Project Report

Prepared by:	Alua Omar and Sagatqyzy Firuza
Prepared for:	Data Collection and Preparation

Review

Code Base	🌐 Google Colab
Project overview	Analyze the relationship between COVID-19 vaccination rates and health outcomes using data from multiple sources.
Data sources	disease.sh API (231 countries, case statistics) Wikipedia (220 countries, vaccination data)
Tools	Python, pandas, BeautifulSoup, scipy, matplotlib

Metodology

API Request → JSON Parsing → DataFrame

↓

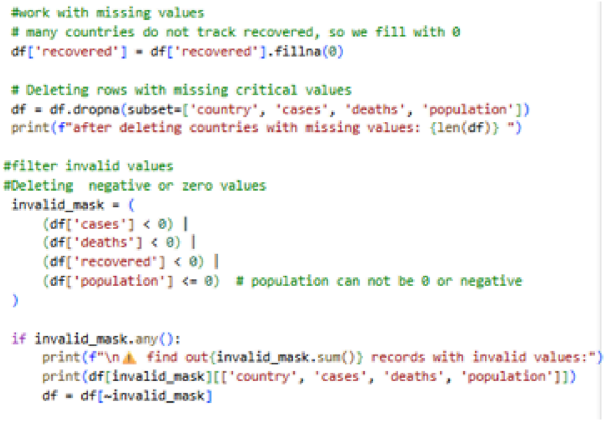
Wikipedia → HTML Scraping → Table Extraction → DataFrame

↓

Country Mapping → Data Merge → Analysis Dataset (195 countries)

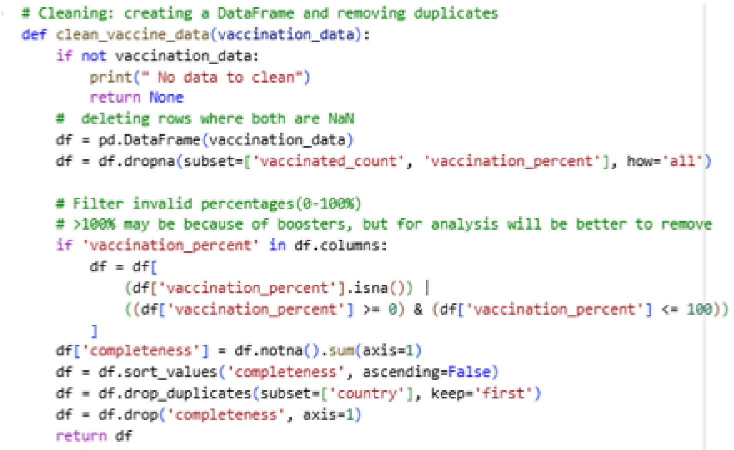
Data cleaning and merging

COVID-19 Dataset:



```
#work with missing values
# many countries do not track recovered, so we fill with 0
df['recovered'] = df['recovered'].fillna(0)

# Deleting rows with missing critical values
df = df.dropna(subset=['country', 'cases', 'deaths', 'population'])
print(f"after deleting countries with missing values: {len(df)} ")

#filter invalid values
#Deleting  negative or zero values
invalid_mask = (
    (df['cases'] < 0) |
    (df['deaths'] < 0) |
    (df['recovered'] < 0) |
    (df['population'] <= 0)  # population can not be 0 or negative
)

if invalid_mask.any():
    print(f"\n⚠️ find out{invalid_mask.sum()} records with invalid values:")
    print(df[invalid_mask][['country', 'cases', 'deaths', 'population']])
    df = df[~invalid_mask]
```

Vaccination Dataset:



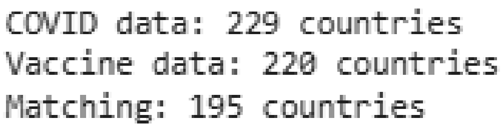
```
# Cleaning: creating a DataFrame and removing duplicates
def clean_vaccine_data(vaccination_data):
    if not vaccination_data:
        print(" No data to clean")
        return None
    #  deleting rows where both are NaN
    df = pd.DataFrame(vaccination_data)
    df = df.dropna(subset=['vaccinated_count', 'vaccination_percent'], how='all')

    # Filter invalid percentages(0-100%)
    # >100% may be because of boosters, but for analysis will be better to remove
    if 'vaccination_percent' in df.columns:
        df = df[
            (df['vaccination_percent'].isna()) |
            ((df['vaccination_percent'] >= 0) & (df['vaccination_percent'] <= 100))
        ]
    df['completeness'] = df.notna().sum(axis=1)
    df = df.sort_values('completeness', ascending=False)
    df = df.drop_duplicates(subset=['country'], keep='first')
    df = df.drop('completeness', axis=1)
    return df
```

✅ No NaN values in key columns

✅ No Inf values

✅ All percentages in valid ranges

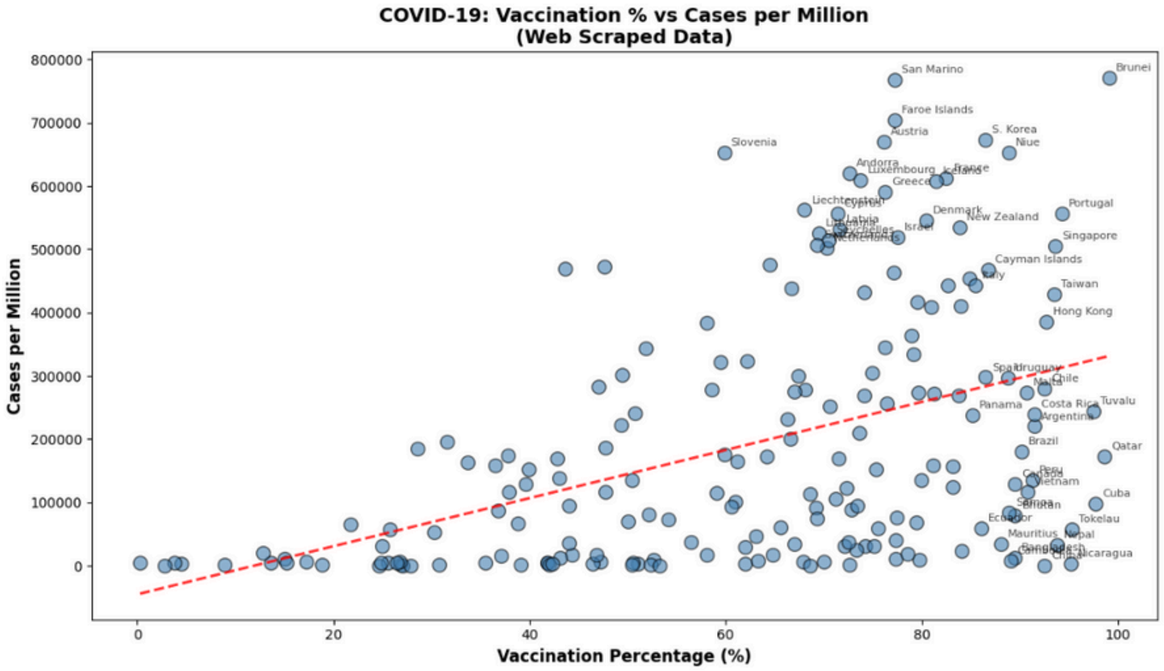
✅ Logical consistency verified



```
COVID data: 229 countries
Vaccine data: 220 countries
Matching: 195 countries
```

The final dataframe contains statistics on infections, deaths, and vaccinations.

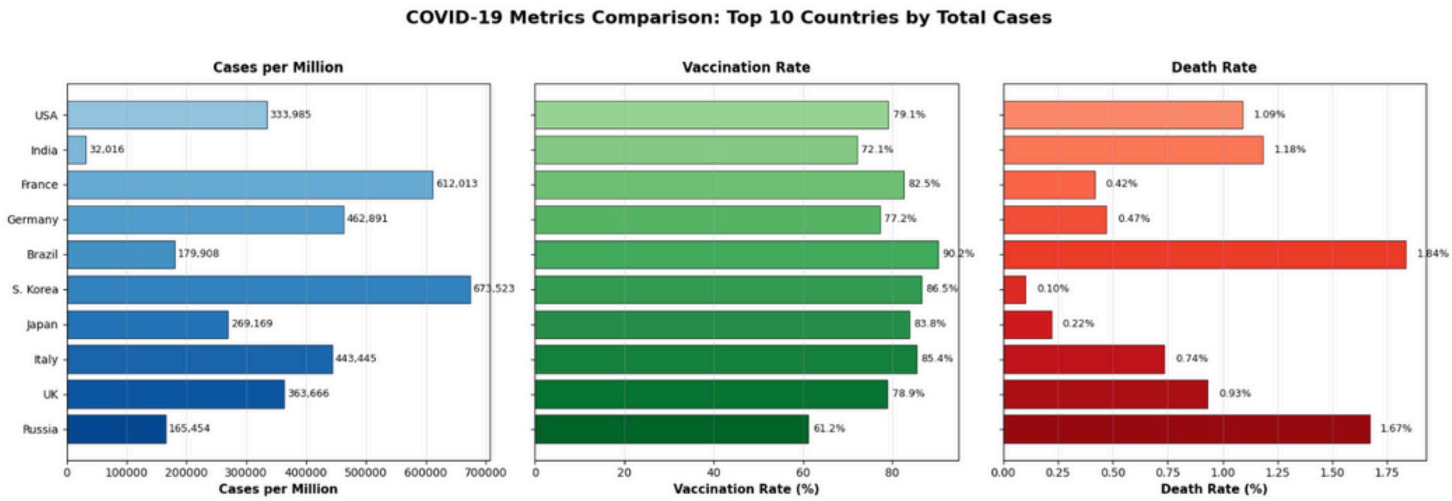
Data analysis and Vizualization



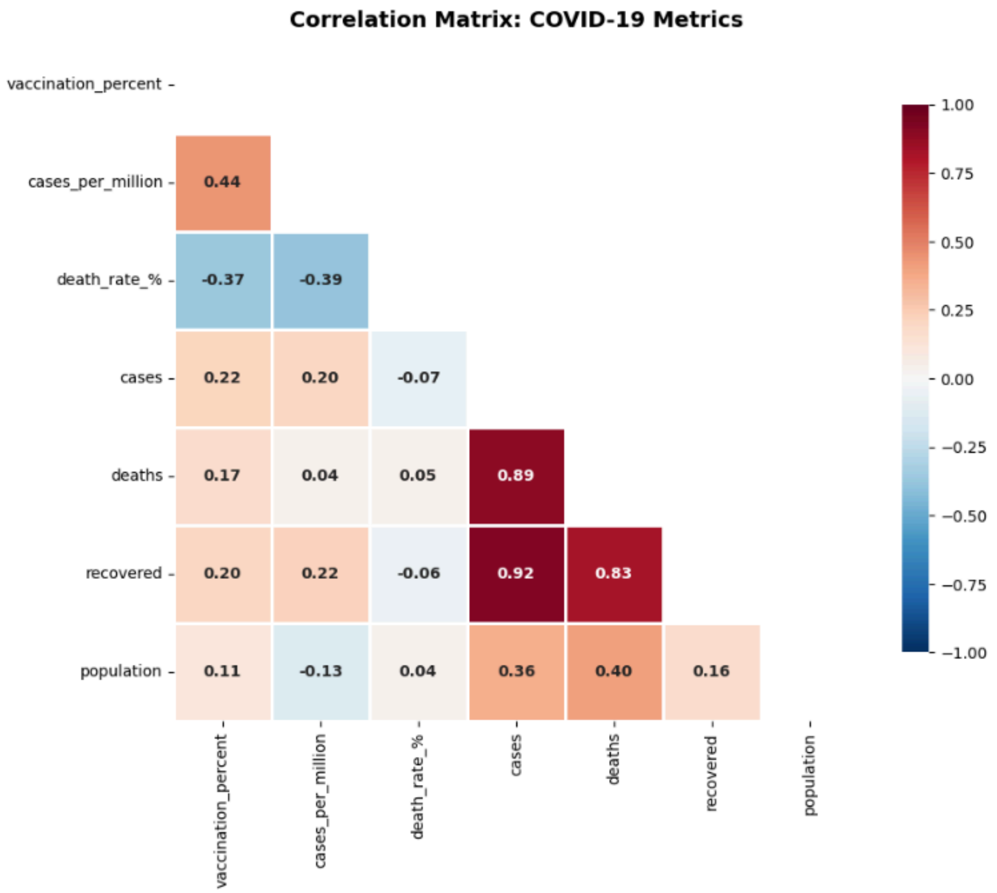
Red dotted line shows positive correlations between Vaccination percentage and Cases per million. But it does not mean that vaccines are not effective and insrease cases. Possible reasons:

1)Countries with high vaccination rates usually have a better healthcare system → more testing → more cases detected

2)After vaccination, countries eased restrictions → more contacts → more cases (but less severe)



By this vizualization we understand that high vaccination rates help, but countries also need strong healthcare systems and effective pandemic response to minimize deaths.



Correlation analysis showed that the increase in the number of cases is closely related to the increase in deaths (r=0.89) and recoveries (r=0.92), which is logical — all indicators are growing simultaneously. A higher vaccination rate is associated with lower mortality (r≈-0.38), which confirms the effectiveness of vaccinations. At the same time, the population has almost no effect on the spread of the disease, which indicates the different scales of the pandemic's impact in large and small countries.