In [40]:
```
import pandas as pd
import numpy as np
```

In [41]:
```
covid=pd.read_csv("https://query.data.world/s/gyrebzvychodiakuv4shr7wk4lqhih")
```

```
C:\Users\srika\anaconda3\lib\site-packages\IPython\core\interactiveshell.py:3
063: DtypeWarning: Columns (8) have mixed types.Specify dtype option on impor
t or set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

In [42]:
```
covid.head()
```

Out[42]:

	Case_Type	People_Total_Tested_Count	Cases	Difference	Date	Combined_Key	Country
0	Confirmed	NaN	0	0	2/3/2020	Switzerland	Sw
1	Deaths	NaN	0	0	3/1/2020	Cyprus	
2	Confirmed	NaN	23	0	4/21/2020	Antigua and Barbuda	Ant
3	Deaths	NaN	0	0	2/11/2020	Jamaica	
4	Confirmed	NaN	0	0	2/6/2020	Belize	

In [43]:
```
covid.shape
```

Out[43]: (718080, 18)

Data Preprocessing

In [44]:
```
covid.isna().sum()
```

Out[44]:
```
Case_Type                                0
People_Total_Tested_Count           715728
Cases                                    0
Difference                               0
Date                                     0
Combined_Key                             0
Country_Region                           0
Province_State                       37128
Admin2                               54264
iso2                                   408
iso3                                   204
FIPS                                 76092
Lat                                  21420
Long                                 21420
Population_Count                     21420
People_Hospitalized_Cumulative_Count    715728
Data_Source                              0
Prep_Flow_Runtime                        0
dtype: int64
```

Case Type, Country,Date does not have any Null Values

Dropping People_Total_Tested_Count,People_Hospitalized_Cumula coulmns as maximum values are Null

In [45]:
```
covid=covid.drop(columns=['People_Total_Tested_Count','People_Hospitalized_Cum
ulative_Count'])
```

In [46]:
```
covid.shape
```

Out[46]:
```
(718080, 16)
```

In [47]:
```
covid.isna().sum()
```

Out[47]:
```
Case_Type               0
Cases                   0
Difference              0
Date                    0
Combined_Key            0
Country_Region          0
Province_State      37128
Admin2              54264
iso2                  408
iso3                  204
FIPS                76092
Lat                 21420
Long                21420
Population_Count    21420
Data_Source             0
Prep_Flow_Runtime       0
dtype: int64
```

In [48]:
```
covid['Province_State']=covid['Province_State'].fillna('May be NA for This Country')
```

Felt Admin and FIPS are also not needed data so dropping off the columns

In [49]:
```
covid=covid.drop(columns=['Admin2','FIPS'])
```

Replacing Lat,Long and Population Count by Mean value

In [50]:
```
covid['Lat']=covid['Lat'].fillna(covid['Lat'].mean())
```

In [51]:
```
covid['Long']=covid['Long'].fillna(covid['Long'].mean())
```

In [52]:
```
covid['Population_Count']=covid['Population_Count'].fillna(covid['Population_Count'].mean())
```

In [53]:
```
covid.isna().sum()
```

Out[53]:
```
Case_Type             0
Cases                 0
Difference            0
Date                  0
Combined_Key          0
Country_Region        0
Province_State        0
iso2                408
iso3                204
Lat                   0
Long                  0
Population_Count      0
Data_Source           0
Prep_Flow_Runtime     0
dtype: int64
```

Dropping off iso2 and iso2 completly as those are less number of rows

In [54]:
```
covid.dropna(subset=['iso2','iso3'],inplace=True)
```

In [55]:
```
covid.shape
```

Out[55]:
```
(717672, 14)
```

In [56]:
```
718080-717672
```

Out[56]:
```
408
```

Original Data Set contains 718080 rows and with preprocessing came down to 717672 which is 408 less than original Data

Visualization

In [ ]:
```
import matplotlib.pyplot as plt
import seaborn as sns
```

In [75]:
```
ind=covid.loc[covid['Country_Region']=='Switzerland','Country_Region']
```

In [76]:
```
ind
```

Out[76]:
```
0         Switzerland
401       Switzerland
582       Switzerland
606       Switzerland
1020      Switzerland
             ...
101010    Switzerland
101240    Switzerland
101693    Switzerland
650585    Switzerland
679257    Switzerland
Name: Country_Region, Length: 204, dtype: object
```

In [ ]:
```
plt.figure(figsize=(16,6))
plt.title("CountryWise Projection") #Only numeric values can be plotted so com
pare non-catgegorical values
#sns.barplot(x=covid['Country_Region'],y=covid['Case_Type'].value_counts())
sns.barplot(x=ind,y=covid['Case_Type'].count_values())
```

In [ ]:
```
sns.set_style('dark')
sns.lineplot(data = covid)
```

In [ ]:
```

```