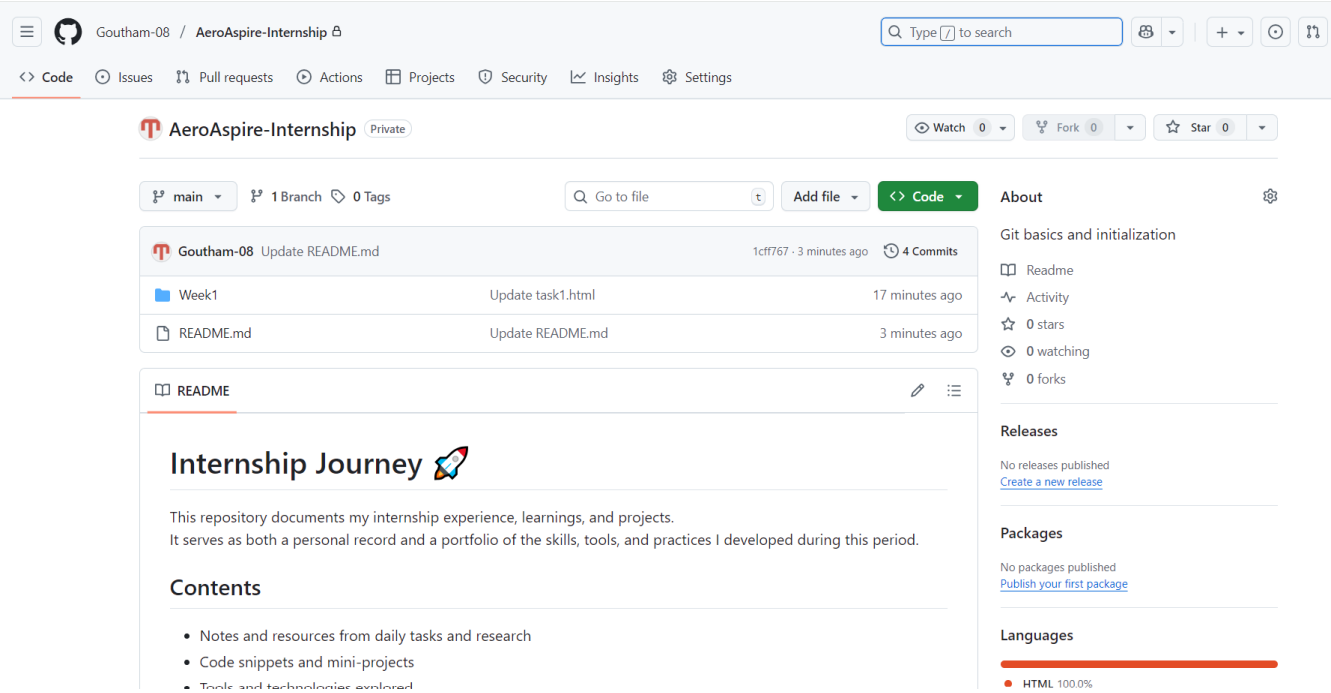
AeroAspire-SDE Intern

Goutham V

Week 1 – Day4 (25rd September)

Task:

Initialize repo; commit daily work; create feature branch; merge after review



I began by initializing a new Git repository inside my project folder using git init, which set up Git version control locally. After preparing the files by adding them to the staging area with git add ., I made my first complete commit titled "initial commit" to capture the initial state of my code. I renamed the branch to "main" using git branch -M main to follow modern naming conventions. Next, I connected the local repository to my GitHub remote repository with git remote add origin <repo-url>. Finally, I pushed my commits to the remote repository using git push -u origin main, successfully syncing my work online.

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire
$ git init
Initialized empty Git repository in C:/Users/Admin/Documents/AeroAspire/.git/

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (master)
$ git commit -m "first commit"
On branch master

Initial commit

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        Week1/

nothing added to commit but untracked files present (use "git add" to track)

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (master)
$ git config --global user.name "your-github-username"

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (master)
$ ^C

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (master)
$ git config --global user.name "Goutham-08"

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (master)
$ git config --global user.email "goutham000333@gmail.com"

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (master)
$ git init
Reinitialized existing Git repository in C:/Users/Admin/Documents/AeroAspire/.git/

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (master)
$ git commit -m "initial commit"
On branch master

Initial commit

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

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Reinitialized existing Git repository in C:/Users/Admin/Documents/AeroAspire/.git/

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (master)
$ git commit -m "initial commit"

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        Week1/

nothing added to commit but untracked files present (use "git add" to track)

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (master)
$ git add .

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (master)
$ git commit -m "initial commit"
[master (root-commit) 43c4b48] initial commit
 5 files changed, 398 insertions(+)
 create mode 100644 Week1/Day1/Goutham_HTML & CSS Basics(pdf).pdf
 create mode 100644 Week1/Day1/flexbox.html
 create mode 100644 Week1/Day1/task1.html
 create mode 100644 Week1/Day2/Day2_GouthamV_week1 (2).pdf
 create mode 100644 Week1/Day2/jscript.html

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (master)
$ git branch -M main

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (main)
$ git remote add origin https://github.com/Goutham-08/AeroAspire-Internship.git

Admin@DESKTOP-DNUBHS4 MINGW64 ~/Documents/AeroAspire (main)
$ git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (10/10), 1.31 MiB | 584.00 KiB/s, done.
Total 10 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/Goutham-08/AeroAspire-Internship.git
```

Before committing my changes, I configured Git with my user name and email using git config --global user.name and git config --global user.email commands. This ensures all commits carry my identity information. I reinitialized the Git repository to confirm proper setup and then staged the project files using git add .. After the initial successful commit named "initial commit," I renamed the branch to "main." I also linked my local repository to my GitHub remote repository using git remote add origin and pushed my commits to the remote with git push -u origin main. This completed syncing the local work to GitHub securely and correctly.

Reflection,

1. What is the workflow from making changes → staging → commit → push?

- **Making Changes:** Modify files in your local working directory.
- **Staging:** Add the changed files to the staging area using git add to prepare them for commit.
- **Commit:** Save the staged changes to the local Git repository with git commit, creating a snapshot with a descriptive message.
- **Push:** Send your local commits to the remote repository (like GitHub) using git push to sync your changes with others.

2. What is a merge conflict: what causes it, and how do you resolve it?

- A merge conflict occurs when Git cannot automatically merge changes from different branches because the same lines in the same file were edited differently.
- Conflicts can arise when:
  - Two branches modify the same line in a file.
  - One branch deletes a file that another branch modifies.
- To resolve:
  - Git highlights conflict areas in the files.
  - Manually edit these files to reconcile the differences.
  - Stage the resolved files using `git add`.
  - Finalize the merge with `git commit`.

3. Describe what happens under the hood with git commit: what objects are stored? (briefly)

- During a commit, Git creates and stores:
  - **Blob objects** that hold the content of each file.
  - **Tree objects** that track the directory structure and link to blobs and other trees.
  - **Commit objects** that point to a tree, store metadata (author, date, message), and reference parent commits.
- This forms an immutable snapshot of the project at that point in time, enabling version tracking.