Financial Data Extraction and Reporting Tool

1. Overview

This notebook is a financial data analysis tool hat:

1. Extracts cryptocurrency price data from the CoinGecko API.

2. Visualizes trends using `matplotlib`.

3. Generates an AI-powered report using OpenAI's GPT-4 model.

The tool is designed for investors, traders, and analysts who need quick insights into cryptocurrency price movements without manual data processing.

2. Key Features

2.1 Data Extraction

- Source: CoinGecko API (free tier available).

- Data Collected:

  - Timestamp (converted to readable dates).

  - Price in selected currency (default: USD).

- Processing:

  - Aggregates raw price data into daily averages using `pandas`.

  - Handles missing data points gracefully.

  - Returns a clean `DataFrame` for further analysis.

Example Output:

Date	Price (USD)
2024-05-01	60,123.45
2024-05-02	61,200.78

2.2 Data Visualization

- Chart Type:Line graph (`matplotlib`).

- Customizations:

  - Title: Asset name + time period (e.g., “Bitcoin Price – Last 30 Days”).

  - Axes Labels:Date (x-axis), Price in USD (y-axis).

  - Styling:

    - Grid lines for readability.

    - Marker points for daily prices.

    - Auto-adjusted date formatting.

- Output:

  - Saved as `trend.png` for reports.

  - Displayed inline in Jupyter/Colab.

Why This Matters:

- Helps users **quickly identify trends (bullish/bearish phases).

- Professional-looking visuals for presentations.

2.3 Report Generation (AI-Powered Insights)

- Input:

  - Price change percentage (e.g., “+15% over 30 days”).

  - Structured prompt for GPT-4.

- AI Analysis:

  - Reasons for Trend:

    - Market sentiment (e.g., ETF approvals, macroeconomic factors).

    - Technical patterns (support/resistance levels).

  - Investment Recommendation:

    - “Hold if long-term bullish, take profits if overbought.”

- Output:

- Plain English summary (easy for non-technical users).

Example Report:

- ”Bitcoin rose 15% in the last 30 days, likely due to increased institutional adoption. Investors might consider holding but monitor resistance at $65K.”

3. Technical Implementation

3.1 Dependencies

Library	**Purpose**
Requests	API calls(CoinGecko)
pandas	Data cleaning & aggregation
matplotlib	Price trend visualization
openai	GPT-4 report generation
datetime	Time formatting

3.2 Configuration

- User-Adjustable Parameters:

ASSET = “bitcoin”. # Can switch to “ethereum”, “solana”, etc.

CURRENCY = “usd” # Supports EUR, GBP, JPY.

DAYS = 30 # Flexible time window (7, 30, 90, etc.).

- API Key Management:

- Securely stored in Google Colab secrets (`userdata.get(‘OPENAI_API_KEY’)`).

3.3 Error Handling

- API Failures:

- Retry logic for rate limits.

- Fallback to cached data if live fetch fails.

- GPT-4 Errors:

- Graceful degradation (e.g., “Report unavailable”).

4. Use Cases

4.1 For Traders

- Quick market snapshot before making trades.

- Compare assets (modify `ASSET` parameter).

4.2 For Analysts

- Automate repetitive reports (daily/weekly summaries).

- Combine with other datasets (e.g., trading volume).

4.3 For Educators

-Teaching financial data analysis in Python.

- Demo of API integrations(CoinGecko + OpenAI).

5. Future Improvements

Feature	Status	Benefit
Multi-Asset Comparison	Planned	Compare BTC vs. ETH trends.
Sentiment Analysis	Research Phase	Add Reddit/Twitter sentiment scores.
Automated Email Reports	Backlog	Schedule daily digests.

6. Conclusion

This tool **bridges raw data and actionable insights** by:

✅ **Automating** data collection & visualization.

✅ **Adding AI interpretation** for faster decision-making.

✅ **Remaining customizable** for different assets/timeframes.