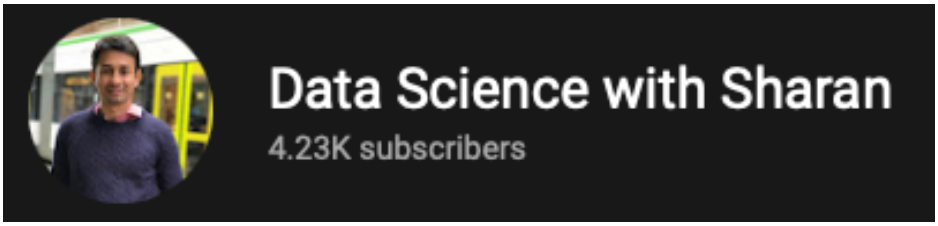


Day 1 - Learning about the popular IDE

We will mostly be using the Jupiter Notebook mostly in this journey. It is important to understand this tool and to best use it. While working on a project it is important to ensure the documentation is clear enough. Jupyter Notebook is not only easy to use but it is also easy to make documentation and presentations.

The first step is to set the environment. Jupyter Notebook is one of the best tools for data science programming. Here are the steps to install Anaconda.

To know about the basic command and getting started with Anaconda. Here is a simple tutorial
Creating an environment is very important while working on a project. There could be different requirements for different projects. There are could be projects where you might have to use a library that is of an older version. This can be made possible using different environments.
Learn to create a new environment using this link here.

Also, here is an article about using Jupyter Notebook. It clearly explains the common functionalities of Jupiter. If you are looking for a cheat sheet, here is one.

Some things to practice are,

- Install Anaconda, it should take care of the installation of Python and Jupiter Notebook
- Create a new environment (On creating a new environment remember to install ipykernel and Jupyter)

If you are looking for a tutorial video. Check out here. I have walked through the installation process.

For Success:

Please remember, to make the best use of this platform. Please share your progress with others.

To give some idea you can talk about some of the below,

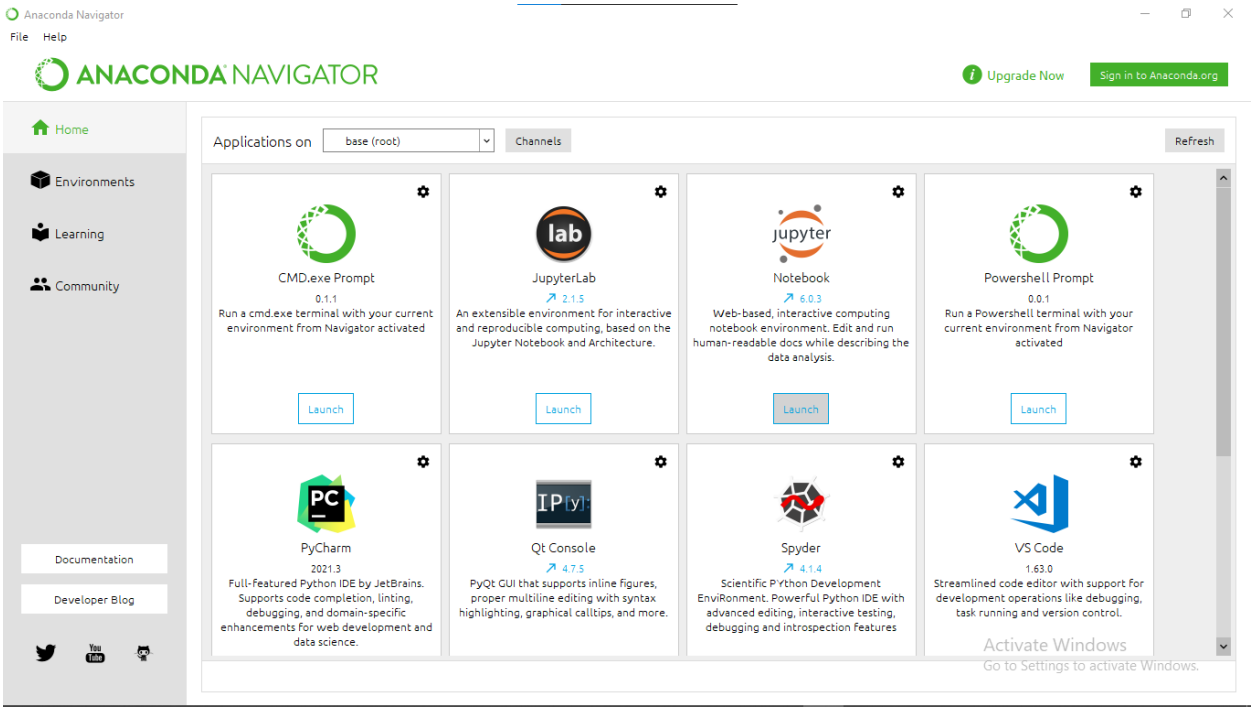
- How easy or difficult it was for you?

- Were there any issues? How did you solve it?

Also, have a Git repository to track and save your work as you learn.

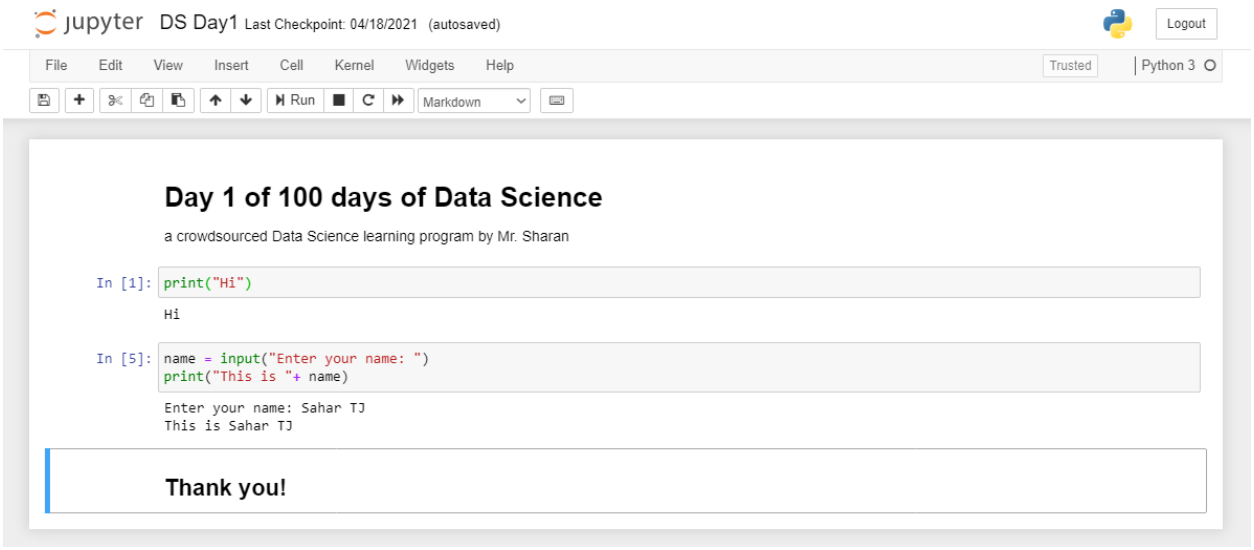
Day 1 - Learning about the popular IDE

In this task we learn how to install Anaconda and use it. Here's a shot how the environment looks like:



We then launch Jupyter and create a new notebook in any folder on the device. It's a Web-based notebook environment.

The Jupyter notebook uses Markdown and Python code.



Some other popularly used IDEs

- **Spyder :** Scientific Python Development Environment (Spyder) is an open-source, cross-platform IDE for Data Science. The IDEs essential building blocks, include advanced editing, code analytical tools, IPython Console, variable explorer, plots, debugger and the help icon, which makes Spyder an ideal choice for data scientists.

- **Pycharm :** PyCharm is an IDE for professional developers and data scientists. It has intelligent coding assistance that allows for smart code completion, code inspections, on-the-fly error highlighting and quick fixes, along with automated code refactorings and rich navigation capabilities.

- **Visual Studio code :** VS Code is one of the most used Python IDEs. The IDE is known for its tools such as IntelliSense that allows features beyond syntax highlighting and smart completions based on variable types, imported modules, and functions definition. In addition, VS Code allows debugging code right from the editor with breakpoints, call stacks and an interactive console. Furthermore, VS Code is extensible and customisable, allowing for the addition of new languages, themes, and debuggers. The IDE also has built-in Git commands. VS Code is available in free and paid versions.

- **Atom:** Atom is a formidable IDE for ML & DS professionals that supports many languages other than Python, such as C, C++, HTML, JavaScript, etc. The IDE includes features such as cross-platform editing, built-in package manager, smart autocompletion, file system browser, and multiple panes. Moreover, its plugins, languages, libraries, and tools are constantly updated, resulting in the Atom interface and experience being customisable and outstanding.