**IBM Data Science Coursera Certificate – Tools for Data Science (Course 2)**

**Week 4**

**Notes:**

**Jupyter Notebooks**

1. Originated as iPython initially.
2. Stands for:

* Julia
* Python
* R

1. A Jupyter Notebook is a browser-based application that allows you to create and share documents containing code, equations, visualizations, narrative text links, and more.
2. It is like a scientist’s lab notebook, where a scientist records all steps to perform their experiments and the results they can reproduce.
3. In the same way, a Jupyter Notebook allows a Data Scientist to record their data experiments and results that others can reuse.
4. Now a Jupyter Notebook file allows you to combine descriptive text, code blocks, and code output in a single file. When you run the code, it generates the output, including plots and tables, within the notebook file. And you can then export the notebook to a PDF or HTML file format that can then be shared with anyone

**JupyterLab**

1. Jupyter Lab is a browser-based application that allows you to access multiple Jupyter Notebook files, other code, and data files.
2. In addition, it extends the functionalities of Jupyter Notebooks by enabling you to work with **multiple notebooks**, text editors, terminals, and custom components in a flexible, integrated, and extensible manner. It is compatible with several file formats like CSV, JSON, PDF, Vega, and so on.
3. Jupyter Notebooks and JypterLab are an open source browser based applications.
4. Jupyter Notebooks can be used with cloud-based services like IBM and Google Colab.
5. They don't require any installation on your local machine.
6. They give you access to the Jupyter Notebook environment and allow you to import and export notebooks using the standard IPython Notebook file format. Also, these services support the Python language and other languages as well.

**Jupyter Kernels**

1. A notebook kernel is a computational engine that executes the code contained in a Notebook file
2. Jupyter Kernels, for many languages, exist.
3. When a Notebook document opens, the related kernel launches automatically.
4. When the Notebook is executed, the kernel performs the computation and produces the results. Depending on your settings, you may need to install other notebook languages in your Jupyter environment.
5. **Python kernel allows you to run python cells**. You can run the Python script in the cells to produce an output. The top right corner of the Notebook shows the name of the kernel. Here it shows the Python kernel.

**Jupyter Architecture**

1. Jupyter architecture implements a two-process model with a kernel and a client. The client is the interface offering the user the ability to send code to the kernel. It is the browser in a Jupyter Notebook.
2. The kernel executes the code and returns the result to the client for display. Jupyter Notebooks represent your code, metadata, contents, and outputs.
3. When you save the Notebook, it is sent from your browser to the Notebook server. It saves the notebook file on a disk as a JSON file with a .ipynb extension.
4. The Notebook server is responsible for saving and loading the notebooks. And the kernel executes the cells of code contained in the Notebook when the user runs them.
5. The Jupyter architecture uses the NBconvert tool to convert files to other formats. For example, if we want to convert a notebook file into an HTML file, the notebook is first modified by a preprocessor, then an exporter converts the notebook to the new file format. Finally, a postprocessor will work on the exported file to give the final output. After conversion, on giving the url of the file, the HTML file displays.

**Anaconda Jupyter**

1. **Computational Notebooks :** These notebookscombine codes, computational output, explanatory text, and multimedia resources into a single document.
2. Jupyter Notebook is a popular type of computational notebook because it supports dozens of programming languages.
3. JupyterLab and VS Code are popular environments for creating and modifying Jupyter Notebooks on a local device.
4. JupyterLab is an open-source, web-based application based on Jupyter Notebook. You can create code, interactive visualizations, text, and equations, just like with Jupyter Notebook.
5. JupyterLab includes expanded features with some of Anaconda's most extensive pre-installed Python libraries, including NumPy, Pandas, and Matplotlib.
6. **Anaconda** is a free and open-source distributor for Python and R, the top languages used in data science and machine learning. Anaconda has fifteen hundred plus libraries.

**VS Code (VisualStudio Code)**

1. It is a free, open-source code editor for operations like debugging and task running.
2. It works on Linux, Windows, and MacOS.
3. It also supports
   1. Multiple languages
   2. Syntax highlighting
   3. And auto-indentation
4. One of the most popular development environment tools

**Cloud-based Jupyter Environments**

1. Jupyter notebooks are the most popular computational notebooks. it supports dozens of programming languages. ​
2. Popular cloud-based environments used to create and modify Jupyter notebooks include:​
   1. **JupyterLite,**
   2. **and​ Google Collaboratory (Google Collab)**
3. JupyterLite is a lightweight cloud-based tool built from JupyterLab components ​that executes entirely in the browser.​
4. JupyterLite does not require a dedicated Jupyter server. ​ Only a web server is required, which means ​we can deploy JupyerLite as a static website.
5. JupyterLite can be used to create interactive graphics and visualizations because it supports many visualization libraries like Altair, Plotly, and ipywidgets​.
6. JupyterLite is a distribution of JupyterLab, ​it includes the latest improvements and features.​
7. **For cloud based Jupyter environments**, Python Pyodide and Python Pyolite are common kernels.
8. The default kernel for JupyterLite is​ Pyolite. ​ Pyolite is a Python kernel based on Pyodide. Pyolite runs in the background, so that intensive computations can execute quickly. ​ Other kernels can also be used with JupyterLite.
9. **Google Colaboratory (or 'GoogleColab')** is a free Jupyter notebook environment that runs entirely in the cloud.​
10. GoogleColab Jupyter notebooks execute on a browser, and Google Collab projects are stored on Google Drive and GitHub. ​
11. You can upload and share notebooks without setup and installation.​ You can also clone projects from GitHub and execute them in GoogleCollab.​
12. Most machine learning and visualization libraries are pre-installed, like scikit-learn and matplotlib.​
13. With GoogleCollab, you can develop many trending data science applications “on the fly”, which is to say, quickly without a lot of setup or preparation.​