**OptionsChainProject Documentation**

## Project Overview

**OptionsChainProject** retrieves options chain data from the Upstox API for a specified underlying symbol and expiry date and calculates margin based on options open interest. This project aims to provide a straightforward approach to fetching and processing financial data using an API.

## Project Structure

plaintext

Copy code

OptionsChainProject/

├── main.py # Main script to execute API call and perform margin calculation

├── get\_options\_data.py # Module to fetch option chain data from Upstox API

├── requirements.txt # Required Python packages

└── README.md # Documentation, including AI assistance notes

## Setup Instructions

**Install Dependencies**: Create a file named requirements.txt with the following content:  
plaintext  
Copy code  
requests

Then, install the required packages using:  
bash  
Copy code  
pip install -r requirements.txt

1. **Obtain Upstox API Access Token**:
   * Register on Upstox’s Developer Console to get API access.
   * Follow the Upstox API Token API documentation to obtain your access token.
   * Replace ACCESS\_TOKEN in get\_options\_data.py with your actual token.

## Code Overview

### get\_options\_data.py

This module is responsible for fetching the options chain data based on the underlying symbol and expiry date. Here’s a breakdown of the code:

python

Copy code

import requests

# Set up your access token and API base URL

ACCESS\_TOKEN = "your\_access\_token" # Replace with your actual access token

BASE\_URL = "https://api.upstox.com/live/options"

def get\_option\_chain(instrument\_key, expiry\_date):

"""Fetches the options chain data for a given instrument and expiry date."""

headers = {

"Authorization": f"Bearer {ACCESS\_TOKEN}",

"Accept": "application/json"

}

params = {

"instrument\_key": instrument\_key,

"expiry\_date": expiry\_date

}

try:

response = requests.get(f"{BASE\_URL}/get-pc-option-chain", headers=headers, params=params)

response.raise\_for\_status()

data = response.json()

if data.get("status") == "success":

return data.get("data", [])

else:

print("Request failed:", data)

except requests.RequestException as e:

print(f"API request error: {e}")

return None

### main.py

This script uses the get\_option\_chain() function to fetch the options chain data and then processes this data to calculate a margin based on open interest:

python

Copy code

from get\_options\_data import get\_option\_chain

def calculate\_margin(options\_data):

"""Calculates margin based on open interest in the options chain data."""

margin = sum(option['put\_options']['market\_data']['oi'] for option in options\_data)

return margin

def main():

instrument\_key = "NSE\_INDEX|Nifty 50" # Example symbol

expiry\_date = "2024-02-15" # Example expiry date

# Fetch options data

options\_data = get\_option\_chain(instrument\_key, expiry\_date)

if options\_data:

margin = calculate\_margin(options\_data)

print(f"Calculated Margin: {margin}")

else:

print("No options data available.")

if \_\_name\_\_ == "\_\_main\_\_":

main()

## Usage

1. Ensure that the ACCESS\_TOKEN is set correctly in get\_options\_data.py.
2. Run the main script to fetch the options chain data and calculate the margin:

bash

Copy code

python main.py

## AI Assistance Documentation

This project utilized AI tools, such as ChatGPT, in several areas:

1. **Function Skeleton Creation**: The initial structure for get\_option\_chain was generated using AI assistance.
2. **Syntax Guidance**: AI helped with correct Python syntax, especially in handling API requests and exceptions.
3. **Financial Terminology**: AI provided explanations for stock market terms such as open interest, call/put options, and strike price.
4. **Debugging Assistance**: AI verified header and parameter formatting for the Upstox API.

Each instance of AI assistance aimed to expedite development and clarify financial concepts and code debugging.

## Example Output

When you run main.py, the script should output the calculated margin based on the options data retrieved:

plaintext

Copy code

Calculated Margin: 3957550

## Troubleshooting

* **API Request Errors**: Ensure your ACCESS\_TOKEN is valid and has the required permissions.
* **No Data Available**: If no options data is returned, check the instrument\_key and expiry\_date values for correctness.

## License

This project is licensed under the MIT License. See the LICENSE file for more details.