Class Task 1

Lab 1: Git Installation & Setup

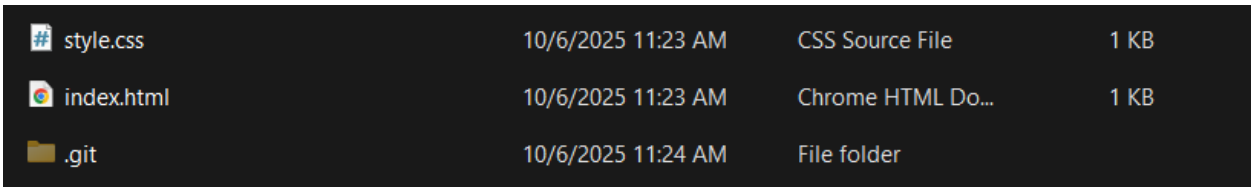
1. **git –version**
2. Git Installed
3. Command for Configure –
   git config --global user.name "Sandip Magart"

   git config --global user.email thapasandip080@gmail.com
4. **Collaboration, Version history, Syncing with remote repositories**
5. **git config --list**

```
PS C:\Users\N I T R O 5> git config --list
diff.astextplain.textconv=astextplain
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
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/etc/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
filter.lfs.required=true
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
user.name=Sandip Magar
user.email=thapasandip080@gmail.com
PS C:\Users\N I T R O 5>
```

Lab 2: Create a Local Website Project

1. 

2. 

3. Index.html

```html
<!DOCTYPE html>
<html>
<head>
    <title>My Website</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <h1>Welcome to My Website</h1>
</body>
</html>
```

4. Style.css

```css
.heading {
    color: blue;
    text-align: center;
}
```

Output:

5. 

Lab 3: Initialize Git Repository

1. **cd path\to\my\project** (Eg: CreativeCollab>>frontend - C:\Users\N I T R O 5\creative-collab\frontend>)
2. **git init** is used to Initialize Git - Creates a hidden .git folder to track version history.
3. **git status -** Shows untracked files (new HTML and CSS files).
4.

```
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   index.html
        new file:   style.css
```

Lab 4: Add and Commit Files

1. The **staging** area in Git is where one prepares and reviews changes before committing them.

2. 3.

```
PS C:\Users\N I T R O 5\OneDrive\Desktop\my-website> git add index.html
PS C:\Users\N I T R O 5\OneDrive\Desktop\my-website> git add style.css
```

```
PS C:\Users\N I T R O 5\OneDrive\Desktop\my-website> git commit -m "Add index.html with welcome heading"
 2 files changed, 14 insertions(+)
 create mode 100644 index.html
 create mode 100644 style.css
```

4.

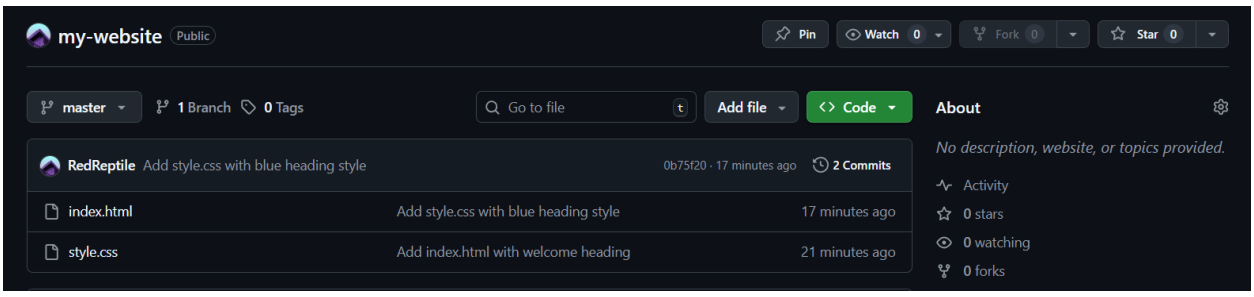
```
PS C:\Users\N I T R O 5\OneDrive\Desktop\my-website> git commit -m "Add style.css with blue heading style"
[master 0b75f20] Add style.css with blue heading style
 1 file changed, 1 insertion(+), 1 deletion(-)
```

5. 6.

```
PS C:\Users\N I T R O 5\OneDrive\Desktop\my-website> git log
commit 0b75f20eccfce9f633ba26d0e7eaec081b98d356 (HEAD -> master)
Author: Sandip Magar <thapasandip080@gmail.com>
Date:   Mon Oct 6 09:58:37 2025 +0545

    Add style.css with blue heading style

commit 0bf7399d8ad6ee884bfd118d9809497a4bc477aa
Author: Sandip Magar <thapasandip080@gmail.com>
Date:   Mon Oct 6 09:54:48 2025 +0545

    Add index.html with welcome heading
```

Lab 5: Create GitHub Repository and Push Code

1. Already Created

2. 

git push -u origin master - pushes your local master branch to origin and sets it as the default upstream for easier future pushes.

Lab 6: Create and Switch Branches

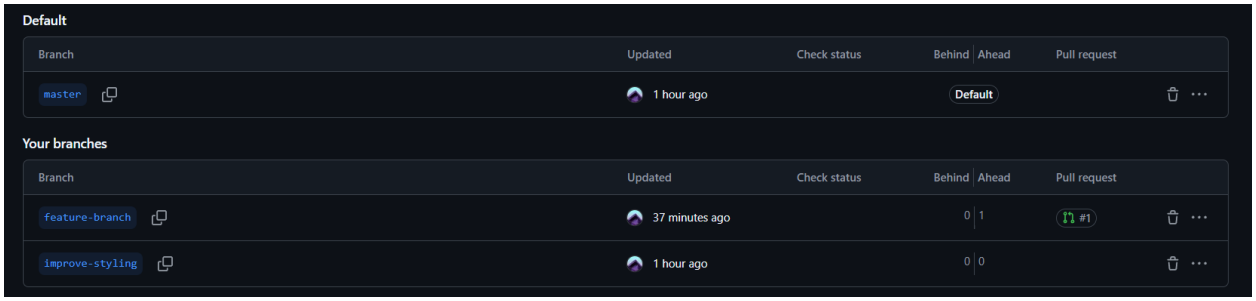
1. A branch in Git is a separate line of development that lets us work on changes independently.
2. Created
3. **git branch –** View All Branch

```
PS C:\Users\N I T R O 5\OneDrive\Desktop\my-website> git branch
* improve-styling
  master
```

4.

```
body {
    background-color: lightgrey;
}

h1 {
    color: blue;
    text-align: center;
}
```

5. Completed
6. Completed
7.



Lab 7: Merge Branches

1. The purpose of merging in Git is to **combine changes from one branch into another**.
2. **git checkout master -**

```
PS C:\Users\N I T R O 5\OneDrive\Desktop\my-website> git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
```

3.

```
PS C:\Users\N I T R O 5\OneDrive\Desktop\my-website> git merge improve-styling
Updating 0b75f20..7664e85
Fast-forward
 style.css | 8 ++++++--
 1 file changed, 6 insertions(+), 2 deletions(-)
PS C:\Users\N I T R O 5\OneDrive\Desktop\my-website> git push
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/RedReptile/my-website.git
   0b75f20..7664e85  master -> master
```

4. Pushed to master branch

5. After merging, the feature branch still exists until you delete it.

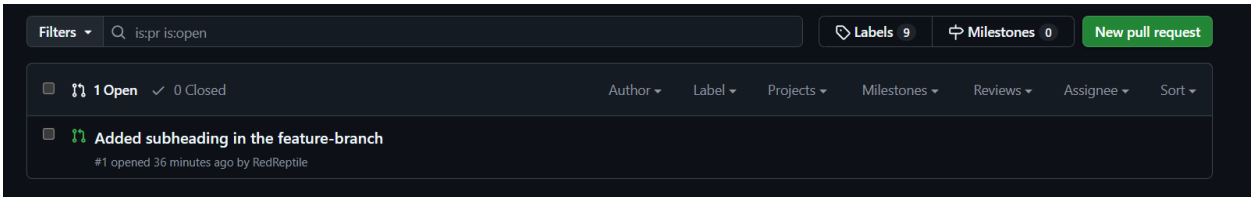
6.

style.css +6 -2

```
@@ -1,4 +1,8 @@
- .heading {
+ body {
+     background-color: lightgrey;
+ }
+
+ h1 {
      color: blue;
      text-align: center;
- }
+ }
```

Lab 8: Collaboration Simulation

1. Forking a repository means making your own copy on GitHub.
2. Forked the repository
3. Clone the forced repository



Lab 9: Reflection Task

1. Git is a tool that **tracks changes in code** and helps teams **collaborate safely and efficiently**.
2. **git add –** it stages changes for commit
   git commit – it saves staged changes with a message
   pit push – it uploads commits to remote repositories
3. **Branch** is the independent version of code it is used to work on features without affecting the master.
4. To see the commit history, we use **git log**
5. Learning how to merge changes without causing conflicts.
6. Learned to manage code versions and collaborate efficiently using Git and GitHub.