In [13]:
```
import pandas as pd
import matplotlib.pyplot as plt
import plotly.express as px
import seaborn as sns
import numpy as np
```

In [2]:
```
birth_df = pd.read_csv('births.csv')
```

In [3]:
```
birth_df
```

Out[3]:

	year	month	day	gender	births
0	1969	1	1.0	F	4046
1	1969	1	1.0	M	4440
2	1969	1	2.0	F	4454
3	1969	1	2.0	M	4548
4	1969	1	3.0	F	4548
...	...	...	...	...	...
15542	2008	10	NaN	M	183219
15543	2008	11	NaN	F	158939
15544	2008	11	NaN	M	165468
15545	2008	12	NaN	F	173215
15546	2008	12	NaN	M	181235

15547 rows × 5 columns

q1= make columns of decates

In [60]:
```
def year_to_decade(year):
    return str(year // 10 * 10)
birth_df['Decade'] = birth_df['year'].apply(year_to_decade)
```

In [62]:
```
# birth_df.drop(columns=['Decades'],inplace=True)
```

In [63]:
```
birth_df
```

Out[63]:

	year	month	day	gender	births	Decade
0	1969	1	1.0	F	4046	1960
1	1969	1	1.0	M	4440	1960
2	1969	1	2.0	F	4454	1960
3	1969	1	2.0	M	4548	1960
4	1969	1	3.0	F	4548	1960
...	...	...	...	...	...	...
15542	2008	10	NaN	M	183219	2000
15543	2008	11	NaN	F	158939	2000
15544	2008	11	NaN	M	165468	2000
15545	2008	12	NaN	F	173215	2000
15546	2008	12	NaN	M	181235	2000

15547 rows × 6 columns

q2=Discriptive statistics

In [64]:
```
birth_df.describe()
```

Out[64]:

	year	month	day	births
count	15547.000000	15547.000000	15067.000000	15547.000000
mean	1979.037435	6.515919	17.769894	9762.293561
std	6.728340	3.449632	15.284034	28552.465810
min	1969.000000	1.000000	1.000000	1.000000
25%	1974.000000	4.000000	8.000000	4358.000000
50%	1979.000000	7.000000	16.000000	4814.000000
75%	1984.000000	10.000000	24.000000	5289.500000
max	2008.000000	12.000000	99.000000	199622.000000

q3 = Checking missing values

In [65]:
```
birth_df.isnull().sum()
```

Out[65]:
```
year        0
month       0
day       480
gender      0
births      0
Decade      0
dtype: int64
```

In [ ]:

In [77]:
```
birth_df.day.value_counts().index.sort_values()
```

Out[77]:
```
Float64Index([ 1.0,  2.0,  3.0,  4.0,  5.0,  6.0,  7.0,  8.0,  9.0, 10.0, 11.0,
              12.0, 13.0, 14.0, 15.0, 16.0, 17.0, 18.0, 19.0, 20.0, 21.0, 22.0,
              23.0, 24.0, 25.0, 26.0, 27.0, 28.0, 29.0, 30.0, 31.0, 99.0],
             dtype='float64')
```

In [ ]:

q4 =trends of male & female every day

In [94]:
```
birth_df.head(20)
```

Out[94]:

	year	month	day	gender	births	Decade
0	1969	1	1.0	F	4046	1960
1	1969	1	1.0	M	4440	1960
2	1969	1	2.0	F	4454	1960
3	1969	1	2.0	M	4548	1960
4	1969	1	3.0	F	4548	1960
5	1969	1	3.0	M	4994	1960
6	1969	1	4.0	F	4440	1960
7	1969	1	4.0	M	4520	1960
8	1969	1	5.0	F	4192	1960
9	1969	1	5.0	M	4198	1960
10	1969	1	6.0	F	4710	1960
11	1969	1	6.0	M	4850	1960
12	1969	1	7.0	F	4646	1960
13	1969	1	7.0	M	5092	1960
14	1969	1	8.0	F	4800	1960
15	1969	1	8.0	M	4934	1960
16	1969	1	9.0	F	4592	1960
17	1969	1	9.0	M	4842	1960
18	1969	1	10.0	F	4852	1960
19	1969	1	10.0	M	5190	1960

In [118…
```
# sns.scatterplot(data=birth_df,x='Decade',y='births',hue='gender')
```

In [115…
```
trends = birth_df.groupby(['Decade', 'gender'])['births'].sum().unstack()
```

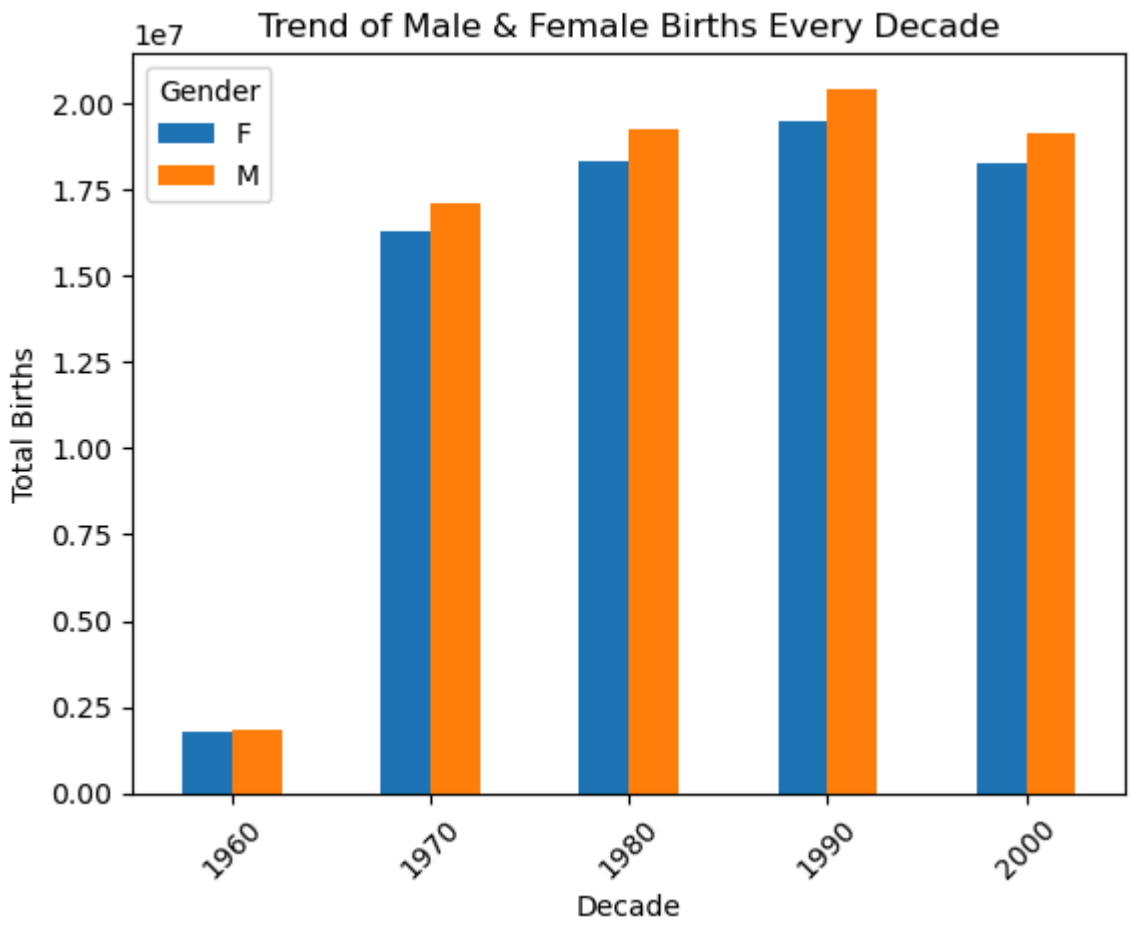
In [116…
```
trends
```

Out[116]:

gender	F	M
Decade		
1960	1753634	1846572
1970	16263075	17121550
1980	18310351	19243452
1990	19479454	20420553
2000	18229309	19106428

In [117…

```
trends.plot(kind='bar')
plt.title('Trend of Male & Female Births Every Decade')
plt.xlabel('Decade')
plt.ylabel('Total Births')
plt.xticks(rotation=45)
plt.legend(title='Gender')
plt.show()
```



In [ ]:

In [119…

```
birth_df
```

Out[119]:

	year	month	day	gender	births	Decade
0	1969	1	1.0	F	4046	1960
1	1969	1	1.0	M	4440	1960
2	1969	1	2.0	F	4454	1960
3	1969	1	2.0	M	4548	1960
4	1969	1	3.0	F	4548	1960
…	…	…	…	…	…	…
15542	2008	10	NaN	M	183219	2000
15543	2008	11	NaN	F	158939	2000
15544	2008	11	NaN	M	165468	2000
15545	2008	12	NaN	F	173215	2000
15546	2008	12	NaN	M	181235	2000

15547 rows × 6 columns

In [123…

```
mean = birth_df['day'].mean()
std = birth_df['day'].std()
```

In [124…

```
mean
```

Out[124]:

```
17.769894471361255
```

In [125…

```
std
```

Out[125]:

```
15.284034179234038
```

In [127…
```
birth_df.shape
```

Out[127]:
```
(15547, 6)
```

In [126…
```
upper_bound = mean + 5*std
lower_bound = mean - 5*std
```

In [131…
```
birth_df= birth_df[(birth_df['day']>=lower_bound) & (birth_df['day']<=upper_bound)]
```

In [143…
```
birth_df.shape
```

Out[143]:
```
(14717, 6)
```

q6=Q.6: Plot births by weekday for several decades.

In [ ]:
```
```

In [ ]:
```
def weekdays(day):
    return str(day // 7)
birth_df['weekdays'] = birth_df['day'].apply(weekdays)
```

In [212…
```
birth_df.head(20)
```

Out[212]:

	year	month	day	gender	births	Decade	weekdays
0	1969	1	1.0	F	4046	1960	0.0
1	1969	1	1.0	M	4440	1960	0.0
2	1969	1	2.0	F	4454	1960	0.0
3	1969	1	2.0	M	4548	1960	0.0
4	1969	1	3.0	F	4548	1960	0.0
5	1969	1	3.0	M	4994	1960	0.0
6	1969	1	4.0	F	4440	1960	0.0
7	1969	1	4.0	M	4520	1960	0.0
8	1969	1	5.0	F	4192	1960	0.0
9	1969	1	5.0	M	4198	1960	0.0
10	1969	1	6.0	F	4710	1960	0.0
11	1969	1	6.0	M	4850	1960	0.0
12	1969	1	7.0	F	4646	1960	1.0
13	1969	1	7.0	M	5092	1960	1.0
14	1969	1	8.0	F	4800	1960	1.0
15	1969	1	8.0	M	4934	1960	1.0
16	1969	1	9.0	F	4592	1960	1.0
17	1969	1	9.0	M	4842	1960	1.0
18	1969	1	10.0	F	4852	1960	1.0
19	1969	1	10.0	M	5190	1960	1.0

In [173…
```
grouped = birth_df.groupby(['Decade', 'weekdays'])['births'].sum().unstack()
grouped
```

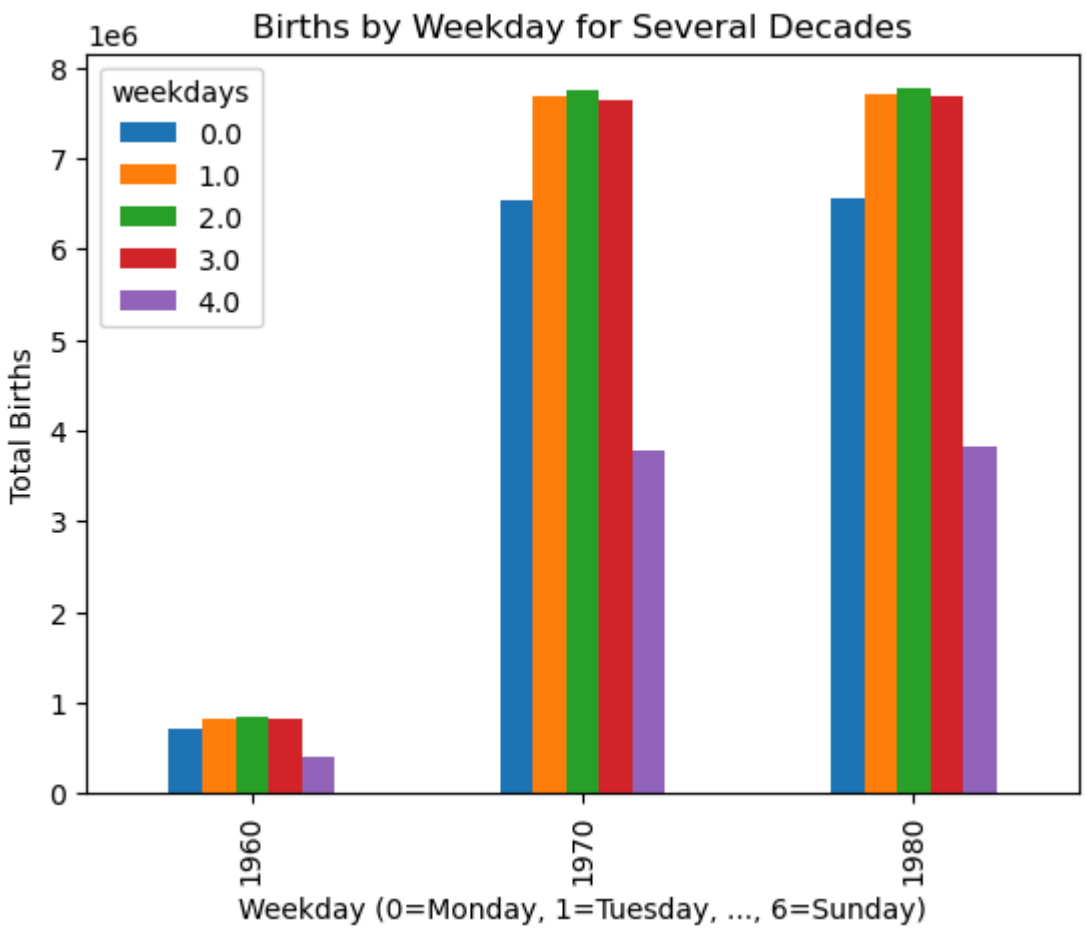
Out[173]:

weekdays	0.0	1.0	2.0	3.0	4.0
Decade					
1960	704732	828616	833222	824950	407516
1970	6530944	7688234	7741338	7640675	3778514
1980	6555045	7695272	7766094	7675594	3815792

In [222…
```
grouped.plot(kind='bar')

plt.title('Births by Weekday for Several Decades')
plt.xlabel('Weekday (0=Monday, 1=Tuesday, ..., 6=Sunday)')
plt.ylabel('Total Births')

plt.show()
```



In [ ]:

In [ ]:

q 7 =group the data by month and day separately

In [218…
```
birth_df.groupby(['Decade','month'])['births'].sum().unstack()
```

Out[218]:

Decade	month 1	2	3	4	5	6	7	8	9	10	11	12
1960	293876	270696	296436	282522	289018	291508	318288	320922	312512	311876	296958	314424
1970	2762078	2554865	2786246	2608957	2714618	2696181	2927375	2987439	2935258	2878587	2715465	2812636
1980	2703211	2537385	2785458	2669424	2781584	2771007	2974632	3001403	2958403	2864909	2683065	2777316

In [219…
```
# grouping with days
```

In [220…
```
birth_df.groupby(['Decade','day'])['births'].sum().unstack()
```

Out[220]:

Decade	day 1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0	...	22.0	23.0	24.0	25.0	26.0
1960	116182	117694	120302	117584	116600	116370	117868	119702	117202	119988	...	119068	116056	118082	118718	118410
1970	1084192	1089203	1093246	1079121	1088763	1096419	1098305	1097123	1096892	1101573	...	1091758	1088499	1084269	1078604	1092591
1980	1086480	1096271	1095539	1089883	1091588	1095284	1097908	1105551	1100961	1105636	...	1106885	1100619	1088149	1086653	1089207

3 rows × 31 columns

In [ ]: