

# CRYPTOCURRENCY PRICE TRACKER

---

KAYATHRI V (TEAM 6)

---

## **PROJECT DESCRIPTION**

Crypto Price Tracker is a Python project that fetches real-time cryptocurrency data using the CoinGecko API. It displays the current price, 24-hour change, and market capitalization of popular cryptocurrencies like Bitcoin, Ethereum, and Dogecoin, and saves the data to an Excel file for easy reference.

## **Features**

1. Fetch Current Prices – Get the latest price of multiple cryptocurrencies.
2. 24-Hour Price Change – View the percentage change in price over the last 24 hours.
3. Market Capitalization – Check the current market cap of each cryptocurrency.
4. Excel Export – Save fetched data to an Excel file for analysis.
5. Multiple Coins Support – Track popular coins like Bitcoin, Ethereum, Dogecoin, and more.
6. Beginner-Friendly – Simple Python script, easy to run and understand.
7. Expandable – Add more coins or upgrade to real-time tracking if needed.

## **Technology Used**

Python – Main programming language for scripting.

Requests – To fetch cryptocurrency data from CoinGecko API.

Pandas – To organize data and export it to Excel.

Openpyxl – To write data into Excel files.

CoinGecko API – Provides real-time cryptocurrency data (price, 24h change, market cap).

## **Outcomes**

Terminal displays current prices, 24-hour change, and market capitalization of selected cryptocurrencies.

Excel file generated with all fetched data and timestamp.

Can track multiple coins at once.

Provides a snapshot of the cryptocurrency market for analysis.

## **Advantages**

Easy to use – Simple Python script, beginner-friendly.

Time-saving – Quickly get cryptocurrency data without visiting multiple sites.

Expandable – Add more coins or convert into real-time tracker with minimal changes.

Data Analysis Ready – Excel export allows further analysis and record keeping.

Market Awareness – Helps investors or enthusiasts monitor trends effectively.

## **Workflow**

Select cryptocurrencies to track (e.g., Bitcoin, Ethereum, Dogecoin).

Fetch current data (price, 24h change, market cap) from CoinGecko API.

Process data using Python & Pandas.

Display data in idle.

Save data to Excel file for reference.

Optional: Add more coins, real-time updates, graphs, or GUI.

## Source code: (utils.py)

```
utils.py - C:/Users/Hi/Desktop/CryptoPriceTracker/utils.py (3.11.9)
File Edit Format Run Options Window Help

import requests

def get_prices(coins, vs_currency="usd"):
    """
    Fetch current prices of coins from CoinGecko API
    """
    url = f"https://api.coingecko.com/api/v3/simple/price?ids={','.join(coins)}&vs_currencies={vs_currency}"
    try:
        response = requests.get(url)
        data = response.json()
        return data
    except Exception as e:
        print("Error fetching prices:", e)
        return {}

|
```

## Main.py

```
main.py - C:/Users/Hi/Desktop/CryptoPriceTracker/main.py (3.11.9)
File Edit Format Run Options Window Help

url = f"https://api.coingecko.com/api/v3/coins/markets?vs_currency={vs_currency}&ids={','.join(coins)}"
try:
    response = requests.get(url)
    data = response.json()
    return data
except Exception as e:
    print("Error fetching data:", e)
    return []

# Fetch coin data
coin_data = get_coin_data(coins, vs_currency)

if coin_data:
    print("----- Current Crypto Data -----")
    data_list = []
    for coin in coin_data:
        name = coin["name"]
        price = coin["current_price"]
        change_24h = coin["price_change_percentage_24h"]
        market_cap = coin["market_cap"]

        print(f"{name}: Price=${price}, 24h Change={change_24h:.2f}%, Market Cap=${market_cap}")

        data_list.append({
            "Coin": name,
            "Price (USD)": price,
            "24h Change (%)": change_24h,
            "Market Cap (USD)": market_cap,
            "Time": datetime.now().strftime("%Y-%m-%d %H:%M:%S")
        })

    print("-----\n")

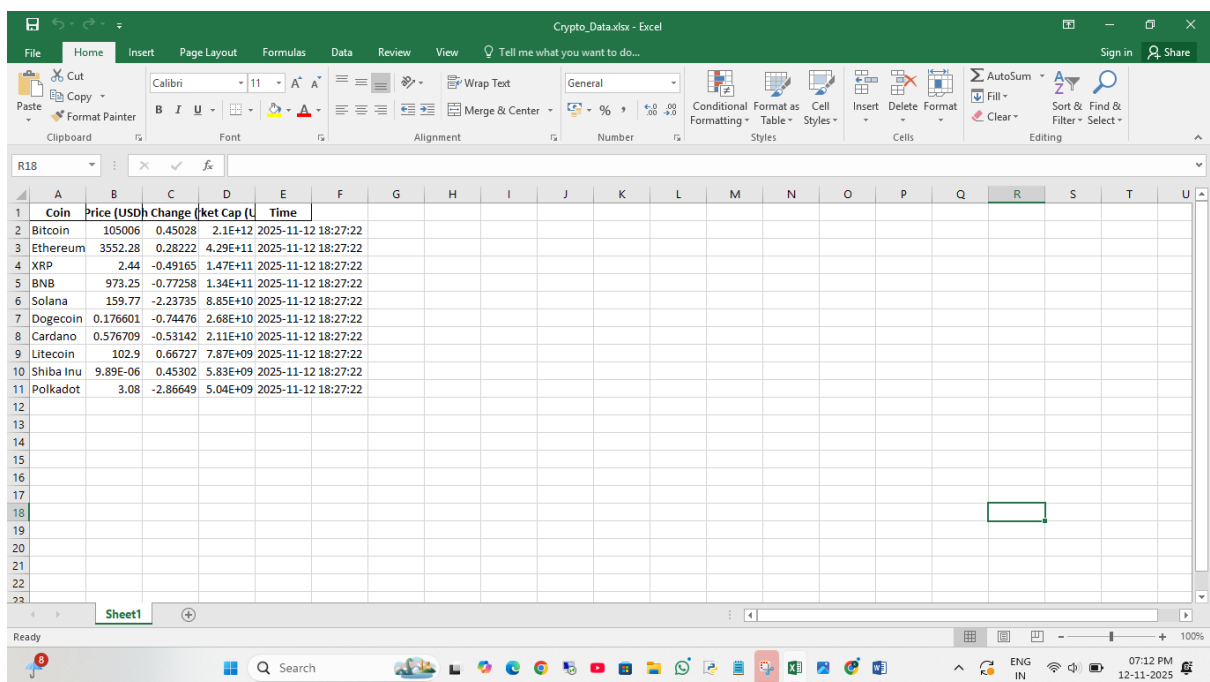
    # Save to Excel
    df = pd.DataFrame(data_list)
    df.to_excel("Crypto_Data.xlsx", index=False)
    print("Data saved to Crypto_Data.xlsx successfully!")

else:
    print("Failed to fetch coin data.")

|
```

```
IDLE Shell 3.11.9
File Edit Shell Debug Options Window Help
Python 3.11.9 (tags/v3.11.9:de54cf5, Apr 2 2024, 10:12:12) [MSC v.1938 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Hi/Desktop/CryptoPriceTracker/main.py
----- Current Crypto Data -----
Bitcoin: Price=$105006, 24h Change=0.45%, Market Cap=$2095752091294
Ethereum: Price=$3552.28, 24h Change=0.28%, Market Cap=$428745314128
XRP: Price=$2.44, 24h Change=-0.49%, Market Cap=$146905861927
BNB: Price=$973.25, 24h Change=-0.77%, Market Cap=$134025690023
Solana: Price=$159.77, 24h Change=-2.24%, Market Cap=$88468425385
Dogecoin: Price=$0.176601, 24h Change=-0.74%, Market Cap=$26786326940
Cardano: Price=$0.576709, 24h Change=-0.53%, Market Cap=$21102548378
Litecoin: Price=$102.9, 24h Change=0.67%, Market Cap=$7865508094
Shiba Inu: Price=$9.89e-06, 24h Change=0.45%, Market Cap=$5828559134
Polkadot: Price=$3.08, 24h Change=-2.87%, Market Cap=$5038529953
-----
Data saved to Crypto_Data.xlsx successfully!
>>>
```

## CSV format screenshot



Coin	Price (USD)	Change	Market Cap (USD)	Time
Bitcoin	105006	0.45028	2.1E+12	2025-11-12 18:27:22
Ethereum	3552.28	0.28222	4.29E+11	2025-11-12 18:27:22
XRP	2.44	-0.49165	1.47E+11	2025-11-12 18:27:22
BNB	973.25	-0.77258	1.34E+11	2025-11-12 18:27:22
Solana	159.77	-2.23735	8.85E+10	2025-11-12 18:27:22
Dogecoin	0.176601	-0.74476	2.68E+10	2025-11-12 18:27:22
Cardano	0.576709	-0.53142	2.11E+10	2025-11-12 18:27:22
Litecoin	102.9	0.66727	7.87E+09	2025-11-12 18:27:22
Shiba Inu	9.89E-06	0.45302	5.83E+09	2025-11-12 18:27:22
Polkadot	3.08	-2.86649	5.04E+09	2025-11-12 18:27:22

## **Module Description**

### 1. main.py

Core script that fetches cryptocurrency data.

Displays current price, 24-hour change, and market capitalization.

Saves data to Excel file (Crypto\_Data.xlsx).

### 2. utils.py (Optional)

Contains helper functions for API requests.

Makes code modular and reusable.

### 3. requirements.txt

Lists all required Python libraries (requests, pandas, openpyxl) for easy setup.

## **Future Enhancements**

Real-time tracking – Update prices every 60 seconds or more frequently.

Graphical representation – Plot price changes using matplotlib.

GUI interface – Use Tkinter or PyQt for a professional look.

Price alerts – Notifications if price crosses a set threshold.

More coin support – Track additional cryptocurrencies as needed.

Historical analysis – Save data over time and analyze trends.

## **Source code:(main.py)**

```
import requests
import pandas as pd
from datetime import datetime

# Coins to track
coins = ["bitcoin", "ethereum", "dogecoin", "litecoin", "ripple",
         "cardano", "solana", "binancecoin", "polkadot", "shiba-inu"]
vs_currency = "usd"

def get_coin_data(coins, vs_currency="usd"):
    """
    Fetch current price, 24h change, market cap from CoinGecko API
    """
    url =
f"https://api.coingecko.com/api/v3/coins/markets?vs_currency={vs_currency}&ids={'.'.join(coins)}"
    try:
        response = requests.get(url)
        data = response.json()
        return data
    except Exception as e:
        print("Error fetching data:", e)
        return []
```

```

# Fetch coin data
coin_data = get_coin_data(coins, vs_currency)

if coin_data:
    print("----- Current Crypto Data -----")
    data_list = []
    for coin in coin_data:
        name = coin["name"]
        price = coin["current_price"]
        change_24h = coin["price_change_percentage_24h"]
        market_cap = coin["market_cap"]

        print(f"{name}: Price=${price}, 24h Change={change_24h:.2f}%,
Market Cap=${market_cap}")

    data_list.append({
        "Coin": name,
        "Price (USD)": price,
        "24h Change (%)": change_24h,
        "Market Cap (USD)": market_cap,
        "Time": datetime.now().strftime("%Y-%m-%d %H:%M:%S")
    })

print("-----\n")

# Save to Excel
df = pd.DataFrame(data_list)

```

```
df.to_excel("Crypto_Data.xlsx", index=False)
print("Data saved to Crypto_Data.xlsx successfully!")
```

else:

```
print("Failed to fetch coin data.")
```

## **Conclusion:**

The Crypto Price Tracker is a simple yet effective Python project that allows users to monitor cryptocurrency prices, 24-hour changes, and market capitalization in real-time. It provides a quick overview of the crypto market and saves the data for future analysis in an Excel file.

This project is beginner-friendly, modular, and easily expandable to include more coins, real-time updates, graphs, and a GUI interface, making it a practical tool for both learning and tracking cryptocurrencies.