PROJECT REPORT

# Development Of Software In Python To Generate Various Graphical Analysis

**Abstract**: Graphs are used to solve many real-life problems. Graphs are used to represent networks. The networks may include paths in a city or telephone network or circuit network. Graphs are also used in social networks like linkedIn, Facebook.Hence the need for development of a software that generates various graphical analysis is very essential.No matter if you want to create interactive, live or highly customized plots, python has an excellent library package. To get a little overview here are a few popular plotting libraries: Matplotlib: low level, provides lots of freedom. Matplotlib was the first Python data visualization library, many other libraries are built on top of itSome libraries like [pandas](https://mode.com/python-tutorial/libraries/pandas/) and [Seaborn](https://mode.com/python-tutorial/libraries/seaborn/) are “wrappers” over matplotlib. They allow you to access a number of matplotlib’s methods with less code.

**Keywords** :Python, Pandas, Seaborn, Matplotlib, Data visualization

# **INTRODUCTION**

Python has already made it easy for data visualisation with two exclusive libraries for visualization, commonly known as *matplotlib and seaborn.Using this libraries we can   
Setup an graphical analysis system.*

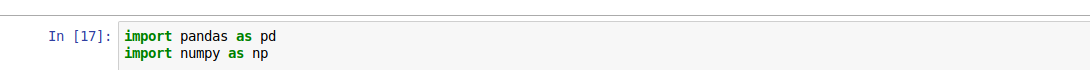
*Matplotlib*: Python based plotting library offers *matplotlib* with a complete 2D support along with limited 3D graphic support. It is useful in producing publication quality figures in interactive environments across platforms. It can also be used for animations as well.

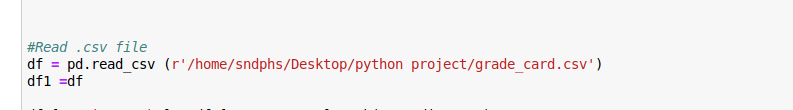
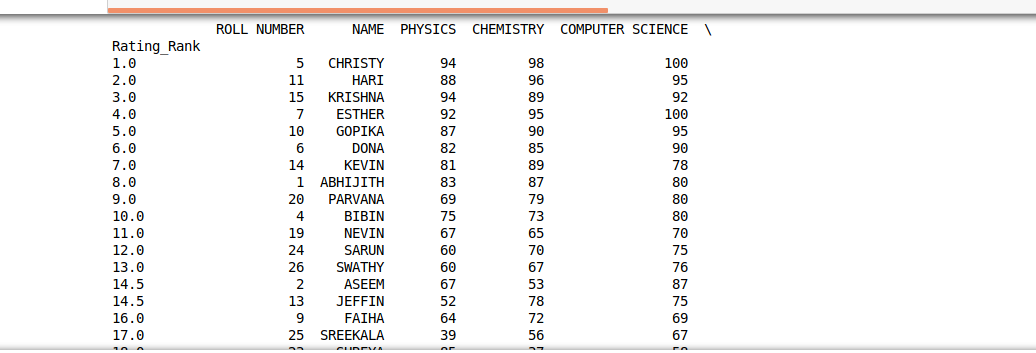
*Seaborn*: Seaborn is a library for creating informative and attractive statistical graphics in python. This library is based on matplotlib. Seaborn offers various features such as built in themes, color palettes, functions and tools to visualize univariate, bivariate, linear regression, matrices of data, statistical time series etc which lets us to build complex visualizations.

*Pandas* : It is a Python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labeled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real world data analysis in Python.

## METHODOLOGY

#### Import Data Set:

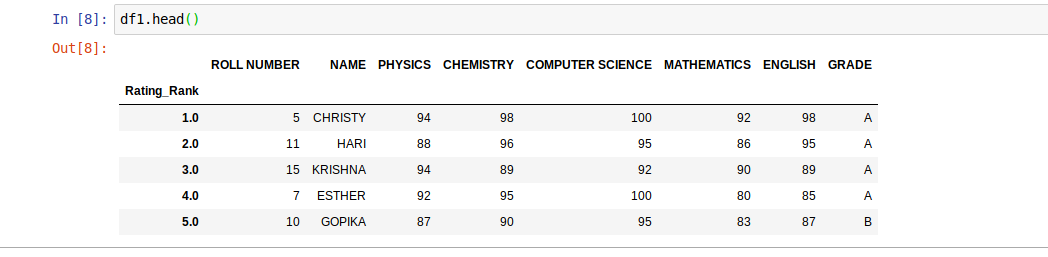
The first step to any data science project is to import your data.Data can be in any of the popular formats - CSV, TXT, XLS/XLSX (Excel), sas7bdat (SAS), Stata, Rdata (R) etc.

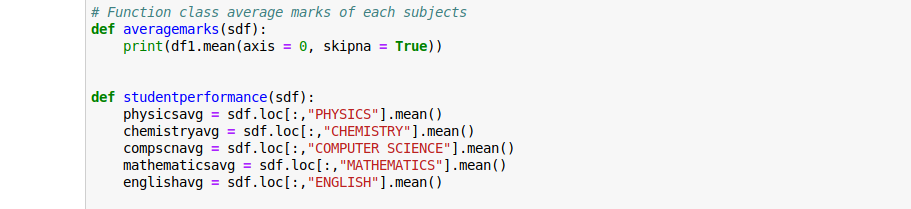


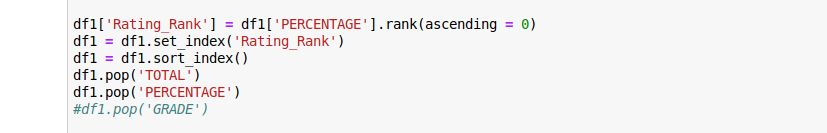
* Exploring the Data & Cleaning Corrupted Data

Python is a great language for doing data analysis, primarily because of the fantastic ecosystem of data-centric python packages. *Pandas* is one of those packages and makes importing and analyzing data much easier.

Data frame objects have many useful attributes that make this easy.

* Let’s get a better look at the data using the *.head()* method, which displays the first five rows, first five index values, of every column within a Pandas data frame object. 
* mean() function can be used to calculate mean/average of a given list of numbers. It returns the mean of the data set passed as parameters.



* Pandas set\_index() is a method to set a List, Series or Data frame as index of a Data Frame.
* Pandas dataframe.sort\_index() function sorts objects by labels along the given axis.
* *pop()* is an inbuilt function in Python that removes and returns last value from the list or the given index value
* VISUALISATION OF DATA

We can analyze data set to visualize through different charts

* Line Plots

Line graphs are plots where a line is drawn to indicate a relationship between a particular set of x and y values.

syntax:plt.plot(x, y)

### 

### Scatter Plots

Alternatively, you might want to plot quantities with 2 positions as data points.

Consider the same data as for line graph, to create scatter plots we just need to modify one line in the above code −

syntax:plt.plot(x, y,'o')

### Histogram

Histograms are very often used in science applications and it's highly likely that you will need to plot them at some point. They are very useful to plot distributions.

Syntax:

df = pd.DataFrame(data, columns = ([ ]))

df.hist()

plt.show()

* Pie chart

A Pie Chart is a circular statistical plot that can display only one series of data. The area of the chart is the total percentage of the given data.Matplotlib API has pie() function in its pyplot module which create a pie chart representing the data in an array. You can define it’s sizes, which parts should explode (distance from center), which labels it should have and which colors it should have.

Syntax:

plt.pie(sizes, explode=explode, labels=labels, colors=colors, ...)

Result Analysis

* Class average marks of each subjects

## CONCLUSION:

Data visualization is the discipline of trying to understand data by placing it in a visual context so that patterns, trends and correlations that might not otherwise be detected can be exposed.

Python offers multiple great graphing libraries that come packed with lots of different features. In this article, we looked at Matplotlib, Pandas visualization and Seaborn.