**Assignment 3 - Visual Mapping**

**Weilin Ruan 50018083 Group5**

**Q1: What is the data type of each data attribute (*date, location, new\_cases*, …)?**

Data Types of Each Data Attribute

* date: String (can be converted to datetime for analysis)
* location: String
* new\_cases: Integer
* new\_deaths: Integer
* total\_cases: Integer
* total\_deaths: Integer
* weekly\_cases: Integer
* weekly\_deaths: Integer
* bi\_weekly\_cases: Integer
* bi\_weekly\_deaths: Integer

**Q2. Describe what you have learned from the data and the visualization. For instance, which country has the most number of total cases/deaths? Which country has the highest case-fatality rate?**

* **Country with Most Total Cases/Deaths**: From the visualization, you can quickly identify which country has the largest values for total\_cases and total\_deaths by looking at the top-right corner.
* **Highest Case-Fatality Rate**: The country with the highest case-fatality rate can be identified by hovering over the data points and comparing the calculated rates.

**Extension of the Scatterplot**

To add further information such as population and continent:

* **Population**: Use the size of each data point to represent the population. Larger circles can indicate larger populations.
* **Continent**: Use colors to differentiate between continents. Assign a different color to each continent to enable quick visual differentiation.

This mapping can be achieved by merging additional data containing population and continent information with the main dataset and using these attributes in the visual encoding.

**Coding**

To complete the task, we will visualize total\_cases and total\_deaths on the latest day in the dataset using a scatter plot. We will also include interactivity to display the country/region name and case-fatality rate on hover.

1. import pandas as pd
2. import plotly.express as px
3. *# Load the data*
4. data\_url = 'https://raw.githubusercontent.com/owid/covid-19-data/master/public/data/jhu/full\_data.csv' *# get newest data*
5. df = pd.read\_csv(data\_url)
6. *# df = pd.read\_csv("full\_data-1.csv") # or use local data*
7. *# Get the latest date in the data*
8. latest\_date = df['date'].max()
9. *# Filter the data for the latest date*
10. latest\_data = df[df['date'] == latest\_date].copy()
11. *# Calculate case-fatality rate using .loc*
12. latest\_data.loc[:, 'case\_fatality\_rate'] = (latest\_data['total\_deaths'] / latest\_data['total\_cases']) \* 100
13. *# Create the scatter plot*
14. fig = px.scatter(
15. latest\_data,
16. x='total\_cases',
17. y='total\_deaths',
18. text='location',
19. hover\_data={
20. 'location': True,
21. 'case\_fatality\_rate': ':.2f',
22. 'total\_cases': True,
23. 'total\_deaths': True
24. },
25. labels={'total\_cases': 'Total Cases', 'total\_deaths': 'Total Deaths'},
26. title='COVID-19 Total Cases vs Total Deaths',
27. log\_x=True,
28. log\_y=True
29. )
30. *# Update layout for better display*
31. fig.update\_traces(marker=dict(size=20, opacity=0.7),
32. selector=dict(mode='markers+text'))
33. fig.update\_layout(
34. hovermode='closest',
35. width=2000,  *# Set the width of the figure*
36. height=1200   *# Set the height of the figure*
37. )
38. *# Show the plot*
39. fig.show()

**Results:**

|  |  |
| --- | --- |
| Figure 1 interaction of the scatterplot | Figure 2 scatter plot snapshot |