# TECHNOCOLABS DATA SCIENCE INTERNSHIP

*Internship Project Report*

On

## OPTIMIZING STOCK TRADING STRATEGY WITH REINFORCEMENT LEARNING


Submitted by:

*Team Member*

*Akshay kumar s*

*Yukti Kapoor*

*Gopika. P*

*Bhushan G. Asati*

*Pushkarani. R*

*Loka akash Reddy*

*Mayur Chilamwar*

*Madiha kazi*

Under the Supervision of:

**Deepthika Shiwani Muralikrishnan**

**Aim: -** The principle focus and aim of our project are to perform exploratory data analysis, train and build a model using a Machine Learning algorithm, and deploy it as a web app. By doing so, it helps us to predict the daily close prices of stocks of different companies using the web app.

**Abstracts: -** The goal of the project is to predict price changes in the future for a given stock. Information that is leveraged to make these predictions includes prices from previous days and financial news headlines related to the company of interest. Reinforcement learning is a branch of Machine learning where we have an **agent** and an **environment**. The environment is nothing but a task or simulation and the Agent is an AI algorithm that interacts with the environment and tries to solve it. In this paper, the Reinforcement Machine Learning model is used to predict the close price using the historical data. Using these trained models, we build a predictor for different companies that predicts the daily close stock prices. This predictor can be used to find the price at which the stock value will close for a specific day by giving inputs like open, high, low prices, the volume of stock, and recent news related to each company.

**Introduction: -** We've always been fascinated by how seemingly unpredictable the stock market. Bear and bull are terms that you will get to hear in the stock market often. A bear run is a term that suggests a decline in the market prices over a long time while a bull run refers to its opposite. These are terms used by traders who deal in intraday trading. Intraday trading is a form of speculation in securities in which a trader buys and sells a financial instrument within the same trading day, such that all market positions are closed before the market closes for that day. A large volume of financial instruments is traded via the Intra Day trading method.

This has been conventionally working with the trade plan and news trends. With the advent of Data Science and Machine Learning, various research approaches have been designed to automate this manual process. This automated trading process will help in giving suggestions at the right time with better calculations. An automated trading strategy that gives maximum profit is highly desirable for mutual funds and hedge funds. The kind of profitable returns that is expected will come with some amount of potential risk. Designing a profitable automated trading strategy is a complex task.

Every human being wants to earn to their maximum potential in the stock market. It is very important to design a balanced and low-risk strategy that can benefit most people. One such approach talks about using reinforcement learning agents to provide us with automated trading strategies based on the basis of historical data.

**Reinforcement Learning: -** Reinforcement learning is a type of machine learning where there are environments and agents. These agents take actions to maximize rewards. Reinforcement learning has a very huge potential when it is used for simulations for training an AI model. There is no label associated with any data, reinforcement learning can learn better with very few data points. All decisions, in this case, are taken sequentially. The best example would be found in Robotics and Gaming. Trading is a continuous task without any endpoint. Trading is also a partially observable Markov Decision Process as we do not have complete information about the traders in the market. Since we don't know the reward function and transition probability, we use model-free reinforcement learning which is Q-Learning.


**Q – Learning: -** Q-learning is a model-free reinforcement learning algorithm. It informs the agent what action to undertake according to the circumstances. It is a value-based method that is used to supply information to an agent for the impending action. It is regarded as an off-policy algorithm as the q-learning function learns from actions that are outside the current policy, like taking random actions, and therefore a policy isn't needed.

Q here stands for Quality. Quality refers to the action quality as to how beneficial that reward will be in accordance with the action taken. A Q-table is created with dimensions an agent interacts with the environment in either of the two ways – exploit and explore. An exploit option suggests that all actions are considered and the one that gives maximum value to the

environment is taken. An explore option is one where a random action is considered without considering the maximum future reward.

**OVERVIEW:**

1. Creating an Environment.
2. Reading and preprocessing the Dataset
3. Training the Dataset
4. Tracking the Portfolio Value
5. Testing the Dataset
6. Plotting the portfolio for the test Dataset

1. **Creating an Environment: -** Environment is the fundamental element in the reinforcement learning problem. It is very important to have the right understanding of the underlying environment with which the RL agent is supposed to interact. This helps us to come up with the right design and learning technique for the agent.

   The reinforcement learning problem is meant to be a straightforward framing of the problem of learning from interaction to achieve a goal. The learner and decision-maker is called the agent. The thing it interacts with, comprising everything outside the agent, is called the environment. These interact continually, the agent selecting actions and the environment responding to those actions and presenting new situations to the agent. The environment also gives rise to rewards, special numerical values that the agent tries to maximize over time. A complete specification of an environment defines a task, one instance of the reinforcement learning problem. More specifically, the agent and environment interact at each of a sequence of discrete time steps, $t = 0, 1, 2, 3, \ldots$. At each time step $t$, the agent receives some representation of the environment's state, $s_t \in \mathcal{S}$, where $\mathcal{S}$ is the set of possible states, and on that basis selects an action, $a_t \in \mathcal{A}(s_t)$, where $\mathcal{A}(s_t)$ is the set of actions available in state $s_t$. One time step later, in part as a consequence of its action, the agent receives a numerical reward, $r_{t+1} \in \Re$, and finds itself in a new state, $s_{t+1}$.

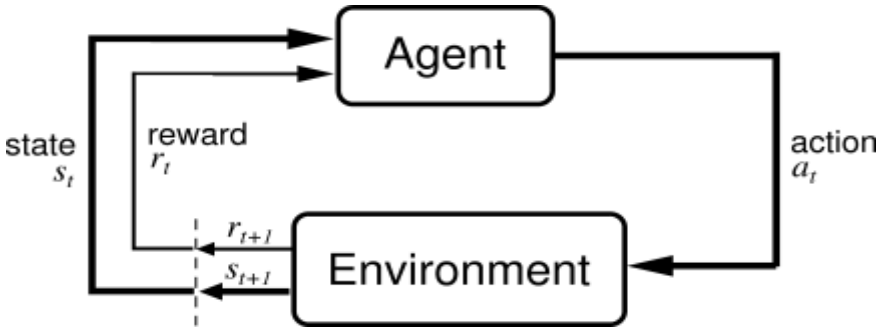
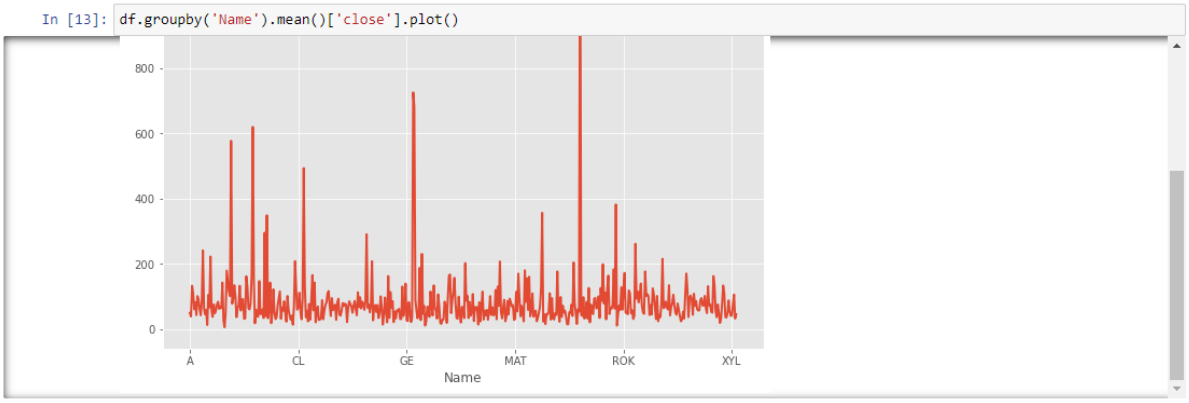
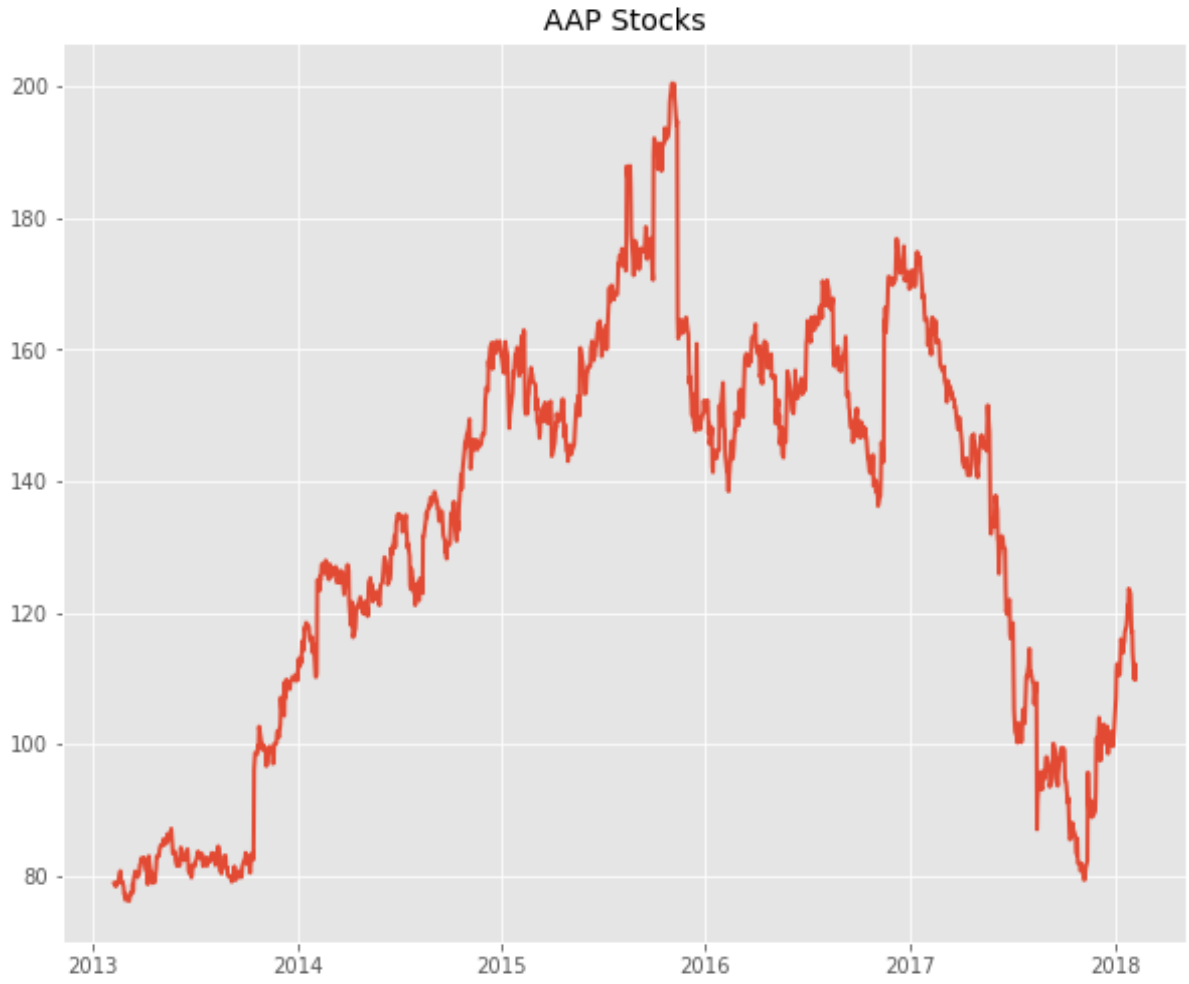



**Figure:-** The agent-environment interaction in reinforcement learning.

2. **Declaring Data Preprocessing Function: -**
    - Reading the Dataset: - The dataset contains data of stock market values from June 2011 to June 2021. In this project, we have worked by combining news headlines and historical data of the company.
    - Format of the dataset: CSV
    - A brief explanation of every column in the dataset is as follows:
    - DATE: Day on which stock is traded
    - CLOSE: Refers to the last price at which a stock trades during a regular trading session
    - VOLUME: The number of shares or contracts traded in a security or an entire market during a given period
    - OPEN: The price of the first trade for any listed stock is its daily opening price.
    - HIGH: High is the highest price at which a stock is traded during the trading day.
    - LOW: Low is the lowest price at which a stock trades over a trading day.
    - Data Cleaning: - Data cleaning refers to identifying and correcting errors in the dataset that may negatively impact a predictive model. Data cleaning is used to refer to all kinds of tasks and activities to detect and repair errors in the data. In this, we removed the duplicate data and renamed column names "Close/Last" to "Close" and” $” to ““. Then converted all different data types into integer or float dtype, explicitly we converted date (object) dtype to date time dtype and removed the NULL values. And also done the Data

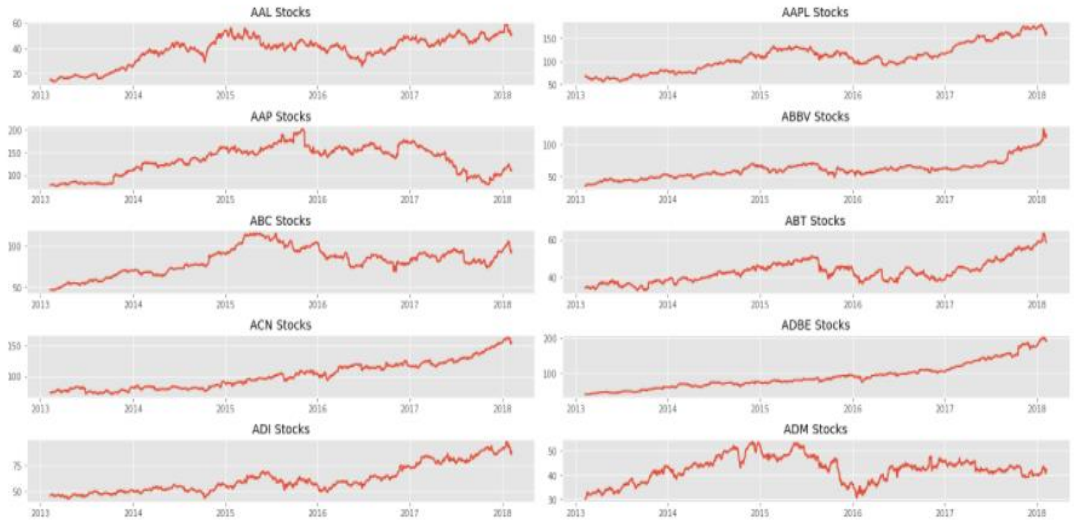
Visualization, helps explore and get to know a dataset and can help with identifying patterns, corrupt data, outliers, and much more.



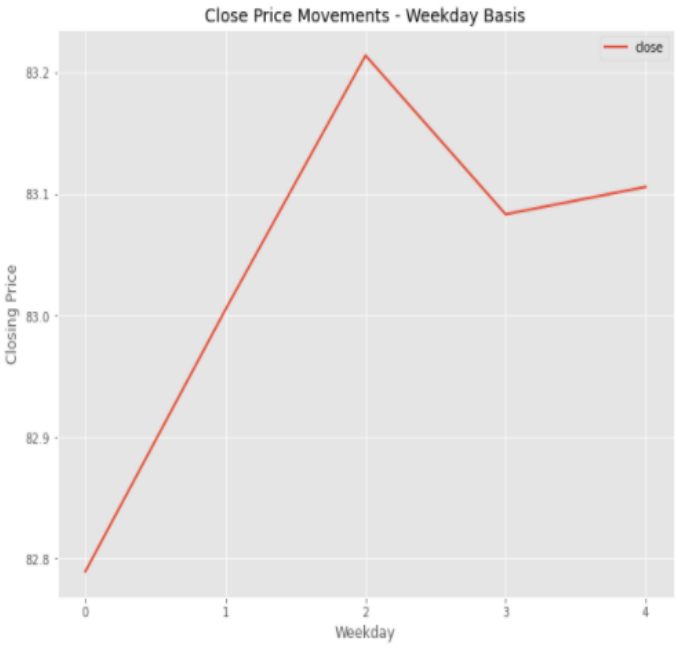
In [23]:
```
fig = plt.figure(figsize=(20, 8))

for i in range(0,10):
    ax = fig.add_subplot(5, 2, i+1)
    stock_plot(list[i])
    plt.tight_layout()
```




In [29]:
```
weekday.groupby('weekday').mean().plot()
plt.xticks([0,1,2,3,4])
plt.title('Close Price Movements - Weekday Basis')
plt.ylabel('Closing Price')
plt.xlabel('Weekday')
```

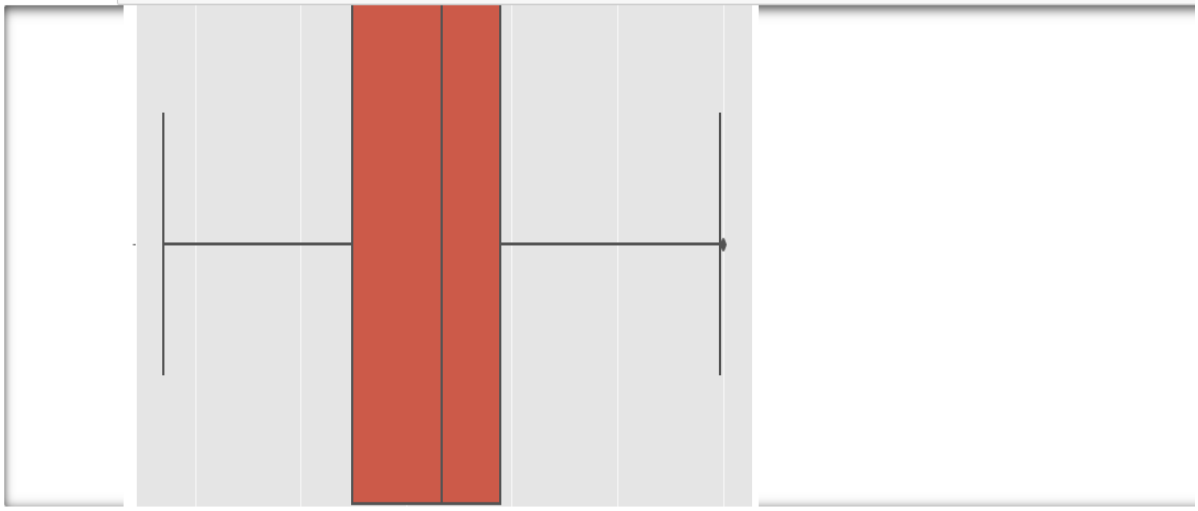
Out[29]: Text(0.5, 0, 'Weekday')



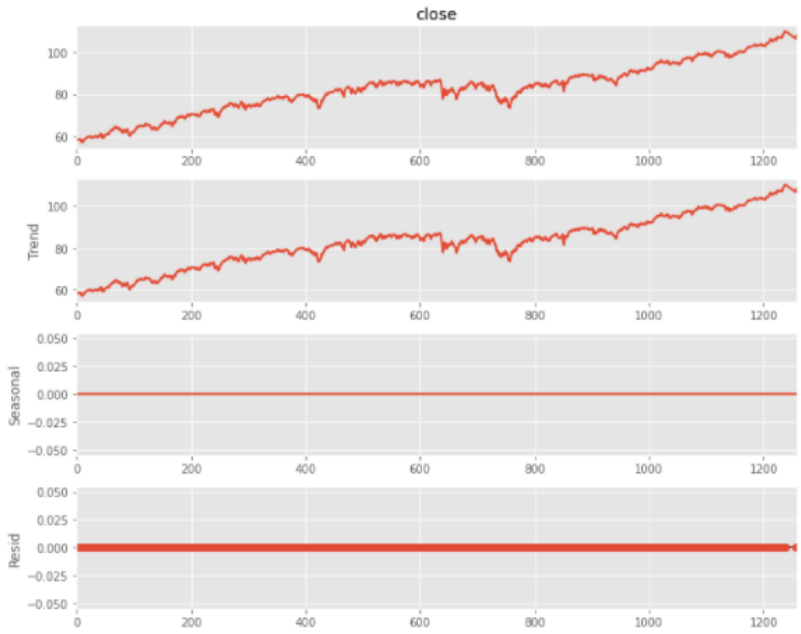
In [30]:
```
df.groupby('month').mean()['close'].plot()
plt.title('Close Price Movements - Monthly Basis')
plt.ylabel('Closing Price')
plt.xlabel('Month')
```

Out[30]: Text(0.5, 0, 'Month')


In [40]:
```
sns.boxplot(clean_df['close'])
```



```
In [57]: x = decomposed.plot()
```




### 3. Training the Dataset: -

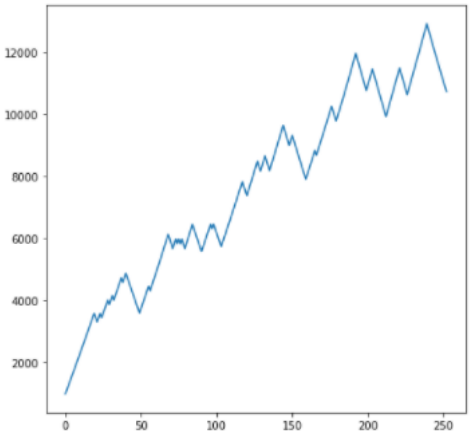
- When the agent is exploring the simulation, it will record experiences.
- Single experience = (old state, action, reward, new state)
- Training our model with a single experience:
- Let the model estimate Q values of the old state
- Let the model estimate Q values of the new state
- Calculate the new target Q value for the action, using the known reward
- Train the model with input = (old state), output = (target Q values)

## 4. Tracking the Portfolio Value

**Tracking the Portfolio Value**

```
In [12]: fig, ax = plt.subplots (figsize = (7, 7))
         plt.plot(net_worth)
         # plt.plot(stocks_train['close'])
```

```
Out[12]: [<matplotlib.lines.Line2D at 0x20ab924ff40>]
```

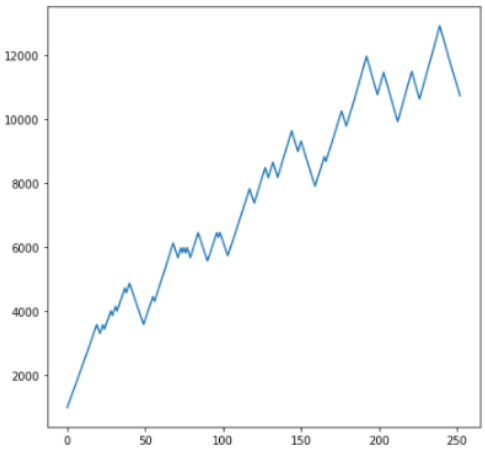



## 5. Testing the Dataset

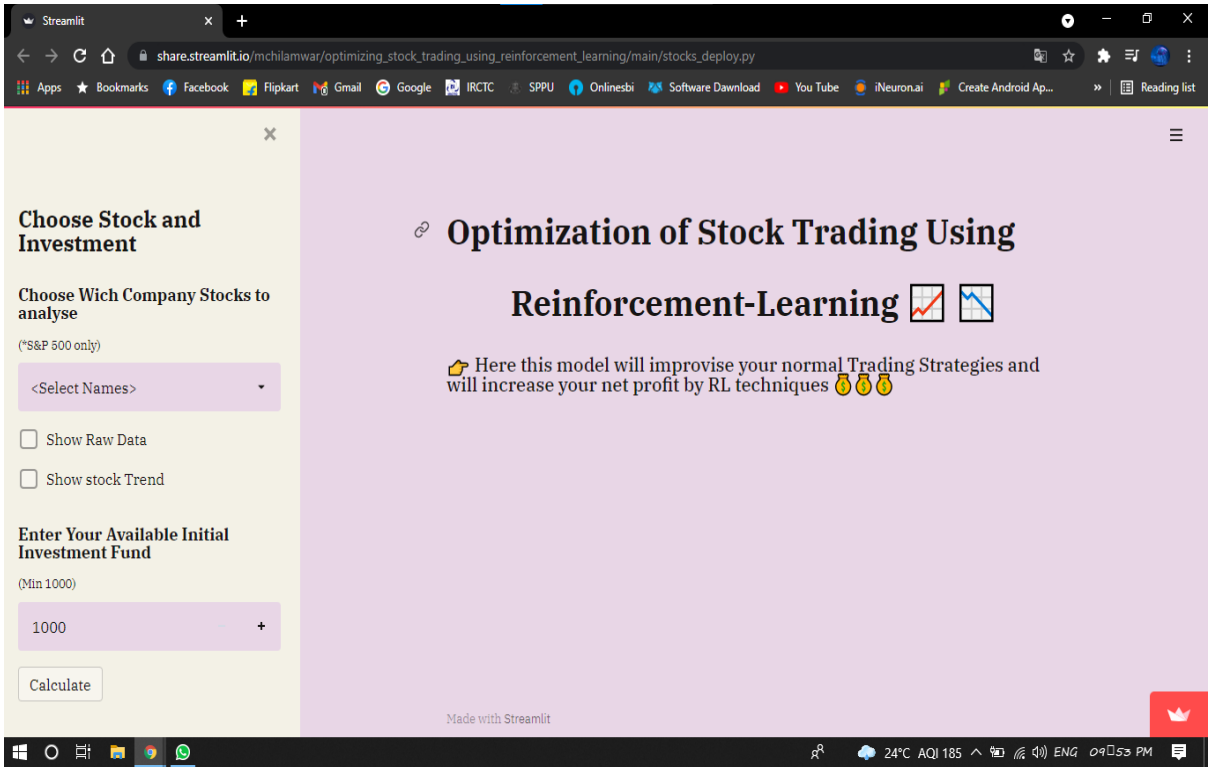
**Plotting the portfolio for the test Dataset**

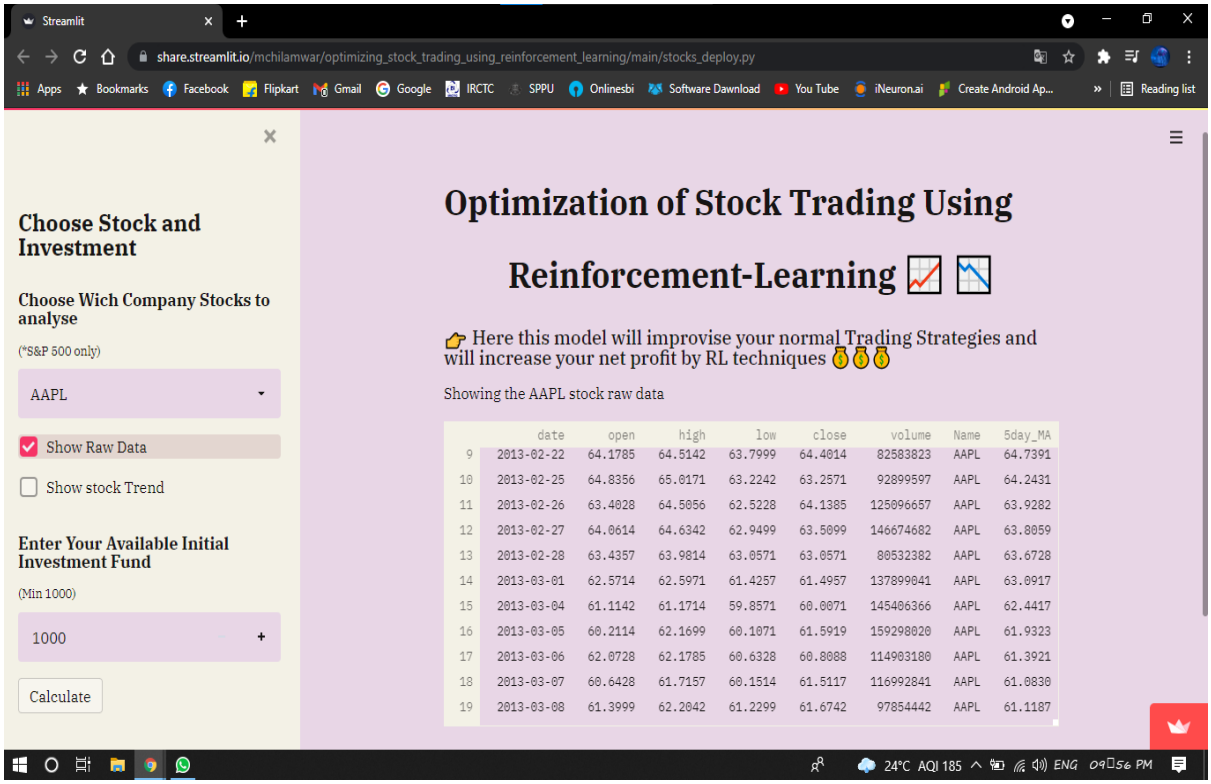
```
In [10]: fig, ax = plt.subplots (figsize = (7, 7))
         plt.plot(net_worth)
```

```
Out[10]: [<matplotlib.lines.Line2D at 0x187d50cbe20>]
```

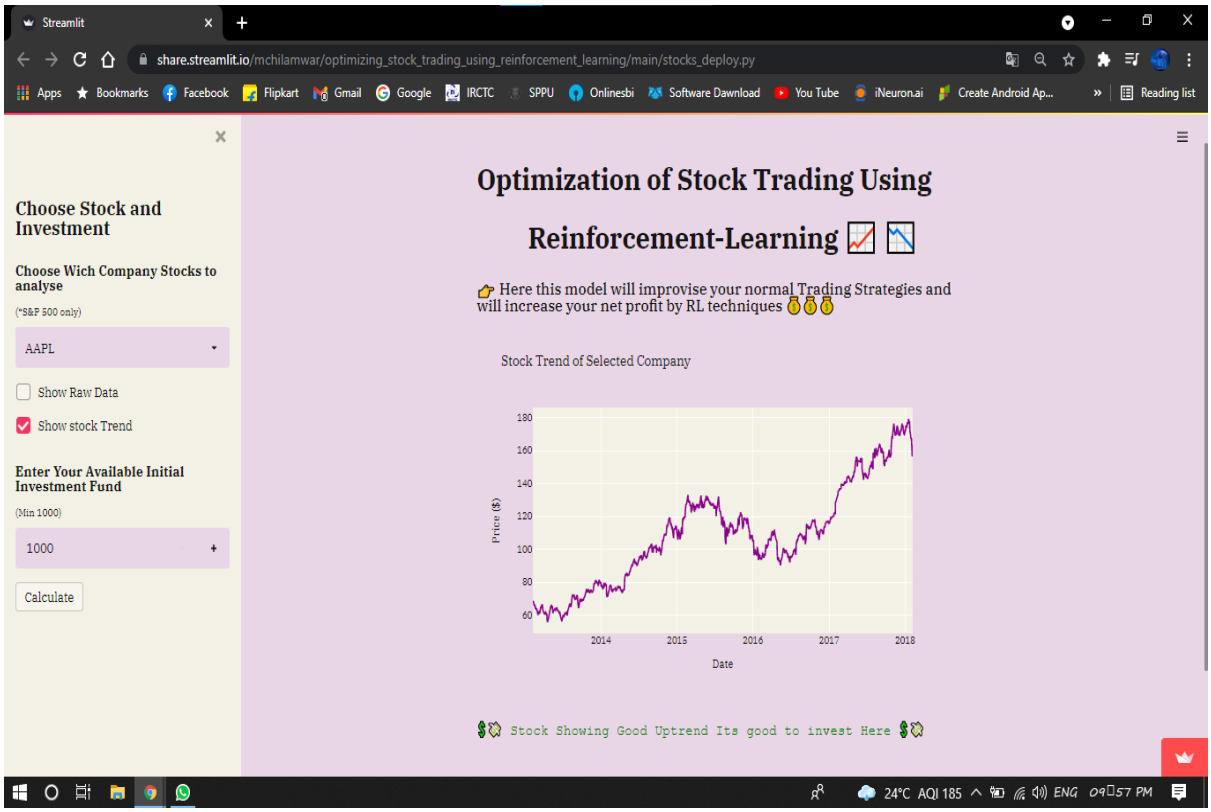


# Output: -




| | date | open | high | low | close | volume | Name | 5day_MA |
|---|---|---|---|---|---|---|---|---|
| 9 | 2013-02-22 | 64.1785 | 64.5142 | 63.7999 | 64.4014 | 82583823 | AAPL | 64.7391 |
| 10 | 2013-02-25 | 64.8356 | 65.0171 | 63.2242 | 63.2571 | 92899597 | AAPL | 64.2431 |
| 11 | 2013-02-26 | 63.4028 | 64.5056 | 62.5228 | 64.1385 | 125096657 | AAPL | 63.9282 |
| 12 | 2013-02-27 | 64.0614 | 64.6342 | 62.9499 | 63.5099 | 146674682 | AAPL | 63.8059 |
| 13 | 2013-02-28 | 63.4357 | 63.9814 | 63.0571 | 63.0571 | 80532382 | AAPL | 63.6728 |
| 14 | 2013-03-01 | 62.5714 | 62.5971 | 61.4257 | 61.4957 | 137899041 | AAPL | 63.0917 |
| 15 | 2013-03-04 | 61.1142 | 61.1714 | 59.8571 | 60.0071 | 145406366 | AAPL | 62.4417 |
| 16 | 2013-03-05 | 60.2114 | 62.1699 | 60.1071 | 61.5919 | 159298020 | AAPL | 61.9323 |
| 17 | 2013-03-06 | 62.0728 | 62.1785 | 60.6328 | 60.8088 | 114903180 | AAPL | 61.3921 |
| 18 | 2013-03-07 | 60.6428 | 61.7157 | 60.1514 | 61.5117 | 116992841 | AAPL | 61.0830 |
| 19 | 2013-03-08 | 61.3999 | 62.2042 | 61.2299 | 61.6742 | 97854442 | AAPL | 61.1187 |


Streamlit
share.streamlit.io/mchilamwar/optimizing_stock_trading_using_reinforcement_learning/main/stocks_deploy.py
Choose Stock and Investment
Choose Wich Company Stocks to analyse
(*S&P 500 only)
AAPL
Show Raw Data
Show stock Trend
Enter Your Available Initial Investment Fund
(Min 1000)
1000
Calculate
Optimization of Stock Trading Using Reinforcement-Learning
Here this model will improvise your normal Trading Strategies and will increase your net profit by RL techniques
Stock Trend of Selected Company
Price ($)
Date
Stock Showing Good Uptrend Its good to invest Here


Streamlit
share.streamlit.io/mchilamwar/optimizing_stock_trading_using_reinforcement_learning/main/stocks_deploy.py
Date
Choose Stock and Investment
Choose Wich Company Stocks to analyse
(*S&P 500 only)
AAPL
Show Raw Data
Show stock Trend
Enter Your Available Initial Investment Fund
(Min 1000)
1300
Calculate
Stock Showing Good Uptrend Its good to invest Here
Progress 100%
...and now we're done!
Change in Portfolio Value Day by Day
Value ($)
Number of Days since Feb 2013
This is your Networth Upmove by Decision Taken By the Model

Project Deployed Link: -

https://share.streamlit.io/mchilamwar/optimizing_stock_trading_using_reinforcement_learning/main/stocks_deploy.py