

```
In [1]: 1 !gdown 1upJyp4hzZVONmwiZv0foKpA28uj2mJwo
```

Downloading...

From: <https://drive.google.com/uc?id=1upJyp4hzZVONmwiZv0foKpA28uj2mJwo> (<https://drive.google.com/uc?id=1upJyp4hzZVONmwiZv0foKpA28uj2mJwo>)

To: /Users/ritnil/Scaler-cohorts/final_vg.csv

100% | ██████████ 2.04M/2.04M [00:03<00:00, 640k B/s]

```
In [3]: 1 import pandas as pd
        2 import numpy as np
        3
        4 data = pd.read_csv('/Users/ritnil/Scaler-cohorts/final_vg.csv')
        5 data
```

Out[3]:

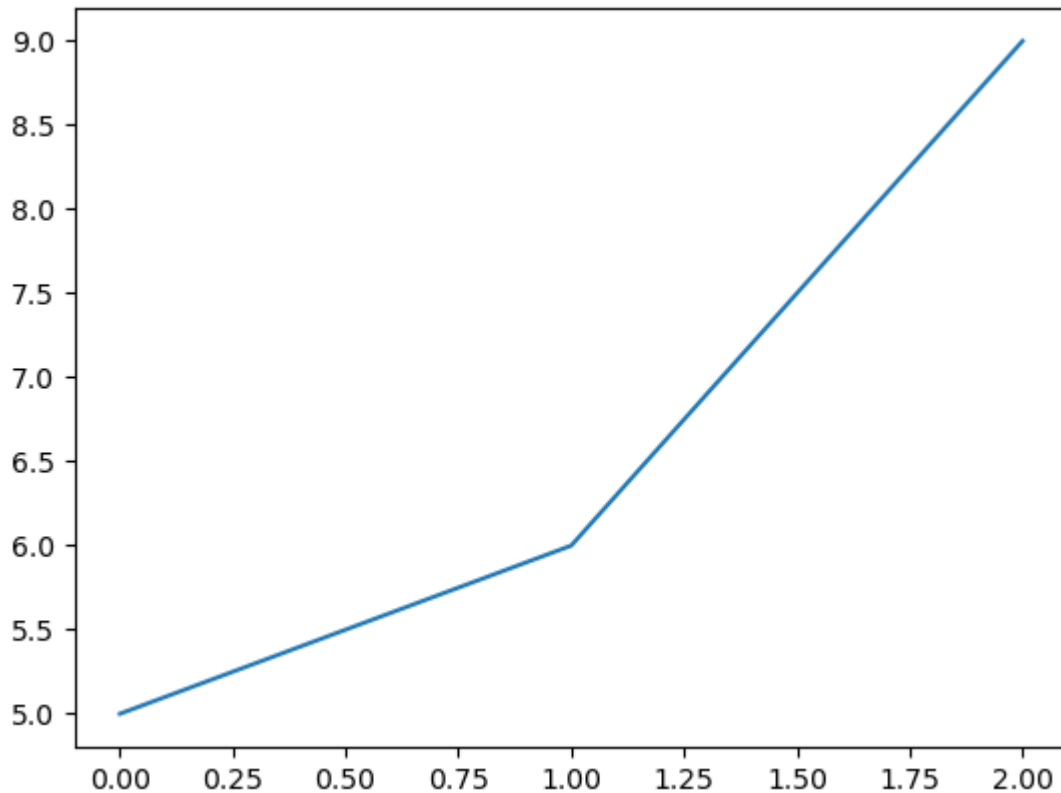
	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	J
0	2061	1942	NES	1985.0	Shooter	Capcom	4.569217	3.033887	3
1	9137	iShin Chan Flipa en colores!	DS	2007.0	Platform	505 Games	2.076955	1.493442	3
2	14279	.hack: Sekai no Mukou ni + Versus	PS3	2012.0	Action	Namco Bandai Games	1.145709	1.762339	1
3	8359	.hack//G.U. Vol.1//Rebirth	PS2	2006.0	Role-Playing	Namco Bandai Games	2.031986	1.389856	3
4	7109	.hack//G.U. Vol.2//Reminisce	PS2	2006.0	Role-Playing	Namco Bandai Games	2.792725	2.592054	1
...	
16647	7925	Zumba Fitness Rush	X360	2012.0	Sports	505 Games	4.409308	3.167419	4
16648	6279	Zumba Fitness: World Party	Wii	2013.0	Misc	Majesco Entertainment	3.033887	2.792725	1
16649	6977	Zumba Fitness: World Party	XOne	2013.0	Misc	Majesco Entertainment	3.228043	2.004268	1
16650	15422	Zwei!!	PSP	2008.0	Role-Playing	Falcom Corporation	1.087977	0.592445	1
16651	12919	Zyuden Sentai Kyoryuger: Game de Gaburincho!!	3DS	2013.0	Action	Namco Bandai Games	1.081046	1.714664	2

16652 rows x 11 columns

```
In [4]: 1 import matplotlib.pyplot as plt
        2 import seaborn as sns
```

```
In [5]: 1 x=[0,1,2]
        2 y=[5,6,9]
        3
        4 plt.plot(x,y)
```

Out[5]: [



```
In [6]: 1 #Univariate analysis
```

```
In [7]: 1 data["Genre"]
```

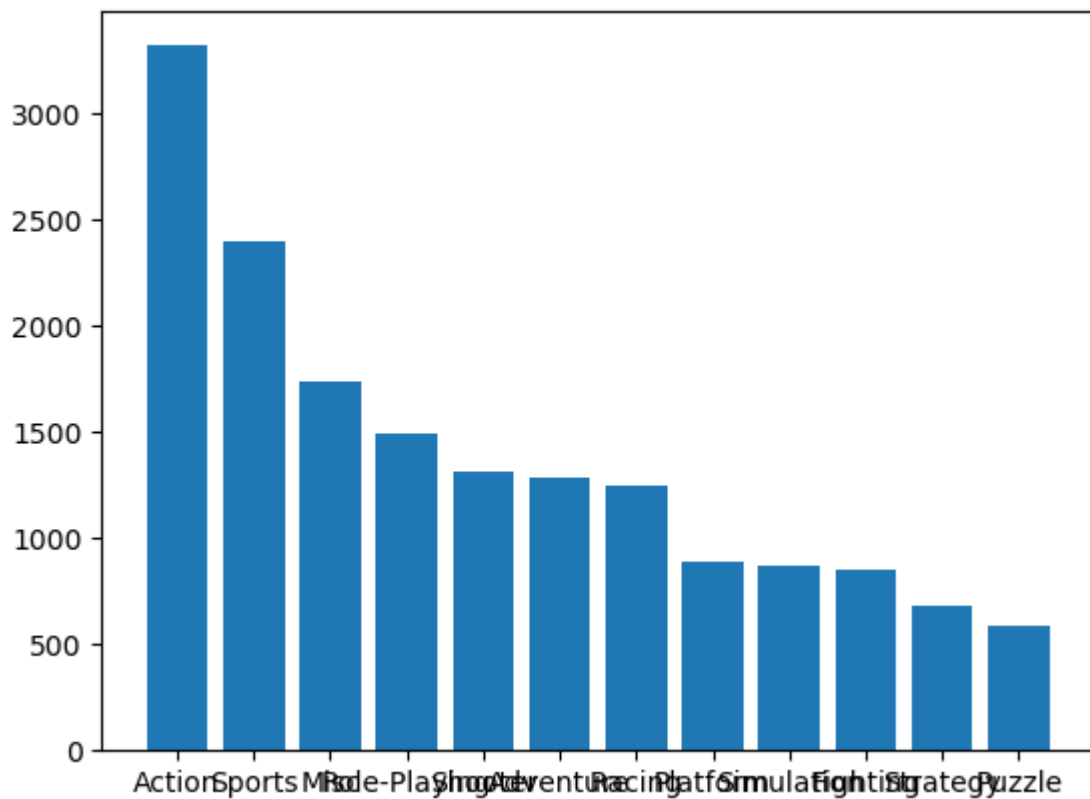
Out[7]: 0 Shooter
1 Platform
2 Action
3 Role-Playing
4 Role-Playing
...
16647 Sports
16648 Misc
16649 Misc
16650 Role-Playing
16651 Action
Name: Genre, Length: 16652, dtype: object

```
In [8]: 1 cat_counts=data["Genre"].value_counts()
        2 cat_counts
```

```
Out[8]: Action          3316
Sports          2400
Misc            1739
Role-Playing    1488
Shooter         1310
Adventure       1286
Racing          1249
Platform        886
Simulation       867
Fighting        848
Strategy        681
Puzzle          582
Name: Genre, dtype: int64
```

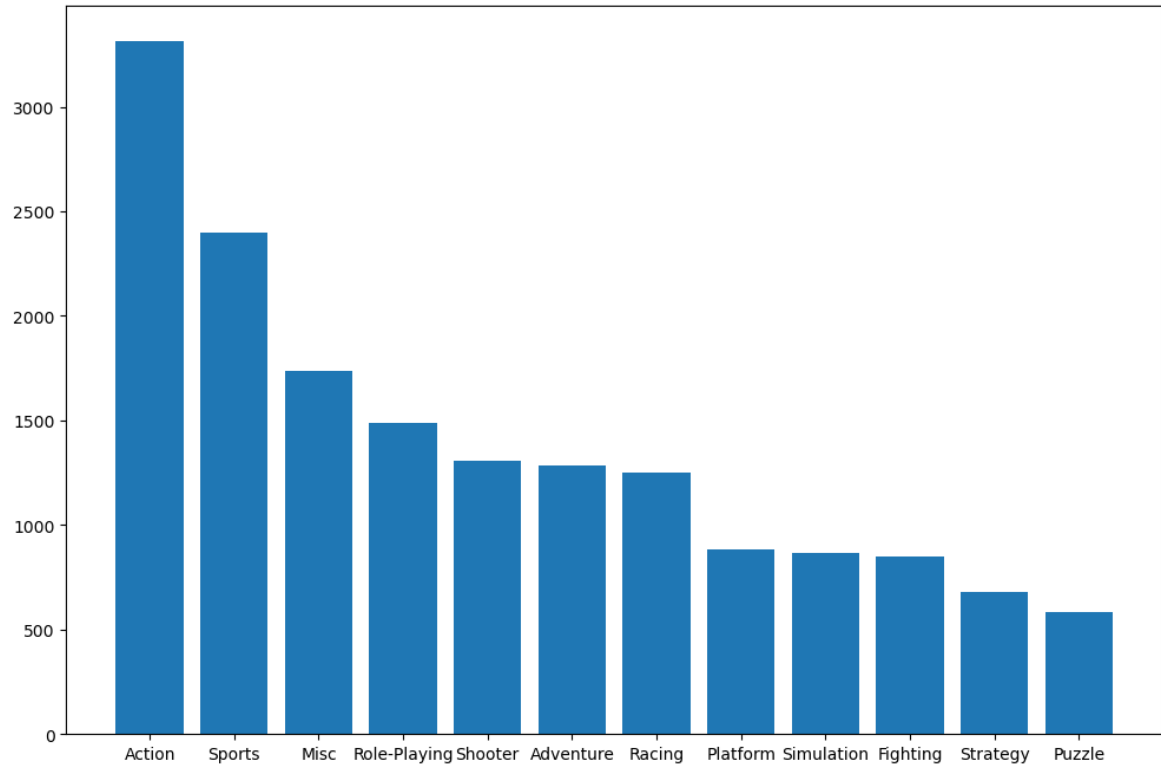
```
In [9]: 1 x=cat_counts.index
        2 y=cat_counts
        3
        4 plt.bar(x,y)
```

```
Out[9]: <BarContainer object of 12 artists>
```

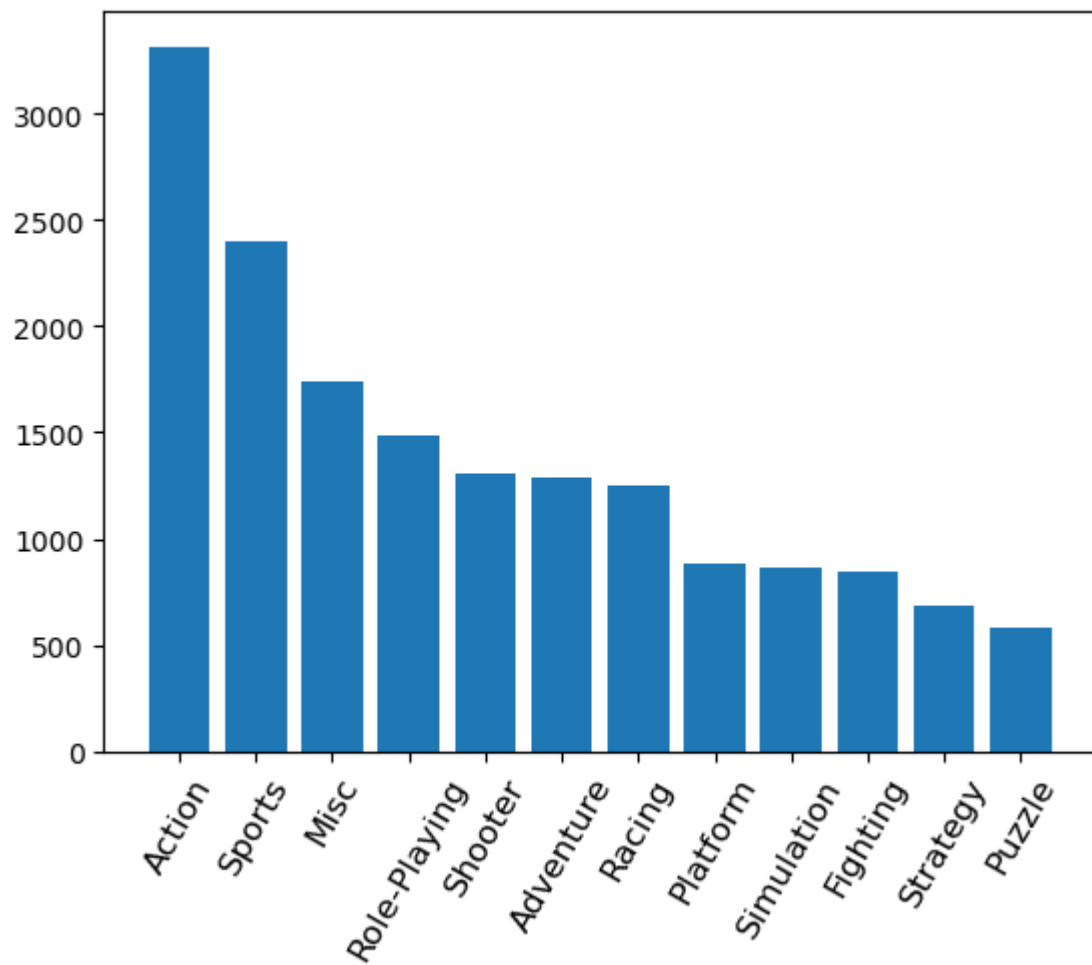


```
In [10]: 1 plt.figure(figsize=(12,8))  
2  
3 x=cat_counts.index  
4 y=cat_counts  
5 plt.bar(x,y)
```

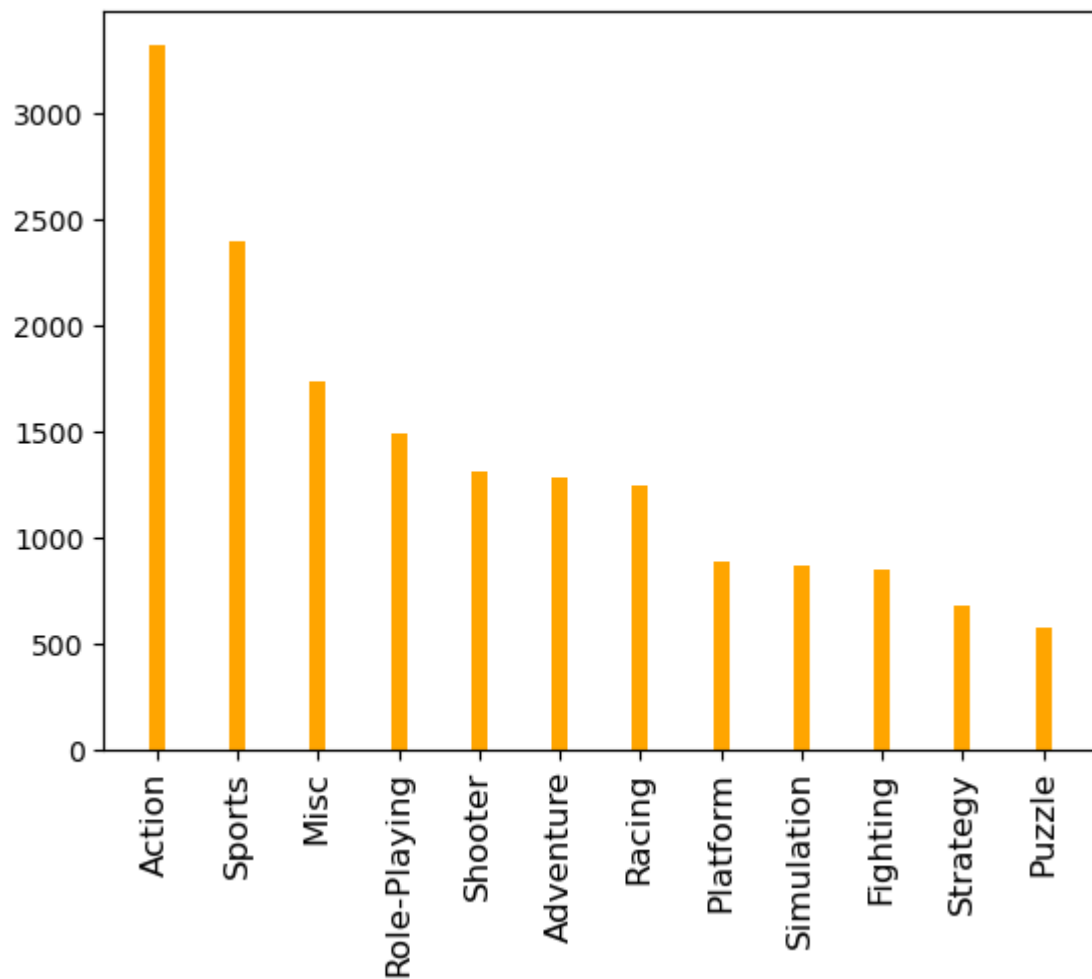
Out[10]: <BarContainer object of 12 artists>



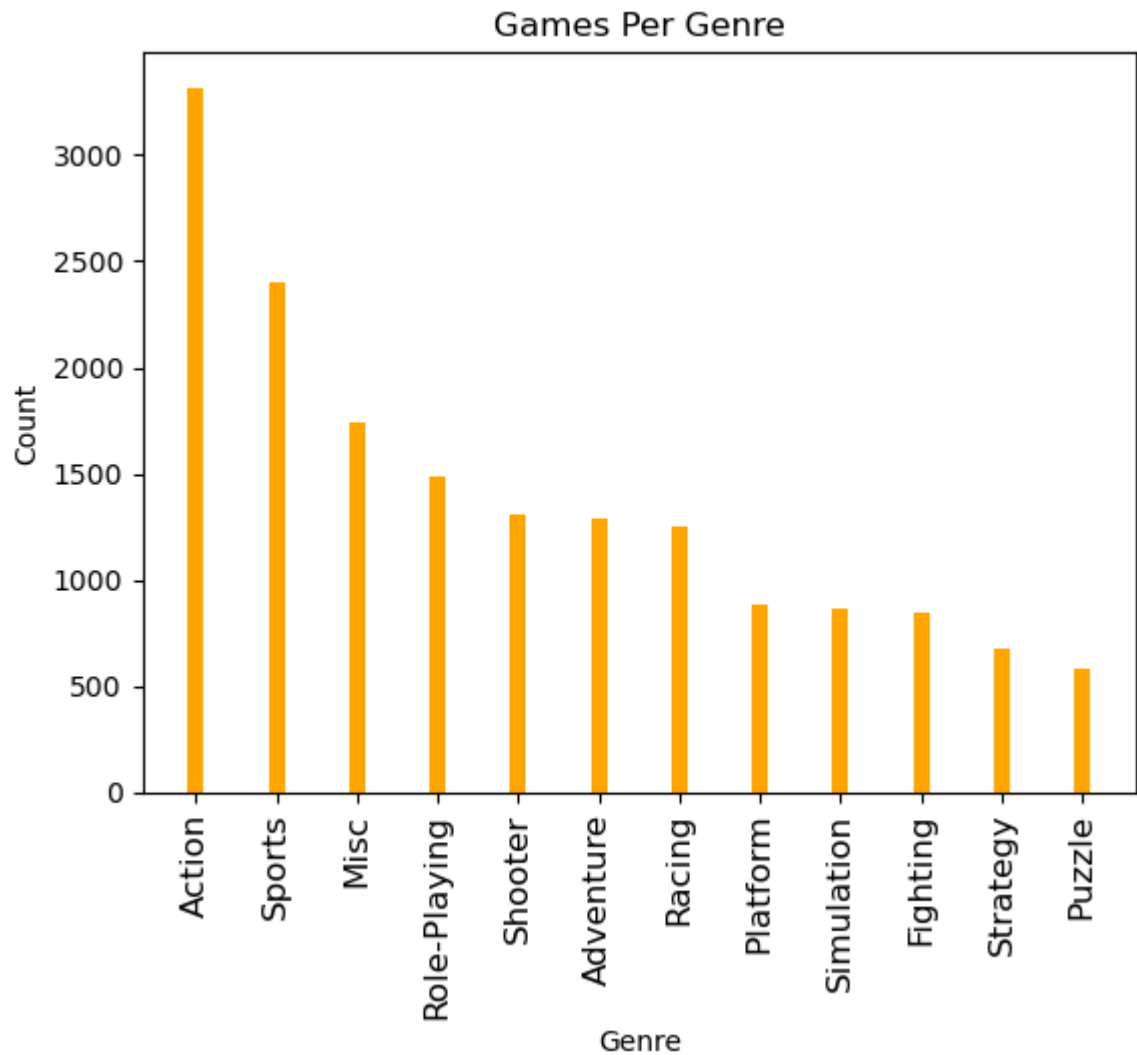
```
In [16]: 1 x=cat_counts.index  
2 y=cat_counts  
3 plt.bar(x,y)  
4 plt.xticks(rotation=60,fontsize=12)  
5 plt.show()
```



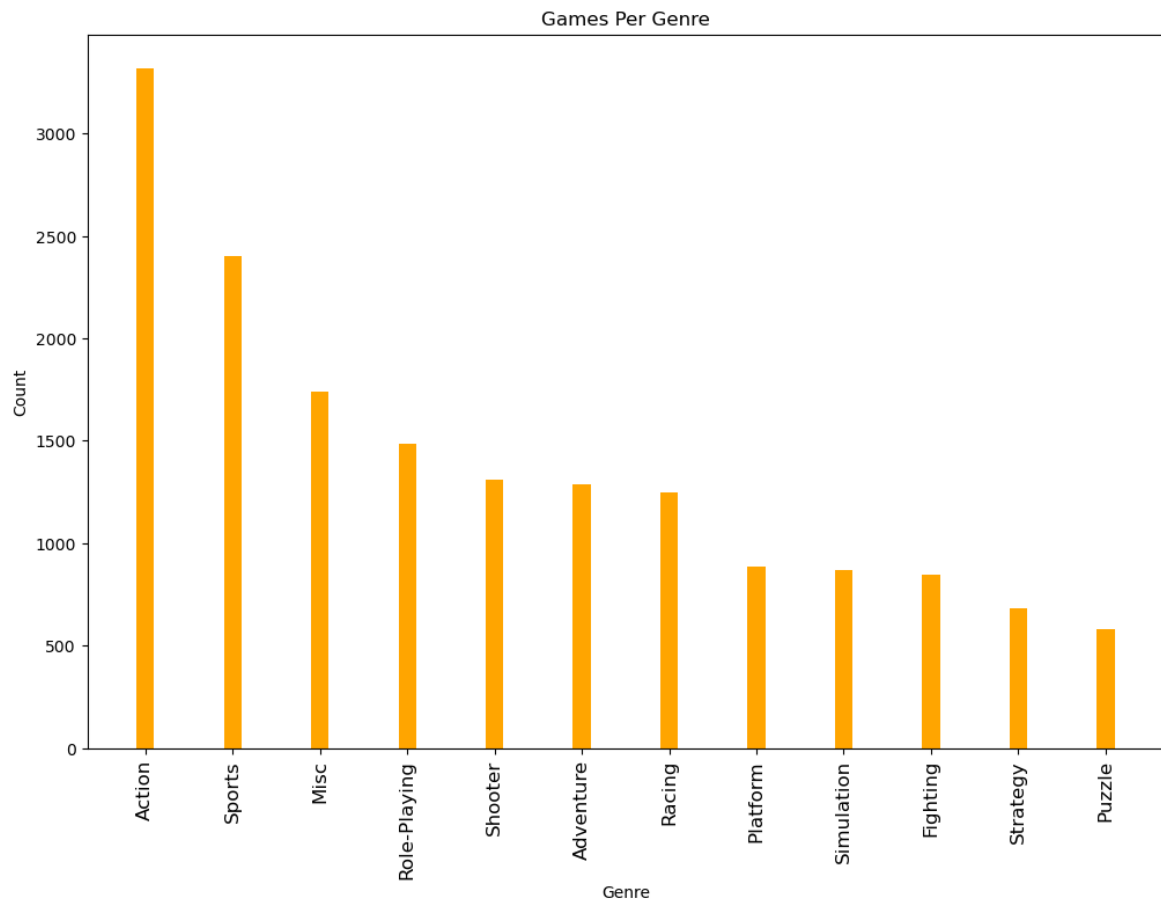
```
In [22]: 1 x=cat_counts.index
          2 y=cat_counts
          3 plt.bar(x,y,width=0.2,color="orange")
          4 plt.xticks(rotation=90,fontsize=12)
          5 plt.show()
```

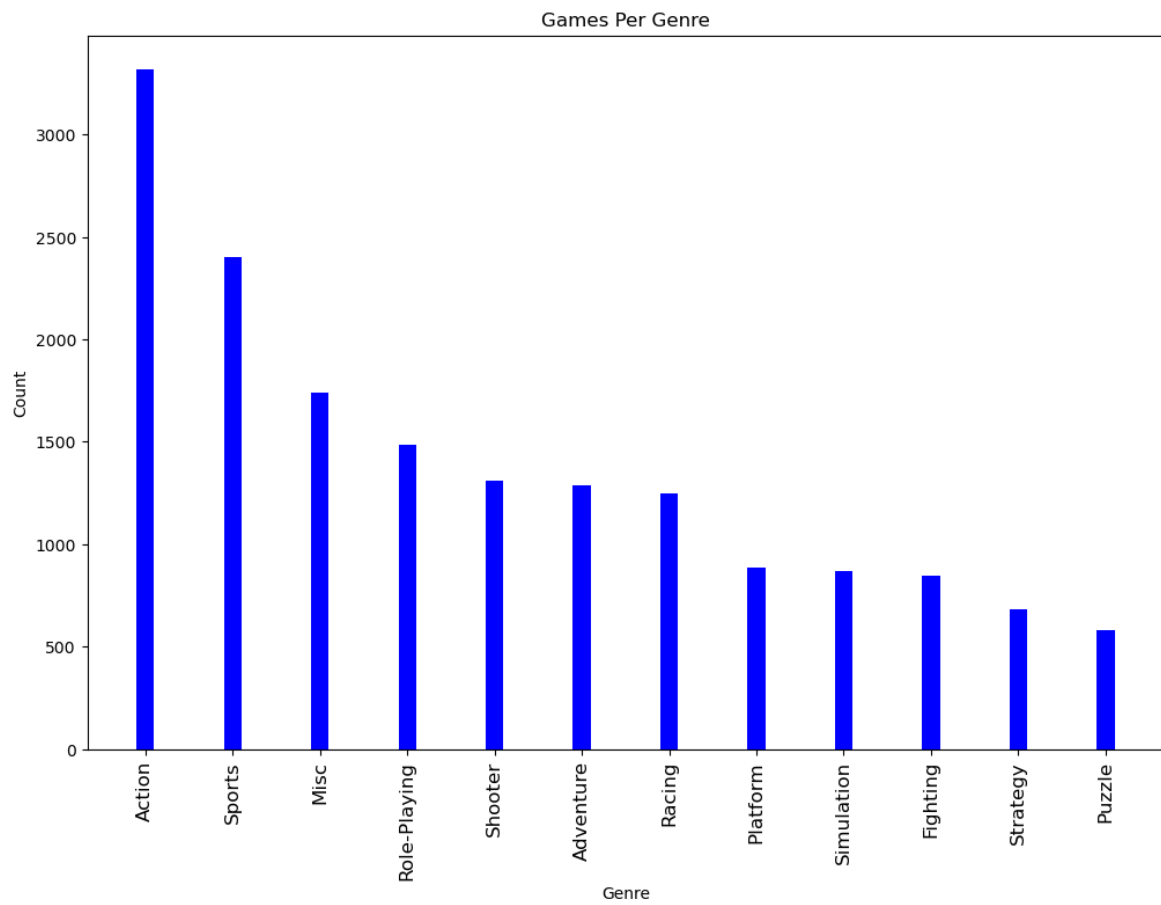


```
In [23]: 1 x=cat_counts.index
2 y=cat_counts
3 plt.bar(x,y,width=0.2,color="orange")
4
5 plt.xlabel("Genre")
6 plt.ylabel("Count")
7 plt.title("Games Per Genre")
8
9 plt.xticks(rotation=90,fontsize=12)
10 plt.show()
```



```
In [26]: 1 plt.figure(figsize=(12,8))
2 x=cat_counts.index
3 y=cat_counts
4 plt.bar(x,y,width=0.2,color="orange")
5 plt.xlabel("Genre")
6 plt.ylabel("Count")
7 plt.title("Games Per Genre")
8 plt.xticks(rotation=90,fontsize=12)
9 plt.show() #Like a print stmt
10
11 plt.figure(figsize=(12,8))
12 x=cat_counts.index
13 y=cat_counts
14 plt.bar(x,y,width=0.2,color="Blue")
15 plt.xlabel("Genre")
16 plt.ylabel("Count")
17 plt.title("Games Per Genre")
18 plt.xticks(rotation=90,fontsize=12)
19 plt.show()
```



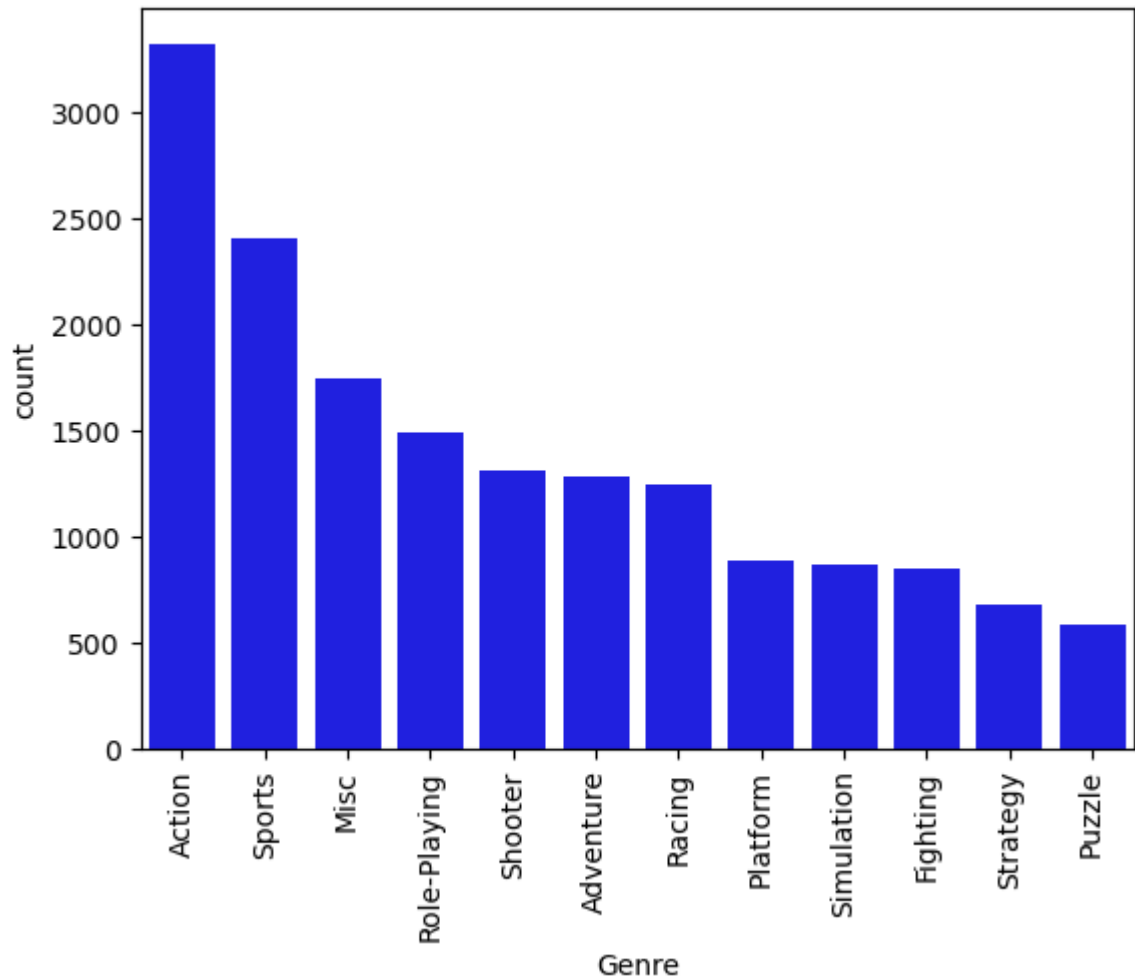


```
In [30]: 1 #AnoLogic
        2 x=3
        3 y=6
        4
        5 print(x)
        6 y
```

3

Out[30]: 6

```
In [31]: 1 sns.countplot(data=data,  
2                 x="Genre",  
3                 order=data["Genre"].value_counts().index,  
4                 color="blue")  
5 plt.xticks(rotation=90)  
6 plt.show()
```



```
In [32]: 1 data["Genre"].value_counts().index
```

```
Out[32]: Index(['Action', 'Sports', 'Misc', 'Role-Playing', 'Shooter', 'Adventure',  
               'Racing', 'Platform', 'Simulation', 'Fighting', 'Strategy', 'Puzzle'],  
              dtype='object')
```

```
In [33]: 1 index_1=np.array(['Action', 'Sports', 'Misc', 'Role-Playing', 'Shooter',  
2                          'Racing', 'Platform', 'Simulation', 'Fighting', 'Strategy', 'Puzzle'])
```

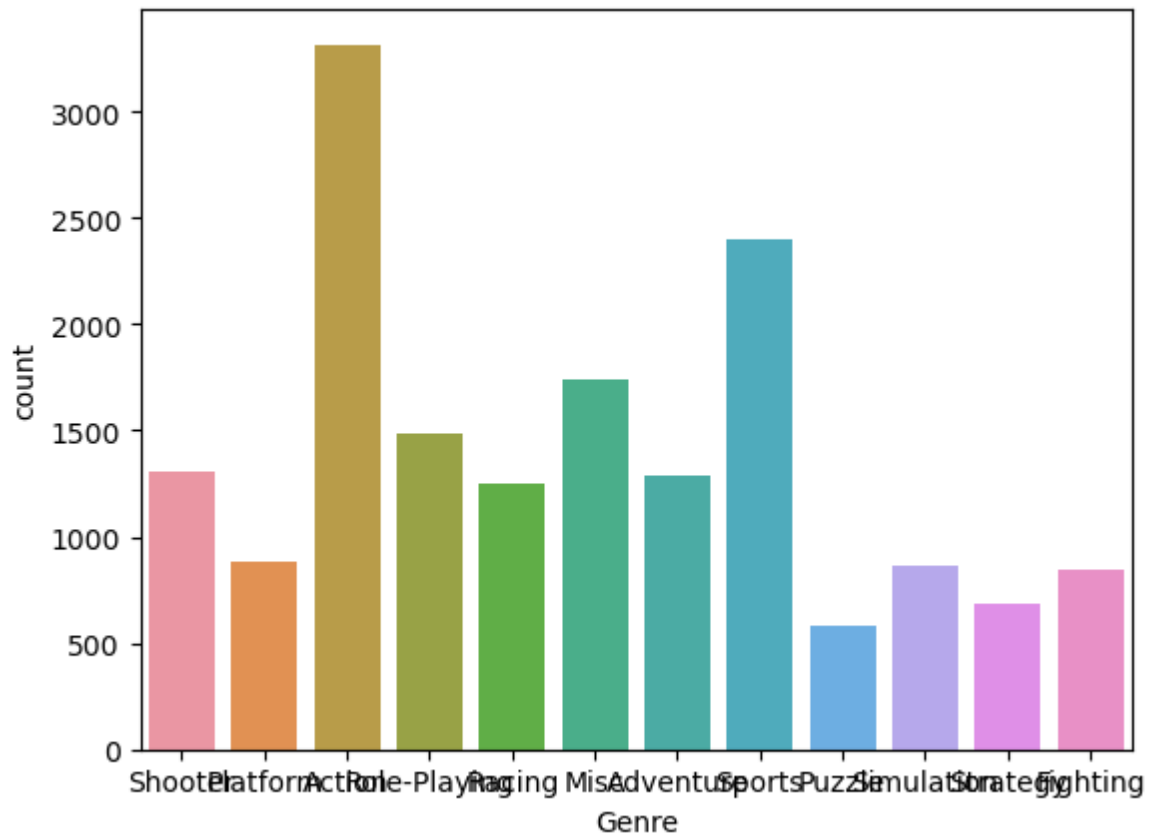
```
In [34]: 1 index_1.sort()
```

```
In [35]: 1 index_1
```

```
Out[35]: array(['Action', 'Adventure', 'Fighting', 'Misc', 'Platform', 'Puzzle',  
              'Racing', 'Role-Playing', 'Shooter', 'Simulation', 'Sports',  
              'Strategy'], dtype='<U12')
```

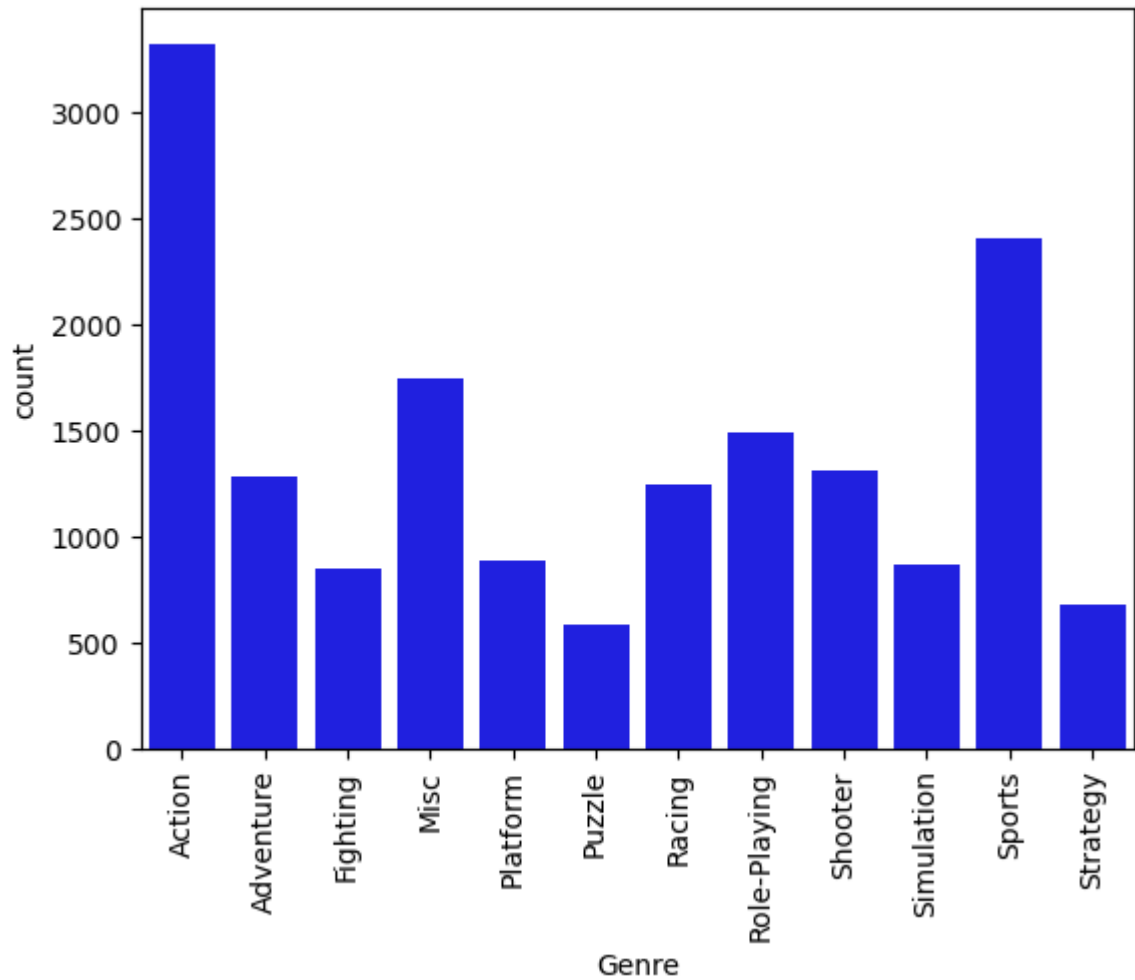
```
In [37]: 1 sns.countplot(data=data,  
2          x="Genre")
```

```
Out[37]: <Axes: xlabel='Genre', ylabel='count'>
```



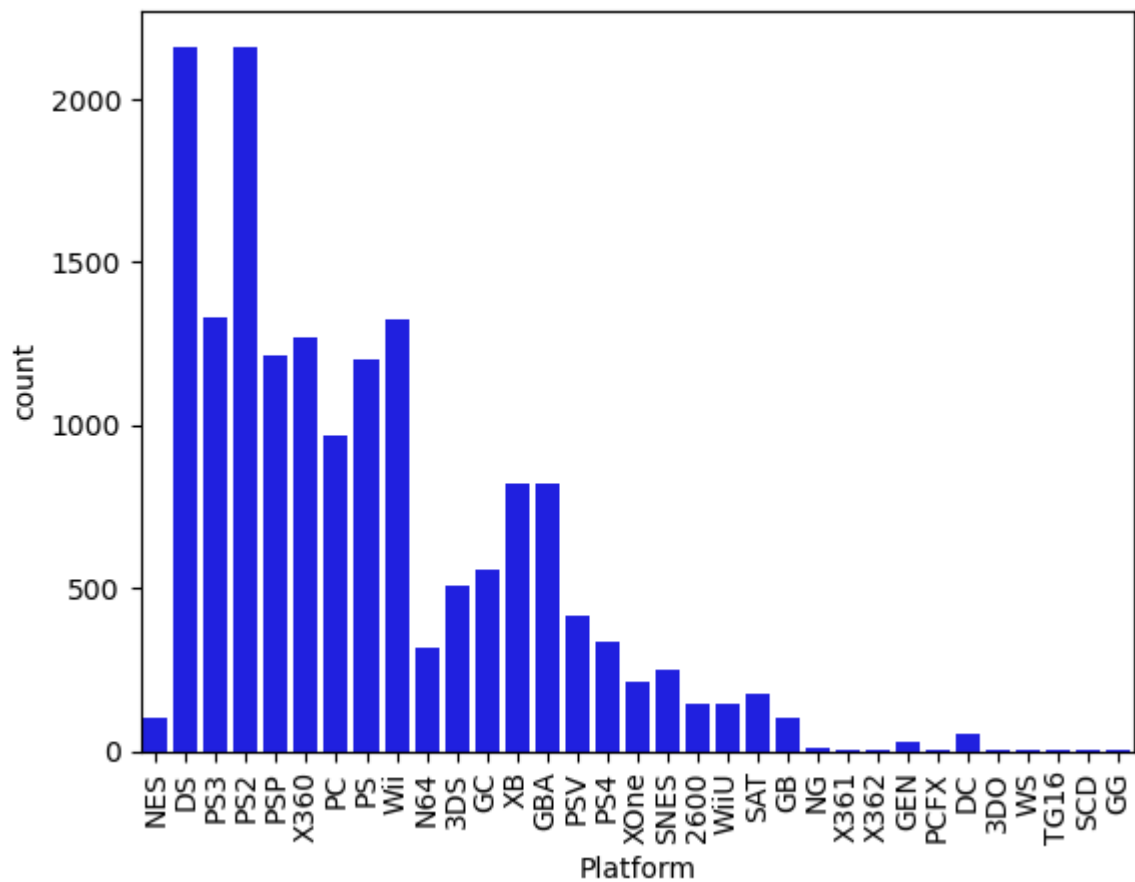
In [39]:

```
1 sns.countplot(data=data,  
2               x="Genre",  
3               order=index_1,  
4               color="blue")  
5 plt.xticks(rotation=90)  
6 plt.show()
```



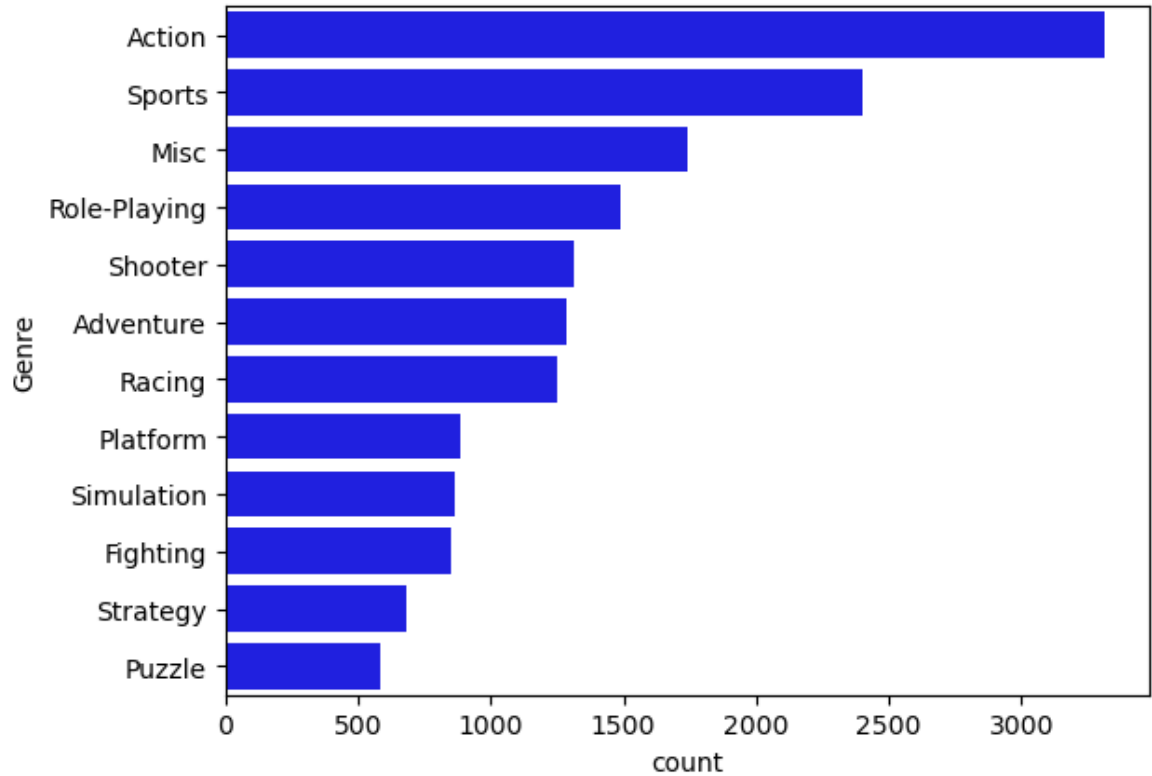
In [40]:

```
1 #Method 1
2
3 sns.countplot(
4     x=data["Platform"],
5     color="blue")
6 plt.xticks(rotation=90)
7 plt.show()
```



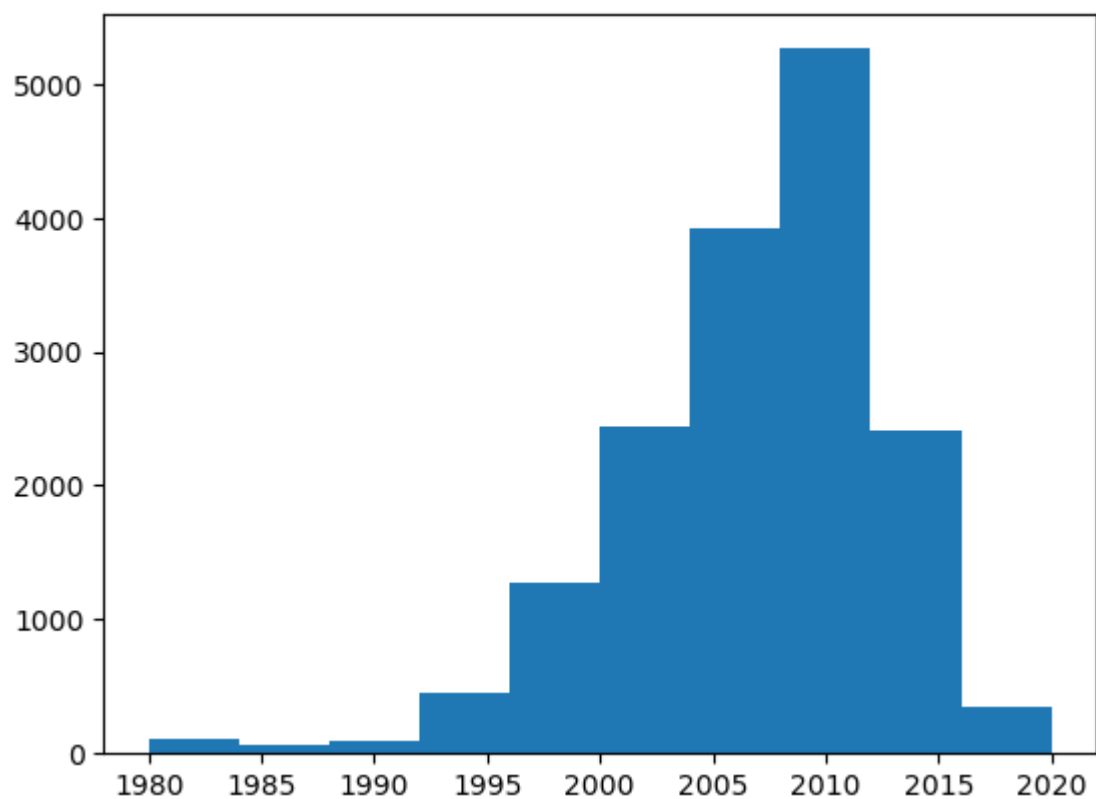
```
In [59]: 1 #Method 2:  
2 sns.countplot(data=data,  
3               y="Genre",  
4               order=data["Genre"].value_counts().index,  
5               color="blue")
```

Out[59]: <Axes: xlabel='count', ylabel='Genre'>

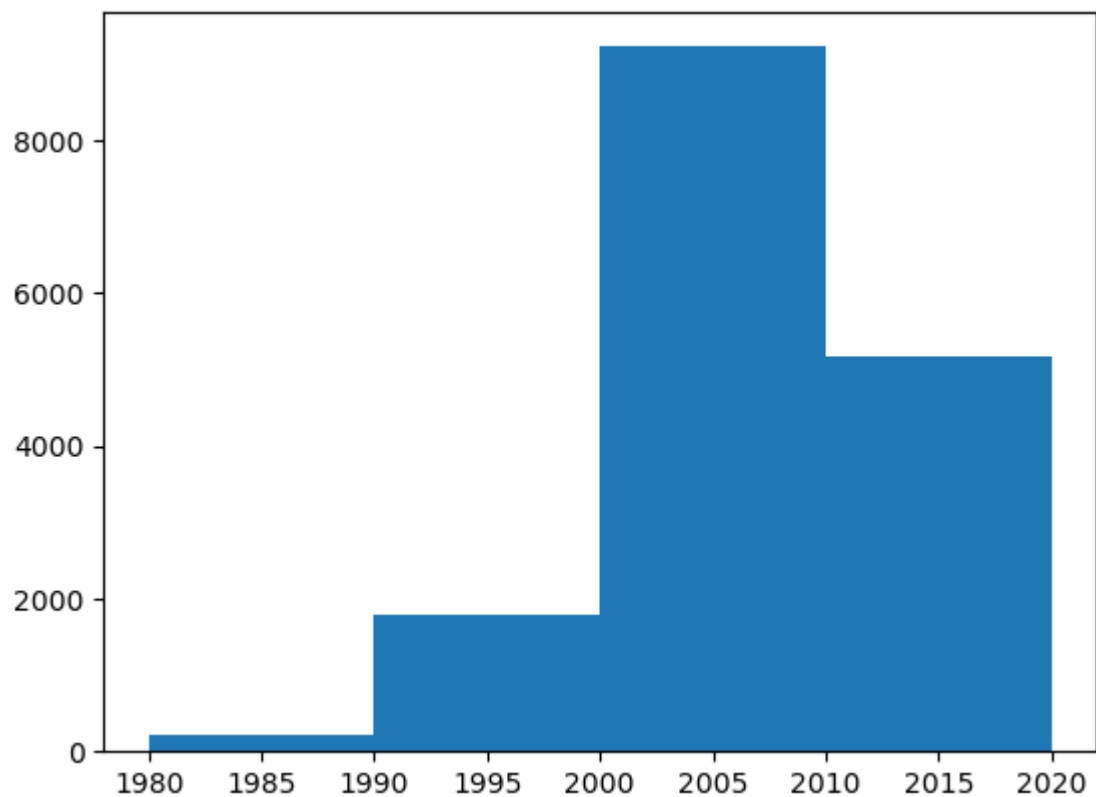


```
In [43]: 1 #Univariate - Numerical
```

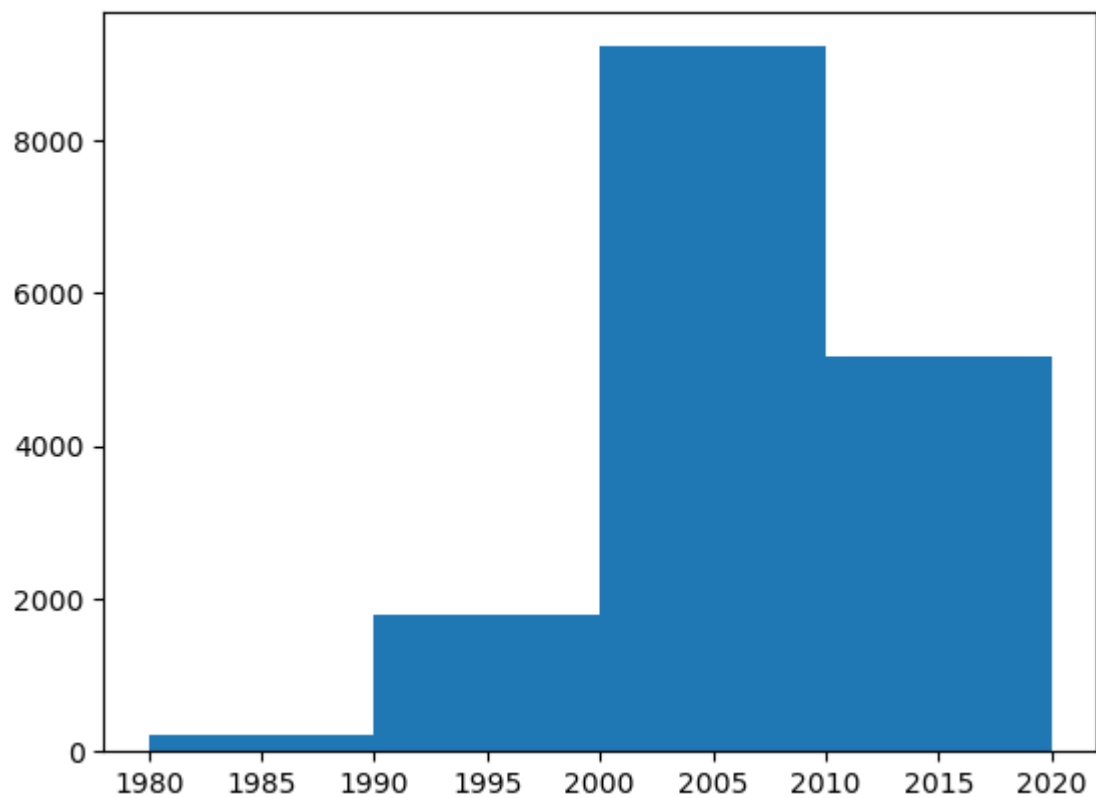
```
In [44]: 1 plt.hist(data["Year"])  
        2 plt.show()
```



```
In [47]: 1 plt.hist(data["Year"],bins=4)  
2 plt.show()
```



```
In [48]: 1 count,bins,patches= plt.hist(data["Year"],bins=4)
```




```
In [49]: 1 count
```

```
Out[49]: array([ 215., 1782., 9222., 5162.])
```

```
In [50]: 1 bins
```

```
Out[50]: array([1980., 1990., 2000., 2010., 2020.])
```

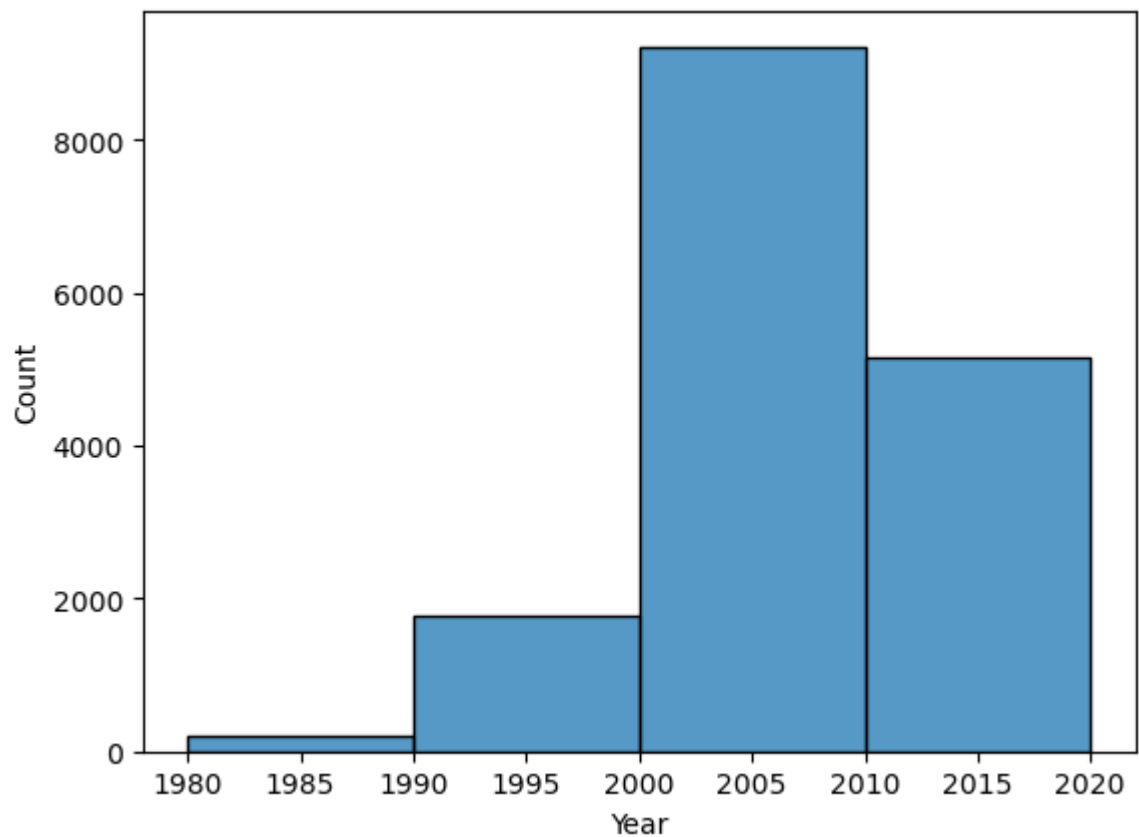
```
In [51]: 1 patches
```

```
Out[51]: <BarContainer object of 4 artists>
```

```
In [ ]: 1 #Seaborn
```

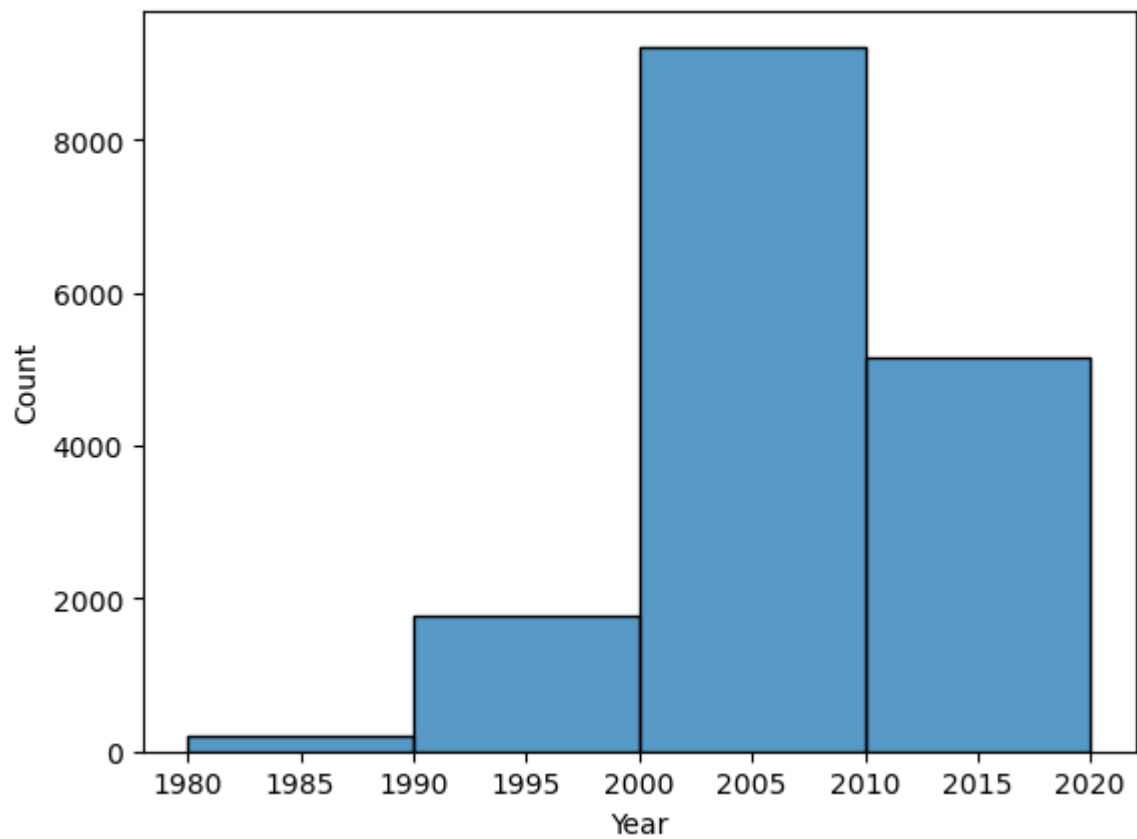
```
In [52]: 1 sns.histplot(data["Year"],bins=4)
```

```
Out[52]: <Axes: xlabel='Year', ylabel='Count'>
```



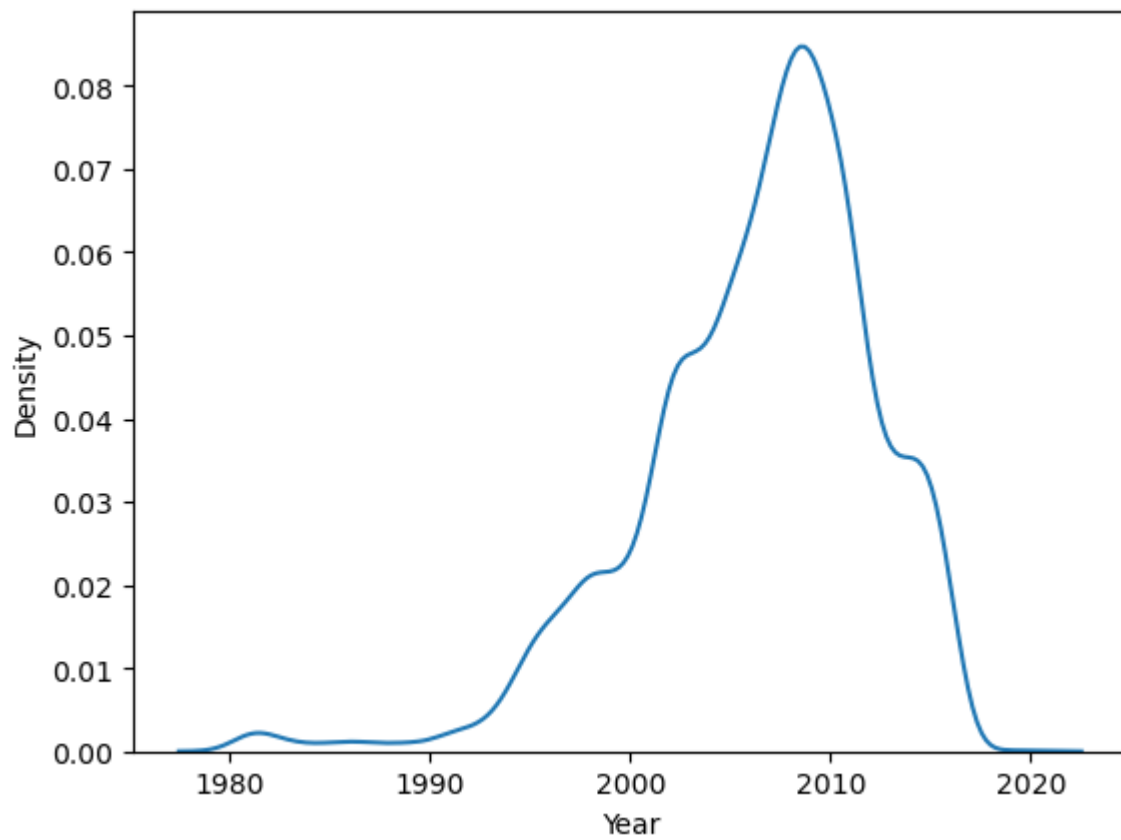
```
In [55]: 1 sns.histplot(data=data,  
2                 x="Year",  
3                 bins=4)
```

```
Out[55]: <Axes: xlabel='Year', ylabel='Count'>
```



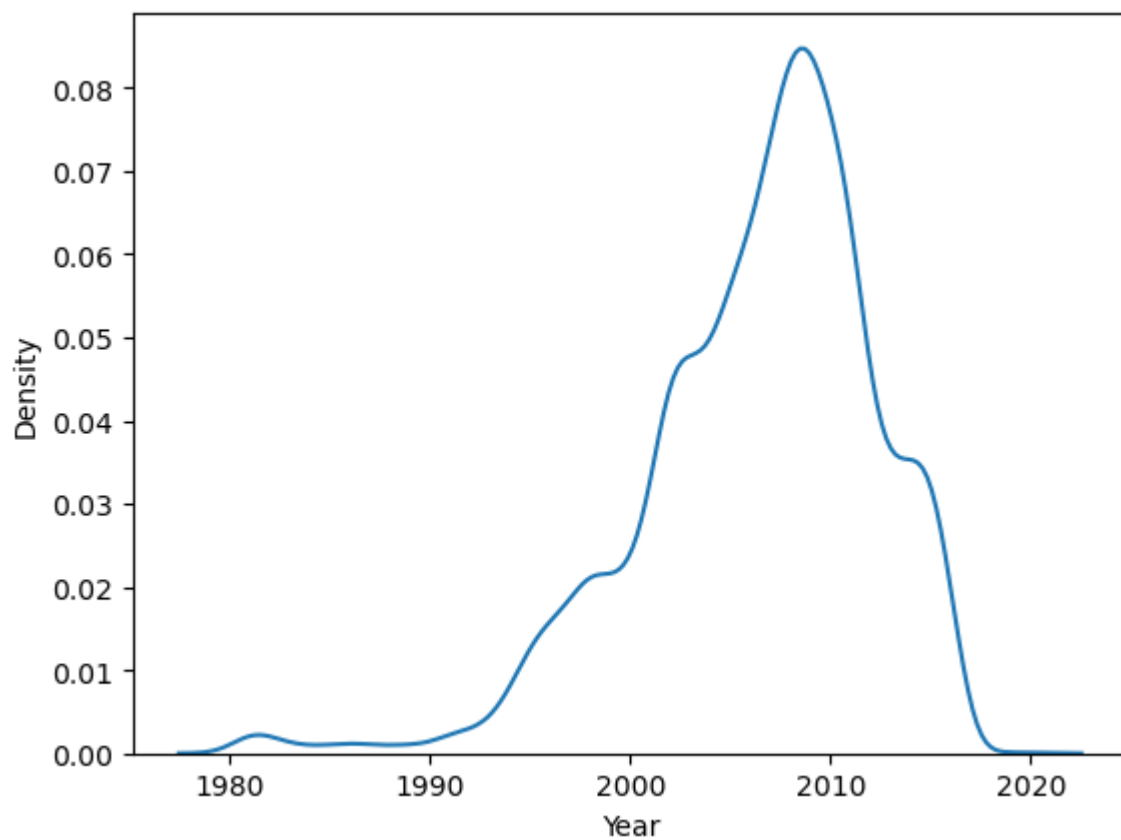
```
In [56]: 1 sns.kdeplot(data=data,  
2             x="Year")
```

```
Out[56]: <Axes: xlabel='Year', ylabel='Density'>
```



```
In [57]: 1 sns.kdeplot(x=data["Year"])
```

```
Out[57]: <Axes: xlabel='Year', ylabel='Density'>
```



```
In [ ]:
```

```
1
```