COVID-19 Data Analytics Using SQL

Introduction

The COVID-19 pandemic had a massive global impact, prompting the need to analyze data for better understanding and decision-making. This project explores COVID-19 trends using SQL to identify patterns in infections, deaths, and other health indicators across countries and continents. Using a Kaggle-sourced dataset, the analysis focuses on querying, summarizing, and interpreting COVID-19 metrics through MySQL Workbench.

Abstract

This project involved analyzing a global COVID-19 dataset using SQL to extract valuable insights. The dataset contained daily reports from multiple countries, including variables such as total cases, new cases, deaths, and reproduction rate. We performed data cleaning, transformation, and analytical querying to understand trends such as top countries by total cases, death rates, daily spikes, and monthly averages. The project also included comparison between countries and trend analysis using SQL window functions. The findings help illustrate how the pandemic evolved across different regions.

Tools Used

- **MySQL Workbench** – for writing and running SQL queries
- **Kaggle Dataset** – source of COVID-19 data (CSV format)
- **Excel / CSV Exports** – for saving query results

Steps Involved in Building the Project

1. **Dataset Import**
   - Downloaded dataset from Kaggle
   - Imported the cleaned .csv file using MySQL Workbench import wizard
2. **Table Creation**
   - Created the covid_analysis database
   - Defined covid_stats table with 18 columns (including dates, cases, deaths, etc.)
3. **Data Cleaning**
   - Checked for NULL values (e.g., in reproduction_rate)
   - Verified date format, total rows, and country consistency

4. **Analytical Querying**

- o Top 10 countries by total cases and deaths

- o Daily new cases and smoothed trends for India and Japan

- o Reproduction rate analysis by country

- o Missing data in countries like Hong Kong

- o Monthly averages using GROUP BY and date formatting

- o Comparison of trends across India, Japan, and Hong Kong

5. **Window Functions**

- o Implemented 7-day rolling average of new cases using SQL window function

6. **Result Export**

- o Exported selected query results (e.g., top countries, daily trends) as CSV files for report and GitHub submission

---

Conclusion

This project demonstrated how SQL can be used effectively for real-world data analysis. By querying the COVID-19 dataset, we were able to uncover critical insights into how different countries were affected during the pandemic. The use of SQL functions like GROUP BY, aggregation, and window functions made the analysis more powerful. This project helped strengthen skills in database design, data cleaning, and analytical reporting, and is useful for presenting in interviews and portfolios.