# SWE Python Intern Assignment

## After all these details of each subtask i will tell you how to run the code.

## Subtask 1: Fetching and Storing Option Chain Data :

In this subtask, I developed a script that fetches option chain data for **NIFTY** and **HDFCBANK** from the NSE India website at regular intervals of 3 minutes.

- **Function `get_cookie(sym)`**: This function retrieves the necessary cookies for authenticating requests based on the selected symbol.
- **Function `gen_headers(sym)`**: This generates headers required for the HTTP request, ensuring that the requests are formatted correctly for the NSE API.
- **Function `fetch_data(url, hdrs, fname)`**: This function handles the actual fetching of data from the API, checking for successful responses and saving the data to JSON files.
- **Function `run(test=False)`**: This function orchestrates the continuous fetching of data in a loop unless in testing mode, where it fetches the data once.

```
Running in testing mode...
Data is being saved to NIFTY_option_chain.json
Data is being saved to HDFCBANK_option_chain.json
Testing is complete.
```

Our fetched data will be saved to two json files. Now every time when 3 minutes complete it overwrites the json files to update the updated data.

## Subtask 2: Parsing and Structuring Data

In this subtask we need to  focus on extracting relevant information from the JSON data and organizing it into a structured format using Pandas.So i written these functions to complete this subtask.

- **Function `load_json_data(filename)`**: This function loads the JSON data from the previously saved files.
- **Function `extract_current_price(data)`**: This extracts the current price of the underlying asset.

- **Function `extract_option_chain_data(option_chain)`**: This processes the option chain data and prepares it in a list of dictionaries containing crucial metrics like `call_open_interest`, `put_open_interest`, `call_ltp`, etc.
- **Function `create_dataframe(rows)`**: This creates a Pandas DataFrame from the structured data.
- **Function `filter_by_expiry(data, expiry_dates)`**: This filters the data to retain only the rows relevant to the next weekly expiry.
- **Function `filter_above_below_current(data, current_price)`**: This function identifies and separates the options that are above and below the current price.

**These are the screenshots of results after running the script :**

```
Rows above current price for NIFTY:
     strikePrice   expiryDate  call_open_interest  put_open_interest  \
223        24250  31-Oct-2024              546885             238658
230        24300  31-Oct-2024              455022             204197
237        24350  31-Oct-2024              204694              43889
244        24400  31-Oct-2024              329263              80583
251        24450  31-Oct-2024              202343              58580

     call_ltp  put_ltp  call_change_in_oi  put_change_in_oi  call_volume  \
223      0.05    44.40             523105            153543     40482544
230      0.05    94.30             331596            -59996     31120610
237      0.05   144.65             107518            -58721     17267797
244      0.05   195.00              47152           -115114     14694181
251      0.05   245.05              17851            -19759      9246152

     put_volume
223    27291942
230    16729744
237     4507637
244     2927424
251      541078
```

Rows below current price for NIFTY:

| | strikePrice | expiryDate | call_open_interest | put_open_interest | call_ltp | put_ltp | call_change_in_oi | put_change_in_oi | call_volume | put_volume |
|---|---|---|---|---|---|---|---|---|---|---|
| 184 | 24000 | 31-Oct-2024 | 43946 | 302701 | 204.60 | 0.05 | 5829 | -86278 | 482236 | 9472555 |
| 196 | 24050 | 31-Oct-2024 | 13973 | 108342 | 154.50 | 0.05 | 4501 | 40511 | 274758 | 7340841 |
| 202 | 24100 | 31-Oct-2024 | 80231 | 288992 | 105.05 | 0.05 | 46976 | 84503 | 2296475 | 17067170 |
| 209 | 24150 | 31-Oct-2024 | 60893 | 265558 | 55.20 | 0.05 | 44213 | 185780 | 6429666 | 31676547 |
| 216 | 24200 | 31-Oct-2024 | 702886 | 763317 | 5.30 | 0.05 | 629732 | 538970 | 45018479 | 64328556 |

Rows above current price for HDFCBANK:

| | strikePrice | expiryDate | call_open_interest | put_open_interest | call_ltp | put_ltp | call_change_in_oi | put_change_in_oi | call_volume | put_volume |
|---|---|---|---|---|---|---|---|---|---|---|
| 65 | 1740 | 31-Oct-2024 | 1987 | 1114 | 0.05 | 1.80 | -1298 | -340 | 6688 | 1228 |
| 68 | 1750 | 31-Oct-2024 | 3655 | 864 | 0.05 | 13.00 | -1299 | -298 | 8252 | 743 |
| 70 | 1760 | 31-Oct-2024 | 1919 | 648 | 0.05 | 20.35 | -1402 | -347 | 4769 | 1066 |
| 72 | 1770 | 31-Oct-2024 | 1468 | 533 | 0.05 | 34.35 | -637 | -45 | 1470 | 134 |
| 74 | 1780 | 31-Oct-2024 | 9757 | 389 | 0.05 | 47.30 | -78 | -38 | 421 | 129 |

Rows below current price for HDFCBANK:

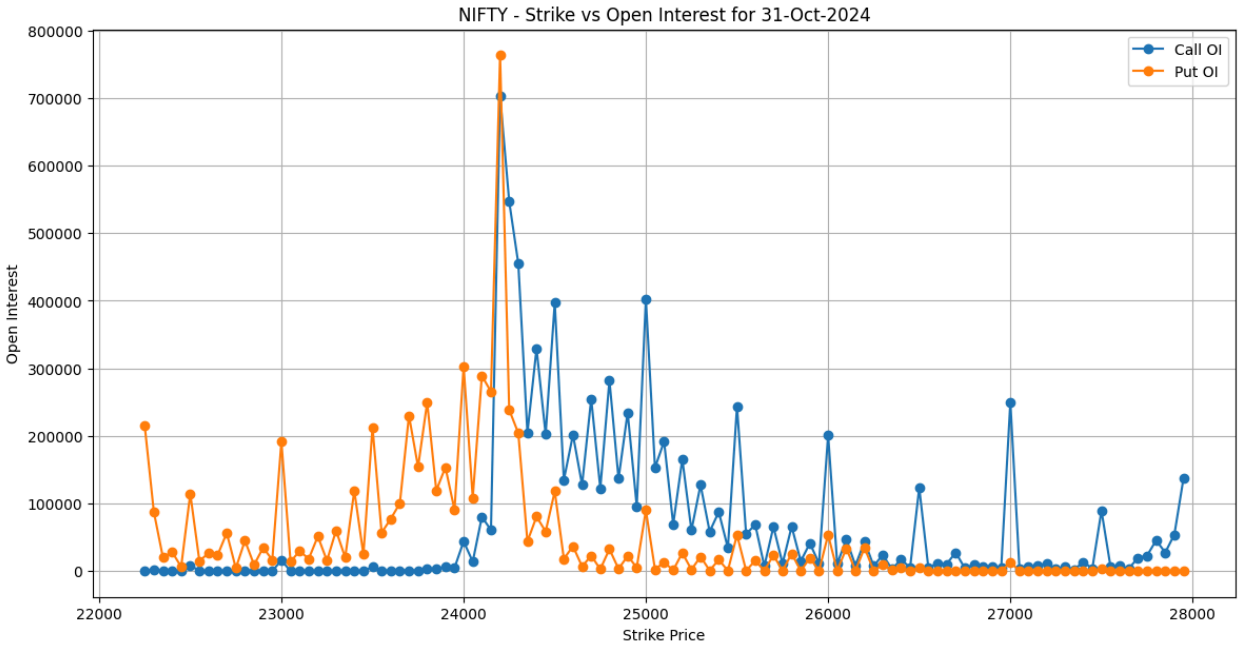
| | strikePrice | expiryDate | call_open_interest | put_open_interest | call_ltp | put_ltp | call_change_in_oi | put_change_in_oi | call_volume | put_volume |
|---|---|---|---|---|---|---|---|---|---|---|
| 53 | 1690 | 31-Oct-2024 | 1812 | 1056 | 49.50 | 0.05 | -21 | -106 | 84 | 182 |
| 55 | 1700 | 31-Oct-2024 | 3503 | 3144 | 34.00 | 0.05 | -469 | -675 | 1165 | 2059 |
| 58 | 1710 | 31-Oct-2024 | 688 | 772 | 27.65 | 0.05 | -19 | -215 | 159 | 1569 |
| 60 | 1720 | 31-Oct-2024 | 1617 | 1382 | 16.80 | 0.05 | -153 | -785 | 1530 | 4221 |
| 63 | 1730 | 31-Oct-2024 | 1974 | 1404 | 4.05 | 0.05 | -212 | -1106 | 5012 | 4299 |

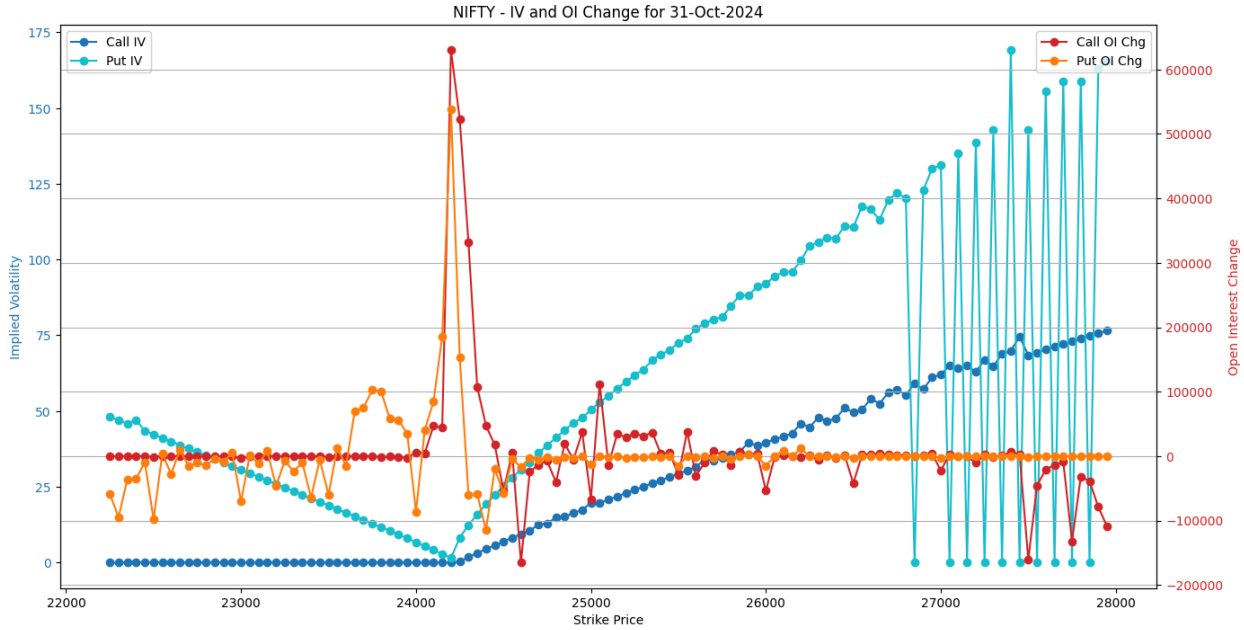
## Subtask 3: Data Visualization

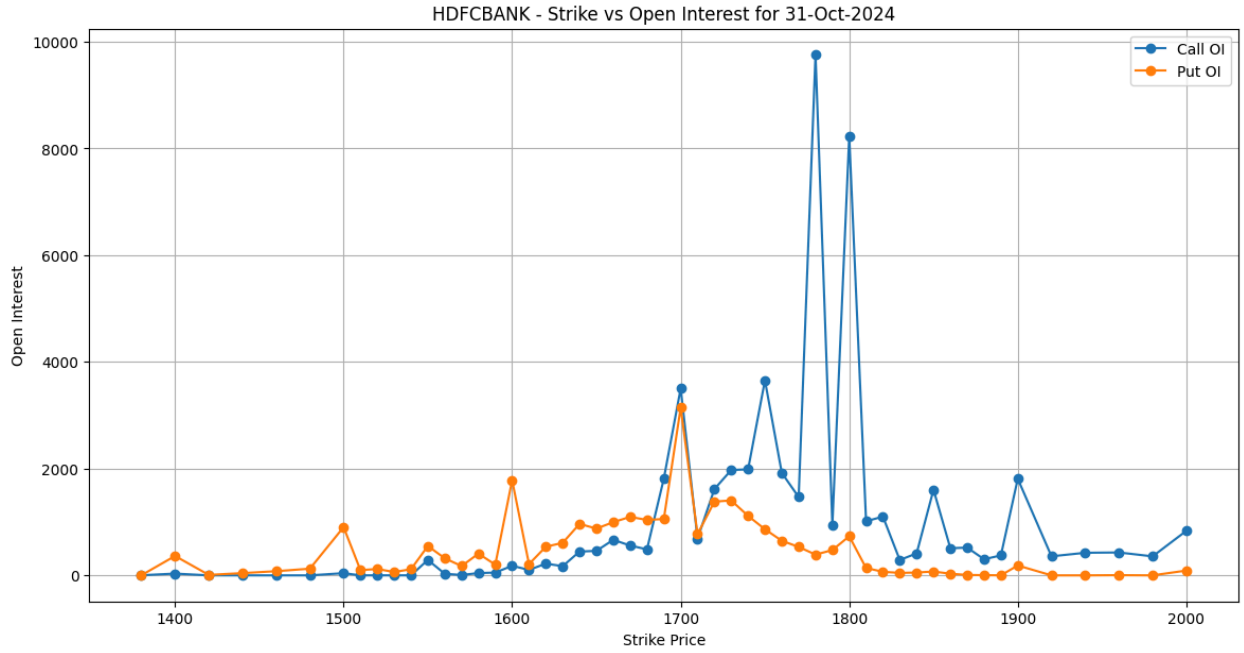
In this subtask, I implemented data visualization using Matplotlib to provide insights into the option chain data.

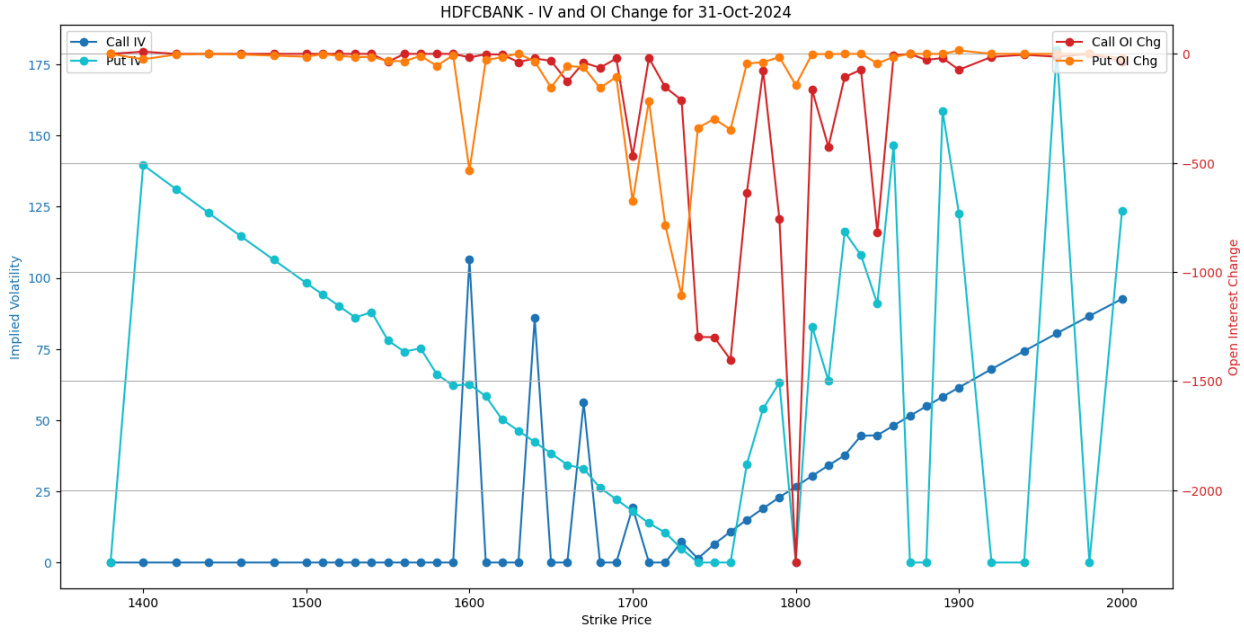
- **Function `load_data(file_path)`**: This loads the JSON data to be visualized.
- **Function `extract_data(data)`**: This prepares the data for visualization, focusing on relevant metrics like open interest and implied volatility.
- **Function `plot_oi(df, sym, expiry)`**: This visualizes the open interest for both call and put options.
- **Function `plot_iv_oi(df, sym, expiry)`**: This creates plots for implied volatility and changes in open interest, providing a comprehensive view of the options market.
- **Overall Structure**: My code is designed to create clear, informative plots based on the filtered DataFrame, enabling easy interpretation of the data.

**These are the screenshots of the result :**




NIFTY - IV and OI Change for 31-Oct-2024
Call IV
Put IV
Call OI Chg
Put OI Chg
Implied Volatility
Open Interest Change
Strike Price
175
150
125
100
75
50
25
0
600000
500000
400000
300000
200000
100000
0
−100000
−200000
22000
23000
24000
25000
26000
27000
28000


HDFCBANK - Strike vs Open Interest for 31-Oct-2024
Call OI
Put OI
Open Interest
Strike Price
10000
8000
6000
4000
2000
0
1400
1500
1600
1700
1800
1900
2000

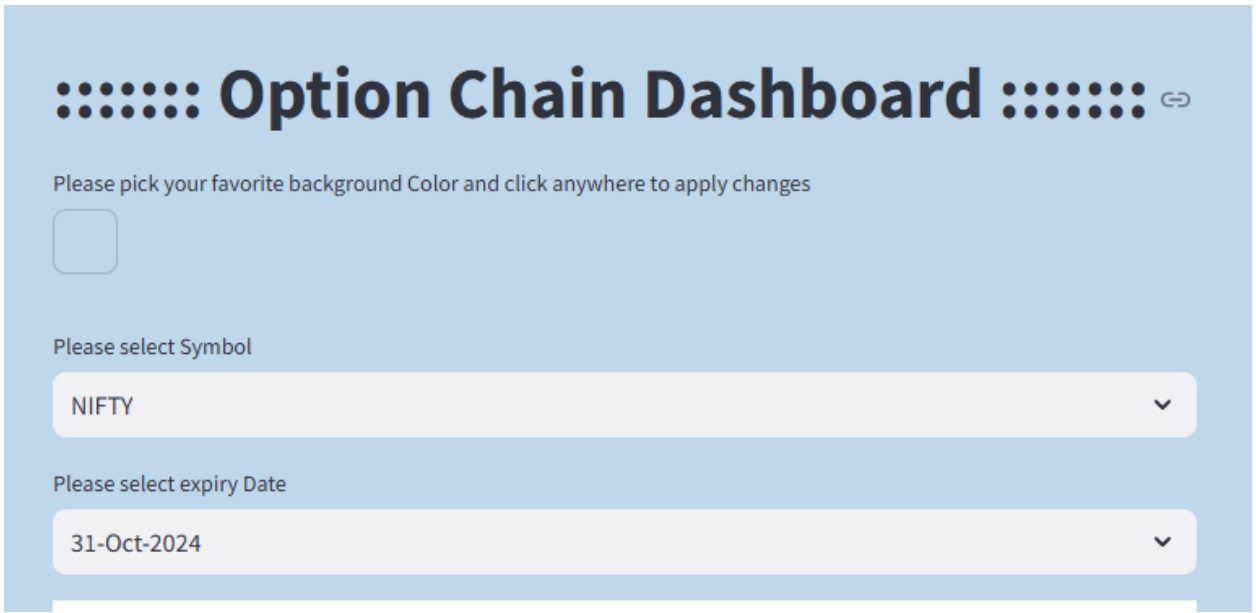

HDFCBANK - IV and OI Change for 31-Oct-2024
Call IV
Put IV
Call OI Chg
Put OI Chg
Implied Volatility
Open Interest Change
Strike Price
0
25
50
75
100
125
150
175
−2000
−1500
−1000
−500
0
1400
1500
1600
1700
1800
1900
2000

### Subtask 4: Interactive Dashboard

The final subtask I developed an interactive dashboard using Streamlit, allowing users to interact with the option chain data dynamically.

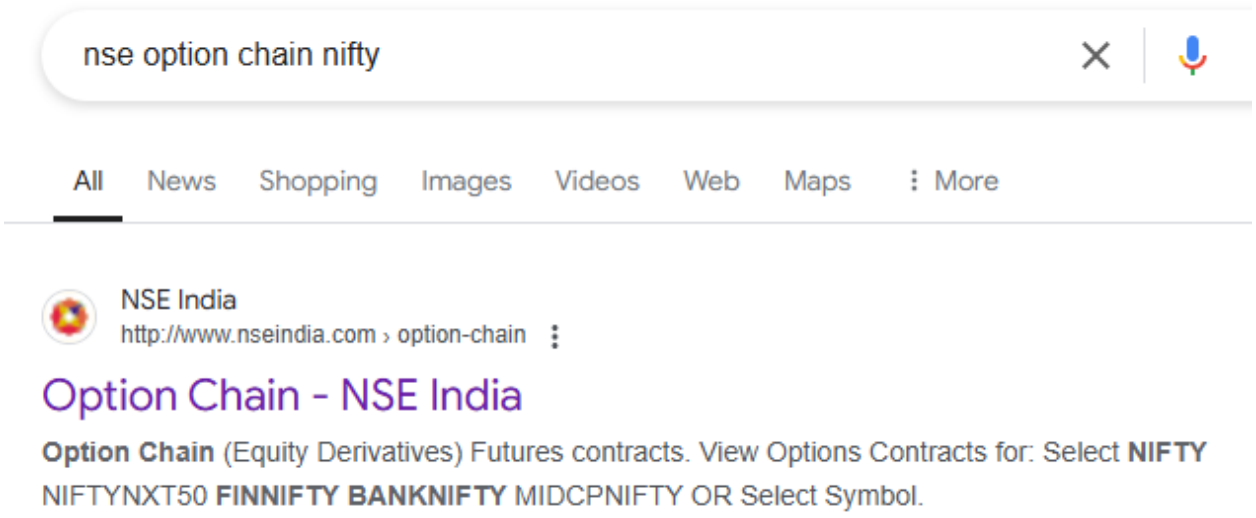
- **Function `load_and_prepare_data(file_path, selected_expiry)`**: This function prepares the option chain data for the selected expiry, ensuring that only relevant information is displayed.
- **Streamlit Interface**: The dashboard includes user-friendly features like color selection for the background and dropdowns for symbol and expiry date selection.
- **Plotting with Plotly**: The dashboard utilizes Plotly for creating interactive visualizations of the option chain data, enhancing user engagement and experience.
- **Dynamic Updates**: The dashboard allows real-time updates and interaction, ensuring that users can see the latest data and visualizations based on their selections.

**The dashboard look like this :**

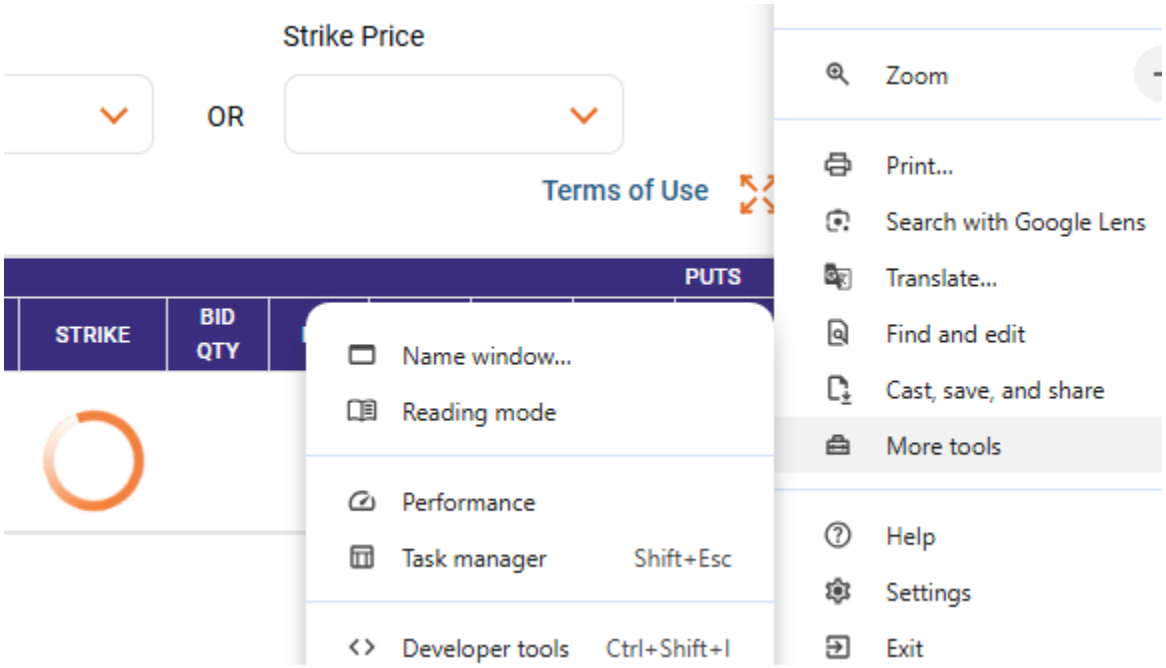


How to get cookies for Nifty :

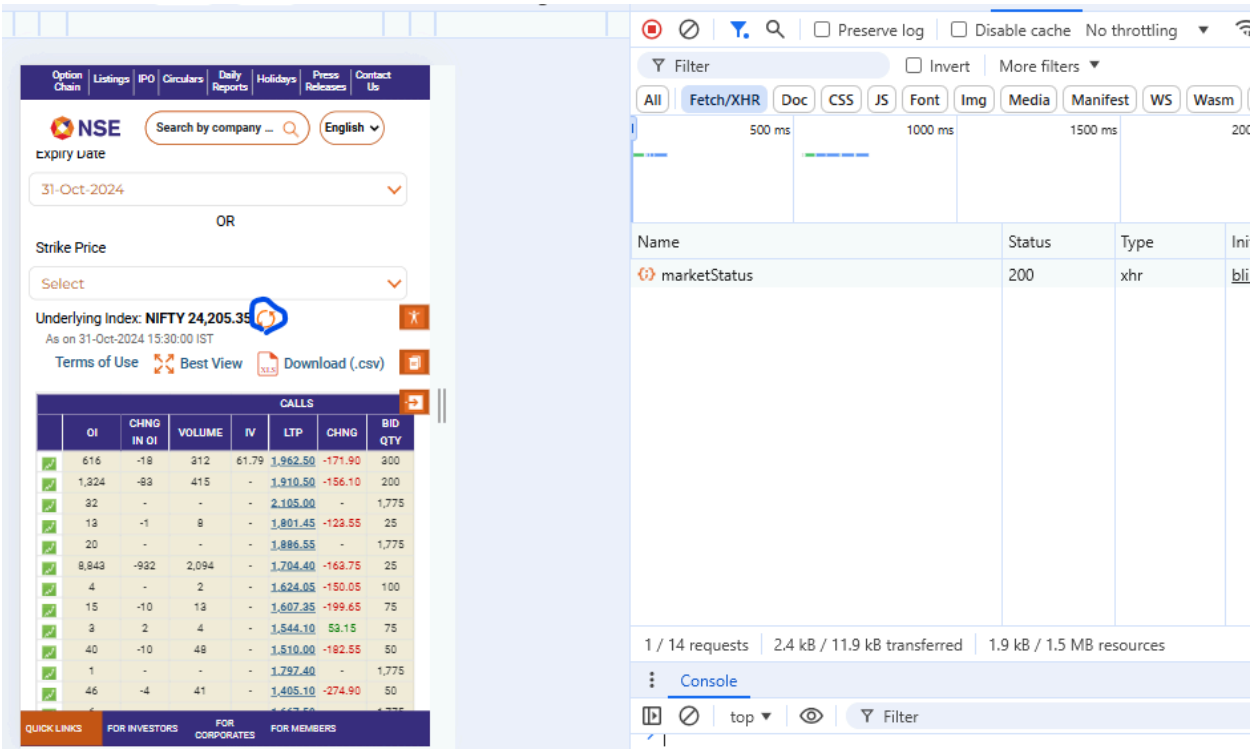
Steps :



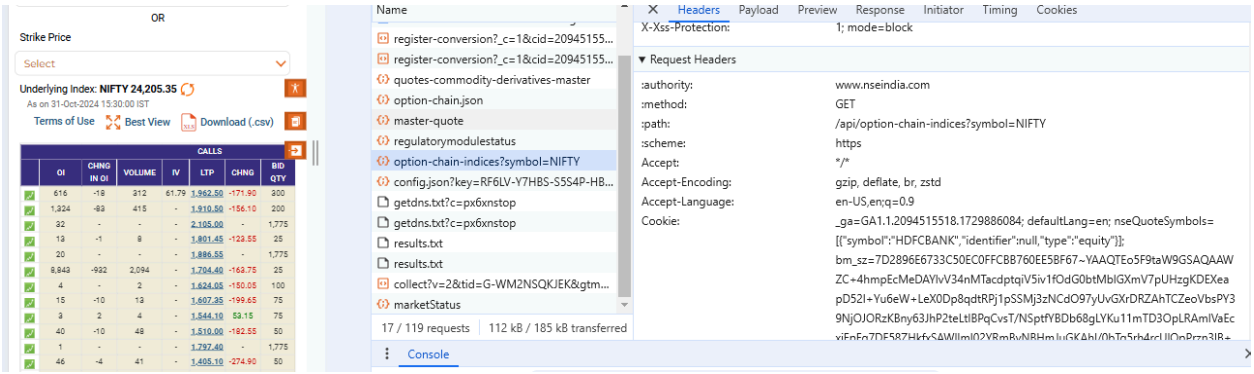

After going to the webpage check for developer tools :



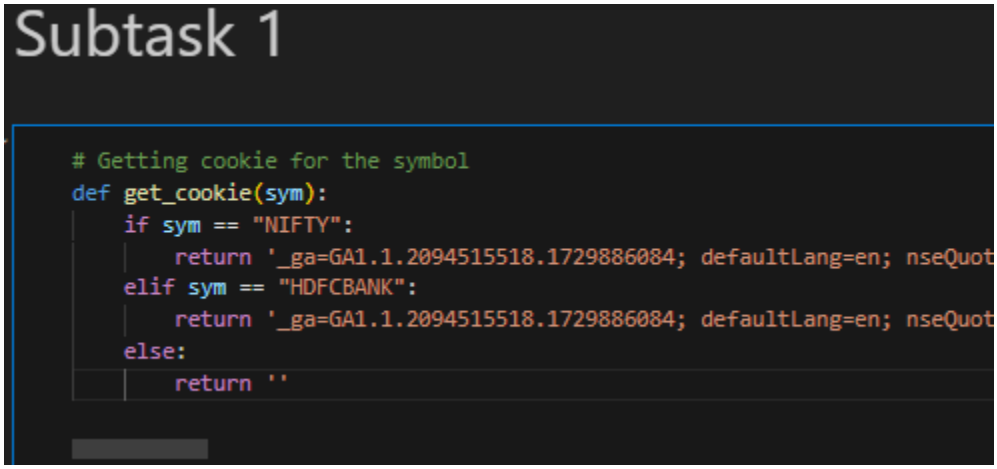
After this go to networks click on refresh blue ticked :




Go to option-chain-indices?symbol=NIFTY and scroll to request headers then copy the cookie :



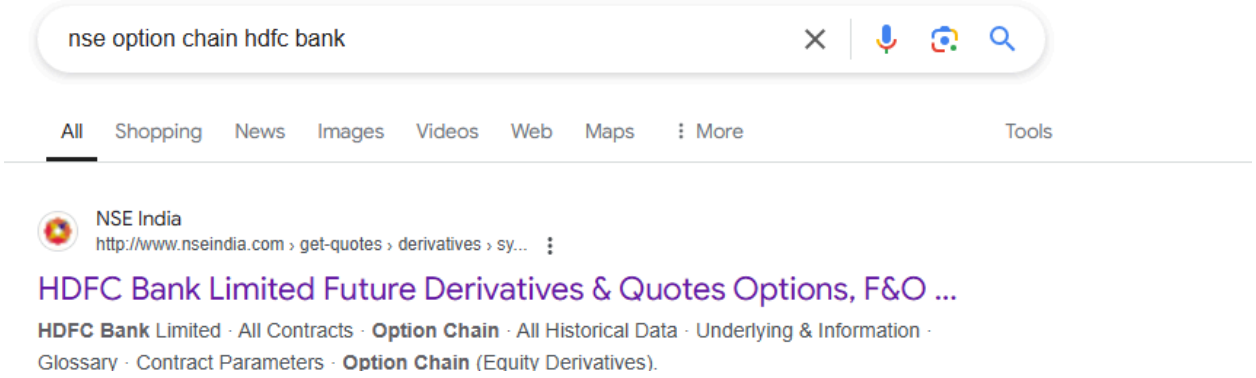
And finally paste this cooki here in the code  (return ‘cookie’) NIFTY part :



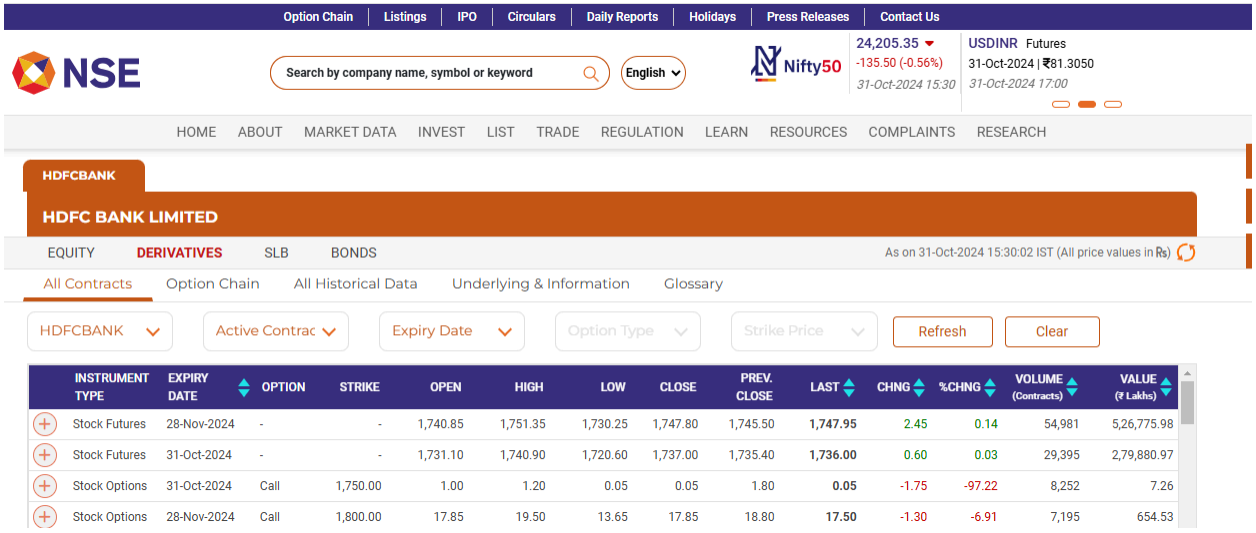

For HDFCBANK part cookie mentioned below , after getting cookie you can paste here in sym = HDFCBANK

### Same for getting HDFCBANK , i will show you some steps :

Go to google and search ;

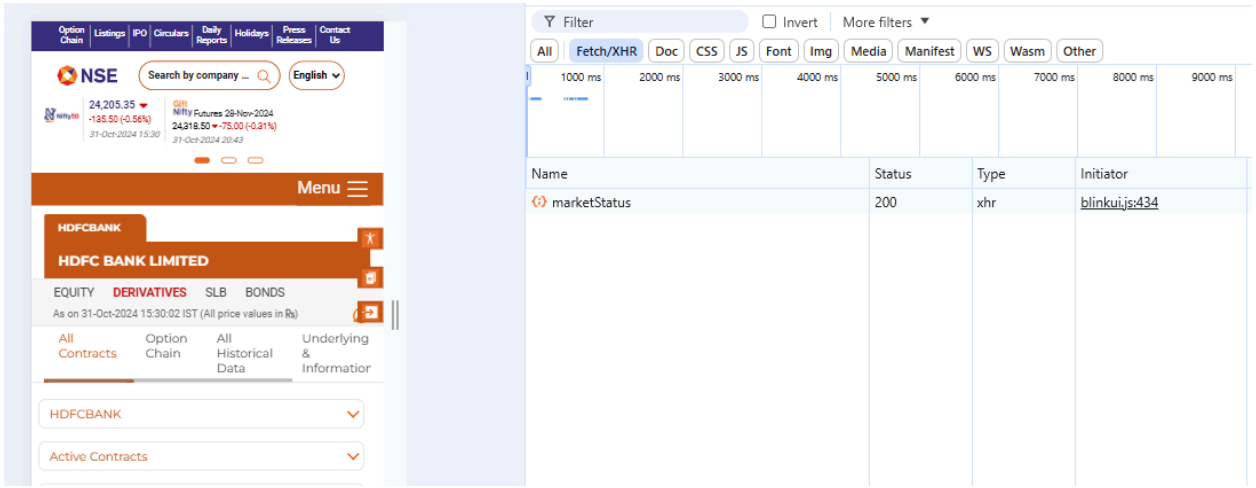



The page :

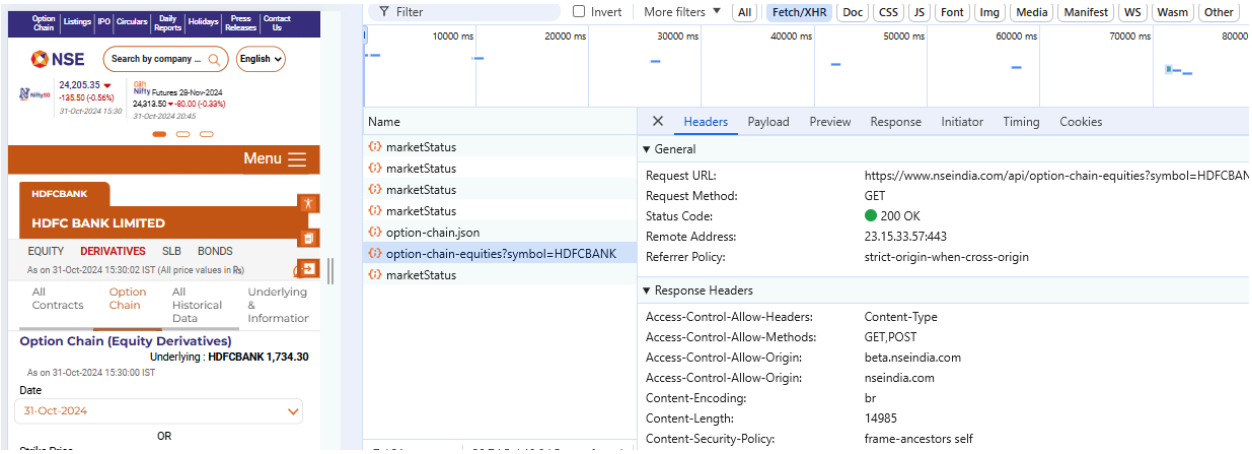



| | INSTRUMENT TYPE | EXPIRY DATE | OPTION | STRIKE | OPEN | HIGH | LOW | CLOSE | PREV. CLOSE | LAST | CHNG | %CHNG | VOLUME (Contracts) | VALUE (₹ Lakhs) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ⊕ | Stock Futures | 28-Nov-2024 | - | - | 1,740.85 | 1,751.35 | 1,730.25 | 1,747.80 | 1,745.50 | **1,747.95** | 2.45 | 0.14 | 54,981 | 5,26,775.98 |
| ⊕ | Stock Futures | 31-Oct-2024 | - | - | 1,731.10 | 1,740.90 | 1,720.60 | 1,737.00 | 1,735.40 | **1,736.00** | 0.60 | 0.03 | 29,395 | 2,79,880.97 |
| ⊕ | Stock Options | 31-Oct-2024 | Call | 1,750.00 | 1.00 | 1.20 | 0.05 | 0.05 | 1.80 | **0.05** | -1.75 | -97.22 | 8,252 | 7.26 |
| ⊕ | Stock Options | 28-Nov-2024 | Call | 1,800.00 | 17.85 | 19.50 | 13.65 | 17.85 | 18.80 | **17.50** | -1.30 | -6.91 | 7,195 | 654.53 |

After going to developer tool , click on option chain side to All Contracts :




Then go to option-chain-equities?symbol=HDFCBANK and fetch cookie as we told for NIFTY :


Option Chain
Listings
IPO
Circulars
Daily Reports
Holidays
Press Releases
Contact Us
NSE
Search by company ...
English
24,205.35
-135.50 (-0.56%)
31-Oct-2024 15:30
Nifty Futures 28-Nov-2024
24,313.50 -80.00 (-0.33%)
31-Oct-2024 20:45
Menu
HDFCBANK
HDFC BANK LIMITED
EQUITY
DERIVATIVES
SLB
BONDS
As on 31-Oct-2024 15:30:02 IST (All price values in Rs)
All Contracts
Option Chain
All Historical Data
Underlying & Informatior
Option Chain (Equity Derivatives)
Underlying : HDFCBANK 1,734.30
As on 31-Oct-2024 15:30:00 IST
Date
31-Oct-2024
OR
Filter
Invert
More filters
All
Fetch/XHR
Doc
CSS
JS
Font
Img
Media
Manifest
WS
Wasm
Other
10000 ms
20000 ms
30000 ms
40000 ms
50000 ms
60000 ms
70000 ms
80000
Name
marketStatus
marketStatus
marketStatus
marketStatus
option-chain.json
option-chain-equities?symbol=HDFCBANK
marketStatus
Headers
Payload
Preview
Response
Initiator
Timing
Cookies
General
Request URL:
https://www.nseindia.com/api/option-chain-equities?symbol=HDFCBAN
Request Method:
GET
Status Code:
200 OK
Remote Address:
23.15.33.57:443
Referrer Policy:
strict-origin-when-cross-origin
Response Headers
Access-Control-Allow-Headers:
Content-Type
Access-Control-Allow-Methods:
GET,POST
Access-Control-Allow-Origin:
beta.nseindia.com
Access-Control-Allow-Origin:
nseindia.com
Content-Encoding:
br
Content-Length:
14985
Content-Security-Policy:
frame-ancestors self