Bloomberg Macroeconomic Data Retrieval Script - README

# Purpose

This script fetches key macroeconomic indicators for USA, China, Japan, South Korea , Eurozone, and Australia from Bloomberg using python’s xbbg library and exports the data to an Excel file. It is designed to segment the process of data retrieval via the Bloomberg terminal so the main dashboard can run on any device as long as relevant packages are installed.

# Script Workflow

1. User Prompt:

* Displays a message box to inform the user that Bloomberg data will be fetched using the show\_messagebox() function.
* Prompts the user to choose a save location for the resulting Excel file using the prompt\_save\_file() function.
* Note: Don’t change these functions as it should not require changing even if data required changes

2. Ticker and Start Date Configuration:

* start\_date = ‘*FILL IN EARLIEST DATE YOU WANT TO LOOK AT YYYY-MM-DD (2018-01-01 as of now)*’
* Defines Bloomberg tickers for major macro indicators (Add country and ticker as needed):

USA:

* GDP YoY : `GDP CYOY Index`
* IP (Industrial Production) YoY: `IP YOY Index`
* Unemployment Rate: `USURTOT Index`
* Core CPI (Consumer Price Index) YoY: `CPURNSA Index`
* PPI (Producer Price Index) YoY: `FDIUFDYO Index`
* Manufacturing PMI (Purchasing Managers' Index) : `NAPMPMI Index`
* 10-Year Treasury Yield: `GT10 Govt`
* 2-Year Treasury Yield: `GT2 Govt`
* High Yield OAS (Option Adjusted Spread): `LF98OAS Index`
* Retail Sales MoM: 'RSTAMOM Index'
* Capacity Utilization: 'CPTICHNG Index'
* Core PCE (Personal Consumption Expenditure) YoY: 'PCE CMOM Index'
* Services PMI (Purchasing Managers' Index) : 'NAPMNMI Index'
* Small Business Sentiment': 'SBOITOTL Index'
* Home Sales: 'ETSLTOTL Index'
* Jobless Claims 4 Weeks Moving Average: 'INJCJC Index'

China:

* GDP YoY: 'CNGDPC$Y Index'
* Core CPI (Consumer Price Index) YoY: 'CNCPIYOY Index'
* Unemployment: 'CNUESRU Index'
* Manufacturing PMI (Purchasing Managers' Index): 'CPMINDX Index'
* Industrial Value Added YoY: 'CHVAIOY Index'
* Retail Sales YoY: 'CNRSCYOY Index'
* Service Production Index: 'CNSF Index'
* PPI (Producer Price Index) YoY: 'CHEFTYOY Index'
* Exports YoY: 'CNFREXPY Index'
* Money Supply M2 YoY: 'CNMS2YOY Index'
* Real Estate Climate Index YoY: 'CHRXCINY Index'
* SHIBOR (Shanghai Interbank Offered Rate) 1M : 'SHIF1M Index'

Japan:

* GDP YoY: 'JGDPAGDP Index'
* Unemployment: 'JNUNRT Index'
* Core CPI (Consumer Price Index) YoY: 'JNCPIYOY Index'
* PPI (Producer Price Index) YoY: 'JNWSDYOY Index'
* Retail Sales YoY: 'JNNETYOY Index'
* Exports\_YoY': 'JNTBEXPY Index'
* 10-Year Treasury Yield: 'GJGB10 Index'
* 2-Year Treasury Yield: 'GJGB2 Index'
* Money Supply M2 YoY: 'JMNSM2Y Index'
* Tankan Business Conditions (Large Enterprises) Manufacturing: 'JNTSMFG Index'
* IP (Industrial Production) YoY: 'JNIPYOY Index'
* Consumer Confidence: 'JCOMACF Index'
* Business Confidence All Industry: 'JSMEALLI Index'

Eurozone:

* GDP YoY: 'EUGNEMUY Index',
* Unemployment: 'UMRTEMU Index',
* Core CPI(Consumer Price Index) YoY: 'CPEXEMUY Index',
* Composite PMI (Purchasing Managers' Index): 'MPMIEZCA Index',
* Money Supply M3 YoY: 'ECMAM3YY Index',
* Capacity Utilization: 'EUUCEMU Index',
* Consumer Confidence: 'EUCCEMU Index',
* Yield Spread: 'EUCBEIOR Index',
* Credit Impulse: 'BCMPCIGD Index',
* IP (Industrial Production) YoY: 'EUIPEMUY Index',
* Retail\_Expectations: 'EUR4EMU Index',
* Economic\_Sentiment: 'EUESEMU Index'

South Korea

* GDP YoY: 'KOGDPYOY Index'
* Unemployment: 'KOEAUERS Index'
* Core CPI (Consumer Price Index) YoY: 'KOCPIYOY Index'
* PPI (Producer Price Index) YoY: 'KOPPIYOY Index'
* Retail Sales YoY: 'KORSTY Index'
* Exports YoY: 'KOEXTOTY Index'
* Manufacturing PMI (Purchasing Managers' Index): 'MPMIKRMA Index'
* 10-Year Treasury Yield: 'GTKRW10Y Govt'
* 2-Year Treasury Yield: 'GTKRW2Y Govt'
* IP (Industrial Production) YoY: 'KOIPIY Index'
* Business Sentiment: 'KOBSCBSI Index'
* Household Debt: 'KOHHD Index'

Australia:

* GDP YoY: 'AUNAGDPY Index’,
* Unemployment: 'AULFUNEM Index',
* Core\_CPI\_YoY: 'ACPMTRNY Index',
* PPI YoY: 'AUPPFYOY Index',
* Retail Sales YoY: 'AURSTYSA Index',
* Exports YoY: 'AUITEXGY Index',
* Composite PMI: 'MPMIAUCA Index',
* 10Y: 'GTAUD10Y Govt',
* 2Y: 'GTAUD2Y Govt',
* Mining Labor YoY: 'AULQMINY Index',
* Money Supply M3 YoY: 'AUM3Y Index',
* Housing Loan Interest Rate: 'AILRHLBS Index',
* Business Conditions: 'NABSCOND Index',
* Business Confidence: 'NABSCONF Index',

3. Data Retrieval using fetch\_macro\_data() function:

* Uses `xbbg.blp.bdh()` to pull all historical `PX LAST` data for each indicator starting from `2018-01-01`.
* All data have date in the rows and tickers for the columns. The data is then merged into their respective DataFrame based on the country.
* Note: Don’t change this function as it should work on any countries’ ticker dictionary and correct country label, as long as ticker exists.

4. Error Handling:

If data for any ticker fails to download, the error is caught and printed in the jupyter notebook terminal, but the script continues with other tickers.

5. Excel Output:

* If any data is successfully fetched, it is saved to the specified Excel file, with each country’s data in the respective country sheets.
* If no data is fetched, an error message box is shown.

# Output

* The resulting Excel file contains 4 sheets for each country, each with a datetime index with each macroeconomic indicator in its own column.
* File is saved at the user-selected path with default filename `macro\_data\_bb.xlsx`.

# Requirements

* Bloomberg Terminal with API access and permission to run Excel Add-In/API calls.
* Python packages:
  + pandas
  + xbbg
  + tkinter
* Jupyter Notebook/VSCode installed on the device

# How to Run

1. Ensure Bloomberg Terminal is running.
2. Open Jupyter Notebook App
3. Click ‘File’ -> ‘New’ -> ‘Terminal’
4. In the line, “PS C:\Users\PMA\Documents>”:
   1. Type to exit folder: cd ..
   2. Type to enter folder: cd “Desktop/Nic/Macroeconomic Dashboard”
   3. Install required Python packages by typing:
      1. pip install xbbg
      2. pip install pandas
   4. Run the script: python fetch\_data\_bb.py
5. Follow the GUI prompts to choose save location for excel file.

# Notes

* The script uses tkinter for GUI dialogs and uses Bloomberg data so it requires the Bloomberg Terminal desktop environment.
* The start date and tickers can be customized at the top of the script under CONFIGURATION and TICKERS (Left side is given name to appear in graph, Right side is the ticker index name in Bloomberg) in the code.
* The script supports 6 countries / regions: USA, China, Japan, South Korea, Eurozone & Australia. Each has its own set of Bloomberg tickers stored in ‘us\_tickers’, ‘cn\_tickers’, ‘jp\_tickers’, ‘sk\_tickers’, ‘eu\_tickers’ & ‘au\_tickers’ dictionaries.
* The script processes both countries' data in parallel and saves them to separate sheets in the same Excel workbook.
  + 'China': China macroeconomic indicators
  + 'USA': USA macroeconomic indicators
  + 'Japan’: Japan macroeconomic indicators
  + ‘South Korea’: South Korea macroeconomic indicators
  + ‘Eurozone’: Eurozone macroeconomic indicators
  + ‘Australia’: Australia macroeconomic indicators