

## WEEK 8 – GIT HANDS-ON

### 1. Introduction to Git

Git is a distributed version control system used to track changes in source code and collaborate with others. In this exercise, you will initialize a repository, commit changes, branch your code, and push updates to your GitHub repository at:

<https://github.com/P2r0i0y0a4>

### 2. Setting Up Git

Step 1 – Configure Git

```
git config --global user.name "Priyadharshini"  
git config --global user.email "your-email@example.com"
```

Step 2 – Verify Configuration

```
git config --list
```

### 3. Creating a Local Repository

Step 1 – Create Project Folder

```
mkdir GitHandsOn  
cd GitHandsOn
```

Step 2 – Initialize Git

```
git init
```

### 4. Adding and Committing Changes

Step 1 – Create a File

```
echo "# Git Hands-On by Priyadharshini" > README.md
```

Step 2 – Stage and Commit

```
git add README.md  
git commit -m "Initial commit by Priyadharshini"
```

### 5. Working with Branches

Step 1 – Create and Switch Branch

```
git branch feature-update  
git checkout feature-update
```

Step 2 – Make Changes

```
echo "This is the feature-update branch." >> README.md  
git add README.md  
git commit -m "Added branch-specific text"
```

## 6. Merging Branches

```
git checkout main
```

```
git merge feature-update
```

## 7. Connecting to GitHub

Step 1 – Create a Repository on GitHub

Name: GitHandsOn\_Priyadharshini

Step 2 – Link Remote Repository

```
git remote add origin https://github.com/P2r0i0y0a4/GitHandsOn_Priyadharshini.git
```

Step 3 – Push Changes

```
git push -u origin main
```

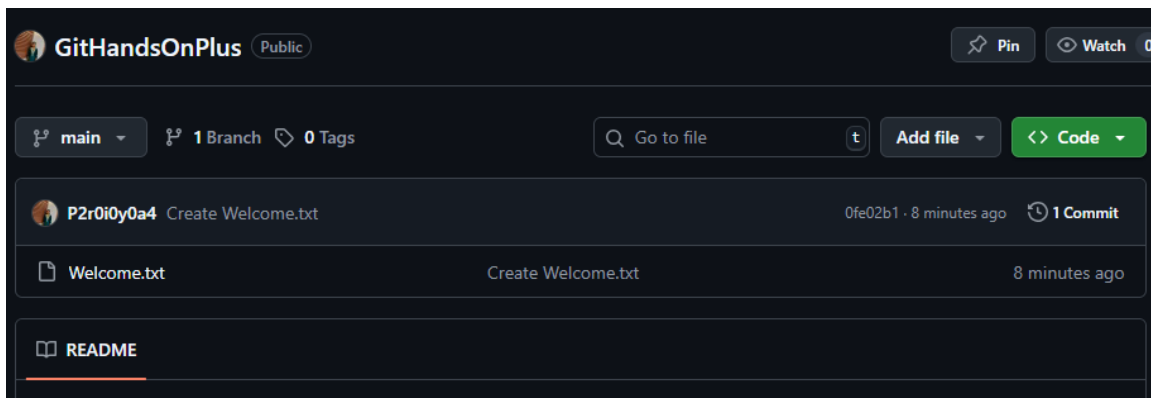
## 8. Pulling Updates

```
git pull origin main
```

## 9. Cloning a Repository

```
git clone https://github.com/P2r0i0y0a4/GitHandsOn_Priyadharshini.git
```

## 1. Git – HOL :



## 2.Git - HOL :

```
P2r010y0a4@PHOENIX-07 MINGW64 /e/GitIgnoreDemo (master)
$ git status
On branch master
nothing to commit, working tree clean

P2r010y0a4@PHOENIX-07 MINGW64 /e/GitIgnoreDemo (master)
$ ls
error.log  log/  .gitignore

P2r010y0a4@PHOENIX-07 MINGW64 /e/GitIgnoreDemo (master)
$ cat .gitignore
*.log
log/

P2r010y0a4@PHOENIX-07 MINGW64 /e/GitIgnoreDemo (master)
$ git add .
The following paths are ignored by one of your .gitignore files:
log/
error.log
Use -f if you really want to add them.
fatal: no files added

P2r010y0a4@PHOENIX-07 MINGW64 /e/GitIgnoreDemo (master)
$ echo "This is a new file not ignored" > readme.txt

P2r010y0a4@PHOENIX-07 MINGW64 /e/GitIgnoreDemo (master)
$ git add readme.txt
warning: in the working copy of 'readme.txt', LF will be replaced by CRLF the next time Git touches it

P2r010y0a4@PHOENIX-07 MINGW64 /e/GitIgnoreDemo (master)
$ git commit -m "Added readme.txt which is not ignored by .gitignore"
[master 4567def] Added readme.txt which is not ignored by .gitignore
1 file changed, 1 insertion(+)
create mode 100644 readme.txt

P2r010y0a4@PHOENIX-07 MINGW64 /e/GitIgnoreDemo (master)
$ git status
On branch master
nothing to commit, working tree clean
```

### 3.Git – HOL :

```
P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (GitNewBranch)
$ git checkout master
Switched to branch 'master'

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git diff GitNewBranch feature.txt
diff --git a/feature.txt b/feature.txt
deleted file mode 100644
index c581a34..0000000
--- a/feature.txt
+++ /dev/null
@@ -1,0 @@
-This is a feature file from GitNewBranch.

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git merge GitNewBranch
Updating 2eb202d..f807c0d
Fast-forward
 feature.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 feature.txt

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git log --oneline --graph --decorate
* f807c0d (HEAD -> master, GitNewBranch) Added feature.txt in GitNewBranch
```

```
P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git config --global merge.tool p4merge

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git config --global mergetool.p4merge.cmd "C:/Program Files/Perforce/p4merge.exe"

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git difftool GitNewBranch

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git merge GitNewBranch
Already up to date.

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git log --oneline --graph --decorate
* f807c0d (HEAD -> master, GitNewBranch) Added feature.txt in GitNewBranch
* 2eb202d Initial commit on master

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git branch -d GitNewBranch
Deleted branch GitNewBranch (was f807c0d).

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git status
On branch master
nothing to commit, working tree clean

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
```

```
P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git commit -m "Added hello.xml in master with different content"
[master 6e4af0d] Added hello.xml in master with different content
1 file changed, 1 insertion(+)
```

```
P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git merge GitWork
Auto-merging hello.xml
CONFLICT (content): Merge conflict in hello.xml
Automatic merge failed; fix conflicts and then commit the result.
```

```
P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master|MERGING)
$ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)
```

```
Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both added:      hello.xml
```

```
no changes added to commit (use "git add" and/or "git commit -a")
```

```
P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (GitWork)
$ git add hello.xml
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (GitWork)
$ git commit -m "Added hello.xml in GitWork branch"
[GitWork 0a0a424] Added hello.xml in GitWork branch
1 file changed, 1 insertion(+)
create mode 100644 hello.xml

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (GitWork)
$ git checkout master
Switched to branch 'master'

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ echo "<message>Hello from Master</message>" > hello.xml

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git add hello.xml
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git commit -m "Added hello.xml in master with different content"
[master 6e4af0d] Added hello.xml in master with different content
1 file changed, 1 insertion(+)
```

## 4. Git - HOL :

```
P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git status
On branch master
nothing to commit, working tree clean

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git init
Reinitialized existing Git repository in E:/GitBranchDemo/.git/

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ echo "Hello Git" > hello.txt

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git add hello.txt
warning: in the working copy of 'hello.txt', LF will be replaced by CRLF the next

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git commit -m "Initial commit"
[master 902f3c1] Initial commit
 1 file changed, 1 insertion(+)
 create mode 100644 hello.txt

P2r0i0y0a4@PHOENIX-07 MINGW64 /e
$ cd GitBranchDemo

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git status
On branch master
nothing to commit, working tree clean

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git checkout -b GitWork
Switched to a new branch 'GitWork'
```



```
P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (GitWork)
$ 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)

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (GitWork)
$ git add hello.xml
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git touches

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (GitWork)
$ git commit -m "Added hello.xml in GitWork branch"
[GitWork 8a0a424] Added hello.xml in GitWork branch
 1 file changed, 1 insertion(+)
 create mode 100644 hello.xml

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (GitWork)
$ git checkout master
Switched to branch 'master'

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ echo "<message>Hello from Master</message>" > hello.xml

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git add hello.xml
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git touches
```

```

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git commit -m "Added hello.xml in master with different content"
[master 6e4af0d] Added hello.xml in master with different content
1 file changed, 1 insertion(+)

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git log --oneline --graph --decorate --all
* 6e4af0d (HEAD -> master) Added hello.xml in master with different content
| * 0a0a424 (GitWork) Added hello.xml in GitWork branch
|/
* f807c0d Initial commit
* 2eb202d Initial commit on master

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git merge GitWork
Auto-merging hello.xml
CONFLICT (content): Merge conflict in hello.xml
Automatic merge failed; fix conflicts and then commit the result.

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master|MERGING)
$ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both added:      hello.xml

no changes added to commit (use "git add" and/or "git commit -a")

```

```

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master|MERGING)
$ git diff GitWork
diff --git a/hello.xml b/hello.xml
index 84f3217..f180b98 100644
--- a/hello.xml
+++ b/hello.xml
@@ -1,5 @@
+<<<<<<< HEAD
<message>Hello from Master</message>
=====
<message>Hello from GitWork</message>
>>>>>> GitWork

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master|MERGING)
$ git add hello.xml

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master|MERGING)
$ git commit -m "Resolved merge conflict manually"
[master 3e21d0b] Resolved merge conflict manually

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ echo "*.orig" > .gitignore

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time
you commit, because the file has been edited by hand.

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git commit -m "Added .orig pattern to .gitignore"
[master 7fd1429] Added .orig pattern to .gitignore
1 file changed, 1 insertion(+)
create mode 100644 .gitignore

```

```
P2r8i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master|MERGING)
$ git commit -m "Resolved merge conflict manually"
[master 3e21d0b] Resolved merge conflict manually

P2r8i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ echo "*.orig" > .gitignore

P2r8i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time you commit

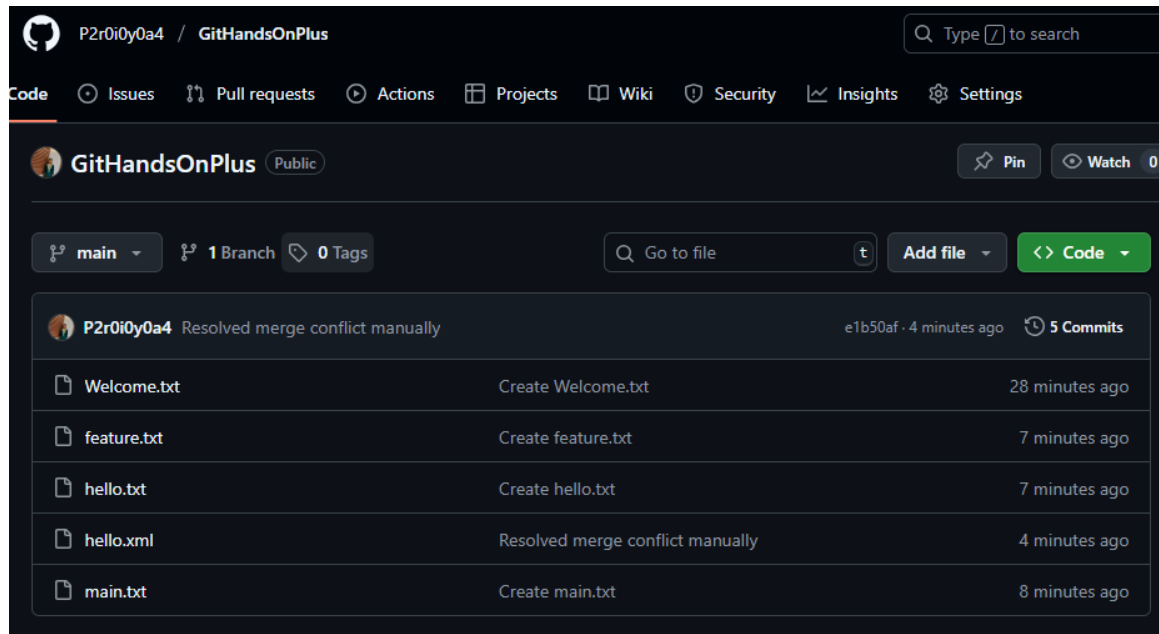
P2r8i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git commit -m "Added .orig pattern to .gitignore"
[master 7fd1429] Added .orig pattern to .gitignore
1 file changed, 1 insertion(+)
create mode 100644 .gitignore

P2r8i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git branch
  GitWork
* master

P2r8i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git branch -d GitWork
Deleted branch GitWork (was 0a0a424).

P2r8i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git log --oneline --graph --decorate
* 7fd1429 (HEAD -> master) Added .orig pattern to .gitignore
* 3e21d0b Resolved merge conflict manually
* 0a0a424 Added hello.xml in GitWork branch
* 6e4af0d Added hello.xml in master with different content
* f807c0d Initial commit
* 2eb202d Initial commit on master
```

## 5.Git – HOL :



P2r0i0y0a4 / GitHandsOnPlus

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

GitHandsOnPlus Public Pin Watch 0

main 1 Branch 0 Tags Go to file Add file Code

P2r0i0y0a4 Resolved merge conflict manually e1b50af · 4 minutes ago 5 Commits

Welcome.txt	Create Welcome.txt	28 minutes ago
feature.txt	Create feature.txt	7 minutes ago
hello.txt	Create hello.txt	7 minutes ago
hello.xml	Resolved merge conflict manually	4 minutes ago
main.txt	Create main.txt	8 minutes ago

```
P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git remote add origin https://github.com/P2r0i0y0a4/githandson.git

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git push -u origin master
Enumerating objects: 21, done.
Counting objects: 100% (21/21), done.
Delta compression using up to 12 threads
Compressing objects: 100% (14/14), done.
Writing objects: 100% (21/21), 1.75 KiB | 179.00 KiB/s, done.
Total 21 (delta 6), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (6/6), done.
To https://github.com/P2r0i0y0a4/githandson.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git remote -v
origin https://github.com/P2r0i0y0a4/githandson.git (fetch)
origin https://github.com/P2r0i0y0a4/githandson.git (push)

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
$ git checkout master
Already on 'master'
Your branch is up to date with 'origin/master'.

P2r0i0y0a4@PHOENIX-07 MINGW64 /e/GitBranchDemo (master)
```