```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set(style="darkgrid")
```

✓ 1.4s Python

```python
pd.__version__
```

✓ 0.0s Python

```
'1.5.3'
```

```python
data = pd.read_csv('vgsales.csv')
```

✓ 0.0s Python

```python
data.head(10)
```

Python

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
0	1	Wii Sports	Wii	2006.0	Sports	Nintendo	41.49	29.02	3.77	8.46	82.74
1	2	Super Mario Bros.	NES	1985.0	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
2	3	Mario Kart Wii	Wii	2008.0	Racing	Nintendo	15.85	12.88	3.79	3.31	35.82
3	4	Wii Sports Resort	Wii	2009.0	Sports	Nintendo	15.75	11.01	3.28	2.96	33.00
4	5	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31.37
5	6	Tetris	GB	1989.0	Puzzle	Nintendo	23.20	2.26	4.22	0.58	30.26
6	7	New Super Mario Bros.	DS	2006.0	Platform	Nintendo	11.38	9.23	6.50	2.90	30.01
7	8	Wii Play	Wii	2006.0	Misc	Nintendo	14.03	9.20	2.93	2.85	29.02
8	9	New Super Mario Bros. Wii	Wii	2009.0	Platform	Nintendo	14.59	7.06	4.70	2.26	28.62
9	10	Duck Hunt	NES	1984.0	Shooter	Nintendo	26.93	0.63	0.28	0.47	28.31

```python
data.sample(10).T
```

Python Python

	9917	6938	4062	14170	10233	7223	15214	4513	12530	694
Rank	9919	6940	4064	14172	10235	7225	15217	4515	12532	696
Name	USA Today Crossword Challenge	Sesame Street: Elmo's Number Journey	Sniper: Art of Victory	Red Faction II	GunGriffon Blaze	ECW Hardcore Revolution	Moujuutsukai to Oujisama Portable	Minecraft	Power Pro Success Legends	Tomb Raider II
Platform	DS	PS	PC	GC	PS2	N64	PSP	WiiU	PSP	PC
Year	2008.0	1998.0	2008.0	2003.0	2000.0	2000.0	2011.0	2016.0	2010.0	1997.0
Genre	Puzzle	Misc	Shooter	Shooter	Simulation	Fighting	Adventure	Misc	Sports	Action
Publisher	Destineer	NewKidCo	City Interactive	THQ	Swing! Entertainment	Acclaim Entertainment	Idea Factory	Microsoft Game Studios	Konami Digital Entertainment	Eidos Interactive
NA_Sales	0.11	0.13	0.0	0.03	0.05	0.18	0.0	0.18	0.0	0.91
EU_Sales	0.0	0.09	0.45	0.01	0.04	0.04	0.0	0.09	0.0	1.25
JP_Sales	0.0	0.0	0.0	0.0	0.0	0.0	0.02	0.14	0.06	0.0
Other_Sales	0.01	0.02	0.03	0.0	0.01	0.0	0.0	0.03	0.0	0.13
Global_Sales	0.12	0.24	0.49	0.03	0.11	0.22	0.02	0.43	0.06	2.29

```python
data.tail(25)
```

Python

```
data.tail(25)
```

Python

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
16573	16576	Mini Desktop Racing	Wii	2007.0	Racing	Popcorn Arcade	0.01	0.00	0.00	0.0	0.01
16574	16577	Yattaman Wii: BikkuriDokkiri Machine de Mou Ra...	Wii	2008.0	Racing	Takara Tomy	0.00	0.00	0.01	0.0	0.01
16575	16578	Neo Angelique Special	PSP	2008.0	Adventure	Tecmo Koei	0.00	0.00	0.01	0.0	0.01
16576	16579	Rugby Challenge 3	XOne	2016.0	Sports	Alternative Software	0.00	0.01	0.00	0.0	0.01
16577	16580	Damnation	PC	2009.0	Shooter	Codemasters	0.00	0.01	0.00	0.0	0.01
16578	16581	Outdoors Unleashed: Africa 3D	3DS	2011.0	Sports	Mastiff	0.01	0.00	0.00	0.0	0.01
16579	16582	PGA European Tour	N64	2000.0	Sports	Infogrames	0.01	0.00	0.00	0.0	0.01
16580	16583	Real Rode	PS2	2008.0	Adventure	Kadokawa Shoten	0.00	0.00	0.01	0.0	0.01
16581	16584	Fit & Fun	Wii	2011.0	Sports	Unknown	0.00	0.01	0.00	0.0	0.01
16582	16585	Planet Monsters	GBA	2001.0	Action	Titus	0.01	0.00	0.00	0.0	0.01
16583	16586	Carmageddon 64	N64	1999.0	Action	Virgin Interactive	0.01	0.00	0.00	0.0	0.01
16584	16587	Bust-A-Move 3000	GC	2003.0	Puzzle	Ubisoft	0.01	0.00	0.00	0.0	0.01
16585	16588	Breach	PC	2011.0	Shooter	Destineer	0.01	0.00	0.00	0.0	0.01
16586	16589	Secret Files 2: Puritas Cordis	DS	2009.0	Adventure	Deep Silver	0.00	0.01	0.00	0.0	0.01
16587	16590	Mezase!! Tsuri Master DS	DS	2009.0	Sports	Hudson Soft	0.00	0.00	0.01	0.0	0.01
16588	16591	Mega Brain Boost	DS	2008.0	Puzzle	Majesco Entertainment	0.01	0.00	0.00	0.0	0.01
16589	16592	Chou Ezaru wa Akai Hana: Koi wa Tsuki ni Shiru...	PSV	2016.0	Action	dramatic create	0.00	0.00	0.01	0.0	0.01
16590	16593	Eiyuu Densetsu: Sora no Kiseki Material Collec...	PSP	2007.0	Role-Playing	Falcom Corporation	0.00	0.00	0.01	0.0	0.01
16591	16594	Myst IV: Revelation	PC	2004.0	Adventure	Ubisoft	0.01	0.00	0.00	0.0	0.01
16592	16595	Plushees	DS	2008.0	Simulation	Destineer	0.01	0.00	0.00	0.0	0.01
16593	16596	Woody Woodpecker in Crazy Castle 5	GBA	2002.0	Platform	Kemco	0.01	0.00	0.00	0.0	0.01
16594	16597	Men in Black II: Alien Escape	GC	2003.0	Shooter	Infogrames	0.01	0.00	0.00	0.0	0.01
16595	16598	SCORE International Baja 1000: The Official Game	PS2	2008.0	Racing	Activision	0.00	0.00	0.00	0.0	0.01
16596	16599	Know How 2	DS	2010.0	Puzzle	7G//AMES	0.00	0.01	0.00	0.0	0.01
16597	16600	Spirits & Spells	GBA	2003.0	Platform	Wanadoo	0.01	0.00	0.00	0.0	0.01

```python
data.shape
```

Python

```
(16598, 11)
```

```python
data.columns
```

Python

```
Index(['Rank', 'Name', 'Platform', 'Year', 'Genre', 'Publisher', 'NA_Sales',
       'EU_Sales', 'JP_Sales', 'Other_Sales', 'Global_Sales'],
      dtype='object')
```

```python
data.info
```

Python

```
<bound method DataFrame.info of         Rank                                            Name Platform  \
0          1                                      Wii Sports      Wii
1          2                                Super Mario Bros.      NES
2          3                                   Mario Kart Wii      Wii
3          4                                Wii Sports Resort      Wii
4          5                          Pokemon Red/Pokemon Blue       GB
...      ...                                             ...      ...
16593  16596                  Woody Woodpecker in Crazy Castle 5      GBA
16594  16597                       Men in Black II: Alien Escape       GC
16595  16598  SCORE International Baja 1000: The Official Game      PS2
16596  16599                                    Know How 2       DS
16597  16600                                 Spirits & Spells      GBA

         Year         Genre   Publisher  NA_Sales  EU_Sales  JP_Sales  \
0      2006.0        Sports    Nintendo     41.49     29.02      3.77
1      1985.0      Platform    Nintendo     29.08      3.58      6.81
2      2008.0        Racing    Nintendo     15.85     12.88      3.79
3      2009.0        Sports    Nintendo     15.75     11.01      3.28
4      1996.0  Role-Playing    Nintendo     11.27      8.89     10.22
...       ...           ...         ...       ...       ...       ...
16593  2002.0      Platform       Kemco      0.01      0.00      0.00
16594  2003.0       Shooter  Infogrames      0.01      0.00      0.00
16595  2008.0        Racing  Activision      0.00      0.00      0.00
16596  2010.0        Puzzle    7G//AMES      0.00      0.01      0.00
16597  2003.0      Platform     Wanadoo      0.01      0.00      0.00
...
16595         0.00          0.01
16596         0.00          0.01
16597         0.00          0.01
```

```
data.describe
```

Python

```
<bound method NDFrame.describe of        Rank                                      Name Platform  \
0         1                                Wii Sports      Wii
1         2                           Super Mario Bros.      NES
2         3                              Mario Kart Wii      Wii
3         4                           Wii Sports Resort      Wii
4         5                     Pokemon Red/Pokemon Blue       GB
...     ...                                         ...      ...
16593  16596           Woody Woodpecker in Crazy Castle 5      GBA
16594  16597                Men in Black II: Alien Escape       GC
16595  16598  SCORE International Baja 1000: The Official Game      PS2
16596  16599                                 Know How 2       DS
16597  16600                              Spirits & Spells      GBA

         Year         Genre   Publisher  NA_Sales  EU_Sales  JP_Sales  \
0      2006.0        Sports    Nintendo     41.49     29.02      3.77
1      1985.0      Platform    Nintendo     29.08      3.58      6.81
2      2008.0        Racing    Nintendo     15.85     12.88      3.79
3      2009.0        Sports    Nintendo     15.75     11.01      3.28
4      1996.0  Role-Playing    Nintendo     11.27      8.89     10.22
...       ...           ...         ...       ...       ...       ...
16593  2002.0      Platform       Kemco      0.01      0.00      0.00
16594  2003.0       Shooter  Infogrames      0.01      0.00      0.00
16595  2008.0        Racing  Activision      0.00      0.00      0.00
16596  2010.0        Puzzle    7G//AMES      0.00      0.01      0.00
16597  2003.0      Platform     Wanadoo      0.01      0.00      0.00
...
16595        0.00         0.01
16596        0.00         0.01
16597        0.00         0.01

[16598 rows x 11 columns]>
```

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...

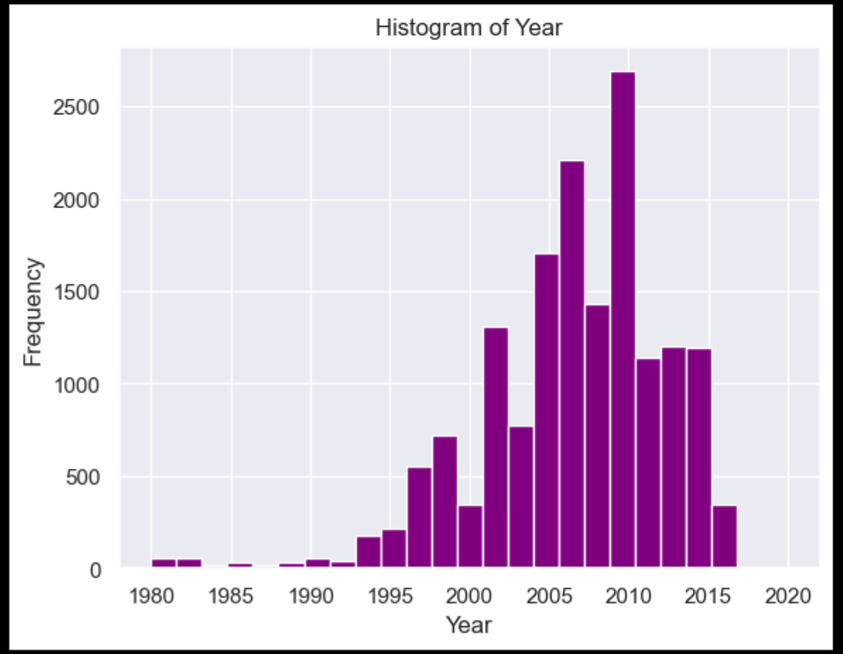
```python
data.describe(include=[object])
```

Python

	Name	Platform	Genre	Publisher
count	16598	16598	16598	16540
unique	11493	31	12	578
top	Need for Speed: Most Wanted	DS	Action	Electronic Arts
freq	12	2163	3316	1351

```python
data.Year.hist(bins=25, color='purple')
plt.title('Histogram of Year')
plt.xlabel('Year')
plt.ylabel('Frequency')
plt.show()
plt.savefig("Histograme.png")
```

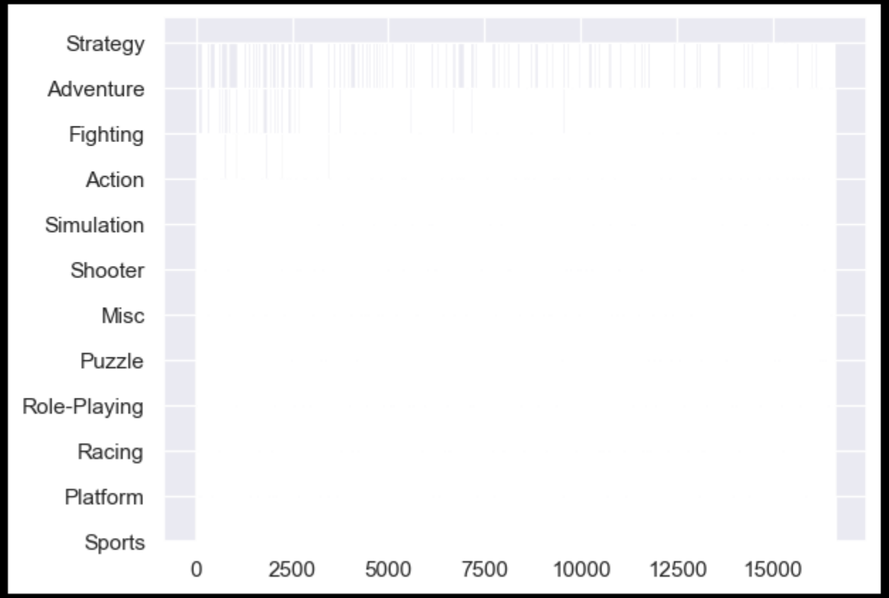
Python



```python
Rank = data['Rank']
Genre = data['Genre']
plt.figure(figsize=(20, 60))
plt.bar(Rank, Genre, color='blue')
plt.title('Video Game Sales by Category')
plt.xlabel('Category')
plt.ylabel('Sales (in millions)')
```

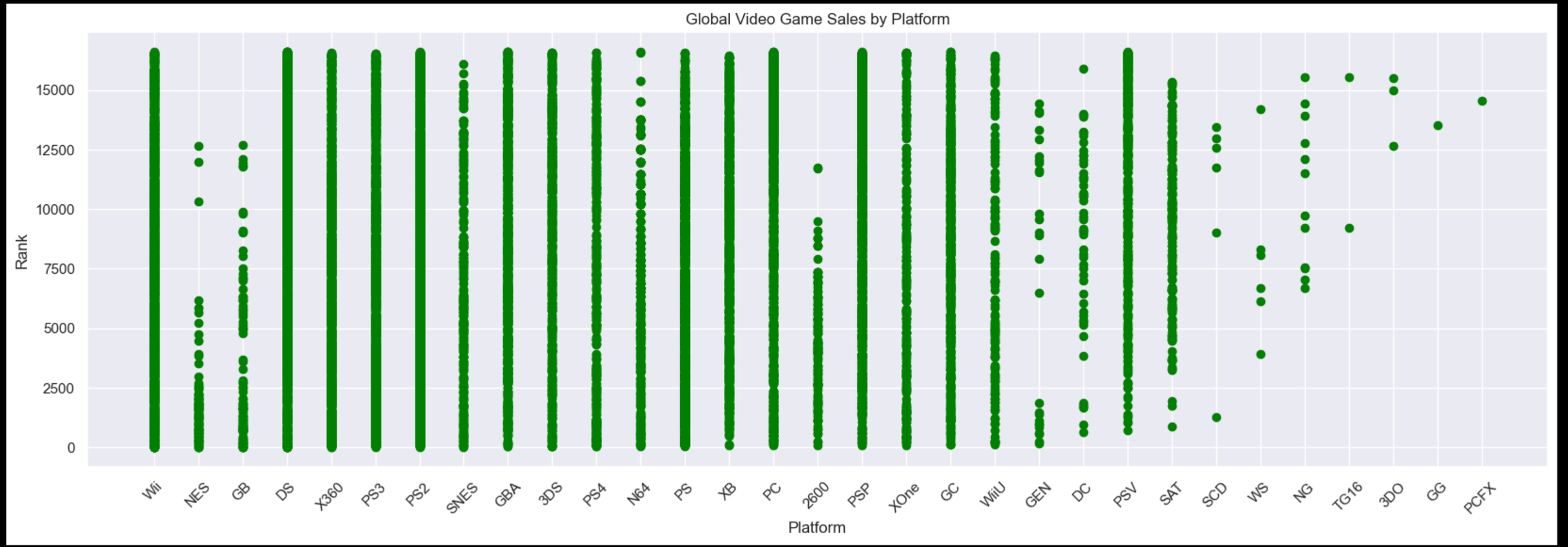
Python

```
<BarContainer object of 16598 artists>
```



```python
Platform = data['Platform']
plt.figure(figsize=(20, 6))
plt.scatter(Platform, Rank,color='Green')
plt.title('Global Video Game Sales by Platform')
plt.xlabel('Platform')
plt.ylabel('Rank')
plt.xticks(rotation=45)
plt.show()
```

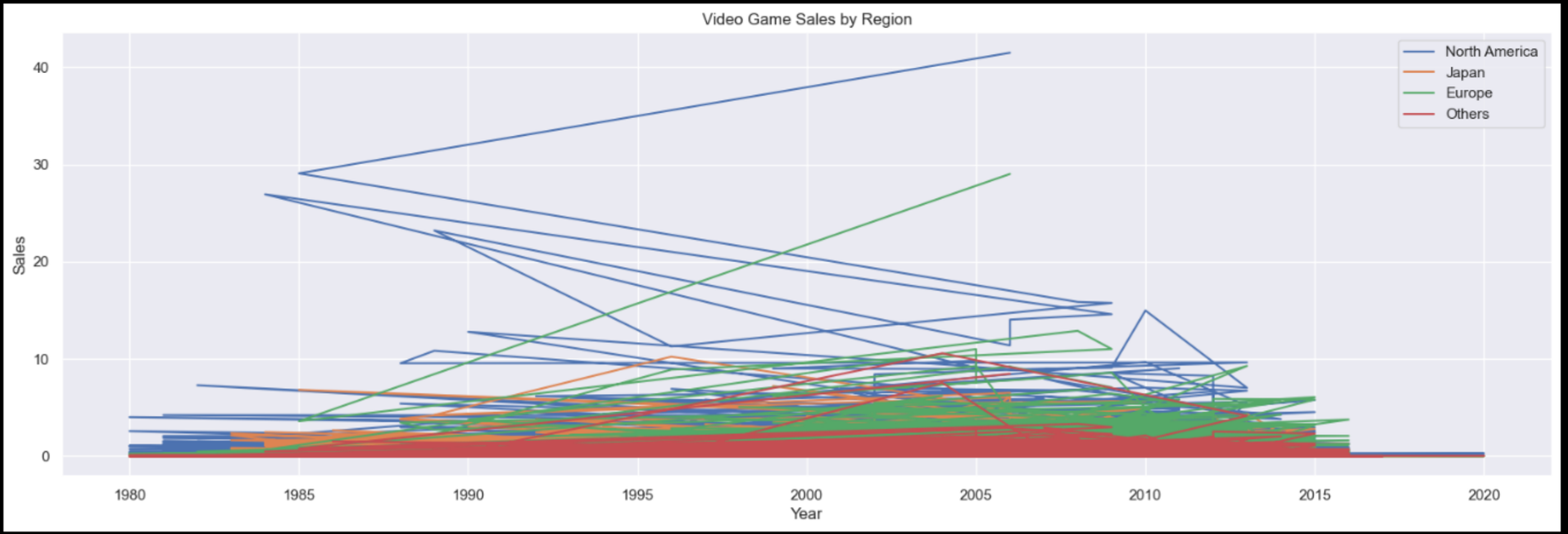
Python




```
<Figure size 640×480 with 0 Axes>
```

```python
NA = data['NA_Sales']
JP = data['JP_Sales']
EU = data['EU_Sales']
Other = data['Other_Sales']

Year = data['Year']

plt.figure(figsize=(20, 6))
plt.plot(Year, NA, label='North America')
plt.plot(Year, JP, label='Japan')
plt.plot(Year, EU, label='Europe')
plt.plot(Year, Other, label='Others')

plt.title('Video Game Sales by Region')
plt.xlabel('Year')
plt.ylabel('Sales')
plt.legend()
plt.savefig("HorzinatalLines.png")
```

Python



```python
data.Publisher.unique()
```

Python

```
array(['Nintendo', 'Microsoft Game Studios', 'Take-Two Interactive',
       'Sony Computer Entertainment', 'Activision', 'Ubisoft',
       'Bethesda Softworks', 'Electronic Arts', 'Sega', 'SquareSoft',
       'Atari', '505 Games', 'Capcom', 'GT Interactive',
       'Konami Digital Entertainment',
       'Sony Computer Entertainment Europe', 'Square Enix', 'LucasArts',
       'Virgin Interactive', 'Warner Bros. Interactive Entertainment',
       'Universal Interactive', 'Eidos Interactive', 'RedOctane',
       'Vivendi Games', 'Enix Corporation', 'Namco Bandai Games',
       'Palcom', 'Hasbro Interactive', 'THQ', 'Fox Interactive',
       'Acclaim Entertainment', 'MTV Games', 'Disney Interactive Studios',
       nan, 'Majesco Entertainment', 'Codemasters', 'Red Orb', 'Level 5',
       'Arena Entertainment', 'Midway Games', 'JVC', 'Deep Silver',
       '989 Studios', 'NCSoft', 'UEP Systems', 'Parker Bros.', 'Maxis',
       'Imagic', 'Tecmo Koei', 'Valve Software', 'ASCII Entertainment',
       'Mindscape', 'Infogrames', 'Unknown', 'Square', 'Valve',
       'Activision Value', 'Banpresto', 'D3Publisher',
       'Oxygen Interactive', 'Red Storm Entertainment', 'Video System',
       'Hello Games', 'Global Star', 'Gotham Games', 'Westwood Studios',
       'GungHo', 'Crave Entertainment', 'Hudson Soft', 'Coleco',
       'Rising Star Games', 'Atlus', 'TDK Mediactive', 'ASC Games',
       'Zoo Games', 'Accolade', 'Sony Online Entertainment', '3DO', 'RTL',
       'Natsume', 'Focus Home Interactive', 'Alchemist',
       'Black Label Games', 'SouthPeak Games', 'Mastertronic', 'Ocean',
       'Zoo Digital Publishing', 'Psygnosis', 'City Interactive',
...
       'EON Digital Entertainment', 'Epic Games', 'Naxat Soft',
       'Ascaron Entertainment', 'Piacci', 'Nitroplus',
       'Paradox Development', 'Otomate', 'Ongakukan', 'Commseed',
       'Inti Creates', 'Takuyo', 'Interchannel-Holon', 'Rain Games',
       'UIG Entertainment'], dtype=object)
```

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...

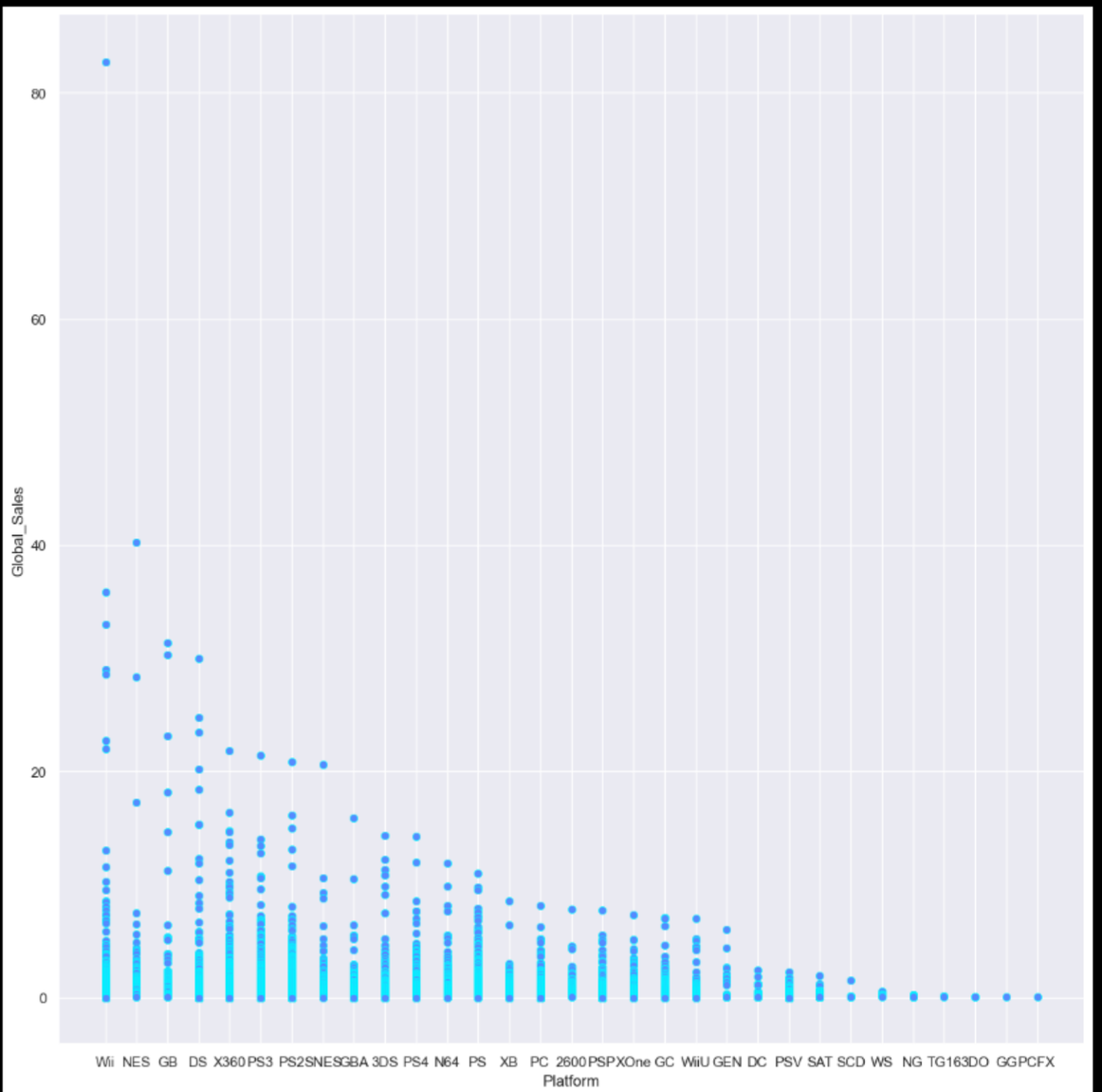
```python
NA.sort_values().head(10)
```

Python

```
11609    0.0
12304    0.0
13775    0.0
10605    0.0
7045     0.0
13772    0.0
13771    0.0
13770    0.0
13769    0.0
13768    0.0
Name: NA_Sales, dtype: float64
```

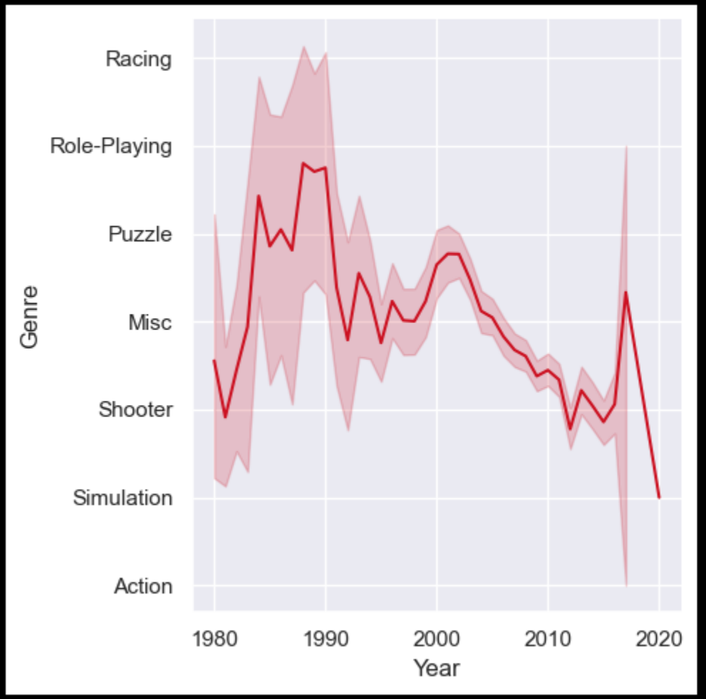
```python
sns.relplot(
    x="Platform" ,
    y="Global_Sales" ,
    data=data ,
    height=12 ,
    color='#568AFF',
    palette="CMRmap_r",
    edgecolor='cyan',

    )
plt.savefig("snsDots.png")
```

Python


Global_Sales
80
60
40
20
0
Wii NES GB DS X360 PS3 PS2 SNES GBA 3DS PS4 N64 PS XB PC 2600 PSP XOne GC WiiU GEN DC PSV SAT SCD WS NG TG16 3DO GG PCFX
Platform

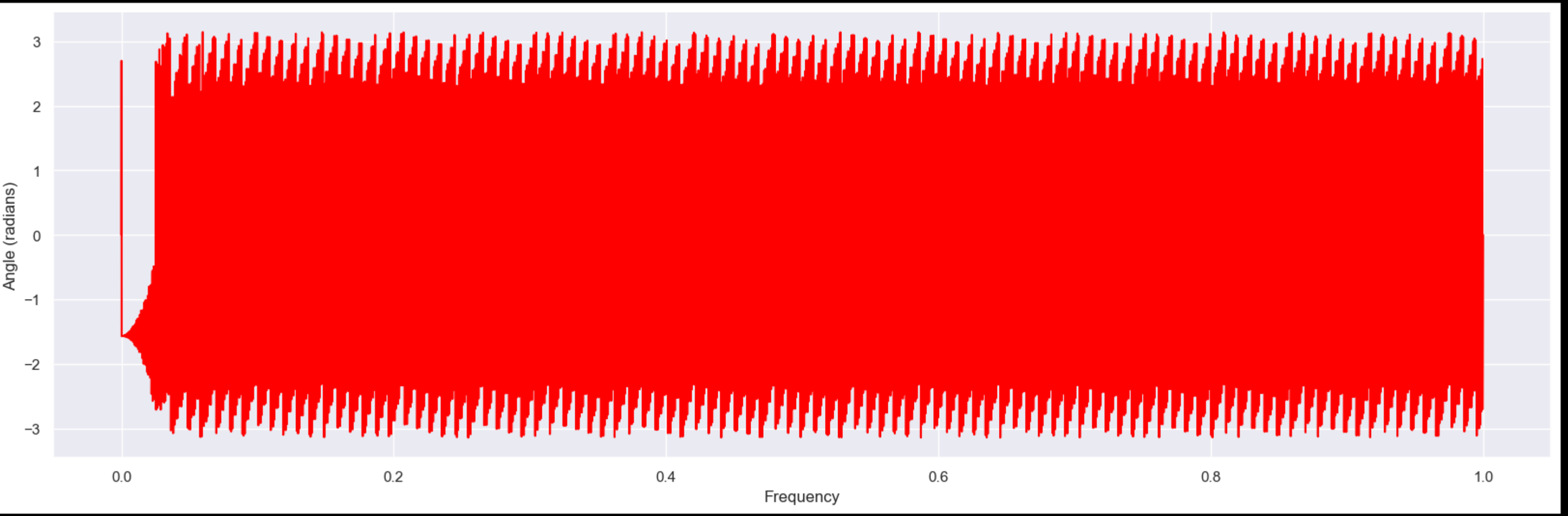
```python
sns.relplot(x="Year",y="Genre",data=data,kind="line",markers=True,color='#CC1222')
plt.savefig("Graph.png")
```

Python



```python
plt.figure(figsize=(20, 6))
plt.angle_spectrum(data=data,x="Rank",color='red' )
plt.savefig("Spectrum.png")
```

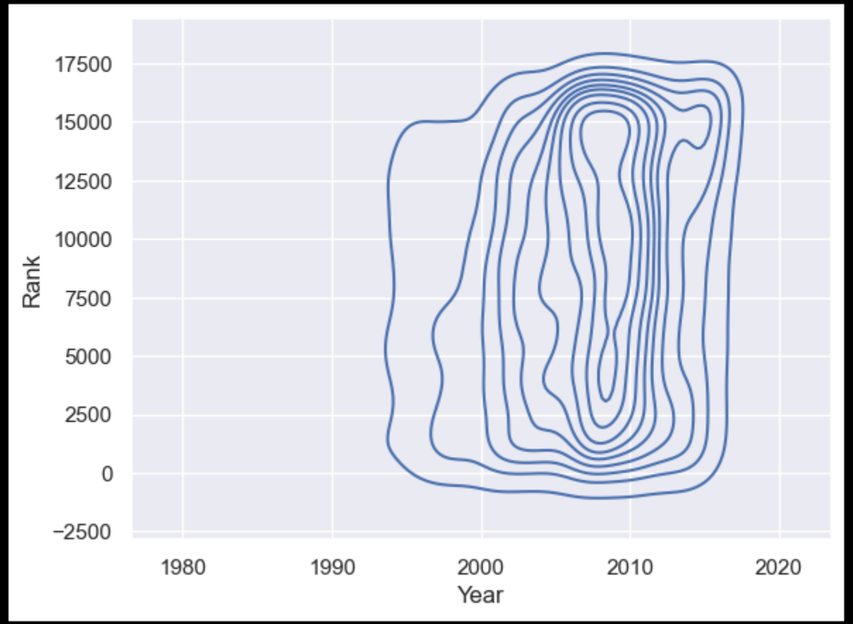
Python



```python
sns.kdeplot(x=data["Year"],y=data["Rank"],height=20)
plt.savefig("kdePot.png")
```

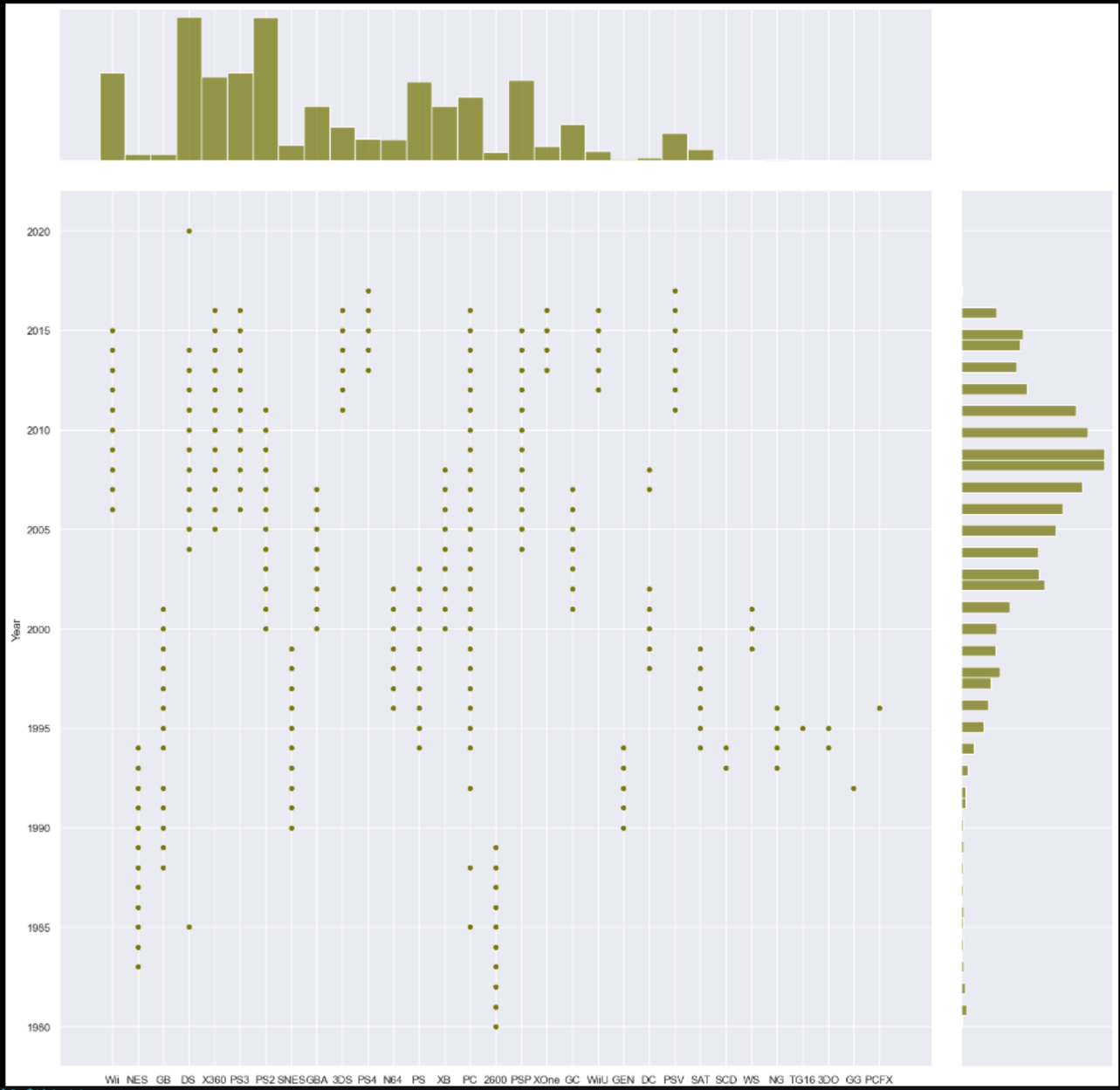
Python

```
c:\Users\main_user_khaled\anaconda3\lib\site-packages\seaborn\distributions.py:1185: UserWarning: The following kwargs were not used by contour: 'height'
  cset = contour_func(
```



```python
sns.jointplot(x="Platform", y="Year", data=data,height=16,color='#777710')
plt.savefig("jointPlot.png")
```

Python



```python
newDATA=data.rename(columns={'NA_Sales':'NORTH_AMERICA_SALES','EU_Sales':'Europe_Sales','JP_Sales':'Japan_Sales'})
```

Python

```python
newDATA
```

Python

	Rank	Name	Platform	Year	Genre	Publisher	NORTH_AMERICA_SALES	Europe_Sales	Japan_Sales	Other_Sales	Global_Sales
0	1	Wii Sports	Wii	2006.0	Sports	Nintendo	41.49	29.02	3.77	8.46	82.74
1	2	Super Mario Bros.	NES	1985.0	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
2	3	Mario Kart Wii	Wii	2008.0	Racing	Nintendo	15.85	12.88	3.79	3.31	35.82
3	4	Wii Sports Resort	Wii	2009.0	Sports	Nintendo	15.75	11.01	3.28	2.96	33.00
4	5	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31.37
...	...	...	...	...	...	...	...	...	...	...	...
16593	16596	Woody Woodpecker in Crazy Castle 5	GBA	2002.0	Platform	Kemco	0.01	0.00	0.00	0.00	0.01
16594	16597	Men in Black II: Alien Escape	GC	2003.0	Shooter	Infogrames	0.01	0.00	0.00	0.00	0.01
16595	16598	SCORE International Baja 1000: The Official Game	PS2	2008.0	Racing	Activision	0.00	0.00	0.00	0.00	0.01
16596	16599	Know How 2	DS	2010.0	Puzzle	7G//AMES	0.00	0.01	0.00	0.00	0.01
16597	16600	Spirits & Spells	GBA	2003.0	Platform	Wanadoo	0.01	0.00	0.00	0.00	0.01

16598 rows × 11 columns

```python
data
```

Python

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
0	1	Wii Sports	Wii	2006.0	Sports	Nintendo	41.49	29.02	3.77	8.46	82.74
1	2	Super Mario Bros.	NES	1985.0	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
2	3	Mario Kart Wii	Wii	2008.0	Racing	Nintendo	15.85	12.88	3.79	3.31	35.82
3	4	Wii Sports Resort	Wii	2009.0	Sports	Nintendo	15.75	11.01	3.28	2.96	33.00
4	5	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31.37
...	...	...	...	...	...	...	...	...	...	...	...
16593	16596	Woody Woodpecker in Crazy Castle 5	GBA	2002.0	Platform	Kemco	0.01	0.00	0.00	0.00	0.01
16594	16597	Men in Black II: Alien Escape	GC	2003.0	Shooter	Infogrames	0.01	0.00	0.00	0.00	0.01
16595	16598	SCORE International Baja 1000: The Official Game	PS2	2008.0	Racing	Activision	0.00	0.00	0.00	0.00	0.01
16596	16599	Know How 2	DS	2010.0	Puzzle	7G//AMES	0.00	0.01	0.00	0.00	0.01
16597	16600	Spirits & Spells	GBA	2003.0	Platform	Wanadoo	0.01	0.00	0.00	0.00	0.01

16598 rows × 11 columns

```python
row=dict({'Rank':16601,'Name':'GTA 6','Platform':'Pc','Genre':'Role-Playing','Publisher':'Rockstar Games','NORTH_AMERICA_SALES':'50','Global_Sales':'100.00'})
newDATA=newDATA.append(row,ignore_index=True)
```

Python

```
C:\Users\main user khaled\AppData\Local\Temp\ipykernel_19892\1112120226.py:2: FutureWarning: The frame.append method is deprecated and will be removed from pandas in a future version. Use pandas.concat instead.
  newDATA=newDATA.append(row,ignore_index=True)
```

```python
newDATA.tail(1)
```

Python

	Rank	Name	Platform	Year	Genre	Publisher	NORTH_AMERICA_SALES	Europe_Sales	Japan_Sales	Other_Sales	Global_Sales	NA_Sales
16599	16601	GTA 6	Pc	NaN	Role-Playing	Rockstar Games	50	NaN	NaN	NaN	100.00	NaN

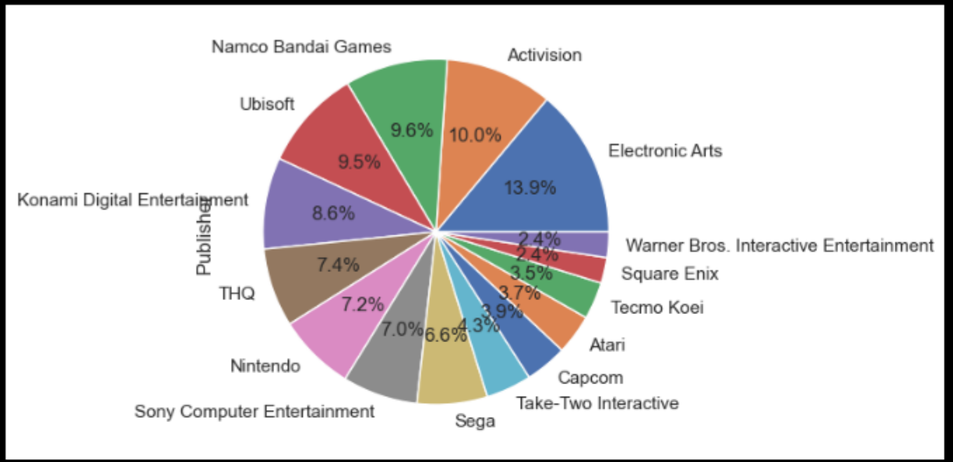
```python
newDATA=newDATA.sort_values(by=['Europe_Sales'],ascending=True)
newDATA.head(10)
```

Python

	Rank	Name	Platform	Year	Genre	Publisher	NORTH_AMERICA_SALES	Europe_Sales	Japan_Sales	Other_Sales	Global_Sales	NA_Sales
1708	1710	Pro Yakyuu Family Stadium '88	NES	1988.0	Sports	Namco Bandai Games	0.1	0.0	1.08	0.0	1.18	NaN
7917	7919	Taiko no Tatsujin: Wii U Version!	WiiU	2013.0	Misc	Namco Bandai Games	0.0	0.0	0.19	0.0	0.19	NaN
7924	7926	Pro Yaky? Spirits 4	PS2	2007.0	Sports	Konami Digital Entertainment	0.0	0.0	0.19	0.0	0.19	NaN
7927	7929	Daigasso! Band Brothers	DS	2004.0	Misc	Nintendo	0.0	0.0	0.19	0.0	0.19	NaN
7824	7826	Tokimeki Memorial Girl's Side 3rd Story	DS	2010.0	Adventure	Konami Digital Entertainment	0.0	0.0	0.19	0.0	0.19	NaN
7929	7931	Thoroughbred Breeder	SNES	1993.0	Simulation	Hect	0.0	0.0	0.19	0.0	0.19	NaN
7823	7825	Jikkyou Powerful Pro Yakyuu '96 Kaimakuban	SNES	1996.0	Sports	Konami Digital Entertainment	0.0	0.0	0.19	0.0	0.19	NaN
7914	7916	3rd Super Robot Wars Z Jigoku Hen	PSV	2014.0	Role-Playing	Namco Bandai Games	0.0	0.0	0.19	0.0	0.19	NaN
7932	7934	Nobunaga no Yabou: Souzou	PS3	2013.0	Strategy	Tecmo Koei	0.0	0.0	0.19	0.0	0.19	NaN
7936	7938	Chibi-Robo! Park Patrol	DS	2007.0	Adventure	Unknown	0.0	0.0	0.19	0.0	0.19	NaN

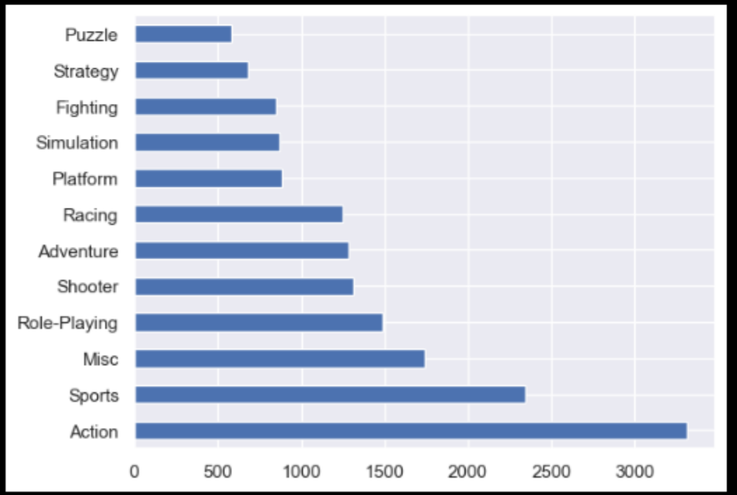
```
data['Publisher'].value_counts().head(15).plot(kind='pie' , autopct='%1.1f%%')
plt.savefig("circle.png")
```

Python



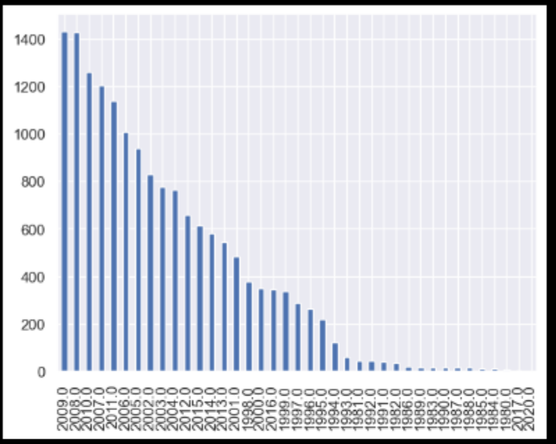

```
data['Genre'].value_counts().head(50).plot(kind='barh')
plt.savefig("barsHorizantel.png")
```

Python



```python
data['Year'].value_counts().head(50).plot(kind='bar',stacked=True)
plt.savefig("VeritcalBars.png")
```

Python




```python
Groupe = data.groupby(['NA_Sales', 'EU_Sales','JP_Sales','Other_Sales','Global_Sales'], as_index=False).mean()
```

Python

```
sers\main user khaled\AppData\Local\Temp\ipykernel_19892\3578637671.py:1: FutureWarning: The default value of numeric_only in DataFrameGroupBy.mean is deprecated. In a future version, numeric_only will default to False. Either specify numeric_only or select only columns which should be valid for the functio
oupe = data.groupby(['NA_Sales', 'EU_Sales','JP_Sales','Other_Sales','Global_Sales'], as_index=False).mean()
```

```python
Groupe
```

Python

	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales	Rank	Year
0	0.00	0.00	0.00	0.00	0.01	16598.000000	2008.000000
1	0.00	0.00	0.00	0.18	0.18	8057.000000	2008.000000
2	0.00	0.00	0.01	0.00	0.01	16282.616613	2010.176849
3	0.00	0.00	0.02	0.00	0.02	15464.473461	2009.704104
4	0.00	0.00	0.02	0.00	0.03	14864.000000	1998.666667
...	...	...	...	...	...	...	...
6769	15.85	12.88	3.79	3.31	35.82	3.000000	2008.000000
6770	23.20	2.26	4.22	0.58	30.26	6.000000	1989.000000
6771	26.93	0.63	0.28	0.47	28.31	10.000000	1984.000000
6772	29.08	3.58	6.81	0.77	40.24	2.000000	1985.000000
6773	41.49	29.02	3.77	8.46	82.74	1.000000	2006.000000

6774 rows × 7 columns

```
sns.lineplot(data = Groupe ,x = 'Year', y='Rank', hue='Rank');
plt.savefig("multiline.png");
```

Python