Task 1.1

GitHub Repository: https://github.com/amos-py/DAT101.git

Discussion Questions:

1. IDE (Visual Studio Code): What are some of the key features of VS Code that users find most helpful for coding? Research how VS Code is compared to other text editors or IDEs?

1.1 - VSCode features things like extensions, autocomplete / AI integration, an integrated terminal, git, and other quality of life features which can make coding more efficient / intuitive.

2. GitHub: In your own words, explain the concept of version control and why it's important in software development. How can GitHub facilitate collaboration and teamwork?

2.1 - Version control is important for creating and tracking backups / previous versions of projects in case (e.g.) an experimental feature should break the current version. One will then be able to roll back to a previous "save". GitHub encourages collaboration / teamwork through features like push, pull, commit, merge etc. which allow multiple people to work on the same project.

3. Folder Structure: Why is a well-organised folder structure crucial for managing programming projects? What potential issues could arise from a poorly structured project?

3.1 - A well-organized folder structure will encourage further organization due to an already established structure, make the files easily findable / accessible both for yourself and the people working on the same projects. Issues from a poorly structured project could range from people not finding the files they are looking for / increasing the time to find the files.

4. Naming Conventions: Why are consistent naming conventions for files and folders important? What are some common best practices for naming files and folders in a programming context?

4.1 - Consistently naming your files / folders in a similar structure will make the files easier to find a developed project, increase the understanding of what exactly the files are for /

their purpose / contents. An example of a good practice for naming files / folders could be to name the files based on their exact purpose / contents. A shared word in the beginning of the file will make sorting / organizing easier as well.

5. Collaboration Challenges: What are some of the potential challenges or misunderstandings that can arise when working on code projects collaboratively? How can these challenges be addressed or mitigated?

5.1 - Some challenges when collaborating on code projects could be different habits of writing code, organizing files / folders. There are also challenges with pushing / pulling where two people could be changing the same code contents, known as a conflict / merge conflict. The parties involved will then have to agree on what will be committed.

6. Best Practices: Based on your group discussion, create a list of 3-5 best practices for using VS Code, GitHub, and organising code projects.

6.1 - Some of the best coding practices could be:

- (When working together with others) agree on a shared file / folder structure and actively work on maintaing this created order.
- Take advantage of VSCode extensions which can improve things like quality of life when coding and general readability / legibility.
- Try to write concise messages with your pushes in GitHub, so version control and potential rollbacks will be easy to keep track of / find.

7. Real-World Application: Can you think of any real-world scenarios where the skills you've learned in Module 1 (using VS Code, GitHub, folder structure, naming conventions) would be directly applicable?

7.1 - These mentioned skills could be useful in other digital arenas (work / school) one actively works on / changes files and folders. Learning how coding (using IDE's like VSCode) works and the sharing / collaboration of this code (GitHub) allows one to maintain good previous habits within e.g. folder structure / naming conventions.