# [P-1]

**A PRELIMENERY REPORT ON**

**" STOCK MARKET PREDICTION "**

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE

IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS

OF

**THIRD YEAR COMPUTER ENGINEERING**

##### SUBMITTED BY

STUDENT NAME ROLL NO:

**MANISH YADAV 19U137**

**PRATIK KULKARNI 19U136**

**SANKET MORALWAR 19U140**



## DEPARTMENT OF COMPUTER ENGINEERING

## STES’S SMT. KASHIBAINAVALECOLLEGE OF ENGINEERING

VADGAON BK, OFF SINHGAD ROAD, PUNE 411041

## SAVITRIBAI PHULE PUNE UNIVERSITY

## 2020 -2021

**[P-2]**

****

**CERTIFICATE**

This is to certify that the project report entitles

**“STOCK MARKET PREDICTION”**

Submitted by

STUDENT NAME ROLL NO:

**MANISH YADAV 19U137**

**PRATIK KULKARNI 19U136**

**SANKET MORALWAR 19U140**

is a bonafide work carried out by him/her under the supervision of **Prof. S.N. Patil**

and it is approved for the partial fulfillment of the requirement of third year computer engineering.

**(Prof. S.N. Patil)** **(Dr. P. N. Mahalle)**

Guide Head,

Department of Computer Engineering Department of Computer Engineering

**(Dr. A. V. Deshpande)**

Principal,

Smt. Kashibai Navle College of Engineering Pune – 41

Place: Pune

Date:

**ACKNOWLEDGEMENT**

We would like to express our sincere indebtedness towards our guide Prof. R. B. Kale, Computer science and Engineering, Smt. Kashibai Navle College, Pune. For her individual guidance, suggestion and supervision throughout the work. Without her kind patronage and guidance, the project would not have taken shape We would also like to express our gratitude and sincere regards for her kind approval of the project of the project and time to time counseling and advices.

We would also like to thank our HOD Dr. P. N. Mahalle, Department of

Computer science and Engineering, Smt. Kashibai Navle College,

Pune. For his expert advice and counseling from time to time.

We owe special thanks to all the faculty members in the department of Computer science and Engineering, Smt. Kashibai Navle College, Pune.

**NAME OF THE STUDENTS**

**MANISH YADAV**

**PRATIK KULKARNI**

**SANKET MORALWAR**

**[P3]**

**CONTENTS**

LIST OF ABBREVATIONS I

LIST OF TABLES II

# CHAPTER TITLE PAGE NO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr. No. | | Chapters | | Page No. |
| 01 | | Abstract | | 3 |
| 02 | | Problem Definition | | 4 |
| 03 | | Motivation | | 5 |
| 04 | | Introduction | | 6 |
| 05 | | Project Scope | | 8 |
| 06 | | Architecture | | 9 |
| 07 | | Software Requirements Specification | | 11 |
|  | 7.1 | Functional Requirements | |  |
|  |  | 7.1.1 | System Feature 1(Functional Requirement) |  |
|  |  | 7.1.2 | System Feature2 (Functional Requirement) |  |
|  |  | 7.1.3 | System Feature3 (Functional Requirement) |  |
|  | 7.2 | System Requirements | |  |
|  |  | 7.2.1 | Software Requirements |  |
|  |  | 7.2.2 | Hardware Requirements |  |
|  | 7.3 | Nonfunctional Requirements | |  |
| 08 | | **Other Specifications** | | 13 |
|  |  | 8.1 | Advantages |  |
|  |  | 8.2 | Disadvantages |  |
|  |  | 8.3 | Applications |  |
| 09 | | **Results & Discussion** | | 16 |
|  |  | 9.1 | Screenshots |  |
| 10 | | **Conclusion** | | 19 |
| 11 | | **References** | | 20 |

**LIST OF ABBREVATIONS**

|  |  |
| --- | --- |
| **Abbreviation** | **Illustration** |
|  |  |
| QR | Quick Response Code |
| IP | Internet Protocol |
| IT | Information Technology |
| TCP | Transmission Control Protocol |
| UPC | Universal Product Code |
| PNG | Portable Network Graphics |
| URL | Uniform Resource Locator |
| GUI | Graphical user Interface |
| DLL | Dynamic Link Library |
| SVG | Scalable Vector Graphics |
| OS | Operating System |
| PIL | Python Imaging Library |
| RGB | Red Green Blue |
| CPU | Central Processing Unit |
| MB | Megabyte |
| GB | Gigabyte |
| RAM | Random Access Memory |
| ISO | International Organization for Standardization |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Table** | **Illustration** | **Page No.** |

**Table 1: Confusion Matrix 7**

**CHAPTER 1**

**ABSTRACT**

Stock Market Prediction using python

The prediction of a stock market direction may serve as an early recommendation system for short-term investors and as an early financial distress warning system for long-term shareholders. Forecasting accuracy is the most important factor in selecting any forecasting methods. Research efforts in improving the accuracy of forecasting models are increasing since the last decade. The appropriate stock selections those are suitable for investment is a very difficult task. The key factor for each investor is to earn maximum profits on their investments.

In this paper Decision Tree is used. is a very specific type of learning algorithms characterized by the capacity control of the decision function, the use of the kernel functions and the scarcity of the solution. In this paper, we investigate the predictability of financial movement with Decision Tree. The results will be used to analyze the stock prices and their prediction in depth in future research efforts

**CHAPTER 2**

**PROBLEM DEFINATION**

To create a python program which will predict upcoming stcok market values. This can be achieved with various methods. One of those methods is to use python’s “Scikit-Learn" library.Where we can also sell or buy stocks

**CHAPTER 3**

**MOTIVATION**

Recently, a lot of interesting work has been done in the area of applying Machine Learning Algorithms for analyzing price patterns and predicting stock price. Most stock traders nowadays depend on Intelligent Trading Systems which help them in predicting prices based on various situations and conditions. Recent researches uses input data from various sources and multiple forms. Some systems use historical stock data, some use financial news articles, some use expert reviews while some use a hybrid system which takes multiple inputs to predict the market.

**CHAPTER 4**

**INTRODUCTION**

In the past decades, there is an increasing interest in predicting markets among economists, policymakers, academics and market makers. The objective of the proposed work is to study and improve the supervised learning algorithms to predict the stock price.

**Technical Objective**

The technical objectives will be implemented in R. The system must be able to access a list of historical prices. It must calculate the estimated price of stock based on the historical data. It must also provide an instantaneous visualization of the market index.

**Experimental Objective**

Two versions of prediction system will be implemented; one using Decision trees and other using Support Vector Machines. The experimental objective will be to compare the forecasting ability of SVM with Decision Trees. We will test and evaluate both the systems with same test data to find their prediction accuracy.

There are many libraries and packages which are available in python and are used to implement this project, they are:

1 tkinter:

The tkinter package (“Tk interface”) is the standard Python interface to the Tk GUI toolkit. Both Tk and tkinter are available on most Unix platforms, as well as on Windows systems. (Tk itself is not part of Python; it is maintained at Active state.)

This module contains the low-level interface to Tk, and should never be used directly by application programmers. It is usually a shared library (or ), but might in some cases be statically linked with the Python interpreter. tkinter includes a number of Python modules, constants being one of the most important. Importing tkinter will automatically import tkinter.

2 sikit-learn:

Scikit-learn is largely written in Python, and uses [numpy](https://en.wikipedia.org/wiki/Numpy) extensively for high-performance linear algebra and array operations

Scikit-learn integrates well with many other Python libraries, such as [matplotlib](https://en.wikipedia.org/wiki/Matplotlib) and [plotly](https://en.wikipedia.org/wiki/Plotly) for plotting, [numpy](https://en.wikipedia.org/wiki/NumPy) for array vectorization, [pandas](https://en.wikipedia.org/wiki/Pandas_(software)) dataframes, [scipy](https://en.wikipedia.org/wiki/SciPy), and many more

3 Tkinter messagebox:

The messagebox module is used to display the message boxes in the python applications. There are the various functions which are used to display the relevant messages depending upon the application requirements.

the tkinter. messagebox module provides a template base class as well as a variety of convenience methods for commonly used configurations. The message boxes are modal and will return a subset of (True, False, OK, None, Yes, No) based on the user’s selection.

4 PIL/Pillow:

Pillow is a Python Imaging Library (PIL), which adds support for opening, manipulating, and saving images

5 Mysql :

Mysql is used to store data in tabular format ,from where data can be accessed easily

**CHAPTER 5**

**PROJECT SCOPE**

This project focuses on making easy, convenient and helpful to buy and sell stocks. This project does the efficient use of the current technology. And it will provide speed and efficiency to the system.

Conventional methods that are used are time consuming and tedious. This project is built for the evolution of the current method that we are using.

**CHAPTER 6**

**ARCHITECTURE**

This project, generation Stock Prediction is implemented with the combination of software and hardware. Software included are Windows/Linux, Python etc. hardware includes intel i5 8th generation, minimum 8GB RAM.

After running the program, a window will pop out which shows four fields that are: ‘Enter Ticker’, ‘Predict For’ and Predict. This is the main application Window of the program. In the ‘Enter Ticker Symbol’, ticker symbol of a company is entered and ‘Predict For’ is used to select the no of days. Then upon clicking the ‘Predict’ the ticker entered in the ‘Enter Ticker Symbol’is used to get get data of that company using nsepy module and predict using sikit-learn module.

**6.1 Creating a Prediction Graph**

The prediction object is designed to be smart about how it constructs prediction graphs. The error correction level defaults to the highest possible level of error correction.

There are many situations where we might wish to have more fine-grained control over how the Prediction graph is generated. We can specify all the properties of prediction through the optional parameters of the skikit-learn’s classification report class.

**6.2 Training set**

It is a subset of the dataset used to build predictive models. Test set or unseen examples is a subset of the dataset to assess the likely future performance of a model. If a model fit to the training set much better than it fits the test set, over fitting is probably the cause.

**6.3 Confusion Matrix**

A confusion matrix shows the number of correct and incorrect predictions made by the classification model compared to the actual outcomes (target value) in the data. The matrix is NxN, where N is the number of target values (classes). Performance of such models is commonly evaluated using the data in the matrix. The following table displays a 2x2 confusion matrix for two classes (Positive and Negative).

|  |  |  |
| --- | --- | --- |
| **Actual** | | |
| **Positiive** | | **Negative** |
| **Prediction** | **Positive** | **True Positive** | | **False Negative** | |
| **Negative** | **False Negative** | | **True Negative** | |
|  | Accuracy (ACC) = (Σ True positive + Σ True negative) / Σ Total population | True positive rate (TPR) = Σ True positive / Σ Condition positive | | False positive rate (FPR) = Σ False positive / Σ Condition negative | |

**Table 1: Confusion Matrix**

**6.3 Accuracy**:

The proportion of the total number of predictions that were correct.

**6.4 Positive Predictive Value or Precision**:

the proportion of positive cases that were correctly identified.

**6.5Negative Predictive Value:**

the proportion of negative cases that were correctly identified.

**6.6Sensitivity or Recall**:

the proportion of actual positive cases which are correctly identified.

**6.7Specificity**:

The proportion of actual negative cases which are correctly identified.

**CHAPTER 7**

**SOFTWARE REQUIREMENTS SPECIFICATION**

**7.1 FUNCTIONAL REQUIREMENTS**

7.3.1 SYSTEM FEATURE 1(FUNCTIONAL REQUIREMENTS)

Input is taken from the user and the Prediction Graph is generated according to the users’ requirements.

7.3.2 SYSTEM FEATURE 2(FUNCTIONAL REQUIREMENTS)

User can buy or sell the stocks

7.3.3 SYSTEM FEATURE 3(FUNCTIONAL REQUIREMENTS)

The background color for all windows in the application will be light green and have a hexadecimal RGB color value of #D2D7CE

**7.2 SYSTEM REQUIREMENTS**

7.2.1 SOFTWARE REQUIREMENTS

* python3 or any latest version of python
* Gedit
* Operating system: Ubuntu or any 32/64-bit Linux/windows operating system
* Mysql

7.2.2 HARDWARE REQUIREMENTS

* Any CPU with Pentium processor or similar.
* 256 MB RAM or more.
* 1 GB Hard Disk or more

**7.3 NONFUNCTIONAL REQUIREMENTS**

* The software should be portable. So, moving from one OS to other OS does not create any problem
* Privacy of information, the export of restricted technologies, intellectual property rights, etc. Should be taken into consideration.

**CHAPTER 8**

**OTHER SPECIFICATION**

**8.1 ADVANTAGES**

* Reduce Biases
* Cross Check results
* Helps Create A schedule
* Consistency gruanteed
* Use of variety of instruments
* No need to hire an advisor

**8.2 DISADVANTAGES**

* **Do no give accurate values**
* **Owners is high risk at time of time involved in invesment**

**8.3 APPLICATIONS**

* Helps to buy and sell stocks while Trading

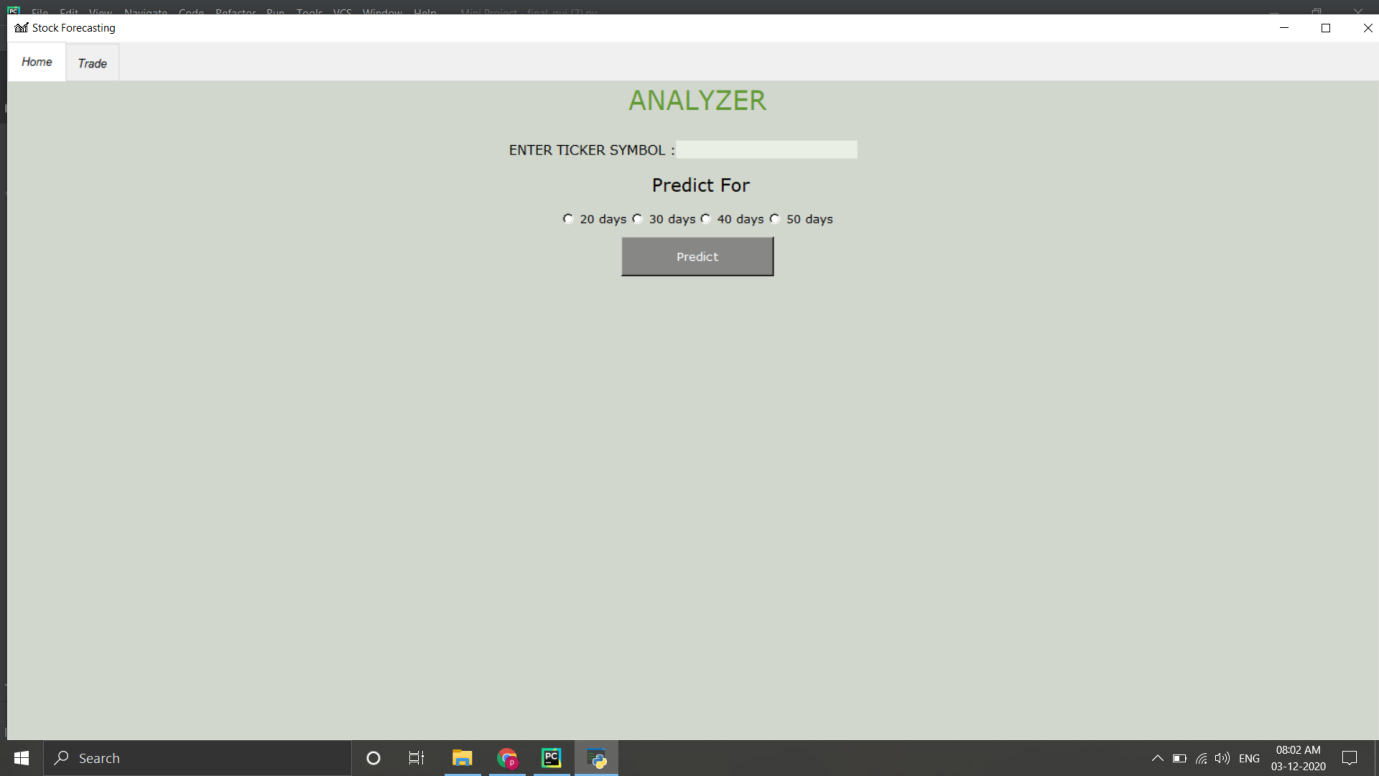
The successful prediction of a stock's future price could yield significant profit. The [efficient-market hypothesis](https://en.wikipedia.org/wiki/Efficient-market_hypothesis) suggests that stock prices reflect all currently available information and any price changes that are not based on newly revealed information thus are inherently unpredictable

**CHAPTER 9**

**RESULT AND DISCUSSUION**

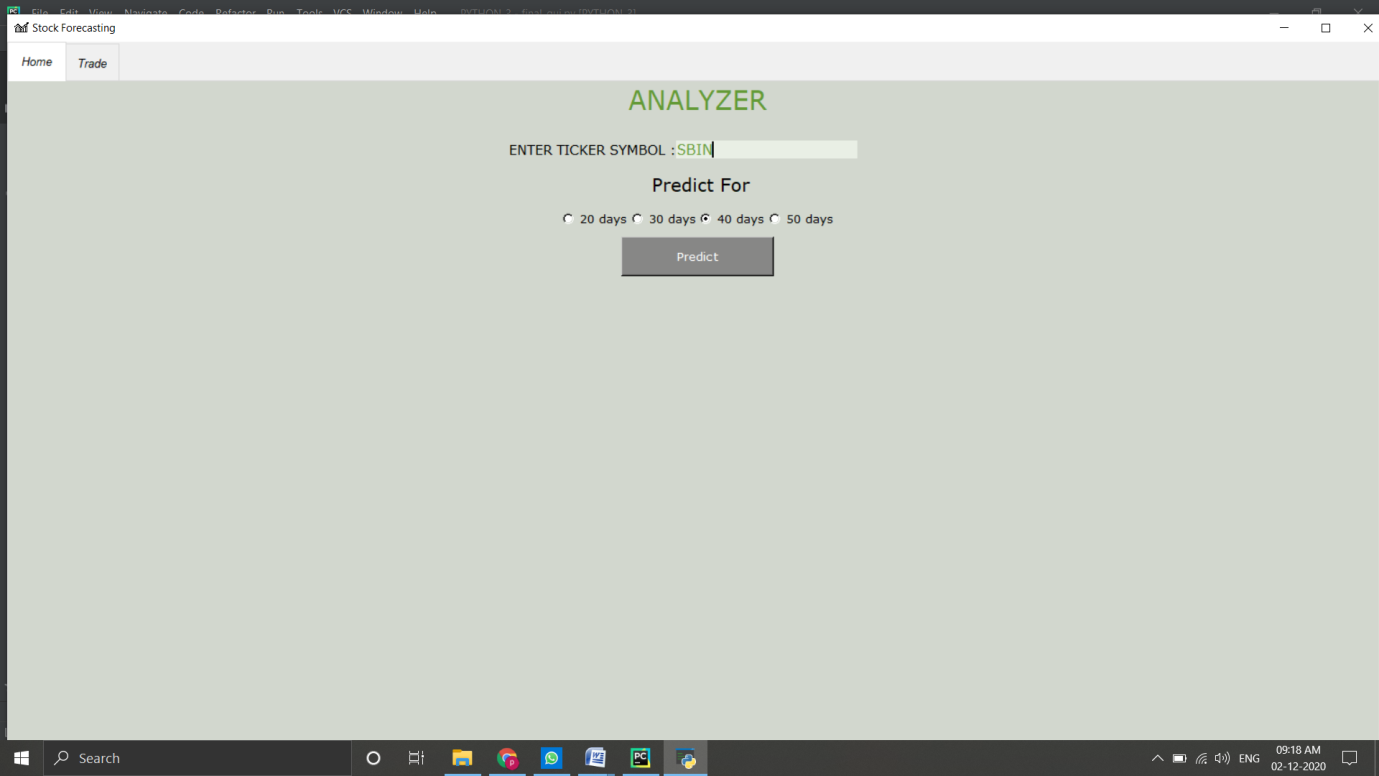
**9.1 SCREENSHOTS**

**Application window:**

****

**Fig 9.1.1 Application Window**

**User Input:**

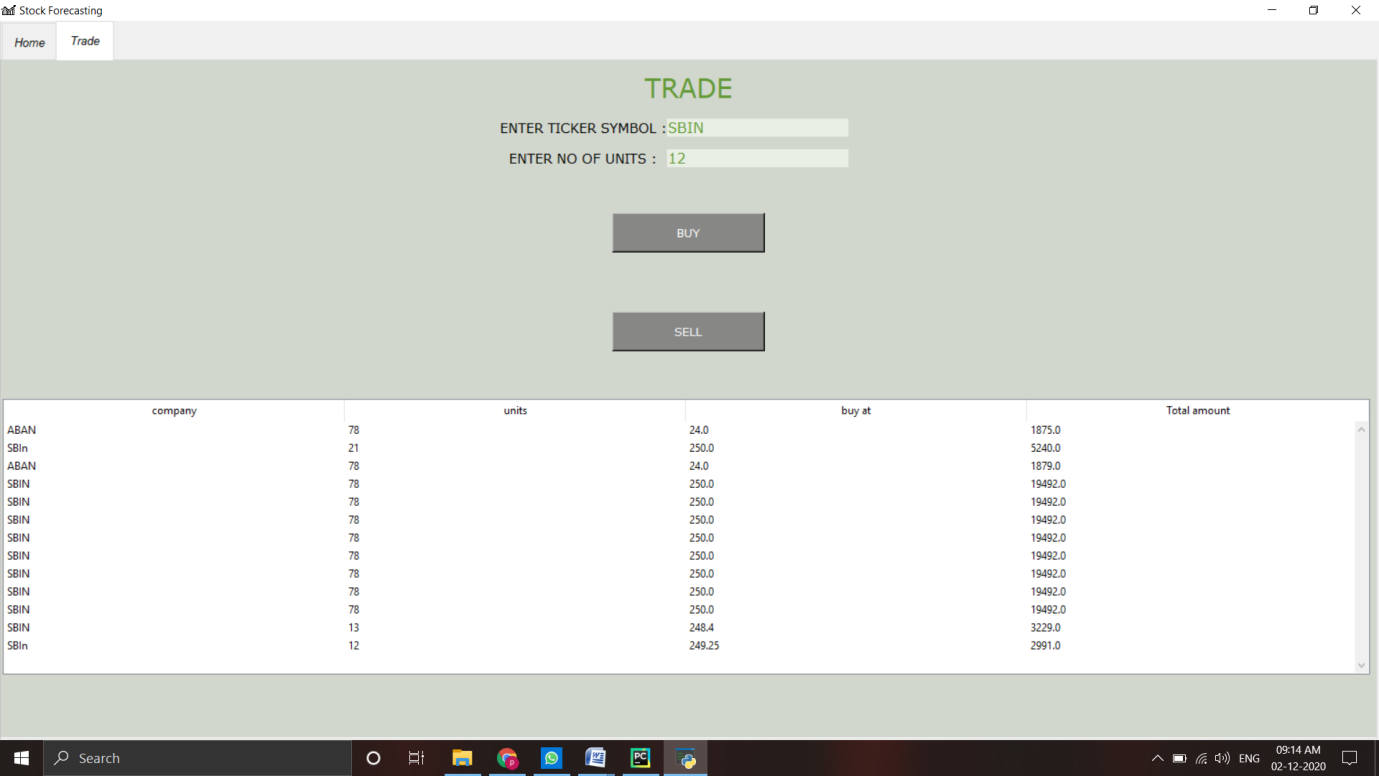
****

**Fig no 9.1.2 User Input:**

**Generated Graph:**

****

**Fig 9.1.3 Generated Graph**

**Trading Window:** ****

**Fig 9.1.5 Trading Window**

**CHAPTER 10**

**CONCLUSION**

In this project, we have shown how to predict stock market via python that facilitates users to easily buy and sell their stocks on Trading sites. The proposed method was developed using entirely open source software such as Python, and Ubuntu. The experimental results show that the prediction graph was successfully generated with higher accuracy. Therefore, the proposed method is considerably a Stock Market Prediction collaborative tools that is available for free use.

It was emphasized on the basic steps, consequently taken during the project’s development course as a particular attention was turned to the basic operative functions performed.

**CHAPTER 11**

**REFERENCES**

* WIKIMEDIAFOUNDATIONINC,2001, [https://en.wikipedia.org/ wiki/ Scikit-learn](https://en.wikipedia.org/%20wiki/Digital_image_processing)
* WIKIMEDIA FOUNDATION INC, 2001, WIKIMEDIA FOUNDATION INC, 2001, <https://en.wikipedia.org/wiki/NumPy>
* [MICROSOFT CORPORATION](https://www.google.com/search?client=firefox-b-d&q=Microsoft&stick=H4sIAAAAAAAAAONgVuLQz9U3MCkuM3nEaMwt8PLHPWEprUlrTl5jVOHiCs7IL3fNK8ksqRQS42KDsnikuLjgmngWsXL6ZiYX5Rfnp5UAAAHap6FOAAAA), 2008, https://scikit-learn.org/stable/tutorial/index.html
* BRUCE ECKEL. (2002), THINKING IN PYTHON
* MATTHES ERIC. (2005), PYTHON CRASH COURSE,2nd EDITION: A HANDS-ON, PROJECT-BASED INTRODUCTION TO PROGRAMMING
* ANTHONY SCOPTAZ, KATHRYAN D. HUFF (O’REILLY,2015), EFFECTIVE COMPUTATION IN PHYSICS: FIELD GUIDE TO RESEARCH WITH PYTHON
* [NATHAN METZLER](https://www.amazon.com/s/ref=dp_byline_sr_book_1?ie=UTF8&field-author=Nathan+Metzler&text=Nathan+Metzler&sort=relevancerank&search-alias=books). (2007), PYTHON GUI PROGRAMMING WITH PYQT: A BEGINNER’S GUIDE TO PYTHON 3 AND GUI APPLICATION DEVELOPMENT