Git

Git HOL-1

- **Setup your machine with Git Configuration**
- **Integrate notepad++.exe to Git and make it a default editor**
- **Add a file to source code repository**

➔solution:

<mark>Setting global configuration</mark>

```
$ git config --global --list
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
user.name=Shom
user.email=abc@gmail.com
core.editor=Notepad
```

<mark>Committing a file</mark>

$ echo "welcome to version control" >> welcome.text

```
$ ls -al
total 5
drwxr-xr-x 1 schow 197609  0 Aug  7 09:02 ./
drwxr-xr-x 1 schow 197609  0 Aug  7 08:48 ../
drwxr-xr-x 1 schow 197609  0 Aug  7 08:49 .git/
-rw-r--r-- 1 schow 197609 27 Aug  7 09:02 welcome.text
```

```
                    MINGW64 ~/Desktop/git and github/GitDemo (master)
$ git add welcome.text
warning: in the working copy of 'welcome.text', LF will be replaced by CRLF the next time Git touches it

                    MINGW64 ~/Desktop/git and github/GitDemo (master)
$ git commit -m "text file"
[master (root-commit) 81dd21d] text file
 1 file changed, 1 insertion(+)
 create mode 100644 welcome.text
```

```
schow@LAPTOP-Q7LUT3G6 MINGW64 ~/Desktop/git and github/GitDemo (master)
$ git remote add origin https://github.com/shomexe/GitDemo.git

schow@LAPTOP-Q7LUT3G6 MINGW64 ~/Desktop/git and github/GitDemo (master)
$ git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 437 bytes | 109.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:      https://github.com/shomexe/GitDemo/pull/new/master
remote:
To https://github.com/shomexe/GitDemo.git
 * [new branch]      master -> master
```

Git HOL-2

- **Implement git ignore command to ignore unwanted files and folders**

➔solution:

created **log folder** and a text file inside it then created a file with **.log** extension

📁 .git	10-08-2025 09:57	File folder	
📁 log	10-08-2025 09:49	File folder	
⚙ .gitignore	10-08-2025 09:57	Git Ignore Source ...	1 KB
📄 abc.log	10-08-2025 09:50	Text Document	1 KB
📄 welcome.text	07-08-2025 09:02	TEXT File	1 KB
📄 welcome2.txt	07-08-2025 09:09	Text Document	1 KB

added .log file and log folder inside **.gitignore** to be ignored by git when new commit is done

```
≡ notes1.txt        ◈ .gitignore  ✕                    ▷  ⬜  ...

C: > Users > schow > Desktop > git and github > GitDemo > ◈ .gitignore
     1    *.log
     2    log/
```

$ git add .
$ git commit -m ".gitignore file added"

```
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), 292 bytes | 97.00 KiB/s, done
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
```

😊 **nagasurya** .gitignore file added		f280e5d · 26 minutes ago	🕘 **3 Commits**
📄 .gitignore	.gitignore file added		26 minutes ago
📄 welcome.text	text file		3 days ago
📄 welcome2.txt	commit done		3 days ago

Git HOL-3

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

➔ solution:

$ git checkout -b NewBranch

$ git branch -a

```
* NewBranch
  main
  master
  remotes/origin/master
```

$ echo "...file content" >> file.txt

$ git add .

$ git commit -m "file added to new branch"

```
[NewBranch 88a6f6f] file added to new branch
 1 file changed, 1 insertion(+)
 create mode 100644 file.txt
```

$ git switch master

$ git diff master..NewBranch

```
diff --git a/file.txt b/file.txt
new file mode 100644
index 0000000..ac62a20
--- /dev/null
+++ b/file.txt
@@ -0,0 +1 @@
+...file content
```

$ git merge NewBranch

```
Updating f280e5d..88a6f6f
Fast-forward
 file.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 file.txt
```

$ git log –oneline –graph –decorate

```
* 88a6f6f (HEAD -> master, NewBranch) file added to new branch
* f280e5d (origin/master, main) .gitignore file added
* d640d24 commit done
* 81dd21d text file
```

$ git branch -d NewBranch

Git HOL-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.**

➔solution:

$ git checkout -b GitWork

$ echo "hello world" > hello.xml

$ git add hello.xml

$ git commit -m "in branch GitWork"

$ git switch master

$ echo "hello world 2" > hello.xml

$ git add .

$ git commit -m "in branch master"

$ git log --oneline --graph --decorate --all

```
* 037c9c8 (HEAD -> master) in branch master
| * a4a6e93 (GitWork) in branch GitWork
|/
* 88a6f6f (origin/master) file added to new branch
* f280e5d (main) .gitignore file added
* d640d24 commit done
* 81dd21d text file
```

$ git diff master GitWork

```
diff --git a/hello.xml b/hello.xml
index d0e1e95..3b18e51 100644
--- a/hello.xml
+++ b/hello.xml
@@ -1 +1 @@
-hello world 2
+hello world
```

$ 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.
```

$ cat hello.xml

```
<<<<<<< HEAD
hello world 2
=======
hello world
>>>>>>> GitWork
```

$ git mergetool (used to resolve conflict by editing file and saving it)

$ git status

```
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

All conflicts fixed but you are still merging.
  (use "git commit" to conclude merge)

Changes to be committed:
        modified:   hello.xml

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

$ git add hello.xml

$ git commit -m "merge conflict resolved"

$ echo "*.orig" >> .gitignore

$ git add .gitignore

$ git commit -m ".gitignore updated"

$ git log --oneline --graph –decorate

```
* 328607e (HEAD -> master) .gitignore updated
*   230e99a merge conflict resolved
|\
| * a4a6e93 (GitWork) in branch GitWork
* | 037c9c8 in branch master
|/
* 88a6f6f (origin/master) file added to new branch
* f280e5d (main) .gitignore file added
* d640d24 commit done
* 81dd21d text file
```

$ git branch -d GitWork

Git HOL-5

- **Execute steps involving clean up and push back to remote Git.**

➜solution:

$ git status

```
On branch master
Your branch is ahead of 'origin/master' by 4 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```

$ git branch -a

```
  main
* master
  remotes/origin/master
```

$ git push

```
Enumerating objects: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (12/12), 1.03 KiB | 1.03 MiB/s, done.
Total 12 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/shomexe/GitDemo.git
   88a6f6f..328607e  master -> master
```

Changes reflected on remote repository

.gitignore	.gitignore updated	24 minutes ago
file.txt	file added to new branch	1 hour ago
hello.xml	merge conflict resolved	26 minutes ago
welcome.text	text file	3 days ago
welcome2.txt	commit done	3 days ago