

python-wk-8

September 29, 2025

```
[11]: import pandas as pd
import urllib.request
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
import plotly.express as px

[ ]: url = "https://covid.ourworldindata.org/data/owid-covid-data.csv"
urllib.request.urlretrieve(url, "owid-covid-data.csv") # save file locally

df = pd.read_csv("owid-covid-data.csv")
print(df.head())

[ ]: df.columns

[ ]: df.head()

[ ]: df.isnull().sum()

[ ]: # Filter countries of interest
countries = ['Kenya', 'United States', 'India']
df_filtered = df[df['location'].isin(countries)].copy()
# Drop rows with missing dates or critical values (e.g., total_cases or
total_deaths)
df_filtered.dropna(subset=['date', 'total_cases', 'total_deaths'], inplace=True)
# Convert date column to date time
df_filtered['date'] = pd.to_datetime(df_filtered['date'])
# Sort by country and date before interpolation
df_filtered.sort_values(by=['location', 'date'], inplace=True)
# Handle missing numeric values by interpolating only numeric columns
numeric_cols = df_filtered.select_dtypes(include=['number']).columns
df_filtered[numeric_cols] = df_filtered[numeric_cols].
interpolate(method='linear')
# Confirm result
print(df_filtered.head())
```

```
[ ]: # Filter countries
countries = ['Kenya', 'United States', 'India']
df = df[df['location'].isin(countries)].copy()
# Clean data
df = df[['location', 'date', 'total_cases', 'new_cases', 'total_deaths']]
df.dropna(subset=['date'], inplace=True)
df['date'] = pd.to_datetime(df['date'])
df.sort_values(by=['location', 'date'], inplace=True)
df[['total_cases', 'new_cases', 'total_deaths']] = df[['total_cases',
'new_cases', 'total_deaths']].interpolate()
```

```
[ ]: plt.figure(figsize=(12, 6))
sns.lineplot(data=df, x='date', y='total_cases', hue='location')
plt.title('Total COVID-19 Cases Over Time')
plt.xlabel('Date')
plt.ylabel('Total Cases')
plt.tight_layout()
plt.show()
```

```
[ ]: plt.figure(figsize=(12, 6))
sns.lineplot(data=df, x='date', y='total_deaths', hue='location')
plt.title('Total COVID-19 Deaths Over Time')
plt.xlabel('Date')
plt.ylabel('Total Deaths')
plt.tight_layout()
plt.show()
```

```
[ ]: plt.figure(figsize=(12, 6))
sns.lineplot(data=df, x='date', y='new_cases', hue='location')
plt.title('Daily New COVID-19 Cases Comparison')
plt.xlabel('Date')
plt.ylabel('New Cases')
plt.tight_layout()
plt.show()
```

```
[ ]: df['death_rate'] = df['total_deaths'] / df['total_cases']
plt.figure(figsize=(12, 6))
sns.lineplot(data=df, x='date', y='death_rate', hue='location')
plt.title('COVID-19 Death Rate Over Time')
plt.xlabel('Date')
plt.ylabel('Death Rate')
plt.tight_layout()
plt.show()
```

```
[ ]: url = "https://covid.ourworldindata.org/data/owid-covid-data.csv"
df = pd.read_csv(url)
# Filter relevant countries
```

```

countries = ['Kenya', 'United States', 'India']
df_vax = df[df['location'].isin(countries)][[
    'location', 'date', 'total_vaccinations', 'people_vaccinated_per_hundred',
    'population'
]].copy()
# Clean up
df_vax.dropna(subset=['date'], inplace=True)
df_vax['date'] = pd.to_datetime(df_vax['date'])
df_vax.sort_values(by=['location', 'date'], inplace=True)
df_vax[['total_vaccinations', 'people_vaccinated_per_hundred']] = df_vax[[
    'total_vaccinations', 'people_vaccinated_per_hundred'
]].interpolate()
plt.figure(figsize=(12, 6))
sns.lineplot(data=df_vax, x='date', y='total_vaccinations', hue='location')
plt.title('Cumulative COVID-19 Vaccinations Over Time')
plt.xlabel('Date')
plt.ylabel('Total Vaccinations')
plt.tight_layout()
plt.show()

```

```

[ ]: plt.figure(figsize=(12, 6))
sns.lineplot(data=df_vax, x='date', y='people_vaccinated_per_hundred',
hue='location')
plt.title('% of Population Vaccinated Over Time')
plt.xlabel('Date')
plt.ylabel('People Vaccinated per 100')
plt.tight_layout()
plt.show()

```

```

[ ]: # Get the most recent vaccination percentage
latest_vax = df_vax.sort_values('date').groupby('location').tail(1)
for _, row in latest_vax.iterrows():
    vaccinated = row['people_vaccinated_per_hundred']
    unvaccinated = 100 - vaccinated
plt.figure(figsize=(5, 5))
plt.pie([vaccinated, unvaccinated],
labels=['Vaccinated', 'Unvaccinated'],
autopct='%1.1f%%',
colors=['green', 'lightgray'])
plt.title(f"{row['location']} - Vaccination Status")
plt.tight_layout()
plt.show()

```

```

[ ]: # Load the dataset
df = pd.read_csv("https://covid.ourworldindata.org/data/owid-covid-data.csv")
# Convert 'date' to datetime
df['date'] = pd.to_datetime(df['date'])

```

```

# Get latest data per country
latest_df = df.sort_values('date').groupby('iso_code', as_index=False).last()
# Filter out aggregate rows (e.g., continents, World)
latest_df = latest_df[latest_df['iso_code'].str.len() == 3]
# Create choropleth
fig = px.choropleth(
    latest_df,
    locations='iso_code',
    color='total_cases',
    hover_name='location',
    color_continuous_scale='Reds',
    title='Total COVID-19 Cases by Country (Latest Available Data)'
)
fig.show()

```