US - Baby Names

Introduction:

We are going to use a subset of US Baby Names from Kaggle.
In the file it will be names from 2004 until 2014

Step 1. Import the necessary libraries

```
import pandas as pd
```

Step 2. Import the dataset from this address.

```
url = "https://raw.githubusercontent.com/thieu1995/csv-files/main/data/pandas/US_Baby_Names_right.csv"
```

Step 3. Assign it to a variable called baby_names.

```
baby_names = pd.read_csv(url)
```

Step 4. See the first 10 entries

```
print(baby_names.head(10))
```

```
   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
```

Step 5. Delete the column 'Unnamed: 0' and 'Id'

```
baby_names = baby_names.drop(columns=['Unnamed: 0', 'Id'])
```

```
print(baby_names.head())
```

```
       Name  Year Gender State  Count
0     Emma  2004      F    AK     62
1  Madison  2004      F    AK     48
2   Hannah  2004      F    AK     46
3    Grace  2004      F    AK     44
4    Emily  2004      F    AK     41
```

Step 6. Is there more male or female names in the dataset?

```
male = baby_names['Gender'].value_counts()['M']
female = baby_names['Gender'].value_counts()['F']

print(male > female)
```

```
False
```

Step 7. Group the dataset by name and assign to names

```
names = baby_names.groupby('Name')

print(names.head())
```

```
            Name  Year Gender State  Count
0           Emma  2004      F    AK     62
```

```
1        Madison  2004      F    AK     48
2         Hannah  2004      F    AK     46
3          Grace  2004      F    AK     44
4          Emily  2004      F    AK     41
...          ...   ...    ...   ...    ...
1004923  Gryffin  2014      M    WI      5
1004950     Kroy  2014      M    WI      5
1004973     Owyn  2014      M    WI      5
1005707   Haylea  2005      F    WV      5
1012216  Coalton  2012      M    WV      7

[65502 rows x 5 columns]
```

Step 8. How many different names exist in the dataset?

```
print(names['Name'].nunique().count())
```

```
17632
```

Step 9. What is the name with most occurrences?

```
max_name = names['Name'].value_counts().idxmax()
print(max_name)
```

```
Riley
```

Step 10. How many different names have the least occurrences?

```
max_name = names['Name'].value_counts().min()
print(max_name)
```

```
1
```

Step 11. What is the median name occurrence?

```
median = names['Name'].value_counts().median()
print(median)
```

```
8.0
```

Step 12. What is the standard deviation of names?

```
standard_daviation = names['Name'].value_counts().std()
print(standard_daviation)
```

```
122.02996350814125
```

Step 13. Get a summary with the mean, min, max, std and quartiles.

```
print(names['Name'].value_counts().describe())
```

```
count    17632.000000
mean        57.644907
std        122.029964
min          1.000000
25%          2.000000
50%          8.000000
75%         39.000000
max       1112.000000
Name: count, dtype: float64
```