

A screenshot of a code editor interface. The top bar shows the title 'DAB111_Final_Project'. The left sidebar lists files: 'ORS', 'Welcome', 'README.md', 'ADME.md', and 'he_'. The main area has tabs for 'Welcome' and 'README.md'. The 'README.md' tab contains the following text:

```
1 # DAB111 Final Project - Group Web Page
2
3 This project showcases our group's work for the DAB111 course. It includes a Flask-based web application, organized folder structure, and a professional presentation of our team and project goals.
4
```

The bottom part of the editor shows a terminal window with the command '\$ git status' and its output:

```
$ git status
Your branch is up to date with 'origin/main'.
nothing to commit, working tree clean
(.venv)
Mulugeta A@DESKTOP-28612CP MINGW64 ~/Documents/Data Analytics-Fall_2025/Documents_Introduction to Python-Note&lab/DAB111_Final_Project (main)
```

DAB111 FINAL PROJECT - TEAM COLLABORATION SUMMARY

https://github.com/Mulugeta972/DAB111_Final_Project/tree/DAB111-Introduction%20to%20Python%20Programming

Instructor: Aishwarya Rajasekaran

Instructor: Keval Patel

By: Wubu Nuerbiya - #0873194

: Mulugeta Negussie - #0599492

DAB111 Final Project – Team Collaboration Summary

Project Overview

This final project was developed for the DAB111 course (Fall 2025) and showcases collaborative work between two team members: **Mulugeta** and **Nurbiya**. The project includes a structured GitHub repository, a Flask-based web application, and supporting documentation. The primary goal was to demonstrate teamwork, technical organization, and presentation — not to analyze or manipulate the dataset itself.

Team Members and Roles

◆ Nurbiya

- Collaborated in setting up the GitHub repository and initial folder structure
- Followed the instructor's guide to help organize the project layout
- Contributed to the homepage and Results page content
- Provided feedback on formatting and accessibility
- Made commits to GitHub reflecting her contributions
- Participated in the decision to replace the original dataset with a smaller open-source dataset
- Processed and pushed the new dataset to GitHub

◆ Mulugeta

- Led the creation and refinement of the GitHub repository and folder structure
- Worked jointly on the first three folders (Website, docs, notebooks) with Nurbiya
- Independently developed and organized the remaining folders (data collection, data processing, database, src)
- Authored the data cleaning notebook and internal documentation
- Managed Git workflow, commits, and file organization
- Finalized the Results page and integrated notebook links
- Participated in the dataset replacement decision and ensured compatibility across the project

Repository Structure

The project is organized into clearly defined folders:

- Website/ – Contains the Flask web app and Results page
- docs/ – Stores internal documentation and team summaries
- notebooks/ – Includes the data cleaning notebook
- data collection/, data processing/, database/, src/ – Support backend and data handling

All work is version-controlled using Git and published to GitHub for transparency and reproducibility.

Supporting Materials

The following visuals are included in the final submission:

- Screenshot of the GitHub repository structure

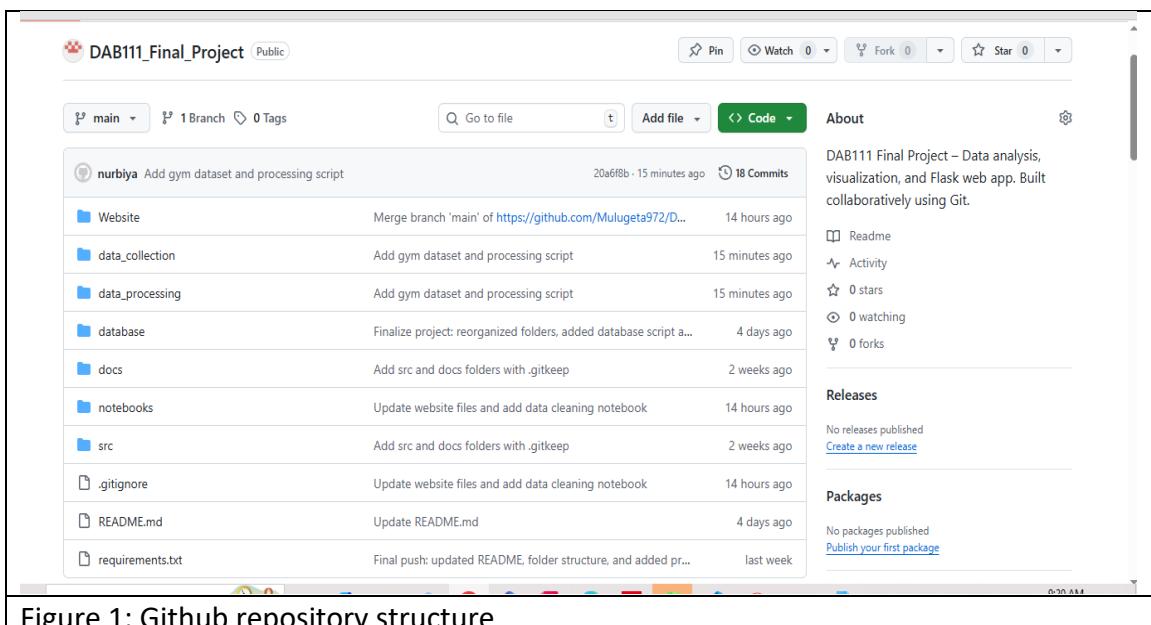


Figure 1: Github repository structure

- Screenshot of the homepage with team introduction

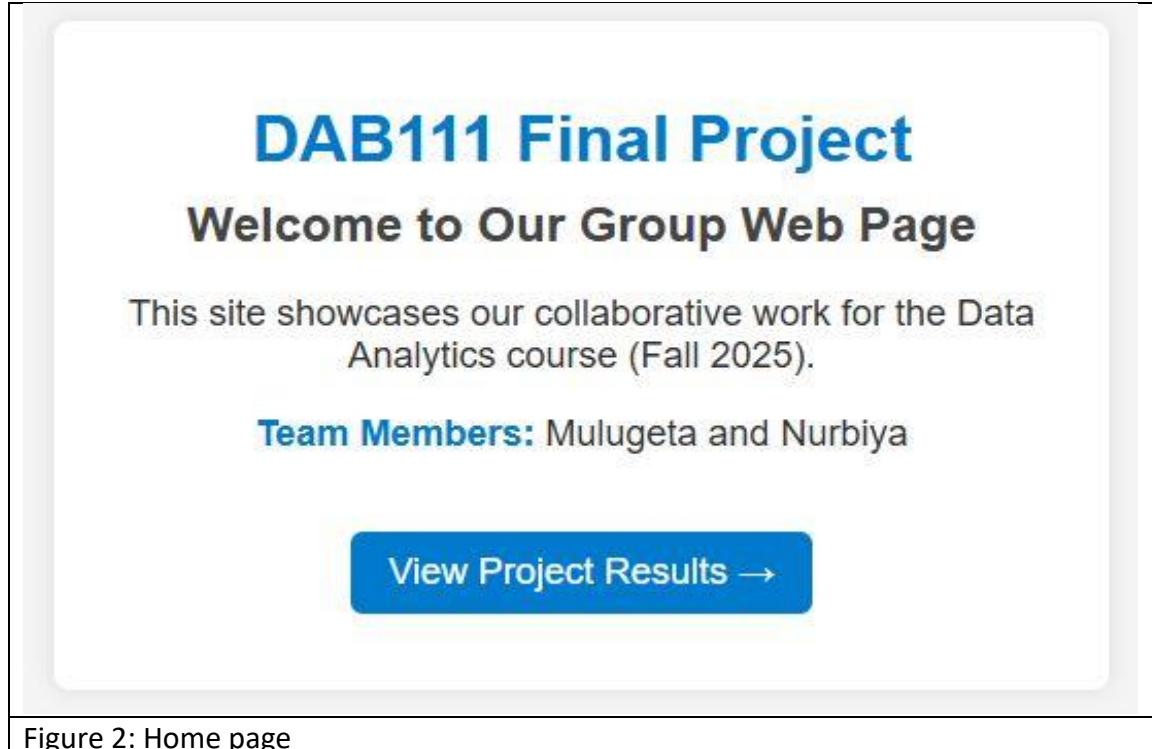


Figure 2: Home page

- Screenshot of the Results page summary

Project Results

This page summarizes the outcomes of our final project for DAB111: Introduction to Python Programming.

Project Overview

We built a Flask-based web application that collects, stores, and presents data using Python and SQLite. The project demonstrates our understanding of core programming concepts, database integration, and web development.

Key Components

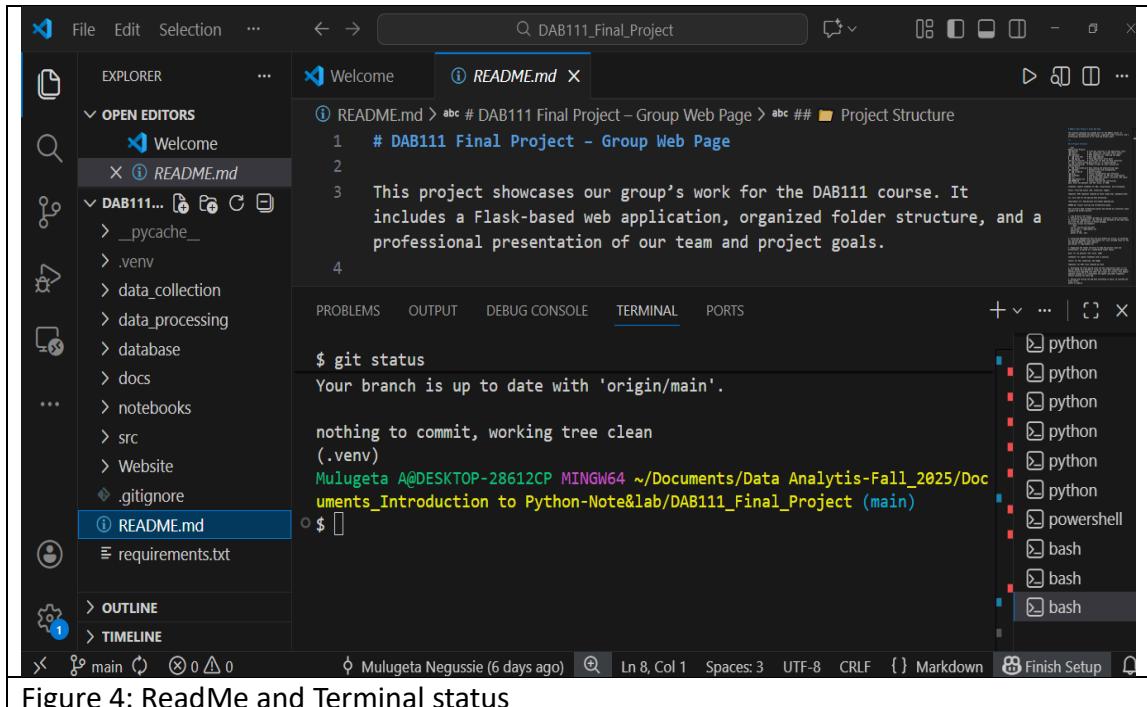
- **Data Collection:** We selected a dataset with multiple variables and data types, suitable for tabular storage.
- **Database:** We created a SQLite database with one table and inserted records using Python's `sqlite3` package.
- **Web Application:** We used Flask to build a website with a homepage, results page, and data display functionality.
- **Project Structure:** Our code is organized into folders for data, notebooks, scripts, templates, and static assets.
- **Version Control:** We used Git and GitHub to collaborate and track changes throughout development.

Learning Outcomes

- How to structure a Python project professionally
- How to use Flask for routing and template rendering
- ~~How to create and query a SQLite database using Python~~

Figure 3: Project Result Page

- Screenshot of the README and terminal status



These images provide a visual summary of the project's structure and presentation.

Final Notes

This document reflects the collaborative effort and division of responsibilities between team members. The project emphasizes clarity, reproducibility, and professional organization. All materials are available in the GitHub repository and linked through the Results page.