

PYTHON ASSIGNMENT BY ASHWINI-08 AND RAKHI-14

```
import pandas as pd
import seaborn as sns
sns.set(color_codes=True)
```

```
music = pd.read_csv('WorldHits.csv')
```

```
music.head()
```

	Track	Artist	Album	Year	Duration	Time_Signature	Danceability	Energy	Key	Loudness	Mode	Speechiness	Acousticness	Instrumentalness
0	Release	Afro Celt Sound System	Volume 2: Release (Real World Gold)	2005	456160	4	0.633	0.828	5	-7.266	0	0.0480	0.0216	0.055800
1	Saor / Free / News from Nowhere	Afro Celt Sound System	Vol. 1: Sound Magic (Real World Gold)	1999	501093	4	0.511	0.524	7	-10.504	1	0.0305	0.0260	0.879000
2	When You're Falling	Afro Celt Sound System	Volume 3 : Further In Time (Real World Gold)	2003	314160	4	0.638	0.822	11	-7.305	1	0.0380	0.0508	0.000025
3	Whirl-Y-Reel 1	Afro Celt Sound System	Vol. 1: Sound Magic (Real World Gold)	1999	441200	4	0.645	0.810	2	-8.133	1	0.0381	0.1320	0.395000
4	Electric Rendezvous	Al Di Meola	Electric Rendezvous	1982	467266	4	0.337	0.658	9	-12.241	1	0.0388	0.1010	0.748000

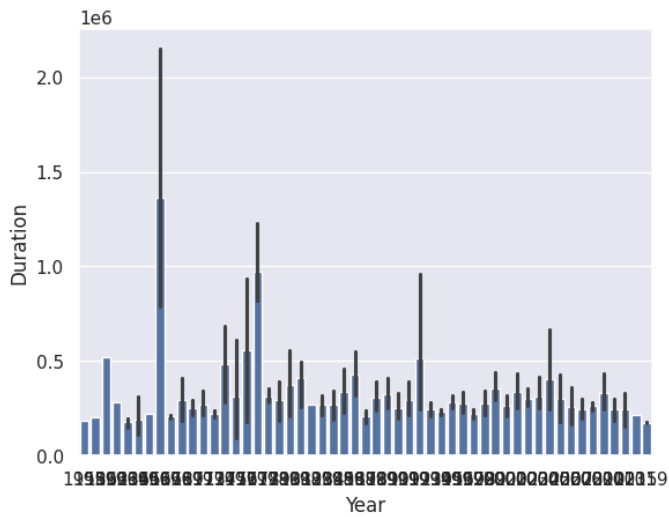
Next steps: [Generate code with music](#) [View recommended plots](#) [New interactive sheet](#)

```
music.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 326 entries, 0 to 325
Data columns (total 18 columns):
#   Column              Non-Null Count  Dtype
---  ---
0   Track               326 non-null   object
1   Artist              326 non-null   object
2   Album               326 non-null   object
3   Year                326 non-null   int64
4   Duration            326 non-null   int64
5   Time_Signature      326 non-null   int64
6   Danceability         326 non-null   float64
7   Energy              326 non-null   float64
8   Key                 326 non-null   int64
9   Loudness            326 non-null   float64
10  Mode                326 non-null   int64
11  Speechiness         326 non-null   float64
12  Acousticness        326 non-null   float64
13  Instrumentalness    326 non-null   float64
14  Liveness            326 non-null   float64
15  Valence             326 non-null   float64
16  Tempo               326 non-null   float64
17  Popularity          326 non-null   int64
dtypes: float64(9), int64(6), object(3)
memory usage: 46.0+ KB
```

```
sns.barplot( y=music['Duration'],x=music['Year']) #ashwini-8
```

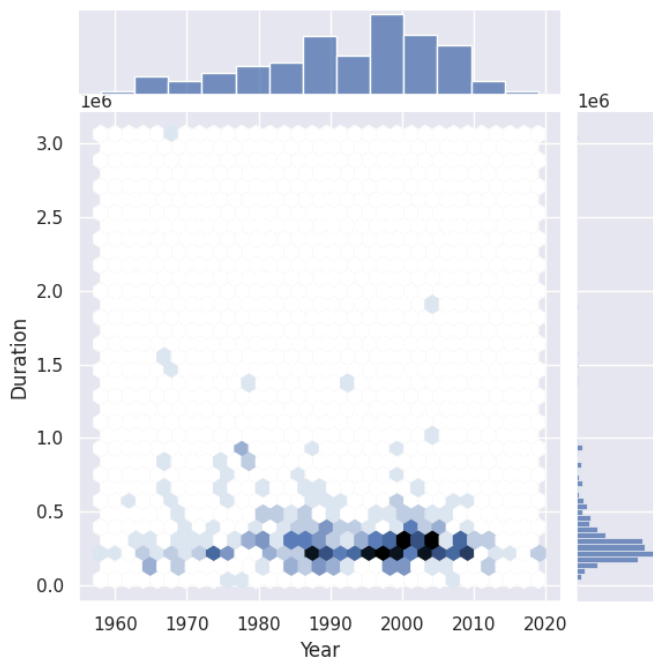
```
<Axes: xlabel='Year', ylabel='Duration'>
```



Start coding or [generate](#) with AI.

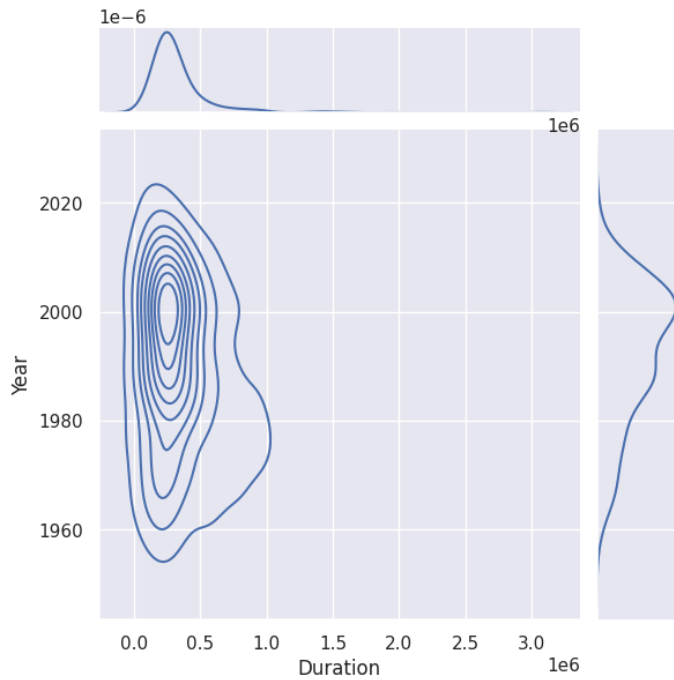
```
sns.jointplot(x=music['Year'], y=music['Duration'], kind="hex") #rakhi-14
```

```
<seaborn.axisgrid.JointGrid at 0x7afef6b17190>
```



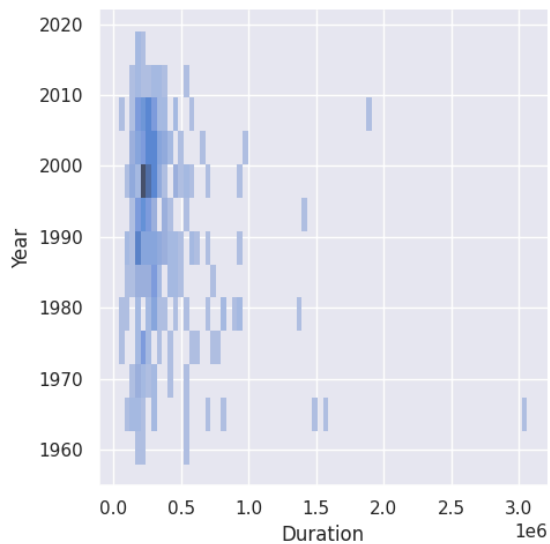
```
sns.jointplot( x=music['Duration'],y= music['Year'], kind="kde") #ashwini-8
```

```
<seaborn.axisgrid.JointGrid at 0x7afef12722c0>
```




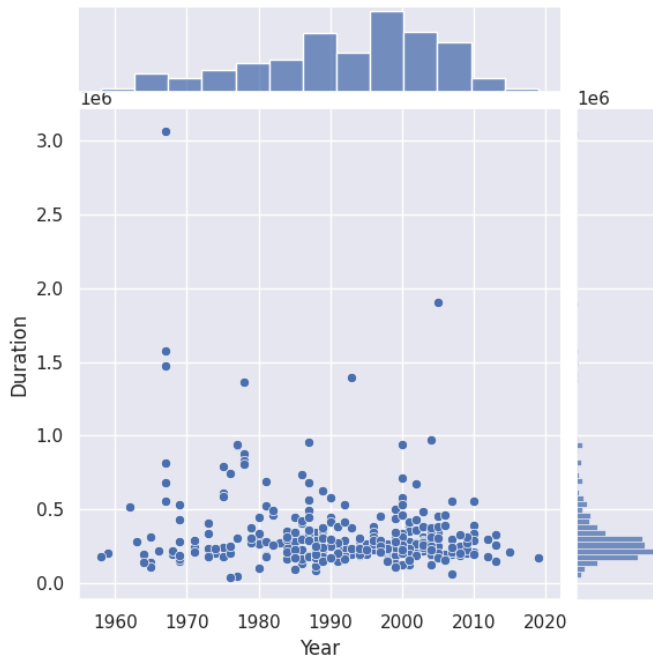
```
sns.displot(x=music['Duration'],y=music['Year'])#rakhi-14
```

```
<seaborn.axisgrid.FacetGrid at 0x7afef0d93790>
```




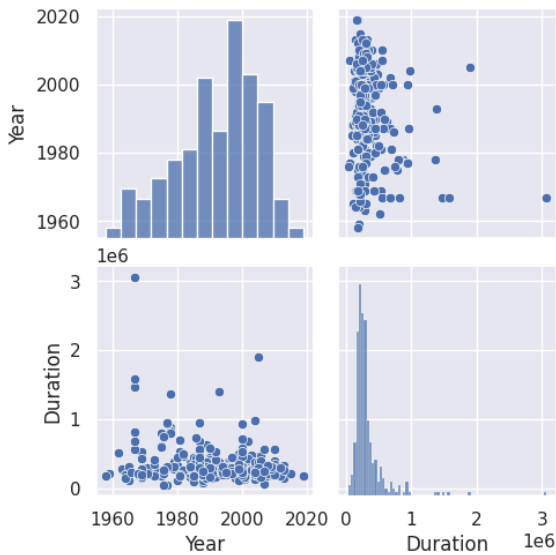
```
sns.jointplot(x=music['Year'],y=music['Duration']) #rakhi-14
```

 <seaborn.axisgrid.JointGrid at 0x7afef0d90880>



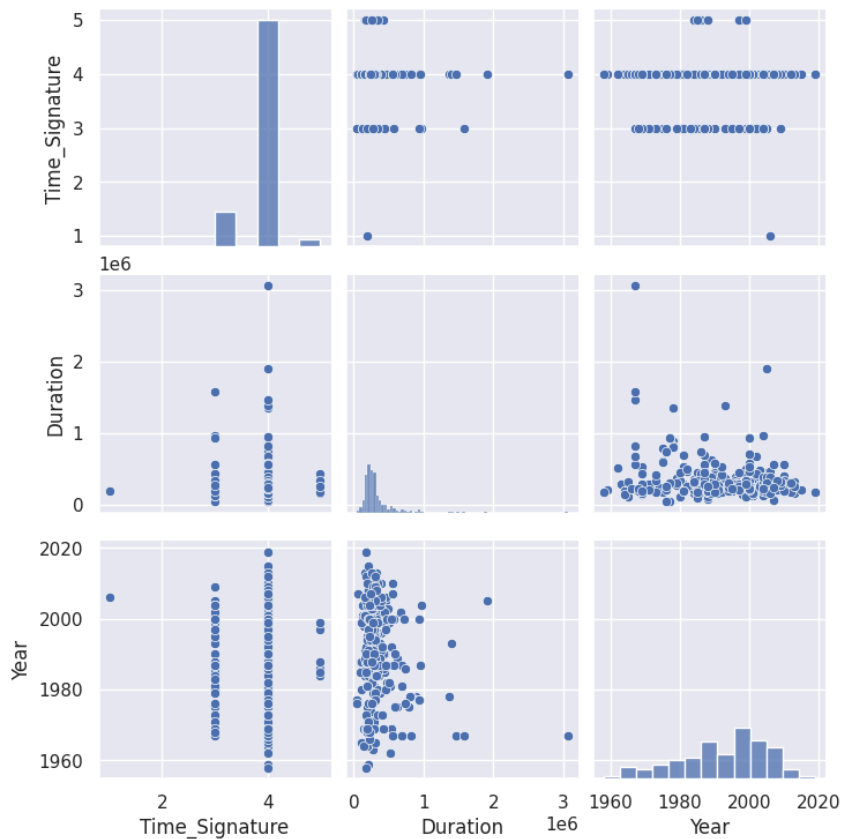
sns.pairplot(music[['Year', 'Duration']])#ashwini-8

 <seaborn.axisgrid.PairGrid at 0x7a5ea3c12f80>



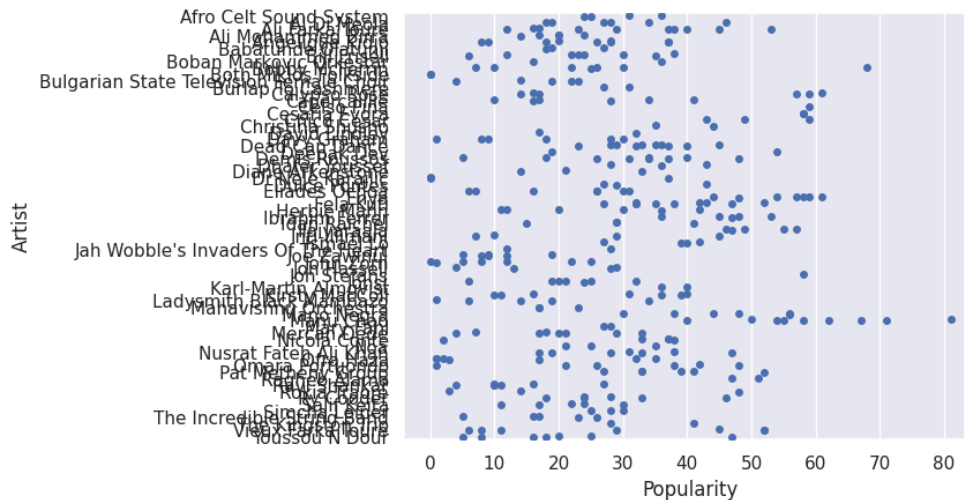
sns.pairplot(music[['Time_Signature', 'Duration', 'Year']]) #rakhi-14

<seaborn.axisgrid.PairGrid at 0x7afe6cd50f0>



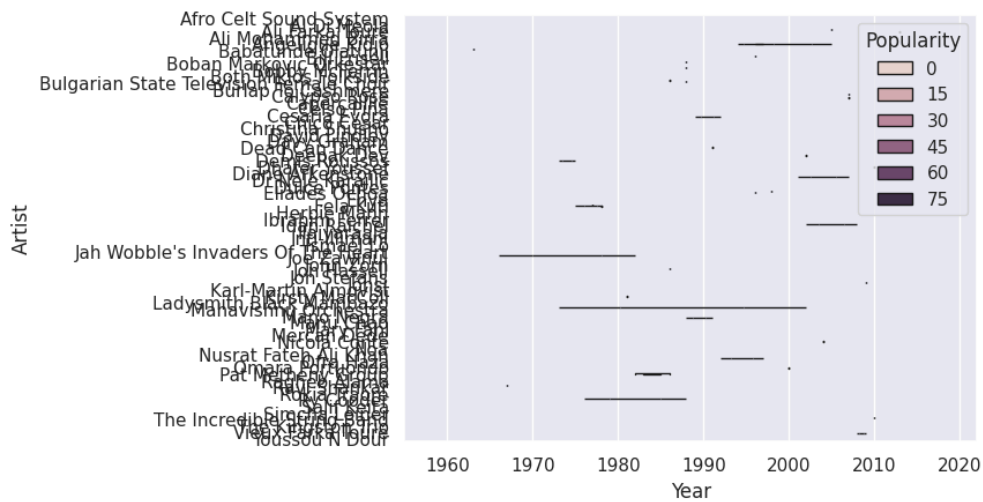
```
sns.stripplot(x=music['Popularity'], y=music['Artist'], jitter=True) #rakhi-14
```

<Axes: xlabel='Popularity', ylabel='Artist'>



```
sns.boxplot(x=music['Year'],y=music['Artist'],hue=music['Popularity']) #rakhi-14
```

```
<Axes: xlabel='Year', ylabel='Artist'>
```



```
sns.countplot(music['Artist']) #ashwini-8
```

```
<Axes: xlabel='count', ylabel='Artist'>
```

