**Lab Reflection Journal – GitHub & Jupyter Notebook**

a. What I Did

During this lab session, I was introduced to two fundamental tools used in modern programming and data science workflows: GitHub and JupyterNotebook.

The session began with the creation of a GitHub account. I went to [github.com](https://github.com), selected a unique username (FranckHCC), and registered with my email address and a secure password. After email verification, I logged into my new account and explored the user dashboard. Following the professor's guide, I created a new repository, added a description, selected the option to include a README file. This process helped me understand how projects are stored and tracked online.

Following that, I moved on to Jupyter Notebook, which I accessed through an online environment such as Google Colab. I created a new .ipynb notebook file and began exploring its features. I learned how to:

* Create and run code cells : the Python code was print("Hello, World!") was
* Add Markdown cells for comments and explanations
* Perform basic data manipulations, print outputs, and display plots

We practiced saving our work and uploading the .ipynb file to our GitHub repository. This was my first hands-on experience with interactive computing with version control, giving me insight into real-world coding workflows. I always had difficulties in other classes but now I am happy for what I learned.

b. What I Learned

This lab introduced me to several key tools and concepts:

Version Control with GitHub

Before this lab, I had only heard of GitHub. Now, I understand how it's used for tracking code changes, collaborating with others, and storing projects in the cloud. Creating repositories, making commits, and uploading files made the concept of version control more concrete.

One of the most important takeaways was the idea that every version of a project is saved, so you can return to earlier stages if needed. I also learned about the importance of using README files to document project goals and setup instructions, which is essential for collaboration.

Interactive Coding with Jupyter Notebook

Jupyter Notebook was not completely new to me. I indeed used it before, but without really understanding anything. Today I learned that it’s a powerful environment for writing and testing Python code interactively. I appreciated the ability to mix code with explanations, which is perfect for data analysis, machine learning, and educational projects.

I have been writing small Python scripts, creating lists and arrays, plotting simple graphs, without knowing that I could share the link online. The immediate feedback from running code cells helped me better understand the results of each command.

Challenges

Initially, I found the GitHub interface a bit confusing, especially understanding the difference between committing changes, pushing files, and cloning repositories. It took some time with many tries, but after following step by step the professor’s guide, I got more comfortable with the workflow.

In Jupyter Notebook, I accidentally deleted a cell once and wasn’t sure how to undo it. I later discovered the undo option under the "Edit" menu, which helped recover my work.

c. Questions or Comments

I enjoyed this lab session. It felt empowering to create a GitHub repository and upload code like a real developer. One suggestion I would make is to provide a visual checklist of steps for both GitHub and Jupyter Notebook.

One question I still have is: after this class, can someone make money using GitHub, and how? Are we going to develop major projects after this class?

My Link

https://github.com/Franckhcc/jupyter-exploration.git