In [ ]:

```
# Data Set Description:: This dataset contains a list of video games with sales gre
It was generated by a scrape of vgchartz.com.

Fields include

Rank - Ranking of overall sales

Name - The games name

Platform - Platform of the games release (i.e. PC,PS4, etc.)

Year - Year of the game's release

Genre - Genre of the game

Publisher - Publisher of the game

NA_Sales - Sales in North America (in millions)

EU_Sales - Sales in Europe (in millions)

JP_Sales - Sales in Japan (in millions)

Other_Sales - Sales in the rest of the world (in millions)

Global_Sales - Total worldwide sales.
```

In [220…

```
import pandas as pd
import numpy as np
import matplotlib
from decimal import Decimal
```

In [221…

```
df=pd.read_csv(r"C:\Users\USER\Desktop\vgsales.csv")
```

In [222…

```
df.head()
```

Out[222…

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales
0	1	Wii Sports	Wii	2006.0	Sports	Nintendo	41.49	29.02	3.77
1	2	Super Mario Bros.	NES	1985.0	Platform	Nintendo	29.08	3.58	6.81
2	3	Mario Kart Wii	Wii	2008.0	Racing	Nintendo	15.85	12.88	3.79
3	4	Wii Sports Resort	Wii	2009.0	Sports	Nintendo	15.75	11.01	3.28
4	5	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo	11.27	8.89	10.22

In [223…

```
# The Data set needs some cleaning and manipulation, converting the Year column to
# the financial columns to (Numeric)
# it also contained sum null values under the publisher columns which i deleted (57
```

In [224…

```
#Converting Year to Interger

df['Year'] = df['Year'].fillna(-1).round().astype(int)
```

In [225…

```
df.head(1)
```

Out[225…

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sa
0	1	Wii Sports	Wii	2006	Sports	Nintendo	41.49	29.02	3.77	8

In [226…

```
df["NA_Sales"] = pd.to_numeric(df["NA_Sales"], errors='coerce')
```

In [227…

```
df["JP_Sales"] = pd.to_numeric(df["JP_Sales"], errors='coerce')
```

In [228…

```
df["EU_Sales"] = pd.to_numeric(df["EU_Sales"], errors='coerce')
```

In [229…

```
df["Global_Sales"] = pd.to_numeric(df["Global_Sales"], errors='coerce')
```

In [230…

```
df["Other_Sales"] = pd.to_numeric(df["Other_Sales"], errors='coerce')
```

In [ ]:

In [231…

```
df[['NA_Sales', 'EU_Sales', 'JP_Sales', 'Other_Sales', 'Global_Sales']].describe()
```

Out[231…

	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
count	16598.000000	16598.000000	16598.000000	16598.000000	16598.000000
mean	0.264667	0.146652	0.077782	0.048063	0.537441
std	0.816683	0.505351	0.309291	0.188588	1.555028
min	0.000000	0.000000	0.000000	0.000000	0.010000
25%	0.000000	0.000000	0.000000	0.000000	0.060000
50%	0.080000	0.020000	0.000000	0.010000	0.170000
75%	0.240000	0.110000	0.040000	0.040000	0.470000
max	41.490000	29.020000	10.220000	10.570000	82.740000

In [232…

```
df.head(5)
```

Out[232…

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales
0	1	Wii Sports	Wii	2006	Sports	Nintendo	41.49	29.02	3.77
1	2	Super Mario Bros.	NES	1985	Platform	Nintendo	29.08	3.58	6.81
2	3	Mario Kart Wii	Wii	2008	Racing	Nintendo	15.85	12.88	3.79
3	4	Wii Sports Resort	Wii	2009	Sports	Nintendo	15.75	11.01	3.28
4	5	Pokemon Red/Pokemon Blue	GB	1996	Role-Playing	Nintendo	11.27	8.89	10.22

In [233…

```
df.dropna(subset=["Publisher"],inplace=True)
```

In [235…

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 16540 entries, 0 to 16597
Data columns (total 11 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Rank          16540 non-null  int64  
 1   Name          16540 non-null  object 
 2   Platform      16540 non-null  object 
 3   Year          16540 non-null  int32  
 4   Genre         16540 non-null  object 
 5   Publisher     16540 non-null  object 
 6   NA_Sales      16540 non-null  float64
 7   EU_Sales      16540 non-null  float64
 8   JP_Sales      16540 non-null  float64
 9   Other_Sales   16540 non-null  float64
 10  Global_Sales  16540 non-null  float64
dtypes: float64(5), int32(1), int64(1), object(4)
memory usage: 1.5+ MB
```

In [357…

```
# Max,Average and Minimum Revenue In North America

df[["NA_Sales"]].agg(["max","mean","min"])
```

Out[357…

	NA_Sales
max	41.490000
mean	0.265079
min	0.000000

In [238…

```
# Top 5 sellers in North America
```

```
df.groupby(["Name"])[["NA_Sales"]].sum().sort_values(by="NA_Sales",ascending=False)
```

Out[238…

Name	NA_Sales
Wii Sports	41.49
Super Mario Bros.	32.48
Duck Hunt	26.93
Tetris	26.17
Grand Theft Auto V	23.46

In [239…

```
# Bottom 5 sellers in North America

df.groupby(["Name"])[["NA_Sales"]].sum().sort_values(by="NA_Sales",ascending=True).
```

Out[239…

Name	NA_Sales
¡Shin Chan Flipa en colores!	0.0
Far East of Eden Shinden	0.0
Far East of Eden Zero	0.0
Farm Frenzy 3	0.0
PachiPara 14: Fuu to Kumo to Super Umi in Okinawa	0.0

In [240…

```
# Maxixmum,Average and Minimum Sales in Europe

df[["EU_Sales"]].agg(["max","mean","min"])
```

Out[240…

	EU_Sales
max	29.020000
mean	0.146883
min	0.000000

In [241…

```
# Top 5 sellers in Europe

df.groupby(["Name"])[["EU_Sales"]].sum().sort_values(by="EU_Sales",ascending=False)
```

Out[241…

	EU_Sales
Name	
Wii Sports	29.02
Grand Theft Auto V	23.04
Mario Kart Wii	12.88
FIFA 15	12.40
Call of Duty: Modern Warfare 3	11.29

In [242…

```
# Bottom 5 sellers in Europe

df.groupby(["Name"])[["EU_Sales"]].sum().sort_values(by="EU_Sales",ascending=True).
```

Out[242…

	EU_Sales
Name	
¡Shin Chan Flipa en colores!	0.0
Okaeri! Chibi-Robo! Happy Richie Oosouji	0.0
Okashi na Shima no Peter Pan: Sweet Never Land	0.0
Flower, Sun, and Rain	0.0
Omega Labyrinth	0.0

In [243…

```
# Maximum,Average and Minimum Sale in Japan

df[["JP_Sales"]].agg(["max","mean","min"])
```

Out[243…

	JP_Sales
max	10.220000
mean	0.077998
min	0.000000

In [244…

```
# Top 5 Sellers In Japan

df.groupby(["Name"])[["JP_Sales"]].sum().sort_values(by="JP_Sales",ascending=False)
```

Out[244…

Name	JP_Sales
Pokemon Red/Pokemon Blue	10.22
Pokemon Gold/Pokemon Silver	7.20
Super Mario Bros.	6.96
New Super Mario Bros.	6.50
Pokemon Diamond/Pokemon Pearl	6.04

In [245…

```
# Bottom 5 Sellers in Japan


df.groupby(["Name"])[["JP_Sales"]].sum().sort_values(by="JP_Sales",ascending=True).
```

Out[245…

Name	JP_Sales
Just Dance 4	0.0
Master of Monsters: Disciples of Gaia	0.0
Master Jin Jin's IQ Challenge	0.0
Mass Effect Trilogy	0.0
Mass Destruction	0.0

In [246…

```
# Maximum,Average and Minimum Sales in Other countries

df[["Other_Sales"]].agg(["max","mean","min"])
```

Out[246…

	Other_Sales
max	10.570000
mean	0.048191
min	0.000000

In [247…

```
# Top 5 sellers in Other Countries

df.groupby(["Name"])[["Other_Sales"]].sum().sort_values(by="Other_Sales",ascending=
```

Out[247…

	Other_Sales
Name	
Grand Theft Auto: San Andreas	10.72
Wii Sports	8.46
Grand Theft Auto V	8.03
Gran Turismo 4	7.53
Call of Duty: Black Ops II	3.88

In [248…

```
# Bottom 5 Sellers in Other Countries

df.groupby(["Name"])[["Other_Sales"]].sum().sort_values(by="Other_Sales",ascending=
```

Out[248…

	Other_Sales
Name	
Mega Man Soccer	0.0
Mezase!! Tsuri Master DS	0.0
Metroid Prime: Federation Force	0.0
Meteos: Disney Magic	0.0
Metal Slug XX	0.0

In [249…

```
# Maximum,Average,Minimum Sales in Global Sales

df[["Global_Sales"]].agg(["max","mean","min"])
```

Out[249…

	Global_Sales
max	82.740000
mean	0.538426
min	0.010000

In [323…

```
# Top 5 Grossing Games Globally

df.groupby(["Name"])[["Global_Sales"]].sum().sort_values(by="Global_Sales",ascendin
```

Out[323…

	Global_Sales
Name	
Wii Sports	82.74
Grand Theft Auto V	55.92
Super Mario Bros.	45.31
Tetris	35.84
Mario Kart Wii	35.82

In [292…

```
# least 5 Grossing games Globally

df.groupby(["Name"])[["Global_Sales"]].sum().sort_values(by="Global_Sales",ascendin
```

Out[292…

	Global_Sales
Name	
Dragon's Dogma Online: Season 2	0.01
Memories Off	0.01
Tom Clancy's Ghost Recon Advanced Warfighter (weekly JP sales)	0.01
Fit & Fun	0.01
Paranormal Pursuit: The Gifted One	0.01

In [261…

```
# How many Games were sold

df["Name"].nunique()
```

Out[261…

```
11442
```

In [262…

```
# Number of times Each Game was Sold (top 5)

df[["Name"]].value_counts().sort_values(ascending=False).head(5)
```

Out[262…

```
Name
Need for Speed: Most Wanted    12
Ratatouille                     9
FIFA 14                         9
Madden NFL 07                   9
LEGO Marvel Super Heroes        9
Name: count, dtype: int64
```

In [264…

```
# Sum of Sales in Each Country and Globally

df[["NA_Sales","EU_Sales","JP_Sales","Other_Sales","Global_Sales"]].sum().sort_valu
```

Out[264…

```
Global_Sales     8905.56
NA_Sales         4384.41
EU_Sales         2429.45
JP_Sales         1290.09
Other_Sales       797.08
dtype: float64
```

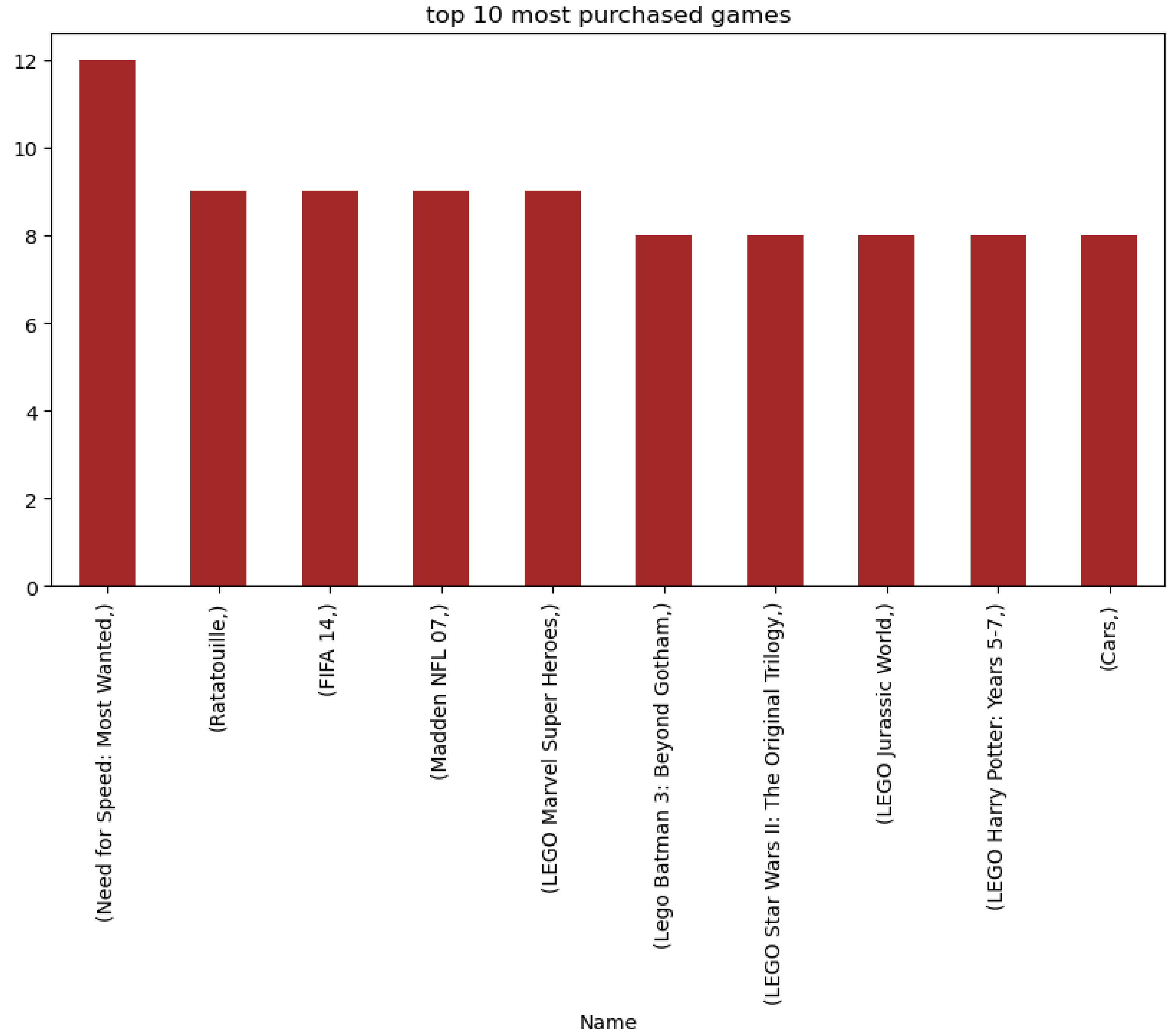
In [294…

```
# Top 10 most purchased Games

df[["Name"]].value_counts().sort_values(ascending=False).head(10).plot(kind="bar",c
                                                                      title= "top 1
```

Out[294…

```
<Axes: title={'center': 'top 10 most purchased games'}, xlabel='Name'>
```



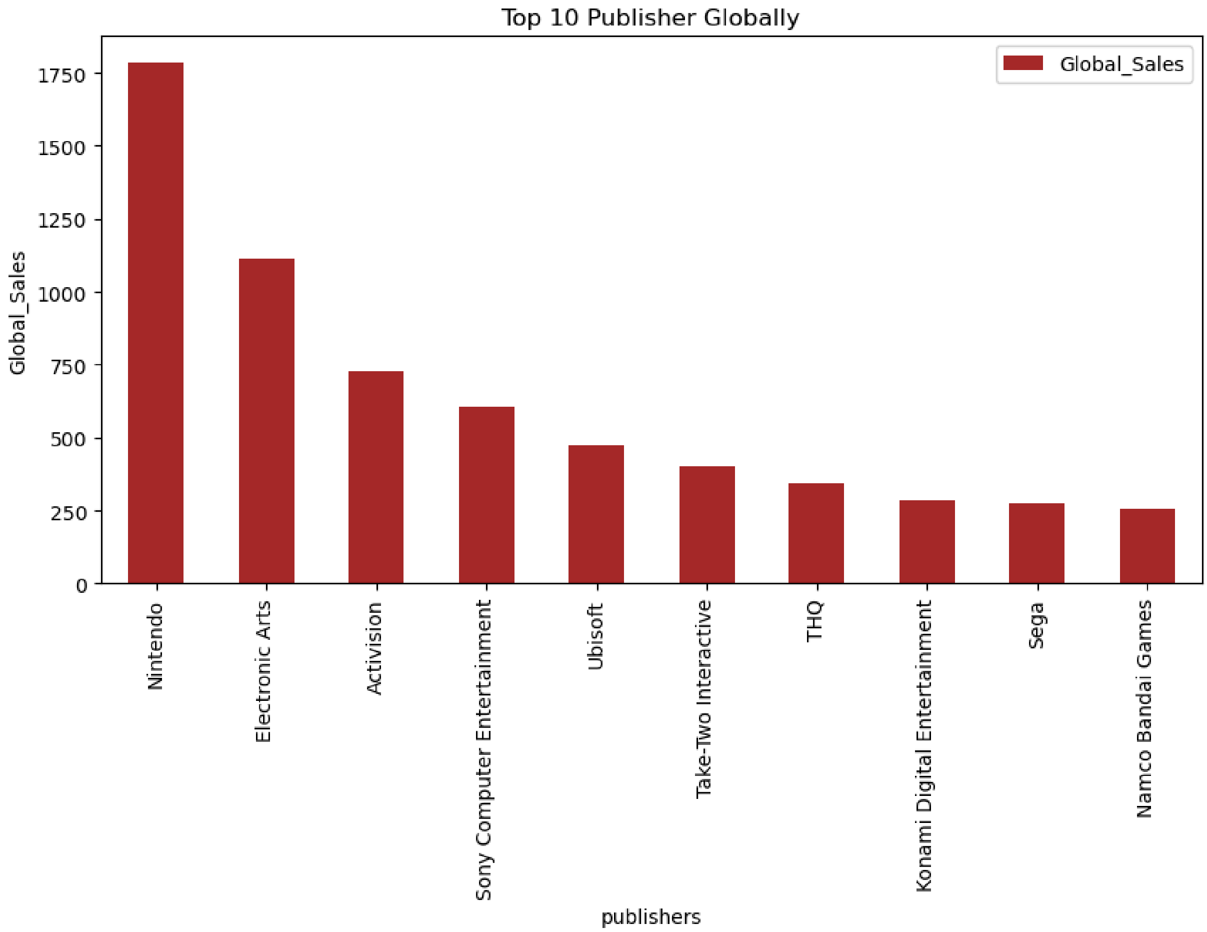
In [354…

```
# Top 10 Publishers Globally

df.groupby(["Publisher"])[["Global_Sales"]].sum().sort_values(by="Global_Sales",asc
```

Out[354…

```
<Axes: title={'center': 'Top 10 Publisher Globally'}, xlabel='publishers', ylabel
='Global_Sales'>
```



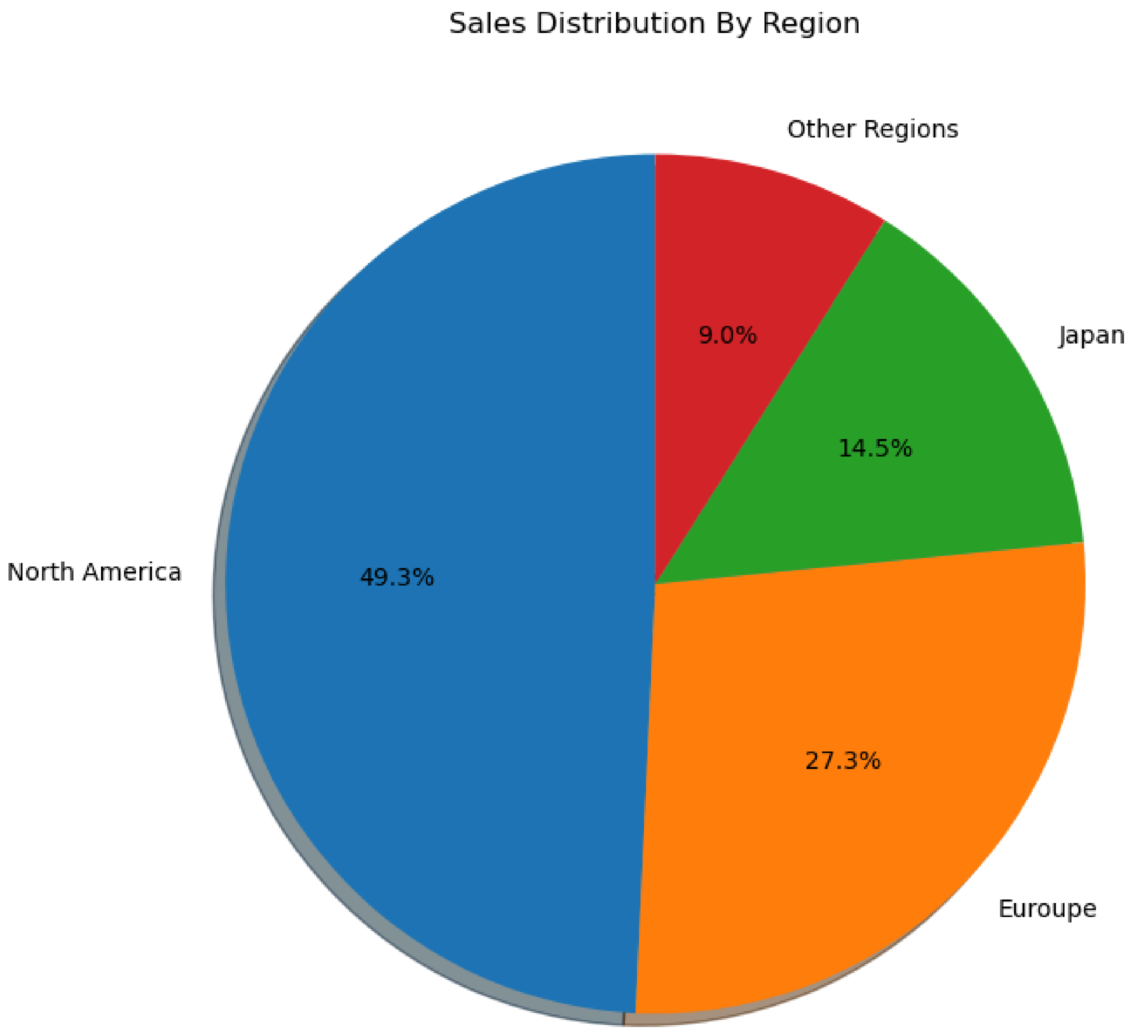
In [352…

```
# Sales Distribution By Region

df[['NA_Sales','EU_Sales','JP_Sales','Other_Sales']].sum().plot(kind="pie",labels=[
                                                                            "Other 
                                                                    startangle=90,s
                                                                    autopct="%1.1f%
```

Out[352…

```
<Axes: title={'center': 'Sales Distribution By Region'}>
```


Sales Distribution By Region

In [297…

```
df.head(1)
```

Out[297…

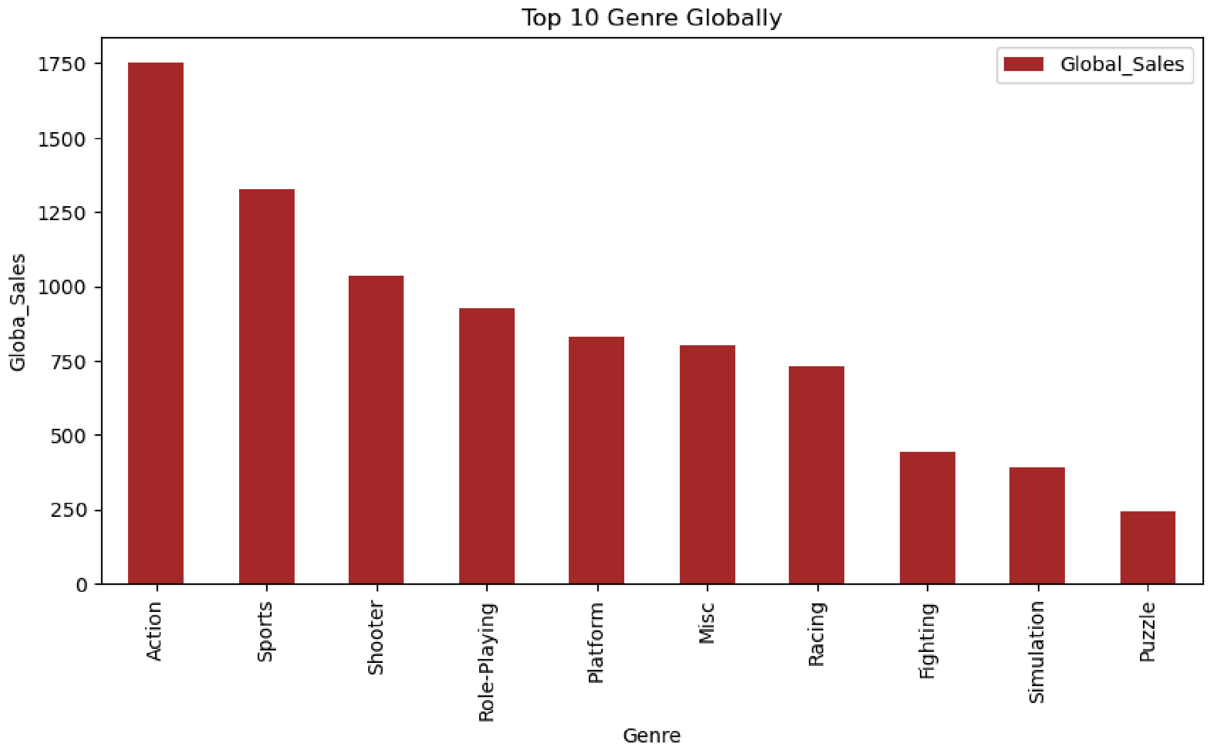
	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sa
0	1	Wii Sports	Wii	2006	Sports	Nintendo	41.49	29.02	3.77	8

In [325…

```
df.groupby(["Genre"])[["Global_Sales"]].sum().sort_values(by="Global_Sales",
                                                           ascending=False).head(10)
```

Out[325…

```
<Axes: title={'center': 'Top 10 Genre Globally'}, xlabel='Genre', ylabel='Globa_Sa
les'>
```

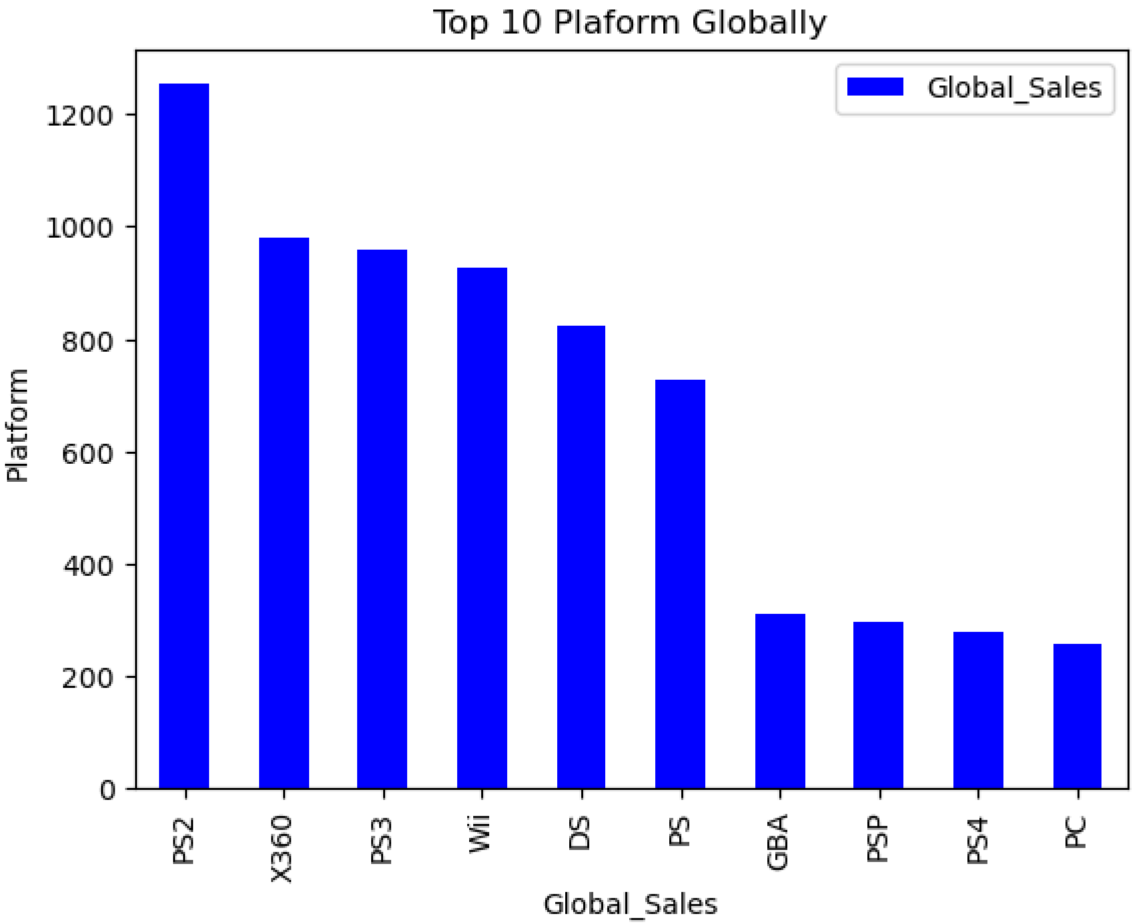


In [356…

```
df.groupby("Platform")[["Global_Sales"]].sum().sort_values(by="Global_Sales",ascend
```

Out[356…

```
<Axes: title={'center': 'Top 10 Plaform Globally'}, xlabel='Global_Sales', ylabel
='Platform'>
```



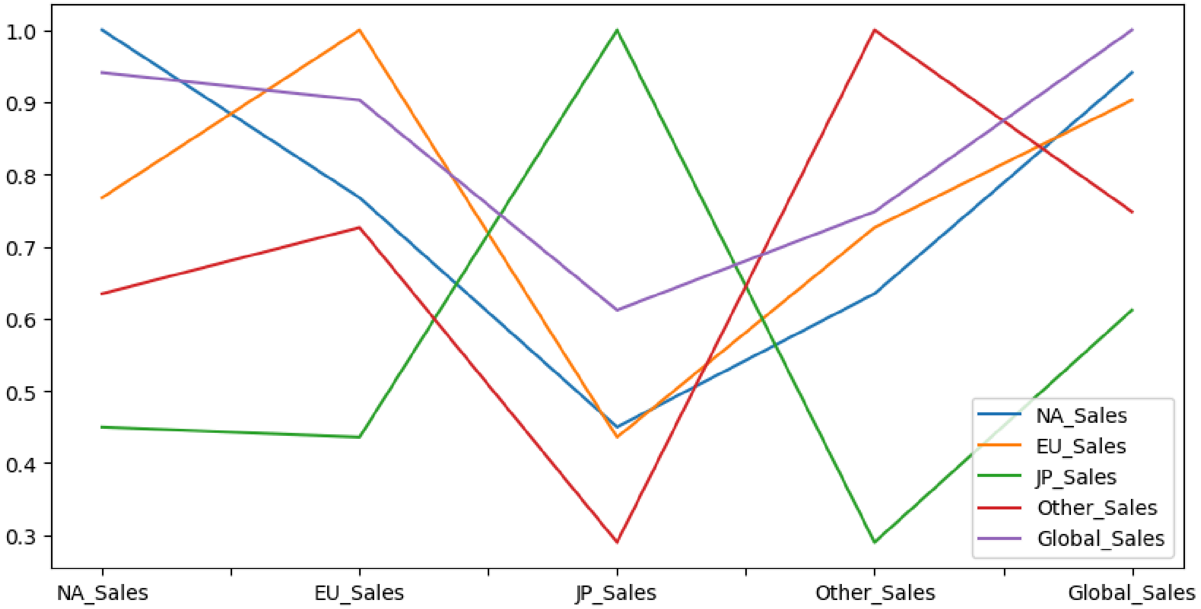
In [333…

```
#Calculate correlation matrix between sales columns and visualize as a line plot

df[['NA_Sales', 'EU_Sales', 'JP_Sales', 'Other_Sales', 'Global_Sales']].corr().plot
```

Out[333…

```
<Axes: >
```

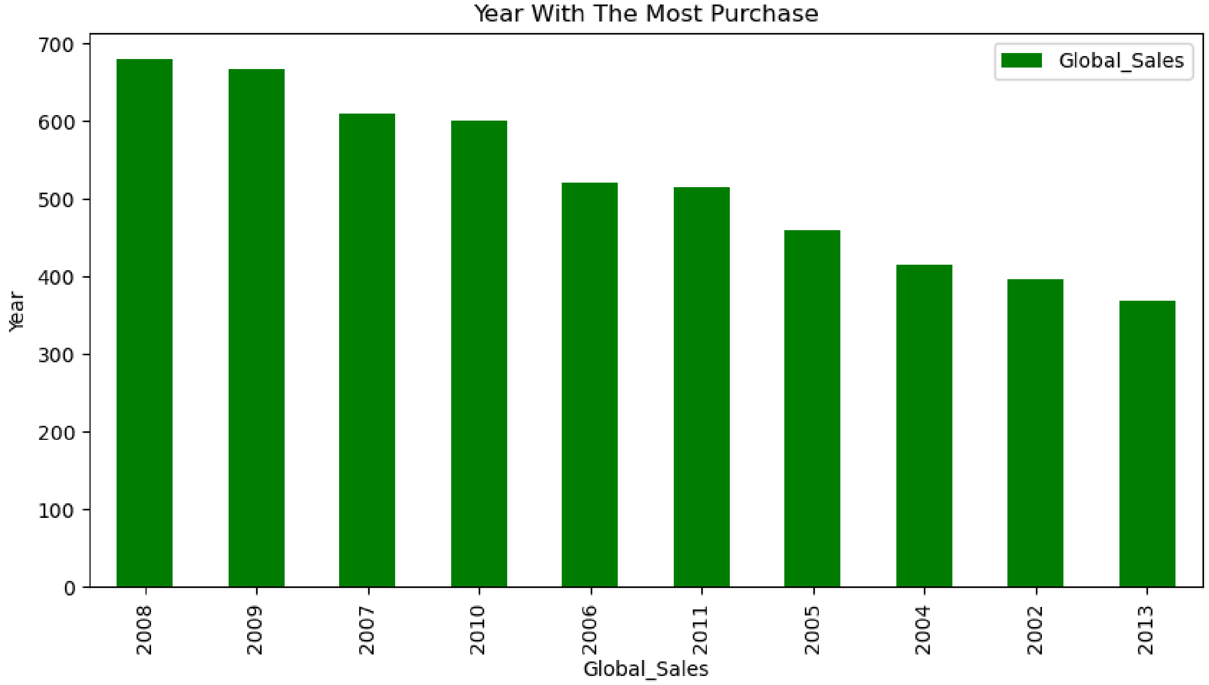


In [355…

```
df.groupby("Year")[["Global_Sales"]].sum().sort_values(by="Global_Sales",ascending=
```

Out[355…

```
<Axes: title={'center': 'Year With The Most Purchase'}, xlabel='Global_Sales', ylabel='Year'>
```


Year With The Most Purchase
Global_Sales
Year
0
100
200
300
400
500
600
700
2008
2009
2007
2010
2006
2011
2005
2004
2002
2013
Global_Sales

In [ ]: