In [2]:
```
## Import the library
import pandas as pd
import numpy  as np
import matplotlib.pyplot as   plt
import seaborn as sns
import plotly.express as px
from ydata_profiling import ProfileReport
%matplotlib inline
import warnings
warnings.filterwarnings("ignore")
plt.style.use('fivethirtyeight')
sns.set()

pd.options.display.float_format = '{:,.2f}'.format
pd.options.display.max_rows = None
pd.options.display.max_columns = None
```

In [3]:
```
baby = pd.read_csv('https://raw.githubusercontent.com/guipsamora/pandas_exercises/master/06_Stats/US_Baby_Names/US_Baby_Names_rig
```

In [4]:
```
baby.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1016395 entries, 0 to 1016394
Data columns (total 7 columns):
 #   Column       Non-Null Count    Dtype 
---  ------       --------------    ----- 
 0   Unnamed: 0   1016395 non-null  int64 
 1   Id           1016395 non-null  int64 
 2   Name         1016395 non-null  object
 3   Year         1016395 non-null  int64 
 4   Gender       1016395 non-null  object
 5   State        1016395 non-null  object
 6   Count        1016395 non-null  int64 
dtypes: int64(4), object(3)
memory usage: 54.3+ MB
```

In [5]:
```
baby.sample(10)
```

Out[5]:

	Unnamed: 0	Id	Name	Year	Gender	State	Count
892918	5056522	5056523	Luis	2007	M	TX	1487
680509	3852676	3852677	Amani	2007	M	NY	13
143859	759096	759097	Mara	2008	F	CO	8
271677	1421934	1421935	Kailee	2005	F	IA	5
873235	4925270	4925271	Annalynn	2011	F	TX	9
110865	682042	682043	Sebastien	2005	M	CA	29
512940	2863621	2863622	Tanner	2004	M	MO	91
744812	4158846	4158847	Mace	2012	M	OK	5
495235	2738376	2738377	Kellen	2008	M	MN	28
873013	4925048	4925049	Emberly	2011	F	TX	11

In [6]:
```
baby.duplicated().sum()
```

Out[6]:
```
0
```

See the first 10 entries.

In [7]:
```
baby.head(10)
```

Out[7]:

	Unnamed: 0	Id	Name	Year	Gender	State	Count
0	11349	11350	Emma	2004	F	AK	62
1	11350	11351	Madison	2004	F	AK	48
2	11351	11352	Hannah	2004	F	AK	46
3	11352	11353	Grace	2004	F	AK	44
4	11353	11354	Emily	2004	F	AK	41
5	11354	11355	Abigail	2004	F	AK	37
6	11355	11356	Olivia	2004	F	AK	33
7	11356	11357	Isabella	2004	F	AK	30
8	11357	11358	Alyssa	2004	F	AK	29
9	11358	11359	Sophia	2004	F	AK	28

Delete the columns ‘Unnamed: 0’ and ‘Id’.

In [10]:
```
baby.drop('Id',axis=1,inplace=True)
```

In [11]:
```
baby.drop('Unnamed: 0',axis=1,inplace=True)
```

In [58]:
```
baby.sample(5)
```

Out[58]:

	Name	Year	Gender	State	Count
404020	Eddie	2012	M	LA	9
520527	Brenden	2012	M	MO	9
81358	Angelee	2008	F	CA	9
358351	Sierra	2008	F	KS	23
526983	Kaydence	2010	F	MS	13

Group the dataset by name, assign to a variable called names, and sort the dataset by highest to lowest count.

In [59]:
```
names=baby.groupby('Name')['Count'].count().sort_values(ascending=False).reset_index().head(10)
names
```

Out[59]:

	Name	Count
0	Riley	1112
1	Avery	1080
2	Jordan	1073
3	Peyton	1064
4	Hayden	1049
5	Taylor	1033
6	Jayden	1031
7	Alexis	984
8	Payton	971
9	Angel	962

How many different names exist in the dataset?

In [25]:
```
baby.Name.nunique()
```

Out[25]:
```
17632
```

What is the name with most occurrences?

In [29]:
```
baby.Name.mode()
```

Out[29]:
```
0    Riley
Name: Name, dtype: object
```

What is the standard deviation of count of names?

In [33]:
```
baby.Count.std()
```

Out[33]:
```
97.39734648625934
```

Get a summary of the dataset with the mean, min, max, std and quartiles.

In [34]:
```
baby.describe()
```

Out[34]:

	Year	Count
count	1,016,395.00	1,016,395.00
mean	2,009.05	34.85
std	3.14	97.40
min	2,004.00	5.00
25%	2,006.00	7.00
50%	2,009.00	11.00
75%	2,012.00	26.00
max	2,014.00	4,167.00