

Table of contents

```
from wbgsapi import data as wbdata
import pandas as pd

# Define indicators
population_indicators = {
    'total_population': 'SP.POP.TOTL',
    'population_growth': 'SP.POP.GROW',
    'urban_population_percent': 'SP.URB.TOTL.IN.ZS',
    'rural_population_percent': 'SP.RUR.TOTL.ZS',
    'fertility_rate': 'SP.DYN.TFRT.IN',
    'birth_rate': 'SP.DYN.CBRT.IN',
    'death_rate': 'SP.DYN.CDRT.IN',
    'infant_mortality_rate': 'SP.DYN.IMRT.IN',
    'under_5_mortality_rate': 'SH.DYN.MORT',
    'adolescent_fertility_rate': 'SP.ADO.TFRT',
    'net_migration': 'SM.POP.NETM',
    'population_density': 'EN.POP.DNST',
    'life_expectancy': 'SP.DYN.LE00.IN'
}

# Countries
country_codes = ['BRA', 'MEX', 'ARG', 'COL']

# Download wide-format data for 2022
df = wbdata.DataFrame(
    population_indicators.values(),
    economy=country_codes,
    time=2022,
    skipBlanks=True,
    labels=True
).reset_index()

# Drop unwanted columns (if any)
df.drop(columns=['economy'], errors='ignore', inplace=True)

# Rename columns using your friendly keys
df.rename(columns=lambda x: {v: k for k, v in population_indicators.items()}.get(x, x).lower)

# Save to CSV
df.to_csv('wdi_population_.csv', index=False)
```

```

import sqlite3

# Connect to or create the database
conn = sqlite3.connect('wdi_data.db')

# Load CSV into DataFrame
df = pd.read_csv('wdi_population_.csv')

# Write the data to SQL
df.to_sql('wdi_data', conn, if_exists='replace', index=False)

```

4

```

import pandas as pd

df = pd.read_csv('wdi_population_.csv')
print(df.columns)

```

```

Index(['country', 'population_density', 'under_5_mortality_rate',
       'net_migration', 'adolescent_fertility_rate', 'birth_rate',
       'death_rate', 'infant_mortality_rate', 'life_expectancy',
       'fertility_rate', 'population_growth', 'total_population',
       'rural_population_percent', 'urban_population_percent'],
      dtype='object')

```