Medium   🔍 Search

Extract all listed companies in Indian Stock Market (NSE & BSE)

Arunesh Kumar Singh · Follow

3 min read · Jan 20, 2024

▶ Listen   ⬆ Share

Image credits: Moneycontrol

> *If you are building apps related to the stock markets, generally the first step is to get the list of all companies listed in the market. Manually, in India it is trivial to do. We can directly go to BSE and NSE exhange official websites and download it.*

1. **For NSE:** Check this link.

2. **For BSE:** Check this <u>link</u>.

It becomes challenging, when we start building apps on top of this, we may need to dynamically fetch it. Suppose, if we want to check the sentiment related to the newly listed company through IPO, the manual list can get outdated. Manually updating it every time would get tedious fast and would be nightmare to maintain.

To automate this, I explored multiple API's related for the Indian market but was not able to directly find a way to achieve it:

1. **yfinance** python api was promising but I couldn't find any direct method to get this list.

2. Also checked **screener.in** but it has no api and export option was available only on paid subscription.

3. There may be some paid options but I wanted open-source solutions .

Automate!

To get this, I wrote a simple python program which fetches the list from both the exchanges and merges it.

If you are trying, please make sure to download all the required *libraries* and *chrome web driver* before proceeding.

The code follows below steps:

1. **Fetch the list from the NSE.** First it seemed trivial to me after all, I just need to make a *get* call through *requests* library. But, simple *requests get* was not working. Turns out it needed a session may be for setting cookies etc. Finally below code worked:

```python
import requests
import pandas as pd
import io
```

```
nse_url = "https://nsearchives.nseindia.com/content/equities/EQUITY_L.csv"

# create Session from 'real' browser
headers = {
    'User-Agent': 'Mozilla/5.0'
}

s = requests.Session()
s.headers.update(headers)

# do a get call now
url = 'https://nsearchives.nseindia.com/content/equities/EQUITY_L.csv'
r = s.get(nse_url)
s.close()

# saving it to pd df for further preprocessing
df_nse = pd.read_csv(io.BytesIO(r.content))
```

2. **Fetch the list from BSE.** It was not trivial to automate it on BSE, manually I needed to fill two form fields selected **segment as equity** and **status as active** and then hit a submit button. After results gets loaded I needed to click a download icon on top left corner.

Automating this required a bit of effort and trial and error, I used selenium and splinter library to automate this.

```
from splinter import Browser
from selenium import webdriver
import time, os

download_dir = "/tmp"
bse_link = "'https://mock.bseindia.com/corporates/List_Scrips.html'"

# change download directory if required
prefs = {"download.default_directory": download_dir};

options = webdriver.ChromeOptions()
options.add_experimental_option("prefs", prefs)

# intiate browser
browser = Browser('chrome', options=options, headless=True)
```

```
# visit link
browser.visit(bse_link)

# fill out form fields
browser.find_by_id('ddlsegment').select("Equity")
browser.find_by_id('ddlstatus').select("Active")

# hit submit button
browser.find_by_id('btnSubmit').click()

# let the table load
browser.is_element_present_by_text("Issuer Name")
time.sleep(5)

# download
browser.find_by_id('lnkDownload').click()

df_bse = pd.read_csv(os.path.join(download_dir, "Equity.csv"))
```

3. Next step was to *merge* this list and refine it according to requirements. My requirement was to get all the names from this list which I can easily get for further upstream processing.

```
# some columns naming convention was different
df_bse = df_bse.rename(columns={
    "Security Name": "NAME OF COMPANY",
    "Security Id": "SYMBOL"
})

# merged on SYMBOL
final_df = pd.merge(df_nse, df_bse, on='SYMBOL',  how="outer")
```

Results

1. I was able to fetch 4253 securities from BSE and 1934 from NSE.

2. 1802 companies were listed on both NSE and BSE.

3. After merging (outer join) final list contained 4385 rows.

This final list I am going to take for further preprocessing and build any financial apps on top of this.

Note:- Hope this article helped you. Please like and leave feedback in comments.

Stock Market   National Stock Exchange   Indian Stock Market

Bombay Stock Exchange   Share Market Tips

Follow

Written by Arunesh Kumar Singh

6 Followers   ·   3 Following

Responses (3)

What are your thoughts?   Respond

Shivam Gupta
Oct 6, 2024

like after deploying it as an API anywhere. It is not able to make a connection with the NSE url.locally it is able to but on a cloud server it is not able to. I tried to deploy it on vercel,netlify,AWS it is not able to build a connection with NSE... more

Reply

Shivam Gupta
Sep 7, 2024

Hi Arunesh, I also wrote the same python program and it worked locally but when i hosted it as an API on vercel then it does not work? where did you host the code?

1 reply   Reply

Lovish Jain he/him
Jun 26, 2024 (edited)

Very nicely explained.

Reply

More from Arunesh Kumar Singh

Arunesh Kumar Singh

Voice Cloning Using AI for Hindi

Nowadays, Voice cloning is one of the popular use cases of the AI especially after Generative AI Boom. This is useful for personalisation...

May 5, 2024

See all from Arunesh Kumar Singh

Recommended from Medium

Shantanu Patra

Automated Daily Market Summary Alerts with Python: A Step-by-Step Guide

Generating an automated email alerts on market summary as on scheduled time.

In InsiderFinance Wire by GunKurnia

Stock Prediction Using News Sentiment Analysis: Capturing Momentum from Market News

⭐ Sep 12, 2024   👋 61

Lists

Staff picks
808 stories · 1616 saves

Stories to Help You Level-Up at Work
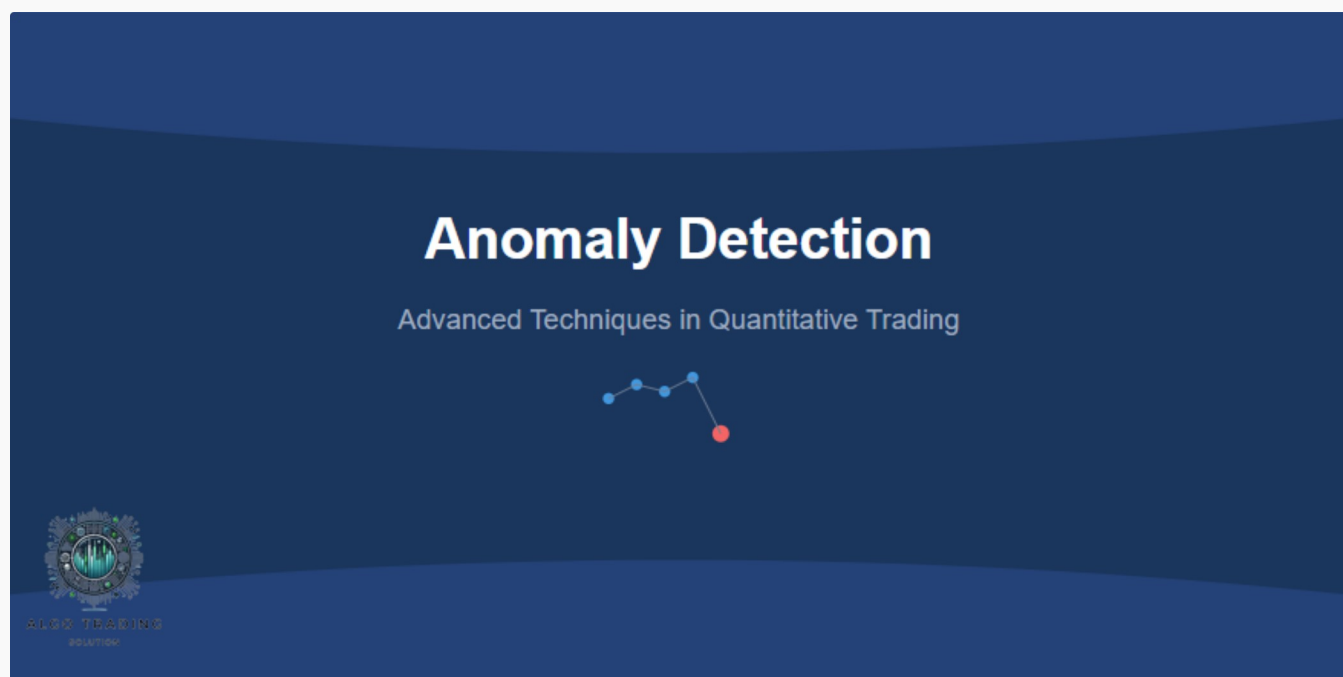19 stories · 933 saves

Self-Improvement 101
20 stories · 3282 saves

Productivity 101

20 stories · 2764 saves

🟦 In Funny AI & Quant by Pham The Anh

Anomaly Detection in Quantitative Trading: Advanced Techniques and Applications

Leveraging Machine Learning for Identifying Market Irregularities and Trading Opportunities in High-Frequency Financial Data

✦ Jan 16   👏 3

Sudarshan Sharma Paudel

Stock Data Collection using Yahoo Finance and Sentiment Analysis for Price Prediction

Combining Historical Data and Social Media Insights for Enhanced Forecasting
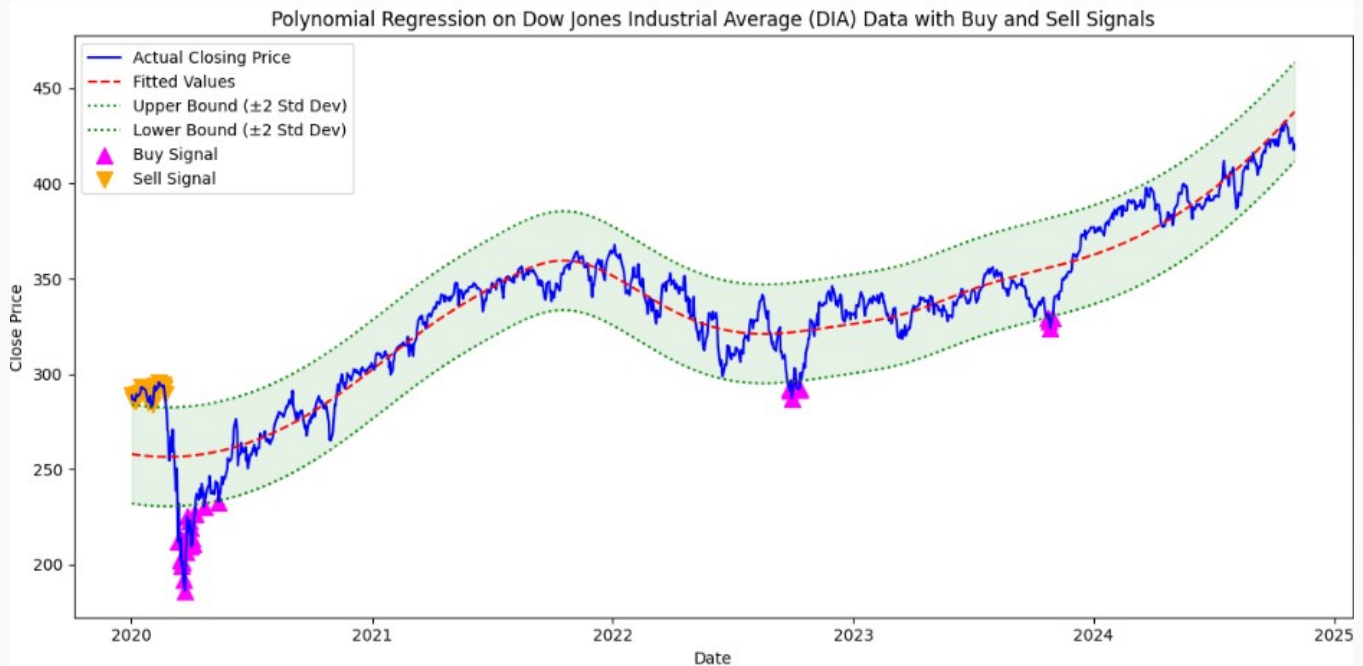
Oct 29, 2024   👏 11

Mdabdullahalhasib

A Complete Guide of Chain with LangChain

Create a simple chain, understand what's happening under the hood, and explore the chain types with implementations.

✦ Oct 4, 2024   👏 104

Unicorn Day

Trading Smarter, Not Harder: How Machine Learning is Revolutionizing Market Predictions

Ever wondered if there's a better way to time the market? Let me introduce you to a fascinating approach that combines the power of AI with...

✦ Nov 5, 2024   👏 396   💬 8

See more recommendations