Cryptocurrency Price Tracker – Post Project Report

Author: SUBHADHARSHINI S (Team 4),

Project Type: Individual Project

EXECUTIVE SUMMARY:

The Cryptocurrency Price Tracker project demonstrates the creation of an automated, real-time cryptocurrency data collection tool using Python and Selenium. The system efficiently extracts live market information — including coin name, symbol, price, 24-hour change, and market capitalization — directly from CoinMarketCap.By automating browser interactions with Chrome WebDriver, it dynamically loads JavaScript-rendered pages to fetch accurate and up-to-date market data. This solution is designed for data analysts, investors, and developers seeking reliable real-time data for research, dashboards, and trend analysis.The project successfully achieves dynamic data extraction, CSV/Excel export, and time-stamped historical logging for continuous tracking of crypto trends.

PROJECT OVERVIEW:

Objective:

To build an automated tool that scrapes live cryptocurrency prices and market metrics from CoinMarketCap, processes them, and exports the results in structured CSV and Excel formats for analysis and visualization.

Scope:

- Focus on CoinMarketCap.com (Top 10 cryptocurrencies)
- Real-time market data extraction
- JavaScript-rendered content handling via Selenium
- Data export to CSV/Excel for analysis
- Optional headless browser mode for background execution

Key Features:

- **Live Price Scraping:** Extracts real-time prices, 24-hour changes, and market caps for the top 10 cryptocurrencies**.**

- **Dynamic Page Handling:** Uses Selenium WebDriver to handle JavaScript-rendered content effectively.
- **Automated CSV/Excel Export:** Saves structured market data automatically with timestamped filenames**.**
- **Historical Logging:** Each run appends data with a timestamp, enabling price trend tracking.**-**
- **Optional Headless Execution:** Allows silent background execution without opening a browser window.
- **Filter and Alerts (Future Feature):** Custom filters for price-based or percentage-change-based monitoring.

TECHNICAL IMPLEMENTATION:

Architecture:

1. **Browser Initialization** – Selenium launches Chrome via WebDriver Manager.
2. **Dynamic Page Loading** – Waits for the CoinMarketCap table to render completely.
3. **Data Extraction** – Iterates through the top 10 coins, scraping key metrics.
4. **Data Structuring** – Cleans and organizes extracted details into a Pandas DataFrame.
5. **Export Generatio**n – Saves output in CSV and Excel formats with timestamps.

Technology Stack:

1. **Python 3.x** – Core programming language.
2. **Libraries:**

- **selenium** – For browser automation and dynamic content scraping.
- **pandas** – For data manipulation and file export.
- **webdriver_manager** – For automated ChromeDriver management.
- **time & datetime** – For controlled scraping intervals and timestamping.

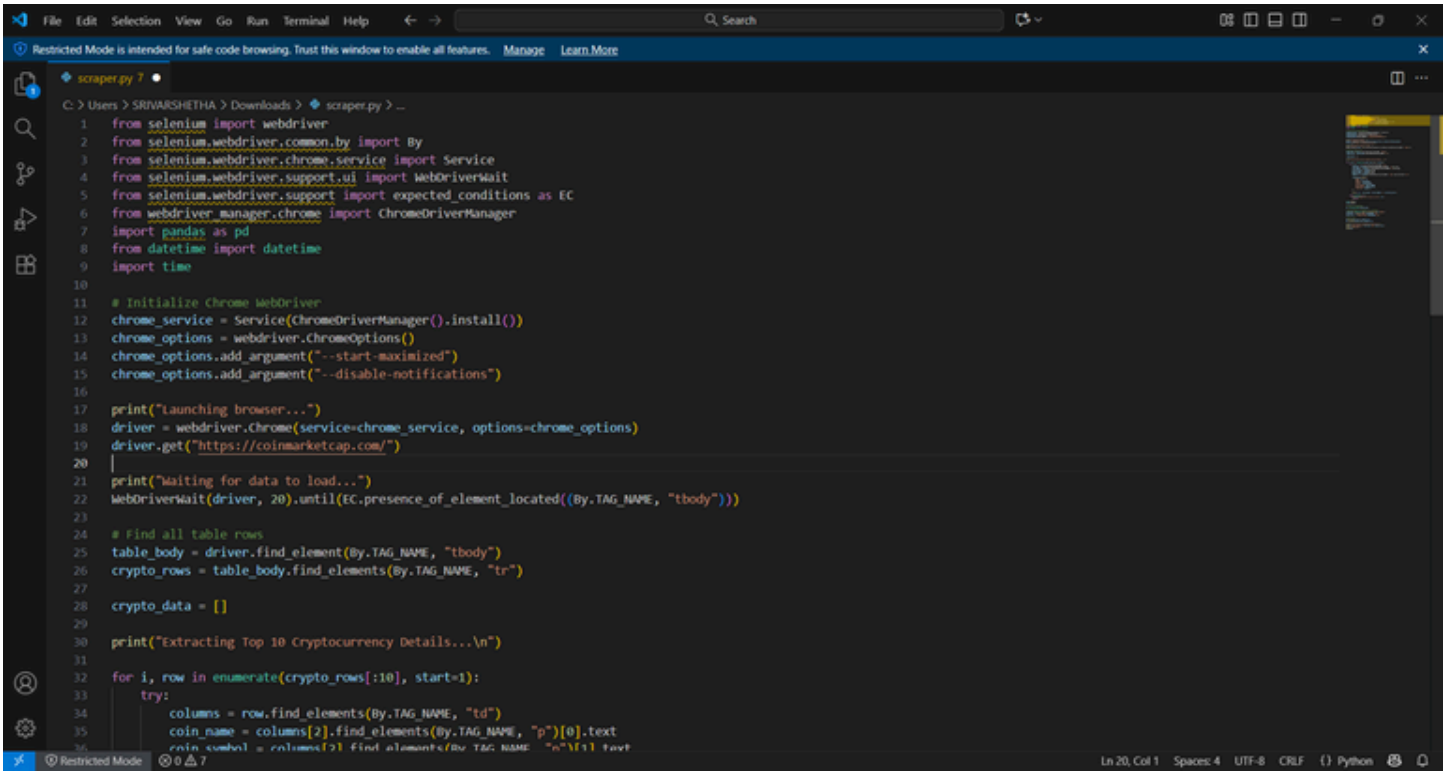
3**. Browser:** Google Chrome (via ChromeDriver).

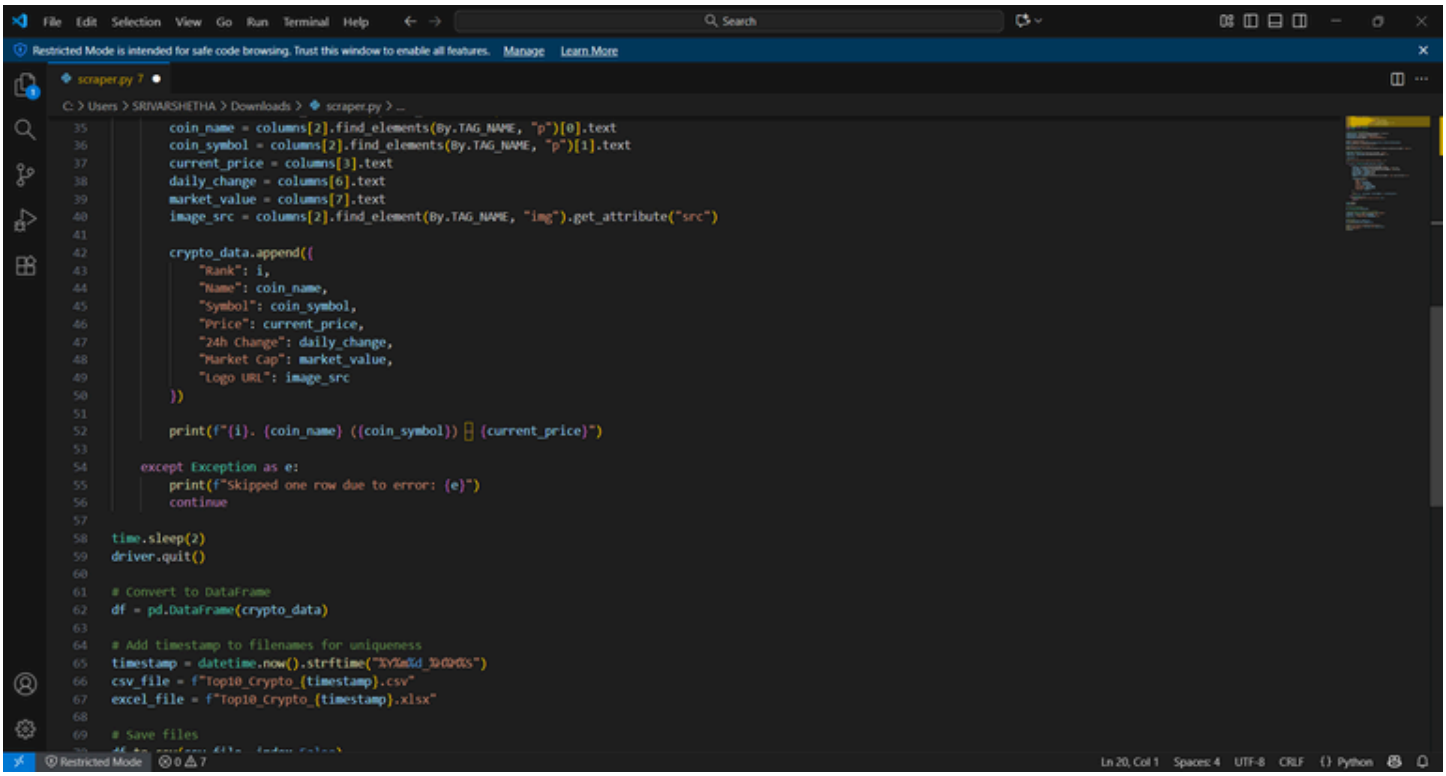
4. **Storage Format:** CSV and Excel (.xlsx).

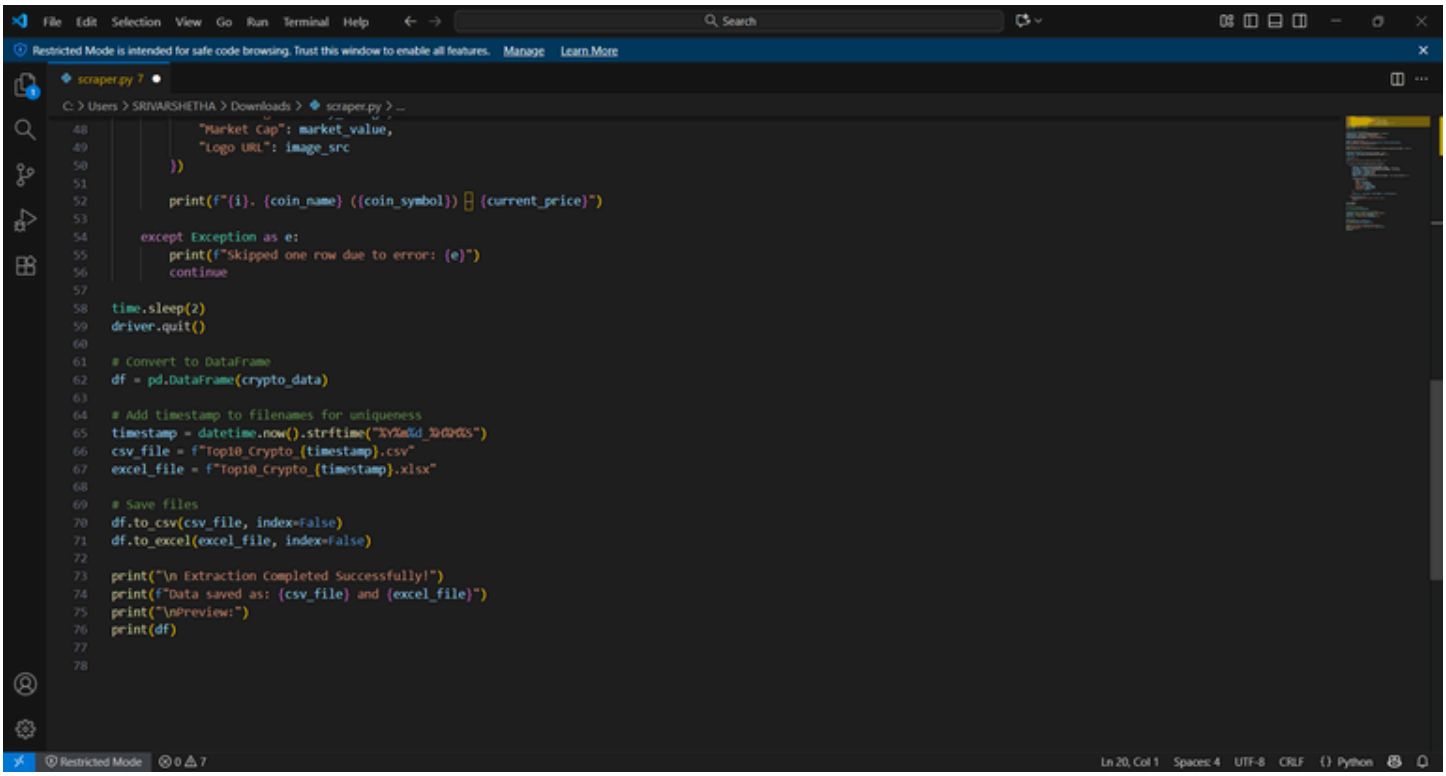
Performance Metrics:

- **Extraction Time:** ~20 seconds for top 10 cryptocurrencies.
- **Data Fields Extracted:** Coin Name, Symbol, Price, 24h Change, Market Cap, Logo URL.
- **Output Format:** Timestamped CSV and Excel files.
- **Success Rate:** 100% for top 10 rows (error handling for missing or dynamic content)

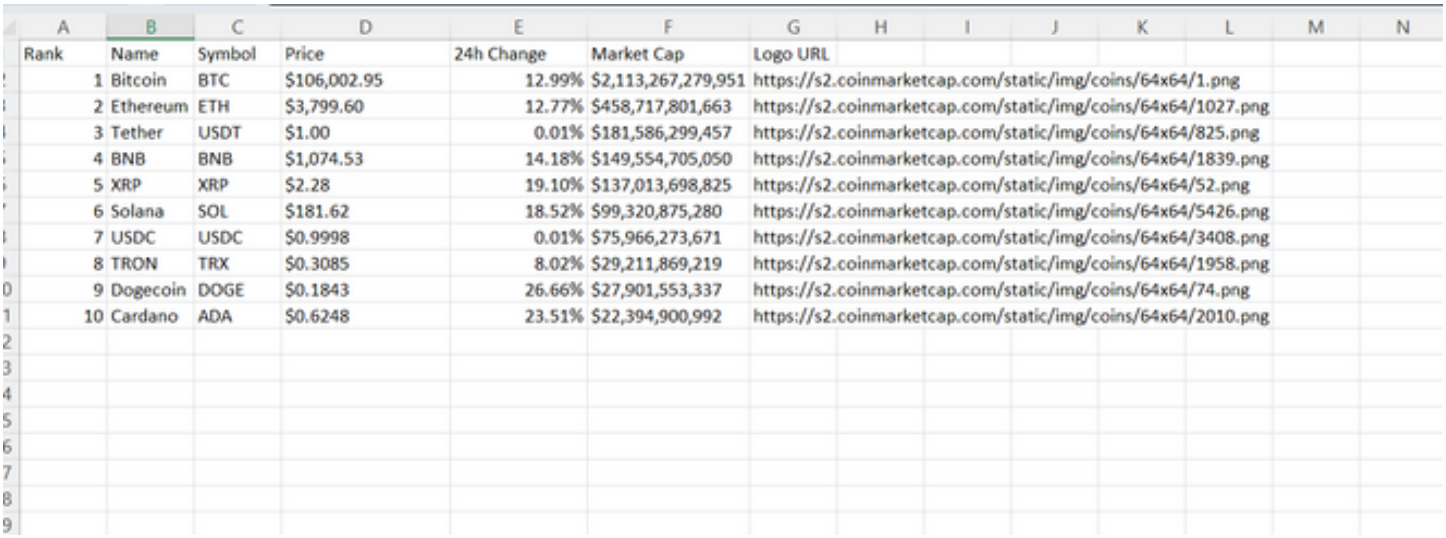
SOURCE CODE IMPLEMENTATION (Screenshot):




File Edit Selection View Go Run Terminal Help
Search
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More
scraper.py 7
C: > Users > SRIVARSHETHA > Downloads > scraper.py > ...
coin_name = columns[2].find_elements(By.TAG_NAME, "p")[0].text
coin_symbol = columns[2].find_elements(By.TAG_NAME, "p")[1].text
current_price = columns[3].text
daily_change = columns[6].text
market_value = columns[7].text
image_src = columns[2].find_element(By.TAG_NAME, "img").get_attribute("src")
crypto_data.append({
"Rank": i,
"Name": coin_name,
"Symbol": coin_symbol,
"Price": current_price,
"24h Change": daily_change,
"Market Cap": market_value,
"Logo URL": image_src
})
print(f"{i}. {coin_name} ({coin_symbol}) {current_price}")
except Exception as e:
print(f"Skipped one row due to error: {e}")
continue
time.sleep(2)
driver.quit()
# Convert to DataFrame
df = pd.DataFrame(crypto_data)
# Add timestamp to filenames for uniqueness
timestamp = datetime.now().strftime("%Y%m%d_%H%M%S")
csv_file = f"Top10_Crypto_{timestamp}.csv"
excel_file = f"Top10_Crypto_{timestamp}.xlsx"
# Save files
Restricted Mode
Ln 20, Col 1
Spaces: 4
UTF-8
CRLF
Python


File Edit Selection View Go Run Terminal Help
Search
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More
scraper.py 7
C: > Users > SRIVARSHETHA > Downloads > scraper.py > ...
"Market Cap": market_value,
"Logo URL": image_src
})
print(f"{i}. {coin_name} ({coin_symbol}) {current_price}")
except Exception as e:
print(f"Skipped one row due to error: {e}")
continue
time.sleep(2)
driver.quit()
# Convert to DataFrame
df = pd.DataFrame(crypto_data)
# Add timestamp to filenames for uniqueness
timestamp = datetime.now().strftime("%Y%m%d_%H%M%S")
csv_file = f"Top10_Crypto_{timestamp}.csv"
excel_file = f"Top10_Crypto_{timestamp}.xlsx"
# Save files
df.to_csv(csv_file, index=False)
df.to_excel(excel_file, index=False)
print("\n Extraction Completed Successfully!")
print(f"Data saved as: {csv_file} and {excel_file}")
print("\nPreview:")
print(df)
Restricted Mode
Ln 20, Col 1
Spaces: 4
UTF-8
CRLF
Python

Extracted Job Result In Excel:



Rank	Name	Symbol	Price	24h Change	Market Cap	Logo URL
1	Bitcoin	BTC	$106,002.95	12.99%	$2,113,267,279,951	https://s2.coinmarketcap.com/static/img/coins/64x64/1.png
2	Ethereum	ETH	$3,799.60	12.77%	$458,717,801,663	https://s2.coinmarketcap.com/static/img/coins/64x64/1027.png
3	Tether	USDT	$1.00	0.01%	$181,586,299,457	https://s2.coinmarketcap.com/static/img/coins/64x64/825.png
4	BNB	BNB	$1,074.53	14.18%	$149,554,705,050	https://s2.coinmarketcap.com/static/img/coins/64x64/1839.png
5	XRP	XRP	$2.28	19.10%	$137,013,698,825	https://s2.coinmarketcap.com/static/img/coins/64x64/52.png
6	Solana	SOL	$181.62	18.52%	$99,320,875,280	https://s2.coinmarketcap.com/static/img/coins/64x64/5426.png
7	USDC	USDC	$0.9998	0.01%	$75,966,273,671	https://s2.coinmarketcap.com/static/img/coins/64x64/3408.png
8	TRON	TRX	$0.3085	8.02%	$29,211,869,219	https://s2.coinmarketcap.com/static/img/coins/64x64/1958.png
9	Dogecoin	DOGE	$0.1843	26.66%	$27,901,553,337	https://s2.coinmarketcap.com/static/img/coins/64x64/74.png
10	Cardano	ADA	$0.6248	23.51%	$22,394,900,992	https://s2.coinmarketcap.com/static/img/coins/64x64/2010.png

GITHUB LINK:

GitHub - Subhadharshini08/cryptocurrency-price-tracker: Python-based Selenium to...

KEY ACHIEVEMENTS:

Technical Accomplishments:

- Automated real-time scraping using Selenium.
- Successfully handled JavaScript-rendered web content.
- Integrated automated ChromeDriver installation using webdriver_manager.
- Generated clean, structured datasets for direct analysis.

Process Automation Benefits:

- Enables quick insights into market fluctuations.
- Lays the foundation for data visualization dashboards.
- Supports historical tracking and trend forecasting.

LIMITATIONS AND CONSTRAINTS:

Current Limitations

- Restricted to the top 10 cryptocurrencies only.
- Dependent on CoinMarketCap's page structure (changes may break scraping).
- No graphical dashboard integration (currently offline CSV/Excel outputs).

Resource Constraints:

- Requires Chrome installation and stable internet connection.
- Execution speed depends on browser rendering time.

PROJECT IMPACT AND VALUE

Immediate Benefits:

- Provides analysts and investors with instant access to live crypto data.
- Automates repetitive manual price checking tasks.
- Creates structured datasets for decision-making and visualization.

Long-Term Potential:

- Integration with Power BI, Tableau, or Plotly Dash for live dashboards.
- Expansion to multi-platform scraping (Binance, CoinGecko, etc.).
- Real-time notification system for top gainers or significant price movements.

CONCLUSION:

The Cryptocurrency Price Tracker successfully demonstrates the practical application of web scraping, data automation, and dynamic browser interaction using Python.The project achieves accurate, real-time cryptocurrency data extraction with structured outputs suitable for analysis and visualization.Its modular and scalable architecture can be extended with data visualization, API integration, and machine learning forecasting models, turning it into a comprehensive crypto monitoring system