# Sucidial Rate

**Problem Statement: (1) Exploratory Data Analysis (2) Machine Learning Model Prediction**
**Data Source: Suicidial Data set taken from Kaggle**

- **My idea is to perform EDA and visualise this data using packages like plotly, matplotlib using Pyspark**
- **To fit a model using Pyspark Machine Learning techniques**

**For the scope of submitting the progress, I have performed EDA using Pandas module. The final project will be presented in Pyspark**

In [1]:
```
import pandas as pd
import missingno as msno
import plotly as py
import plotly.graph_objs as go
import plotly.express as px
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot
init_notebook_mode(connected=True)
```

In [2]:
```
#Loading the csv file into pandas
df = pd.read_csv("C:\\Users\\Anuja\\OneDrive\\Desktop\\Data 603\\Suicidal_Pyspark\\mast
```

## Exploratory Data Analysis

Checking the information about the dataframe

In [9]:
```
print(df.info())
print(df.describe())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 27820 entries, 0 to 27819
Data columns (total 12 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   country             27820 non-null  object 
 1   year                27820 non-null  int64  
 2   sex                 27820 non-null  object 
 3   age                 27820 non-null  object 
 4   suicides_no         27820 non-null  int64  
 5   population          27820 non-null  int64  
 6   suicides/100k pop   27820 non-null  float64
 7   country-year        27820 non-null  object 
 8   HDI for year        8364 non-null   float64
 9    gdp_for_year ($)   27820 non-null  object 
 10  gdp_per_capita ($)  27820 non-null  int64  
 11  generation          27820 non-null  object 
dtypes: float64(2), int64(4), object(6)
memory usage: 2.5+ MB
None
                year   suicides_no    population  suicides/100k pop  \
count  27820.000000  27820.000000  2.782000e+04       27820.000000
```

```
mean    2001.258375    242.574407  1.844794e+06          12.816097
std        8.469055    902.047917  3.911779e+06          18.961511
min     1985.000000      0.000000  2.780000e+02           0.000000
25%     1995.000000      3.000000  9.749850e+04           0.920000
50%     2002.000000     25.000000  4.301500e+05           5.990000
75%     2008.000000    131.000000  1.486143e+06          16.620000
max     2016.000000  22338.000000  4.380521e+07         224.970000

       HDI for year  gdp_per_capita ($)
count   8364.000000        27820.000000
mean       0.776601        16866.464414
std        0.093367        18887.576472
min        0.483000          251.000000
25%        0.713000         3447.000000
50%        0.779000         9372.000000
75%        0.855000        24874.000000
max        0.944000       126352.000000
```

Number of rows: 27820

Number of columns: 12

Data types: Object, Float, Integer

1. Checking for Null Values

In [4]:
```
df.isnull().sum()
```

Out[4]:
```
country                 0
year                    0
sex                     0
age                     0
suicides_no             0
population              0
suicides/100k pop       0
country-year            0
HDI for year        19456
 gdp_for_year ($)       0
gdp_per_capita ($)      0
generation              0
dtype: int64
```

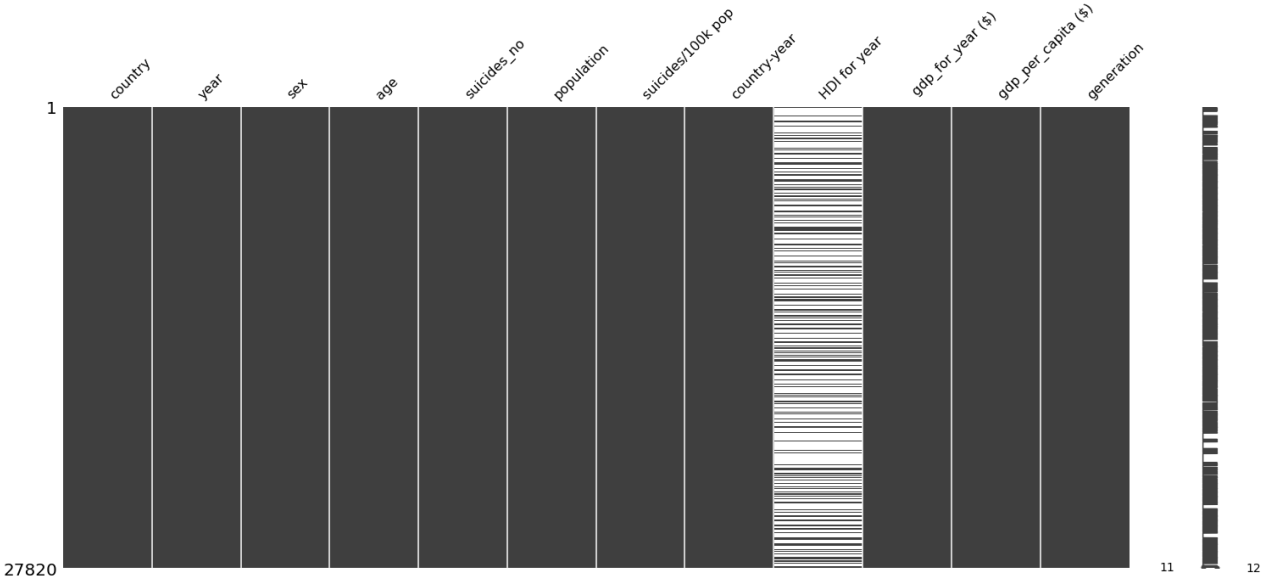
In [5]:
```
#percentage of null values in all columns of the dataframe:
df.isnull().sum() * 100 / len(df)
```

Out[5]:
```
country              0.000000
year                 0.000000
sex                  0.000000
age                  0.000000
suicides_no          0.000000
population           0.000000
suicides/100k pop    0.000000
country-year         0.000000
HDI for year        69.935298
 gdp_for_year ($)    0.000000
gdp_per_capita ($)   0.000000
generation           0.000000
dtype: float64
```

Let's visualise it using missigno library

In [10]:
```
msno.matrix(df)
```

Out[10]: <AxesSubplot:>




Column 'HDI for year' (Human Development Index is a statistic composite index of life expectancy, education, and per capita income indicators) contains 19456 null values.

Also, about 69% of the values in 'HDI for year' column is empty. Therefore dropping the column 'HDI for year'

In [ ]:
```
#df.drop('HDI for year', axis =1)
```

The column 'sex' seems a categorical variable. Let's see it's type. If the datatype is not category let's convert it into a category

In [ ]:
```
print("Originially the type of column sex is", df['sex'].dtype)
df['sex'] = df['sex'].astype('category')
print("After changing the datatype of column sex it is", df['sex'].dtype)
```

1. Unique Values

In [11]:
```
unique_value = [col for col in df.columns if df[col].nunique() <= 1]
print(f"There are {len(unique_value)} columns in dataframe with one unique value")
```

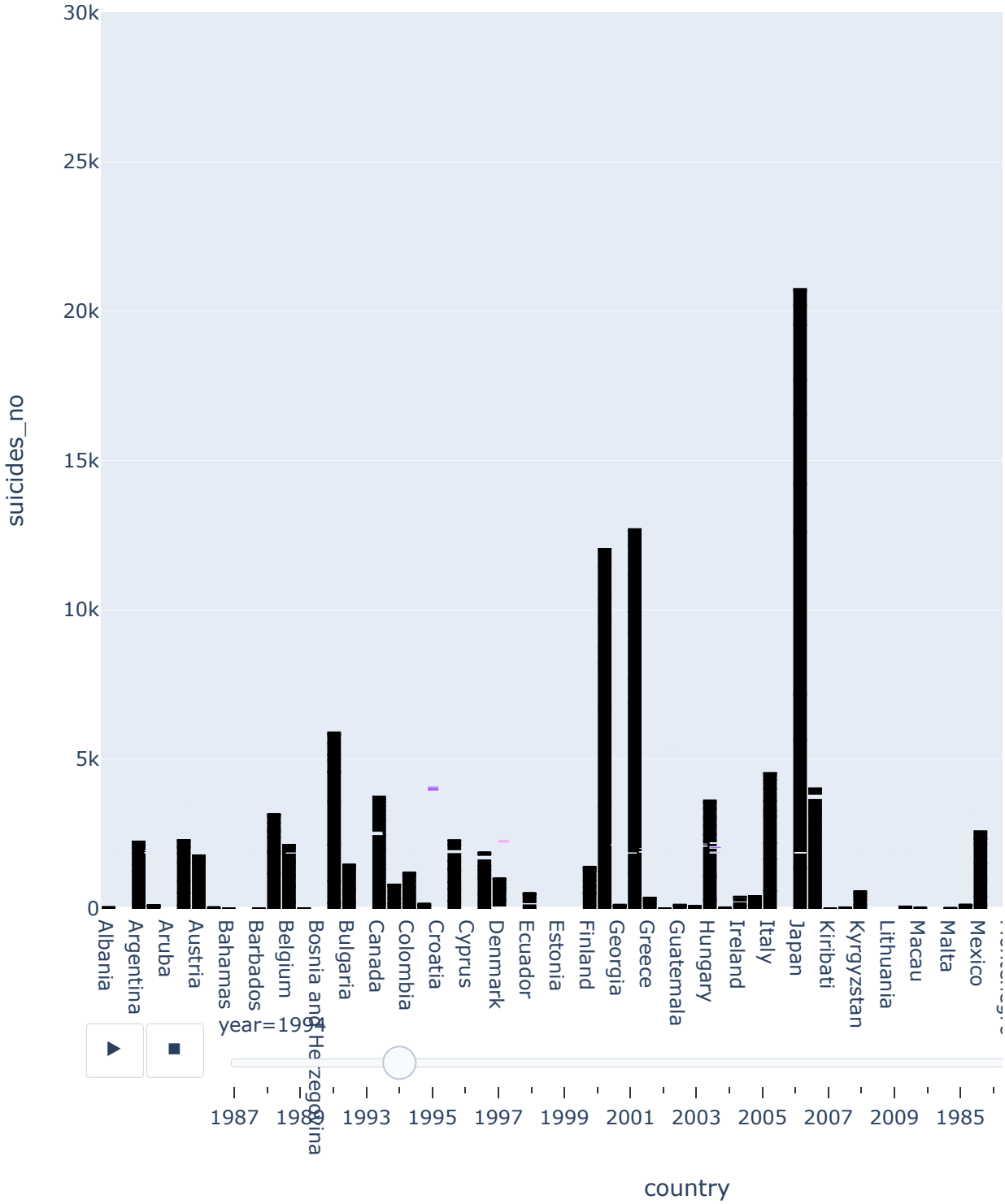
```
There are 0 columns in dataframe with one unique value
```

Let's visualise some of the columns and get insights from them:

In [6]:
```
fig = px.bar(df, x="country", y="suicides_no", color="country",
  animation_frame="year", animation_group="suicides_no", width=1000)
fig.update_layout(yaxis_range=[0,30000], height = 800, title = " Number of suicides per

py.offline.iplot(fig)
```

## Number of suicides per country




In [7]:
```
country_group = df.groupby('country').count()['suicides_no'].reset_index().sort_values(
country_group.style.background_gradient(cmap = 'Reds')
```

Out[7]:

| | country | suicides_no |
|---|---|---|
| **57** | Mauritius | 382 |
| **6** | Austria | 382 |
| **61** | Netherlands | 382 |

| | country | suicides_no |
|---|---|---|
| **41** | Iceland | 382 |
| **15** | Brazil | 372 |
| **82** | Singapore | 372 |
| **28** | Ecuador | 372 |
| **86** | Spain | 372 |
| **71** | Puerto Rico | 372 |
| **58** | Mexico | 372 |
| **36** | Greece | 372 |
| **20** | Colombia | 372 |
| **19** | Chile | 372 |
| **53** | Luxembourg | 372 |
| **56** | Malta | 372 |
| **73** | Republic of Korea | 372 |
| **12** | Belgium | 372 |
| **43** | Israel | 372 |
| **97** | United Kingdom | 372 |
| **98** | United States | 372 |
| **44** | Italy | 372 |
| **46** | Japan | 372 |
| **2** | Argentina | 372 |
| **16** | Bulgaria | 360 |
| **38** | Guatemala | 360 |
| **42** | Ireland | 360 |
| **21** | Costa Rica | 360 |
| **5** | Australia | 360 |
| **33** | France | 360 |
| **64** | Norway | 360 |
| **89** | Sweden | 358 |
| **32** | Finland | 348 |
| **18** | Canada | 348 |
| **94** | Turkmenistan | 348 |
| **62** | New Zealand | 348 |
| **88** | Suriname | 336 |

| | country | suicides_no |
|---|---|---|
| **13** | Belize | 336 |
| **95** | Ukraine | 336 |
| **99** | Uruguay | 336 |
| **77** | Saint Lucia | 336 |
| **91** | Thailand | 334 |
| **74** | Romania | 334 |
| **70** | Portugal | 324 |
| **75** | Russian Federation | 324 |
| **92** | Trinidad and Tobago | 324 |
| **67** | Paraguay | 324 |
| **1** | Antigua and Barbuda | 324 |
| **25** | Czech Republic | 322 |
| **50** | Kyrgyzstan | 312 |
| **47** | Kazakhstan | 312 |
| **35** | Germany | 312 |
| **40** | Hungary | 310 |
| **37** | Grenada | 310 |
| **49** | Kuwait | 300 |
| **39** | Guyana | 300 |
| **78** | Saint Vincent and Grenadines | 300 |
| **10** | Barbados | 300 |
| **66** | Panama | 300 |
| **3** | Armenia | 298 |
| **29** | El Salvador | 288 |
| **23** | Cuba | 288 |
| **69** | Poland | 288 |
| **8** | Bahamas | 276 |
| **83** | Slovakia | 264 |
| **0** | Albania | 264 |
| **100** | Uzbekistan | 264 |
| **34** | Georgia | 264 |
| **26** | Denmark | 264 |
| **22** | Croatia | 262 |

| | country | suicides_no |
|---|---|---|
| **52** | Lithuania | 262 |
| **9** | Bahrain | 252 |
| **11** | Belarus | 252 |
| **90** | Switzerland | 252 |
| **84** | Slovenia | 252 |
| **30** | Estonia | 252 |
| **51** | Latvia | 252 |
| **85** | South Africa | 240 |
| **80** | Serbia | 216 |
| **81** | Seychelles | 216 |
| **45** | Jamaica | 204 |
| **7** | Azerbaijan | 192 |
| **68** | Philippines | 180 |
| **24** | Cyprus | 178 |
| **72** | Qatar | 178 |
| **4** | Aruba | 168 |
| **31** | Fiji | 132 |
| **87** | Sri Lanka | 132 |
| **48** | Kiribati | 132 |
| **60** | Montenegro | 120 |
| **55** | Maldives | 120 |
| **93** | Turkey | 84 |
| **63** | Nicaragua | 72 |
| **96** | United Arab Emirates | 72 |
| **65** | Oman | 36 |
| **76** | Saint Kitts and Nevis | 36 |
| **79** | San Marino | 36 |
| **14** | Bosnia and Herzegovina | 24 |
| **17** | Cabo Verde | 12 |
| **27** | Dominica | 12 |
| **54** | Macau | 12 |
| **59** | Mongolia | 10 |

**Mauritius, Austria, Netherlands, Iceland have the highest suicidal number. While Mongolia**

**has the least**