

Week - 8

Git

Handson - 1:

```
Windows PowerShell
Directory: C:\Users\admin\GitDemo

Mode                LastWriteTime         Length Name
----                -
d-----          10-08-2025    22:02             .git

PS C:\Users\admin\GitDemo> echo "Welcoms to git dems">welcome.txt
PS C:\Users\admin\GitDemo> ls

Directory: C:\Users\admin\GitDemo

Mode                LastWriteTime         Length Name
----                -
-a-----          10-08-2025    22:04             44 welcome.txt

PS C:\Users\admin\GitDemo> cat welcome.txt
Welcoms to git dems
PS C:\Users\admin\GitDemo> 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)
PS C:\Users\admin\GitDemo> git add welcome.txt

Windows PowerShell

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)
PS C:\Users\admin\GitDemo> git add welcome.txt
PS C:\Users\admin\GitDemo> git commit
[master (root-commit) 49f1064] GIT This is the first assignment of GIT
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 welcome.txt
PS C:\Users\admin\GitDemo> git status
On branch master

nothing to commit, working tree clean
PS C:\Users\admin\GitDemo> git remote add origin https://gitlab.com/abinayashrrr2005-group/gitdemo
PS C:\Users\admin\GitDemo> git pull origin master
info: please complete authentication in your browser...
warning: redirecting to https://gitlab.com/abinayashrrr2005-group/gitdemo.git/
fatal: couldn't find remote ref master
PS C:\Users\admin\GitDemo> git push origin master
warning: redirecting to https://gitlab.com/abinayashrrr2005-group/gitdemo.git/
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 275 bytes | 91.08 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: To create a merge request for master, visit:
remote:   https://gitlab.com/abinayashrrr2005-group/gitdemo/-/merge_requests/new?merge_request%5Bsource_branch%5D=master
remote:
To https://gitlab.com/abinayashrrr2005-group/gitdemo
 * [new branch]      master -> master
PS C:\Users\admin\GitDemo>
```

C:\Users\admin\.gitconfig - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ? + ▾

length: 105 lines: 6 Ln: 1 Col: 1 Pos: 1 Unix (LF) UTF-8 IN

```
1 [user]
2   name = Abhi
3   email = abinayashrrr2005@gmail.com
4 [core]
5   editor = notepad++ -multiInst -nosession
6
```

C:\Users\admin\GitDmo\welcome.txt - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ? + ▾

length: 21 lines: 2 Ln: 1 Col: 1 Pos: 1 Windows (CR LF) UTF-16 LE BOM IN

```
1 Welcome to git demo
2
```

Handson - 2:

Create a **“.log”** file and a **log folder** in the working directory of Git. Update the **.gitignore** file in such a way that on committing, these files (.log extensions and log folders) are ignored.

Verify if the git status reflects the same about working directory, local repository and git repository.

```
Windows PowerShell
PS C:\Users\admin\GitDemo> git init
Reinitialized existing Git repository in C:/Users/admin/GitDemo/.git/
PS C:\Users\admin\GitDemo> echo "Welcome to Git">welcome.txt
PS C:\Users\admin\GitDemo> ls

Directory: C:\Users\admin\GitDemo

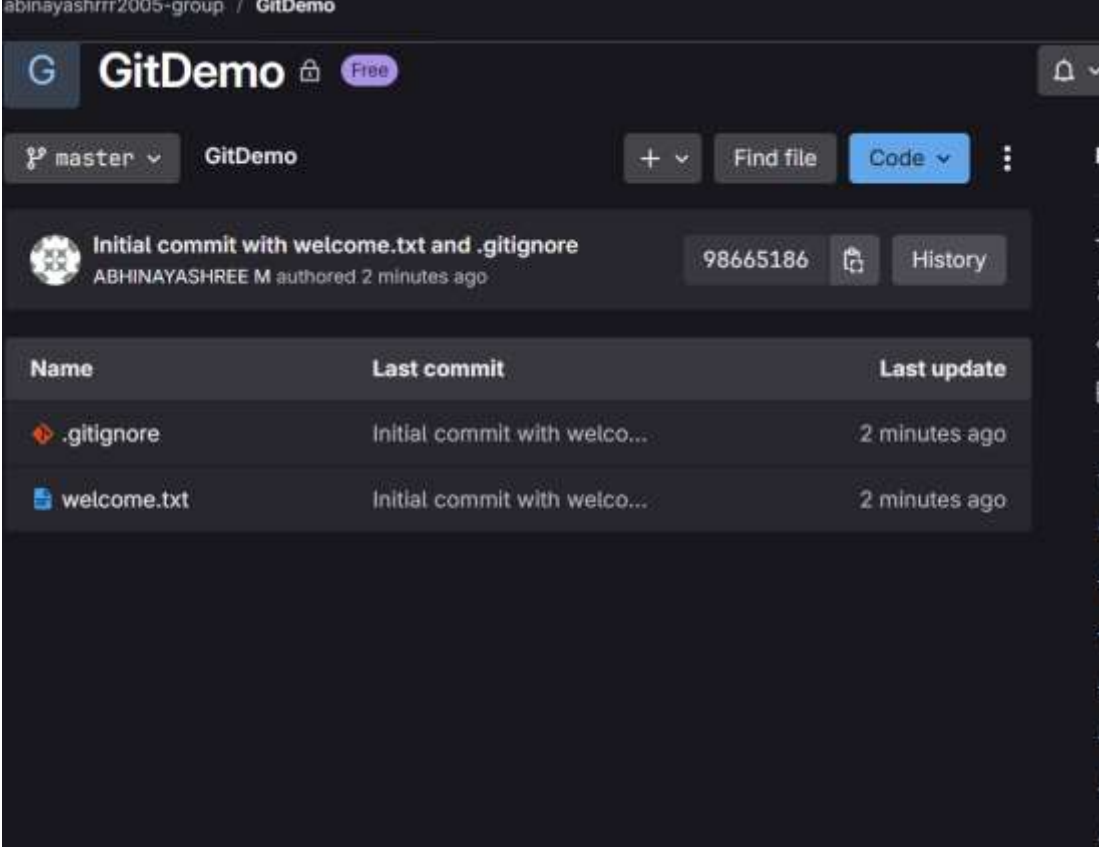
Mode                LastWriteTime         Length Name
----                -
-a-----         12-08-2025    10:02             34 welcome.txt

PS C:\Users\admin\GitDemo> cat welcome.txt
Welcome to Git
PS C:\Users\admin\GitDemo> echo "this is a debug log">debug.log
PS C:\Users\admin\GitDemo> mkdir log

Directory: C:\Users\admin\GitDemo

Mode                LastWriteTime         Length Name
----                -
d-----         12-08-2025    10:03             log

PS C:\Users\admin\GitDemo> echo "log_data">log/app.log
PS C:\Users\admin\GitDemo> notepad .gitignore
PS C:\Users\admin\GitDemo> git status
On branch master
abinyashrrr2005-group / GitDemo
```



The screenshot shows the Git web interface for a repository named 'GitDemo'. The interface is in dark mode. At the top, there's a header with the repository name 'GitDemo' and a 'Free' badge. Below the header, there's a navigation bar with 'master' selected, a 'Find file' button, and a 'Code' button. The main content area shows the 'Initial commit with welcome.txt and .gitignore' by 'ABHINAYASHREE M' authored 2 minutes ago. Below this, there's a table with two rows: one for '.gitignore' and one for 'welcome.txt', both showing the 'Initial commit with welco...' and '2 minutes ago'.

| Name | Last commit | Last update |
|-------------|------------------------------|---------------|
| .gitignore | Initial commit with welco... | 2 minutes ago |
| welcome.txt | Initial commit with welco... | 2 minutes ago |

Handson - 3:

Branching:

1. Create a new branch **"GitNewBranch"**.
2. List all the local and remote branches available in the current trunk. Observe the "*" mark which denote the current pointing branch.
3. Switch to the newly created branch. Add some files to it with some contents.
4. Commit the changes to the branch.
5. Check the status with **"git status"** command.

Merging:

Switch to the master

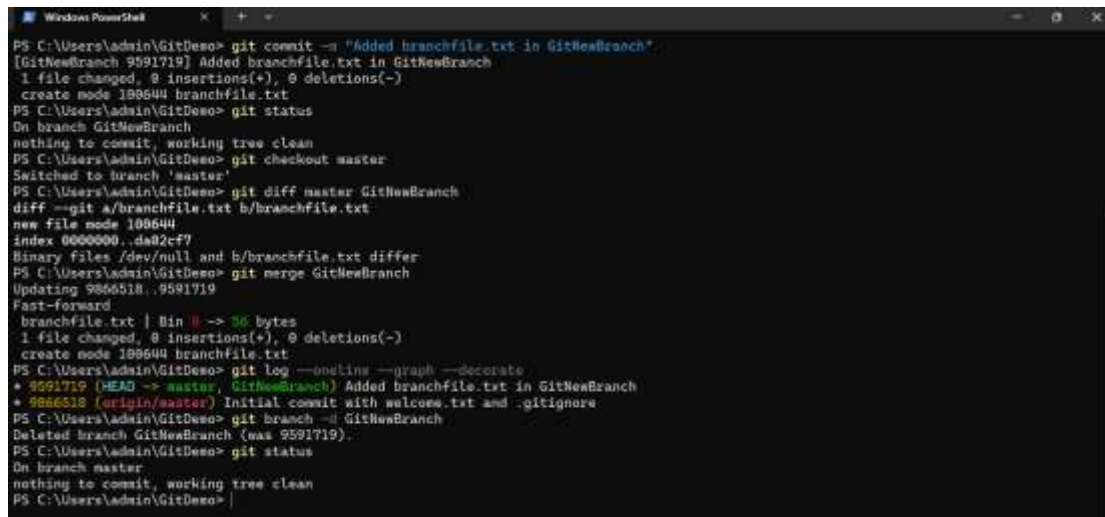
List out all the differences between trunk and branch. These provide the differences in command line interface.

List out all the visual differences between master and branch using **P4Merge tool**.

Merge the source branch to the trunk.

Observe the logging after merging using **"git log --oneline --graph --decorate"**

Delete the branch after merging with the trunk and observe the git status.



```
PS C:\Users\admin\GitDemo> git commit -m "Added branchfile.txt in GitNewBranch"
[GitNewBranch 9591719] Added branchfile.txt in GitNewBranch
1 file changed, 8 insertions(+), 0 deletions(-)
create mode 100644 branchfile.txt
PS C:\Users\admin\GitDemo> git status
On branch GitNewBranch
nothing to commit, working tree clean
PS C:\Users\admin\GitDemo> git checkout master
Switched to branch 'master'
PS C:\Users\admin\GitDemo> git diff master GitNewBranch
diff --git a/branchfile.txt b/branchfile.txt
new file mode 100644
index 0000000..da02cf7
Binary files /dev/null and b/branchfile.txt differ
PS C:\Users\admin\GitDemo> git merge GitNewBranch
Updating 9866518..9591719
Fast-forward
 branchfile.txt | Bin 0 -> 36 bytes
1 file changed, 8 insertions(+), 0 deletions(-)
create mode 100644 branchfile.txt
PS C:\Users\admin\GitDemo> git log --oneline --graph --decorate
* 9591719 (HEAD -> master, GitNewBranch) Added branchfile.txt in GitNewBranch
+ 9866518 (origin/master) Initial commit with welcome.txt and .gitignore
PS C:\Users\admin\GitDemo> git branch -d GitNewBranch
Deleted branch GitNewBranch (was 9591719).
PS C:\Users\admin\GitDemo> git status
On branch master
nothing to commit, working tree clean
PS C:\Users\admin\GitDemo>
```

Handson - 4:

1. Verify if master is in clean state.
2. Create a branch **"GitWork"**. Add a file **"hello.xml"**.
3. Update the content of **"hello.xml"** and observe the status
4. Commit the changes to reflect in the branch
5. Switch to master.
6. Add a file **"hello.xml"** to the master and add some different content than previous.
7. Commit the changes to the master
8. Observe the log by executing **"git log --oneline --graph --decorate --all"**
9. Check the differences with Git diff tool
10. For better visualization, use P4Merge tool to list out all the differences between master and branch
11. Merge the bran to the master
12. Observe the git mark up.
13. Use 3-way merge tool to resolve the conflict
14. Commit the changes to the master, once done with conflict
15. Observe the git status and add backup file to the .gitignore file.
16. Commit the changes to the .gitignore

17. List out all the available branches
18. Delete the branch, which merge to master.
19. Observe the log by executing “**git log --oneline --graph --decorate**”

```

PS C:\Users\admin\GitDemo> git status
On branch master
nothing to commit, working tree clean
PS C:\Users\admin\GitDemo> git checkout -b GitWork
git: 'checkout' is not a git command. See 'git --help'.

The most similar command is
  checkout
PS C:\Users\admin\GitDemo> git checkout -b GitWork
Switched to a new branch 'GitWork'
PS C:\Users\admin\GitDemo> echo "<greeting>Hello from Gitwork branch</greeting>">hello.xml
PS C:\Users\admin\GitDemo> echo "<greeting>Hello from updated GitWork branch</greeting>"> hello.xml
PS C:\Users\admin\GitDemo> 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)
PS C:\Users\admin\GitDemo> git add hello.xml
PS C:\Users\admin\GitDemo> git commit -m "Updated hello.xml in GitWork branch"
[GitWork bea95fd] Updated hello.xml in GitWork branch
1 file changed, 8 insertions(+), 0 deletions(-)
create mode 100644 hello.xml
PS C:\Users\admin\GitDemo> git checkout master
Switched to branch 'master'
PS C:\Users\admin\GitDemo> echo "<greeting>Hello from master branch</greeting>"> hello.xml
PS C:\Users\admin\GitDemo> git add hello.xml
PS C:\Users\admin\GitDemo> git commit -m "Added hello.xml in master branch"
[master 87a6588] Added hello.xml in master branch
1 file changed, 8 insertions(+), 0 deletions(-)
create mode 100644 hello.xml
PS C:\Users\admin\GitDemo> git log --oneline --graph --decorate --all

* 87a6588 (HEAD -> master) Added hello.xml in master branch
| * bea95fd (GitWork) Updated hello.xml in GitWork branch
|
* 9591719 Added branchfile.txt in GitNewBranch
* 9866518 (origin/master) Initial commit with welcome.txt and .gitignore
PS C:\Users\admin\GitDemo> git diff master GitWork
diff --git a/hello.xml b/hello.xml
index 66aa6d3..ea6de39 100644
Binary files a/hello.xml and b/hello.xml differ
PS C:\Users\admin\GitDemo> git merge GitWork
warning: Cannot merge binary files: hello.xml (HEAD vs. GitWork)
Auto-merging hello.xml
CONFLICT (add/add): Merge conflict in hello.xml
Automatic merge failed; fix conflicts and then commit the result.
PS C:\Users\admin\GitDemo> git add hello.xml
PS C:\Users\admin\GitDemo> git commit -m "Resolved"
[master d29953a] Resolved
PS C:\Users\admin\GitDemo> echo "<*.orig>">.gitignore
PS C:\Users\admin\GitDemo> git add .gitignore
PS C:\Users\admin\GitDemo> git commit -m "Ignore backup files"
On branch master
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)
PS C:\Users\admin\GitDemo> git commit -a -m "Ignore backup files"
On branch master
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)
PS C:\Users\admin\GitDemo> git branch
  GitWork
* master
PS C:\Users\admin\GitDemo> git branch -d GitWork
Deleted branch GitWork (was bea95fd).
PS C:\Users\admin\GitDemo> git log --oneline --graph --decorate
* d29953a (HEAD -> master) Resolved
|
| * bea95fd Updated hello.xml in GitWork branch
| * 87a6588 Added hello.xml in master branch
|
* 9591719 Added branchfile.txt in GitNewBranch
* 9866518 (origin/master) Initial commit with welcome.txt and .gitignore
PS C:\Users\admin\GitDemo>

```

Handson - 5 :

Verify if master is in clean state.







List out all the available branches.

Pull the remote git repository to the master

Push the changes, which are pending from “Git-T03-HOL_002” to the remote repository.

Observe if the changes are reflected in the remote repository.

```
PS C:\Users\admin\GitDemo> git status
On branch Git-T03-HOL_002
nothing to commit, working tree clean
PS C:\Users\admin\GitDemo> git branch -a
* Git-T03-HOL_002
  master
  remotes/origin/Git-T03-HOL_002
  remotes/origin/master
PS C:\Users\admin\GitDemo> git checkout master
Switched to branch 'master'
PS C:\Users\admin\GitDemo> git pull origin master
From https://gitlab.com/abinayashrrr2005-group/GitDemo
  * branch      master       -> FETCH_HEAD
Already up to date.
PS C:\Users\admin\GitDemo> git checkout Git-T02-HOL_002
error: pathspec 'Git-T02-HOL_002' did not match any file(s) known to git
PS C:\Users\admin\GitDemo> git checkout Git-T03-HOL_002
Switched to branch 'Git-T03-HOL_002'
PS C:\Users\admin\GitDemo> git status
On branch Git-T03-HOL_002
nothing to commit, working tree clean
PS C:\Users\admin\GitDemo> git add .
PS C:\Users\admin\GitDemo> git commit -m "your commit message for pending changes"
On branch Git-T03-HOL_002
nothing to commit, working tree clean
PS C:\Users\admin\GitDemo> git push origin Git-T03-HOL_002
Everything up-to-date
PS C:\Users\admin\GitDemo>
```

| Name | Last commit | Last update |
|--|--|----------------|
|  .gitignore | Initial commit with welcome.txt and .gitign... | 26 minutes ago |
|  branchfile.txt | Added branchfile.txt in GitNewBranch | 18 minutes ago |
|  .gitignore | yes | 6 minutes ago |
|  hello.xml | Added hello.xml in master branch | 13 minutes ago |
|  hol002.txt | Initial commit to Git-T03-HOL_002 branch | 3 minutes ago |
|  welcome.txt | Initial commit with welcome.txt and .gitign... | 26 minutes ago |