**Data Analysis with Python: Part 1- Importing Datasets**

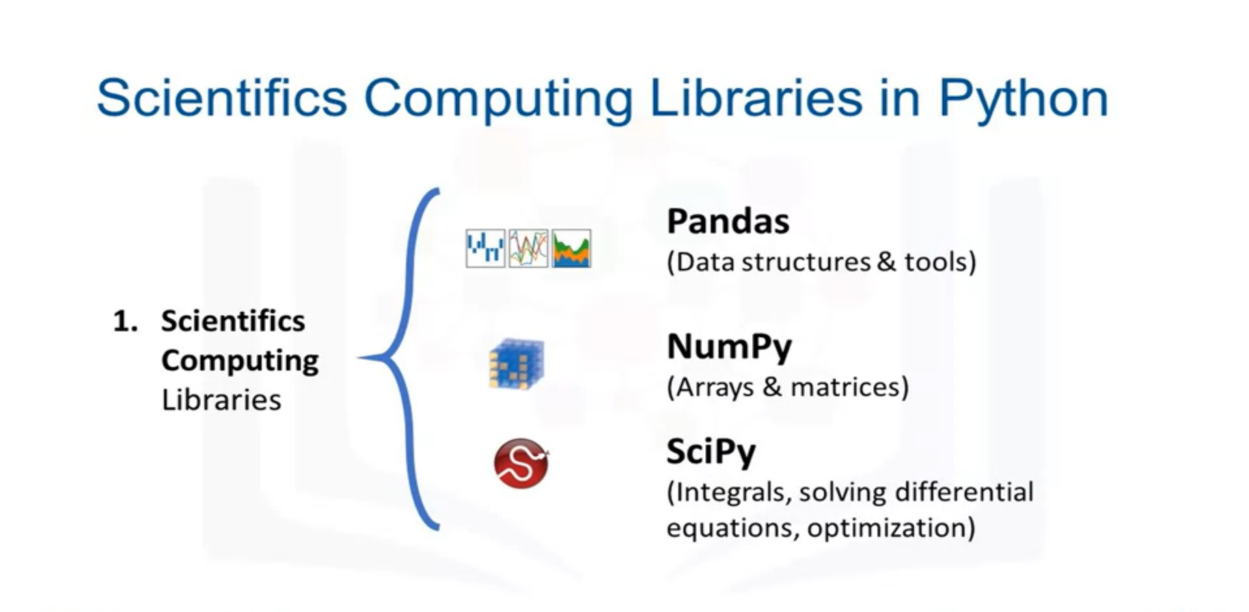
**Introduction**

Before we begin, we should first understand the importance of data analysis. As you know, data is collected everywhere around us. Whether it’s collected manually by scientists or collected digitally every time you click on a website or your mobile device. Data analysis and in essence data science helps us unlock the information and insights from raw data to answer our questions. So, data analysis plays an important role by helping us to discover useful information from the data, answer questions, and even predict the future or the unknown.

**Python packages for Data Science:**

A Python library is a collection of functions and methods that allow you to perform lots of actions without writing any code. The libraries usually contain built in modules providing different functionalities which you can use directly. As these libraries offer broad range of facilities, we have divided them into 3 categories:

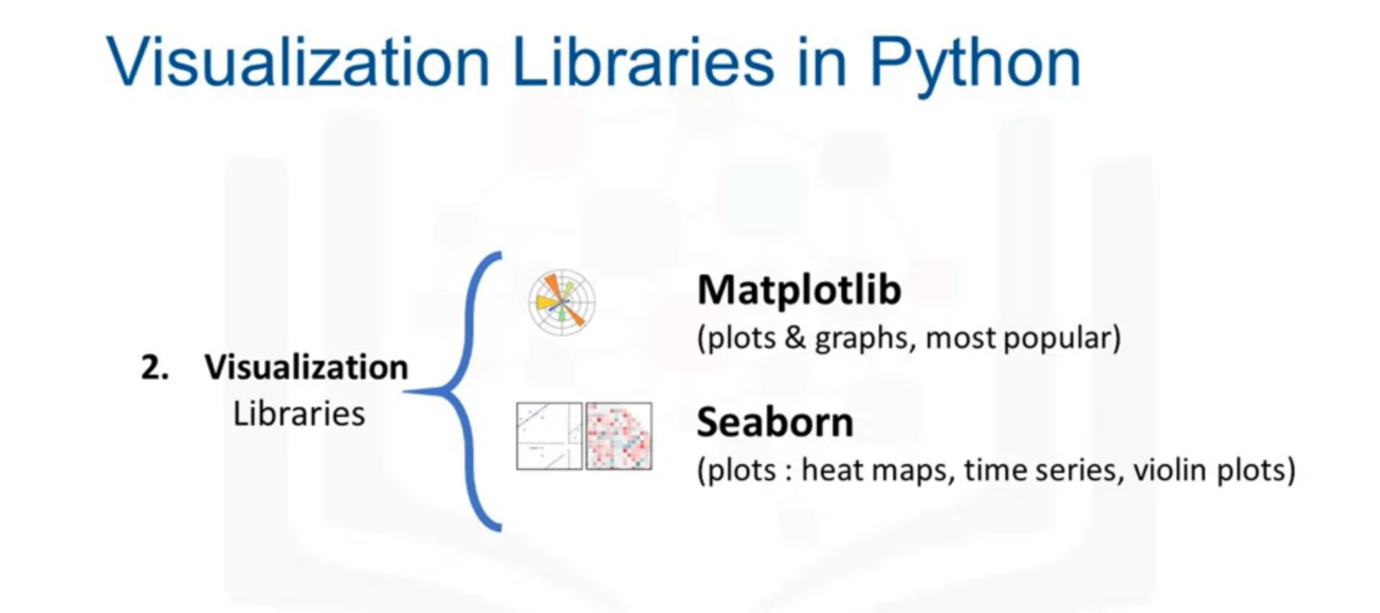
* Scientific computing libraries
* Visualization libraries
* Algorithmic libraries

1.Scientific computing libraries:

* **Pandas**: It offers data structure and tools for effective data manipulation and analysis. It provides facts, access to structured data. The primary instrument of Pandas is the two-dimensional table consisting of column and row labels, which are called a data frame. It is designed to provide easy indexing functionality.
* **NumPy:** It is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays
* **SciPy:** It is a free and open-source Python library used for scientific computing and technical computing. It contains modules for optimization, linear algebra, integration, interpolation, special functions, signal and image processing and other tasks common in science and engineering.

2. Visualisation Libraries:

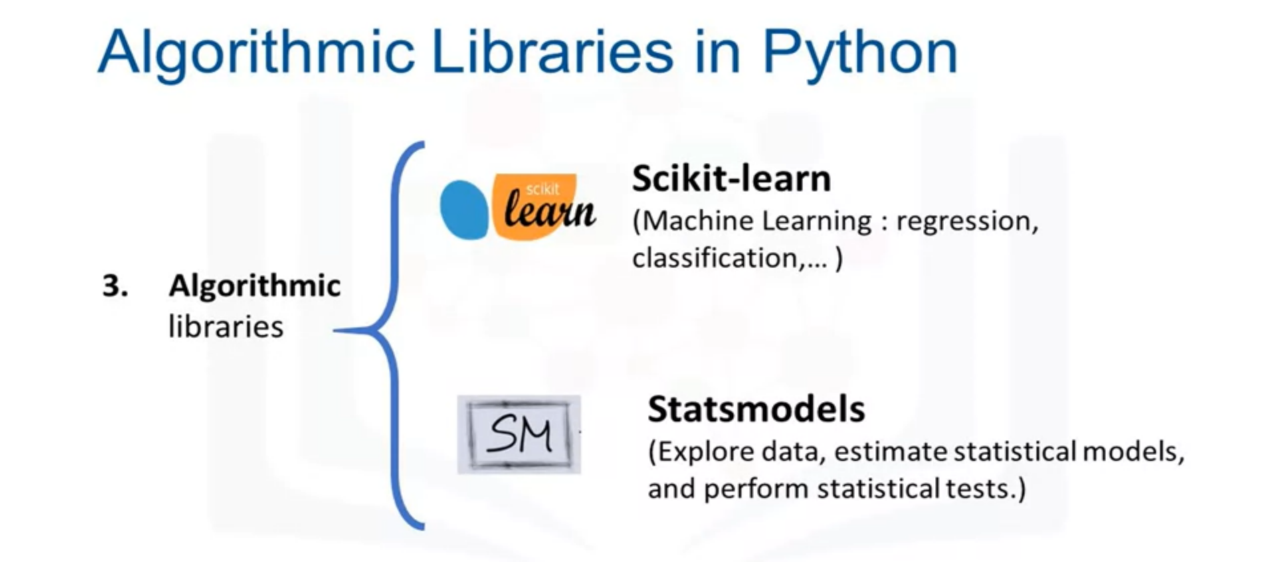
Using data visualization methods is the best way to communicate with others. Showing them meaningful results of analysis. These libraries enable you to create graphs, charts and maps.



* **Matplotlib**: The Matplotlib package is the most well-known library for data visualization. Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It is great for making graphs and plots. The graphs are also highly customizable.
* **Seaborn**: Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures. Seaborn helps you explore and understand your data. It’s very easy to generate various plots such as heat maps, time series and violin plots.

3. Algorithmic Libraries:

With machine learning algorithms, we're able to develop a model using our data set and obtain predictions. The algorithmic libraries tackle the machine learning tasks from basic to complex.



* **Scikit-learn:** Scikit-learn library contains tools statistical modelling, including regression, classification, clustering, and so on. It features various algorithms like support vector machine, random forests, and k-neighbours, and it also supports Python numerical and scientific libraries like NumPy and SciPy.
* **Statsmodels:** Statsmodels is a Python module that provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests, and statistical data exploration.

**Importing Datasets:**

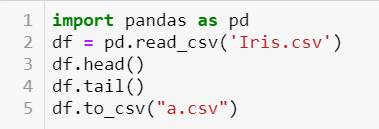
Data acquisition is a process of loading and reading data into notebook from various sources. To read any data using Python's pandas package, there are two important factors to consider, format and file path.

* Format is the way data is encoded. We can usually tell different encoding schemes by looking at the ending of the file name. Some common encodings are: CSV, JSON, XLSX, HDF and so forth.
* File path tells us where the data is stored. Usually, it is stored either on the computer we are using or online on the internet.

There are various kinds of files and each format of files have different way of importing them.

**Importing a csv file:**

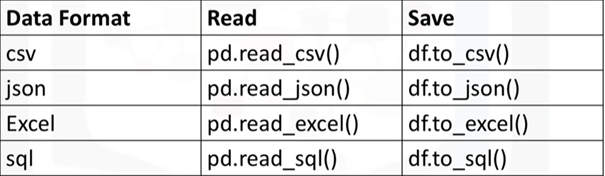
CSV stands for Comma seperated values. In pandas we use function **pandas.read\_csv()** to load csv files seperated by columns in a dataframe. Reading data from csv files can be done in three simple steps. First import pandas, then load the file in dataframe using **read\_csv()** function and finally display the dataframe



Head function gives us the first five rows of dataframe, while tail() gives us the last five rows

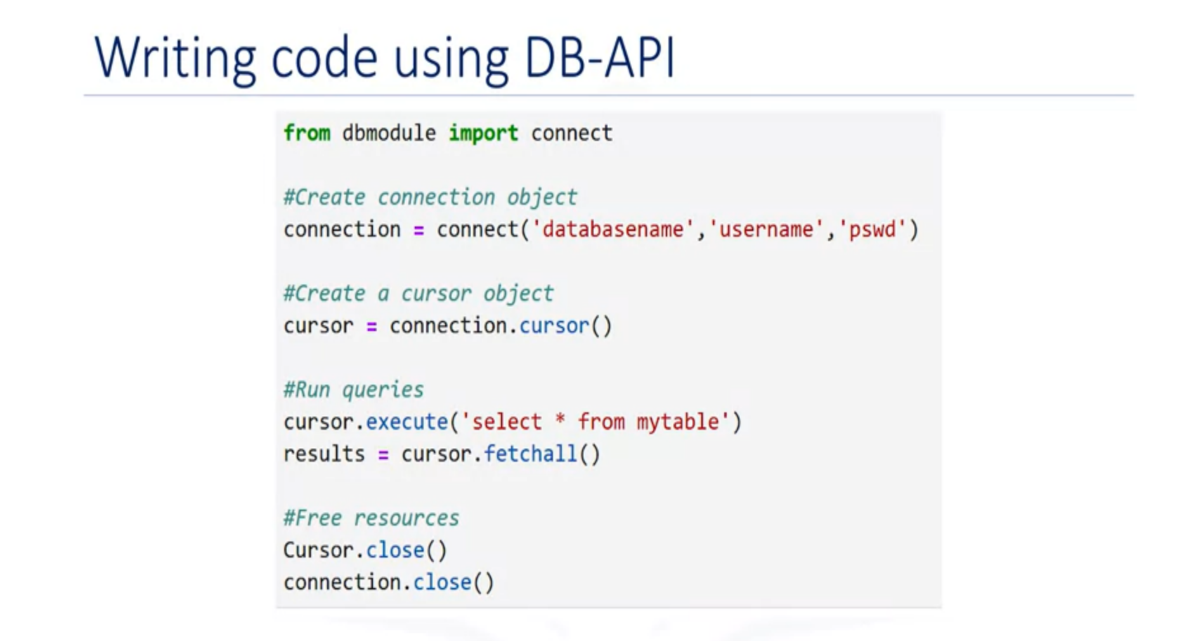
We can also export the dataframe to any csv file using ‘to\_csv()’ function.

Apart from this, pandas support importing of different file types. The code syntax for reading and exporting datasets is very similar to read or save CSV file. Each column shows a different method to read and save files in different formats.



**Accessing Databases using Python:**

The Python code connects to the database using API calls. We will explain the basics of SQL APIs and Python DB APIs. An application programming interface is a set of functions that you can call to get access to some type of servers. The SQL API consists of library function calls as an application programming interface, API, for the DBMS. To pass SQL statements to the DBMS, an application program calls functions in the API, and it calls other functions to retrieve query results and status information from the DBMS.



You use connection objects to connect to a database and manage your transactions. Cursor objects are used to run queries. You open a cursor object and then run queries. The cursor works similar to a cursor in a text processing system where you scroll down in your result set and get your data into the application. Cursors are used to scan through the results of a database.

Here are the methods used with connection objects. The cursor() method returns a new cursor object using the connection. The commit() method is used to commit any pending transaction to the database. The rollback() method causes the database to roll back to the start of any pending transaction. The close() method is used to close a database connection.