US STOCK RISK ANALYZER USNIG MACHINE LEARNING – HOW TO RUN

1. Project Setup

- Save all project files in a single folder.
- Ensure the folder contains all required Python files (data collection, preprocessing, feature engineering, labeling, model training, and Flask app files), as well as an HTML template folder for the UI.

2. Install Dependencies

Install all required Python libraries using pip. The key dependencies include:

- Flask
- yfinance
- pandas
- scikit-learn
- joblib
- pandas-ta (for technical indicators)
- numpy

Tip: Create and activate a virtual environment before installation for better project management.

3. Data Preprocessing

- Historical stock data is automatically collected from Yahoo Finance using the provided ticker symbol.
- The data is cleaned and missing values are removed.
- Technical indicators like Moving Averages, RSI, MACD, and Bollinger Bands are added.

4. Train the Model

- Use a predefined list of stock tickers to train the risk classification model.
- The model analyzes features such as daily return, volatility, and moving averages to determine risk.
- The trained model is saved in the models directory.

5. Launch the Flask App

- Run the Flask application.
- This opens a web interface where users can input a stock ticker.

6. Use the Web Interface

- Enter a stock ticker (e.g., AAPL, MSFT) and submit.
- The app fetches the stock data, processes it, and returns:
    - Current stock price
    - Daily return
    - Volatility
    - Predicted risk level (Low, Medium, High)
- A price chart for the last 30 days is also displayed.

7. API Access (Optional)

- The app also provides a REST API endpoint.
- You can access the risk analysis by sending a GET request to /api/analyze/<ticker>.

Output

- Web-based dashboard with risk predictions and price charts.
- JSON response for API users with risk level and recent stock trends.