THE EUROPEAN CENTRAL BANK DATA PORTAL

An Open Data Platform for Economic and Financial Insights

by Hannes Bähr

CONTENT

1. The ECB and its Data Portal
2. Data Sets and Data Formats
3. Catalogue System and API
4. Data quality in the 5-star Open Data plan

1. THE ECB AND IT'S DATA PORTAL

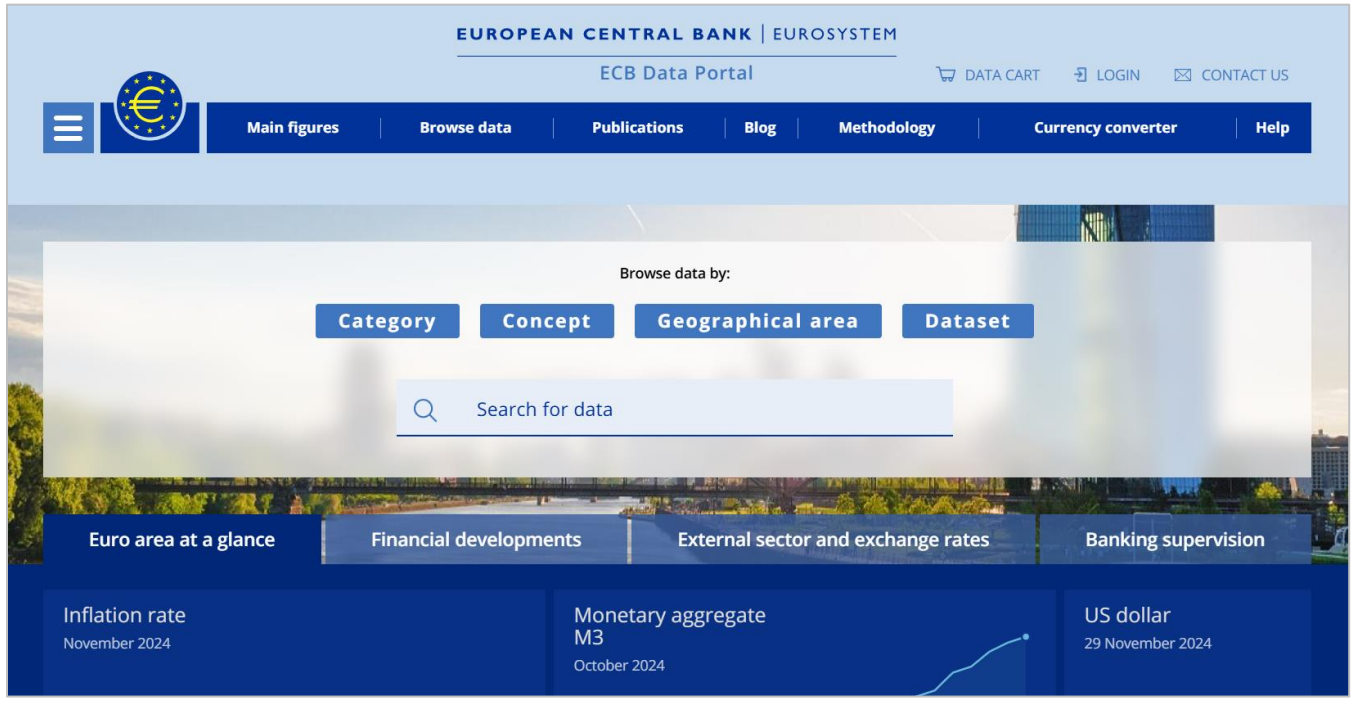
Who is the ECB?

- founded on June 1st 1998 it's the monetary authority of the **E**conomic and **M**onetary **U**nion
- managing the euro, ensuring price stability, implementing EU economic and monetary policies
- sets key interest rates, manages Eurozone's currency reserves, ensures financial markets and institutions are well supervised
- main goal: stable prices, thereby supporting economic growth and job creation

1. THE ECB AND ITS DATA PORTAL

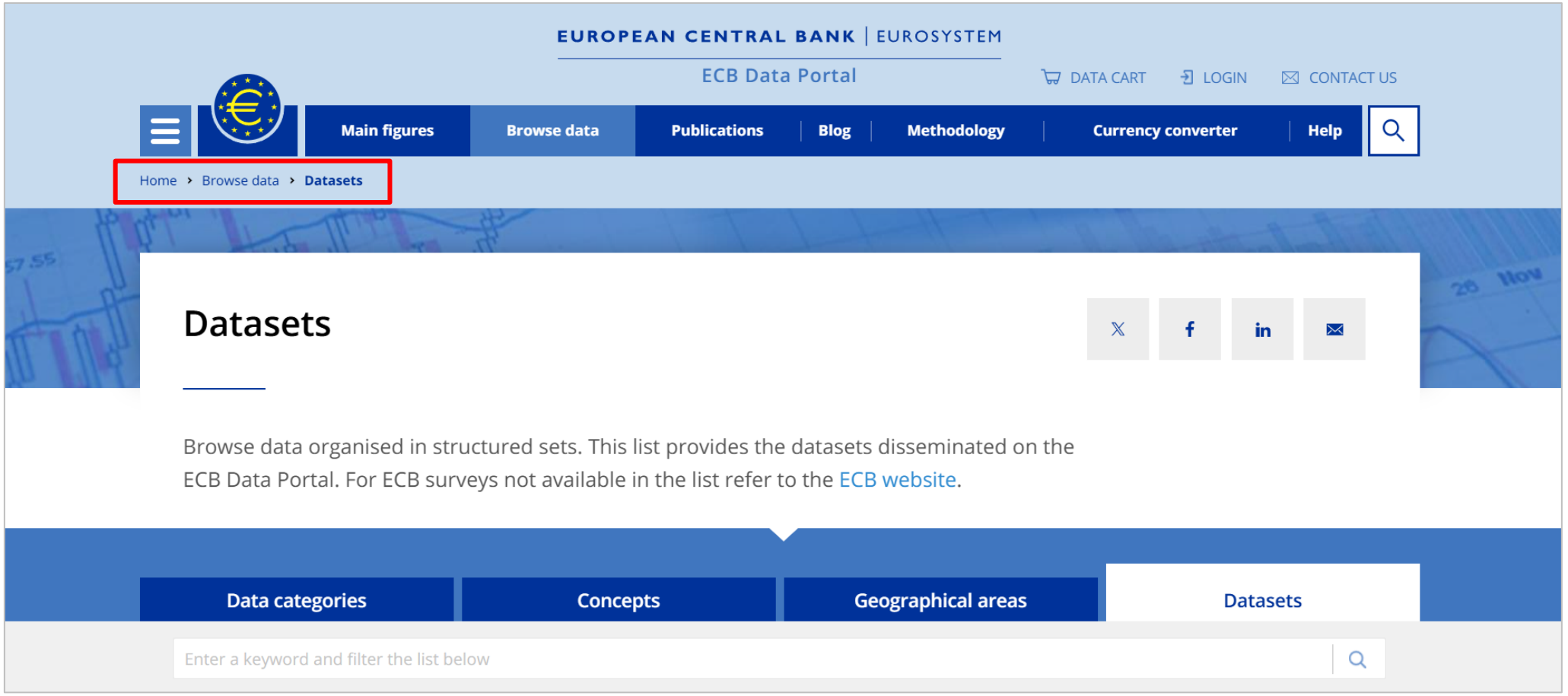
The ECB Data Portal

- ECB's online service for statistical data, related metadata and publications
- **user-friendly:** various tools for searching, visualizing, customizing, and downloading ECB's data
- tutorial: https://www.youtube.com/playlist?list=PLnVAEZuF9FZmxUaFiWcIp0efM9e7bz1F5



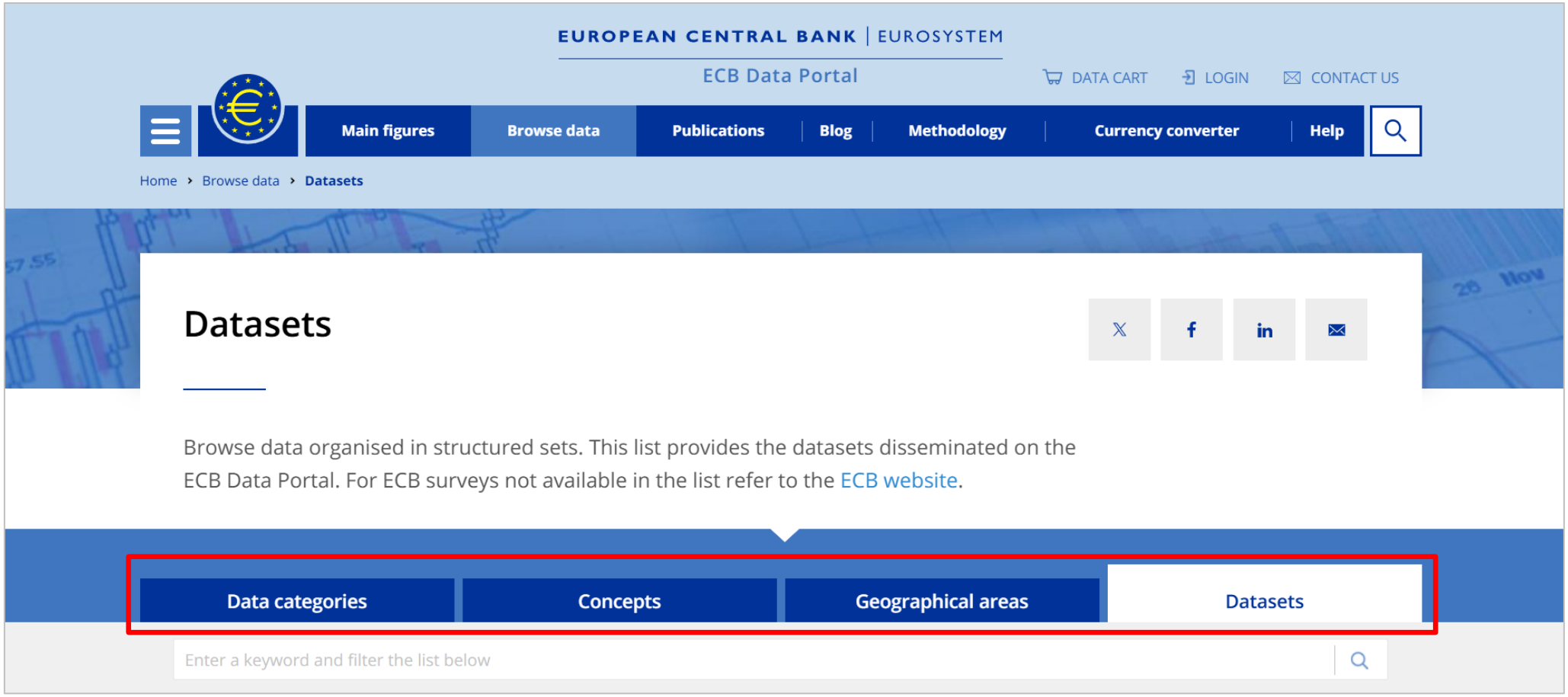
2. DATA SETS AND DATA FORMATS

Data Sets (https://data.ecb.europa.eu/data/datasets)



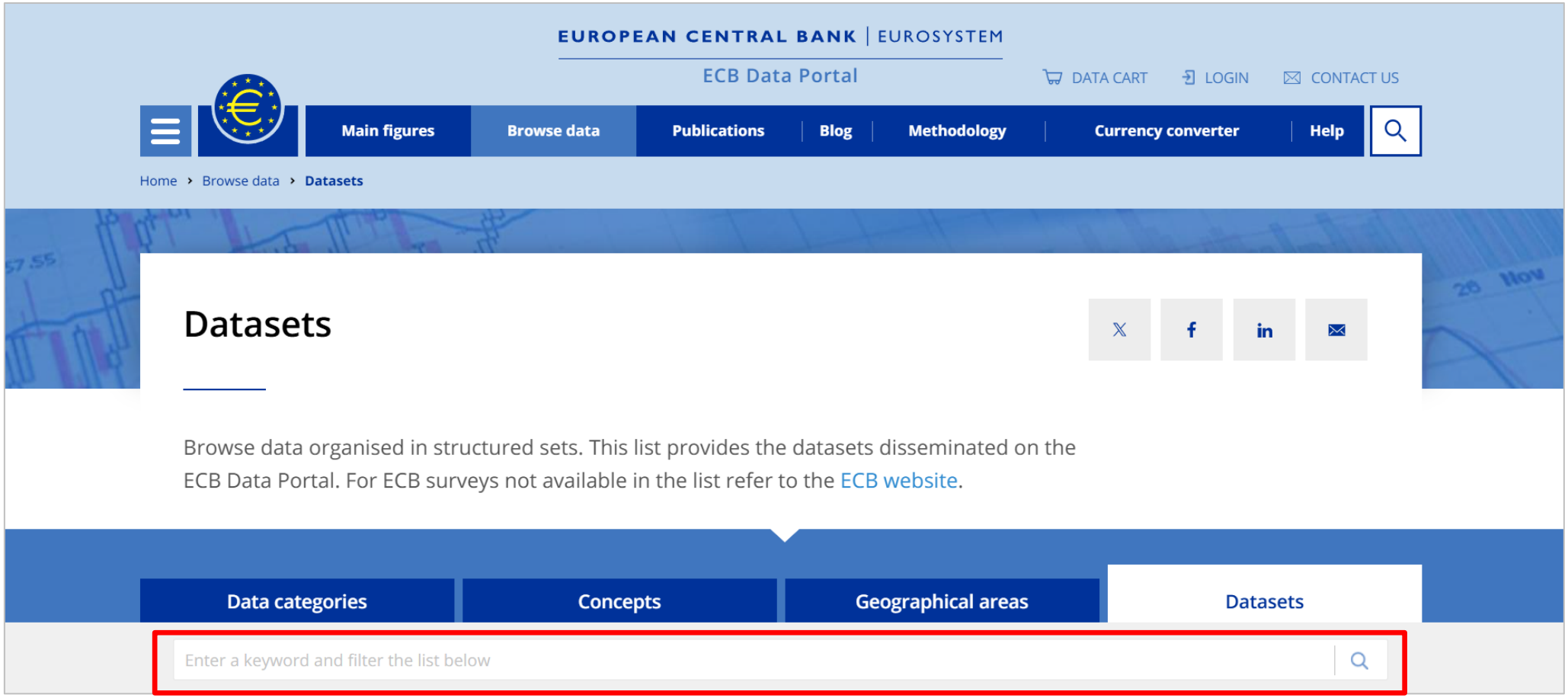
2. DATA SETS AND DATA FORMATS

Data Sets (https://data.ecb.europa.eu/data/datasets)



2. DATA SETS AND DATA FORMATS

Data Sets (https://data.ecb.europa.eu/data/datasets)



2. DATA SETS AND DATA FORMATS

Data Sets

- economic and financial data: inflation, interest rates, government finance, balance of payments, financial institutions' data…
- **searchable by categories, concepts, geograpic region, datsets**
- mainly aggregated and country specific data of Eurozone, European Union, European Economic Area
- but also data from countries all over the world in varying amounts

Data Formats

- statistical data formats: **SDMX-JSON,** SDMX-ML (default, XML-based)
- tabular data formats: CSV (incl. PivotTable-ready), XLS/XLSX
- graphical data formats: PNG, PDF, PPT (for visualizations and exports)

3. CATALOGUE SYSTEM AND API ACCES

Catalogue System

- **not** a standard open-source catalogue system like CKAN
- custom-designed system to organize financial and economic datasets
- metadata driven search: browse by categories, concepts, indicators, and geographical areas
- datasets follow SDMX standards for interoperability and contain detailed metadata (sources, update frequency, methods)

3. CATALOGUE SYSTEM AND API ACCES

API access

- RESTful API following SDMX standards
- features include customizable queries, metadata retrieval and multi-format outputs
- suitable for automating data retrieval and integration

Example API request on the following slides!

3. CATALOGUE SYSTEM AND API ACCES

Step 1: Import Package, define paramters and make request

```
import pandas as pd
import requests
import matplotlib.pyplot as plt

# Define parameters and API URL:
base_url = "https://data-api.ecb.europa.eu/service/data"
flow_ref = "EXR"                 # exchange rates dataflow
series_key = "D.USD.EUR.SP00.A"  # Daily USD/EUR exchange rates
start_period = "2023-01-01"
end_period = "2023-12-31"
url = f"{base_url}/{flow_ref}/{series_key}?startPeriod={start_period}&endPeriod={end_period}&format=jsondata"  # full URL with parameters

# Fetch the data:
response = requests.get(url)
if response.status_code == 200:
    data = response.json()
    print("Data retrieved successfully.")
else:
    print(f"Failed to fetch data. Status Code: {response.status_code}")
    print(response.text)
    raise SystemExit("Exiting script due to error.")

data
```

3. CATALOGUE SYSTEM AND API ACCES

```
…   Data retrieved successfully.

…   {'header': {'id': '0f2ad931-a51c-4902-b2e5-8b799b9116f5',
      'test': False,
      'prepared': '2024-12-09T06:28:21.288+01:00',
      'sender': {'id': 'ECB'}},
     'dataSets': [{'action': 'Replace',
       'validFrom': '2024-12-09T06:28:21.288+01:00',
       'series': {'0:0:0:0:0': {'attributes': [0,
          None,
          0,
          None,
          None,
          None,
          None,
          None,
          None,
          None,
          None,
          None,
          None,
          None,
          0,
          None,
          0,
          None,
          0,
          0,
          0,
          0,
    ...
         'values': [{'id': 'F', 'name': 'Free'}]},
        {'id': 'OBS_PRE_BREAK',
         'name': 'Pre-break observation value',
         'values': []},
        {'id': 'OBS_COM', 'name': 'Observation comment', 'values': []}]}}}
    Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
```

Output of the code from the previous slide!

3. CATALOGUE SYSTEM AND API ACCES

Step 2: Extract relevant data

```python
# Extract series data and observations:
series_data = data['dataSets'][0]['series']
structure = data['structure']
time_periods = structure['dimensions']['observation'][0]['values']  # extract time values

observations = []  # lists for observations
dates = []         # list for dates

for series_key, series_content in series_data.items():
    series_observations = series_content['observations']
    for obs_key, obs_values in series_observations.items():
        date_index = int(obs_key)             # use obs_key as index into time_periods
        date = time_periods[date_index]['id']  # map index to actual date
        value = obs_values[0]                 # first element in the observation array
        if value is not None:
            dates.append(date)
            observations.append(value)

df = pd.DataFrame({'Date': pd.to_datetime(dates), 'Exchange Rate': observations})
df.head(5)
```

[8] ✓ 0.0s

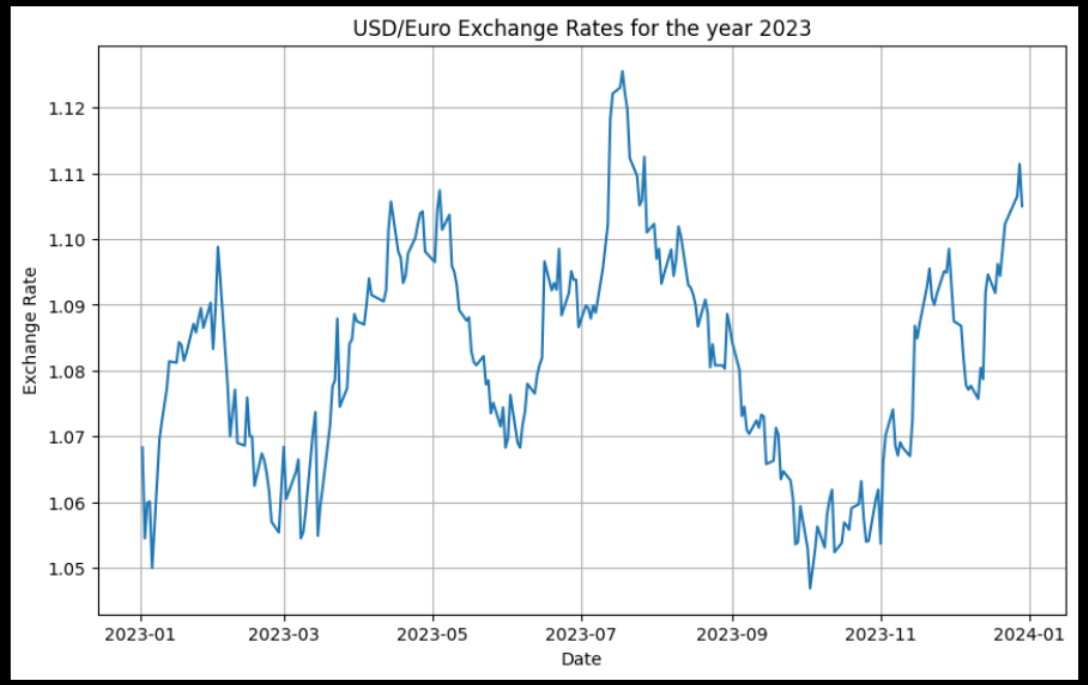
	Date	Exchange Rate
0	2023-01-02	1.0683
1	2023-01-03	1.0545
2	2023-01-04	1.0599
3	2023-01-05	1.0601
4	2023-01-06	1.0500

3. CATALOGUE SYSTEM AND API ACCES

Step 3: Plot a time series

```
plt.figure(figsize=(10, 6))
plt.plot(df['Date'], df['Exchange Rate'])
plt.title("USD/Euro Exchange Rates for the year 2023")
plt.xlabel("Date")
plt.ylabel("Exchange Rate")
plt.grid(True)
plt.show()
```

[13] ✓ 0.3s



4. DATA QUALITY IN THE 5-STAR OPEN DATA PLAN

Deployment schema for Open Data by Tim Berners-Lee:

*	Make your stuff available on the Web (whatever format) under an open license.
**	Make it available as structured data (e.g., Excel instead of image scan of a table).
***	Make it available in a non-proprietary open format (e.g., CSV instead of Excel).
****	Use URIs to denote things, so that people can point at your stuff.
*****	Link your data to other data to provide context.

4. DATA QUALITY IN THE 5-STAR OPEN DATA PLAN

Deployment schema for Open Data by Tim Berners-Lee:

*	Make your stuff available on the Web (whatever format) under an open license.
**	Make it available as structured data (e.g., Excel instead of image scan of a table).
***	Make it available in a non-proprietary open format (e.g., CSV instead of Excel).
****	Use URIs to denote things, so that people can point at your stuff.
*****	Link your data to other data to provide context.

THANKS FOR YOUR ATTENTION!

ANY QUESTIONS?