.txt', LF will be replaced by CRLF the next time Git touches it
sehv@SEHV MINGW64 ~/GitDemo (GitNewBranch)
$ git commit -m "Add branchFile.txt in GitNewBranch"
[GitNewBranch 61dc4b1] Add branchFile.txt in GitNewBranch
1 file changed, 1 insertion(+)
create mode 100644 branchFile.txt
sehv@SEHV MINGW64 ~/GitDemo (GitNewBranch)
$ git status
On branch GitNewBranch
nothing to commit, working tree clean
sehv@SEHV MINGW64 ~/GitDemo (GitNewBranch)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
sehv@SEHV MINGW64 ~/GitDemo (master)
$

29°C
Partly cloudy
```

```
MINGW64/c/Users/sejvi/GitDemo
CONFLICT (add/add): Merge conflict in hello.xml
Automatic merge failed; fix conflicts and then commit the result.
$ git mergetool
merging:
hello.xml

Normal merge conflict for 'hello.xml':
  (local): created file
  (remote): created file
The merge tool 'p4merge' is not available as 'p4merge'
$ git add hello.xml
$ git commit -m "Resolve merge conflict between master and Gitwork"
[master 6c77af] Resolve merge conflict between master and Gitwork
$ echo ".orig" >> .gitignore
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it
$ git commit -m "Ignore merge backup files"
[master 3aedcc] Ignore merge backup files
1 file changed, 1 insertion(+), 1 deletion(-)
$ git branch
  Git-T03-HDL_002
  GitNewBranch
  Gitwork
* master
$ git branch -d Gitwork
Deleted branch Gitwork (was b47f46a).
$ git log --oneline --graph --decorate
* 3aedcc (HEAD -> master) Ignore merge backup files
* 6c77af Resolve merge conflict between master and Gitwork
* b47f46a Add hello.xml in Gitwork Branch
* 5a012ab Add hello.xml in master with different content
/ efc90a6 (origin/master, Git-T03-HDL_002) Add test.txt
/ 896daa3 to ignore .log files
/ 07e35d0 Add welcome.txt
$
```

4) Implement conflict resolution when multiple users are updating the trunk (or master) in such a way that it results into a conflict with the branch's modification.

```
MINGW64/c/Users/sejvi/GitDemo
$ cd GitDemo
$ git checkout master
Already on 'master'
Your branch is up to date with 'origin/master'.
$ git status
on branch master
Your branch is up to date with 'origin/master'.
nothing to commit, working tree clean
$ git checkout -b Gitwork
Switched to a new branch 'Gitwork'
$ echo "<greeting>hello from branch/greeting-" > hello.xml
$ git status
on branch Gitwork
untracked files:
  (use "git add <file>..." to include in what will be committed)
    hello.xml
nothing added to commit but untracked files present (use "git add" to track)
$ git add hello.xml
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git touches it
$ git commit -m "Add hello.xml in Gitwork Branch"
[Gitwork b47f46a] Add hello.xml in Gitwork Branch
1 file changed, 1 insertion(+)
create mode 100644 hello.xml
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
$ echo "<greeting>hello from master/greeting-" > hello.xml
$ git add hello.xml
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git touches it
```

```
MINGW64/c/Users/sehvi/GitDemo
$ cd GitDemo
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
$ echo "greeting1111 from master/greeting" > hello.txt
$ git add hello.txt
Changes to be committed:
  new file   : hello.txt
warning: in the working copy of 'hello.txt', LF will be replaced by CRLF the next time git touches it
$ git commit -m "Add hello.txt in master with different content"
[master 62ca2b2] Add hello.txt in master with different content
1 file changed, 1 insertion(+)
$ git config --global merge.tool p4merge
$ git merge --no-commit p4merge
$ git merge GitDemo
Merging GitDemo
CONFLICT (add/add): Merge conflict in hello.txt
Automatic merge failed. Fix conflicts and then commit the result.
$ git merge --abort
$ git merge --continue
$ git merge --continue
Normal merge conflict for 'hello.txt':
  (branch) created file
  (branch) created file
  the merge tool p4merge is not available as 'p4merge'
$ git add hello.txt
$ git commit -m "Resolve merge conflict between master and GitDemo"
[master 62f7f9f] Resolve merge conflict between master and GitDemo
1 file changed, 1 insertion(+)
$ echo "greeting" > .gitignore
$ git add .gitignore
Changes to be committed:
  new file   : .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time git touches it
$ git commit -m "Ignore merge backup files"
[master 1a6001c] Ignore merge backup files
1 file changed, 1 insertion(+), 1 deletion(-)
$ git checkout master
$ git branch
  Git-T03-HOL_002
  GitDemoBranch
  GitDemo
  master
$ git branch -d GitDemo
Deleted branch GitDemo (was 1a7f464).
$ git log --pretty=format:"%h %s" --graph --decorate
* 1a7f464 Add hello.txt in GitDemo branch
* 62f7f9f Resolve merge conflict in master with different content
* 62ca2b2 Add hello.txt in master with different content
* 1a6001c Ignore merge backup files
* 1a7f464 Add hello.txt
$ git push origin master
error: pathspec 'Git-T03-HOL_002' did not match any file(s) known to git
$ git checkout -b Git-T03-HOL_002
Switched to a new branch 'Git-T03-HOL_002'
$ git push origin Git-T03-HOL_002
error: src refspec Git-T03-HOL_002 does not match any
error: failed to push some refs to 'https://gitlab.com/Sei-vi-Murugan/GitDemo.git'
```

5) Execute steps involving clean up and push back to remote Git.

```
MINGW64/c/Users/sehvi/GitDemo
$ cd GitDemo
$ git status
On branch master
Your branch is up to date with 'origin/master'.
nothing to commit, working tree clean
$ git branch -a
* master
+ remotes/origin/master
$ git checkout master
Already on 'master'
Your branch is up to date with 'origin/master'.
$ git pull origin master
From https://gitlab.com/Sei-vi-Murugan/GitDemo
+ branch      master    -> FETCH_HEAD
Already up to date.
$ git checkout Git-T03-HOL_002
error: pathspec 'Git-T03-HOL_002' did not match any file(s) known to git
$ git push origin Git-T03-HOL_002
error: src refspec Git-T03-HOL_002 does not match any
error: failed to push some refs to 'https://gitlab.com/Sei-vi-Murugan/GitDemo.git'
$ git fetch origin
$ git checkout Git-T03-HOL_002
error: pathspec 'Git-T03-HOL_002' did not match any file(s) known to git
$ git checkout -b Git-T03-HOL_002
Switched to a new branch 'Git-T03-HOL_002'
$
```

```
MINGW64/c/Users/sehvi/GitDemo
sehvi@sehvi MINGW64 ~/GitDemo (master)
$ git commit -m "Add test.txt"
[master efc90a6] Add test.txt
1 file changed, 1 insertion(+)
create mode 100644 test.txt

sehvi@sehvi MINGW64 ~/GitDemo (master)
$ git push origin master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), 342 bytes | 342.00 KiB/s, done.
Writing objects: 100% (3/3), 342 bytes | 342.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://gitlab.com/Sei-vi-Murugan/GitDemo.git
39bdaa3..efc90a6 master -> master

sehvi@sehvi MINGW64 ~/GitDemo (master)
$ git branch GitNewBranch

sehvi@sehvi MINGW64 ~/GitDemo (Git-T03-HOL_002)
$ git branch -m
git-T03-HOL_002
GitNewBranch
master
renotes/origin/master

sehvi@sehvi MINGW64 ~/GitDemo (Git-T03-HOL_002)
$ git checkout GitNewBranch
Switched to branch 'GitNewBranch'

sehvi@sehvi MINGW64 ~/GitDemo (GitNewBranch)
$ echo "This is new branch file" > branchfile.txt

sehvi@sehvi MINGW64 ~/GitDemo (GitNewBranch)
$ git add branchfile.txt
warning: in the working copy of 'branchfile.txt', LF will be replaced by CRLF the
next time Git touches it.

sehvi@sehvi MINGW64 ~/GitDemo (GitNewBranch)
$ git commit -m "Add branchfile.txt in GitNewBranch"
[GitNewBranch 61dc4b1] Add branchfile.txt in GitNewBranch
1 file changed, 1 insertion(+)
create mode 100644 branchfile.txt

sehvi@sehvi MINGW64 ~/GitDemo (GitNewBranch)
$ git status
On branch GitNewBranch
nothing to commit, working tree clean

sehvi@sehvi MINGW64 ~/GitDemo (GitNewBranch)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

sehvi@sehvi MINGW64 ~/GitDemo (master)
$
```

29°C  
Partly cloudy



Search



ENG  
IN



17:00

10-08-2025