



# Tubes 2

## *Data Analysis*

---



# Anggota kelompok

- Albert (16521192)
- Daven Darmawan Sendjaya (16521322)
- Jimly Firdaus (16521312)
- Muhammad Naufal (16521442)




A close-up, slightly blurred photograph of a person's hands holding a black and gold video game controller. The controller features a D-pad, several colored buttons (red, green, blue, yellow), and a central touchpad. The background is dark and out of focus, showing hints of a person's clothing.

# Video games sales

---

# Latar belakang

- ◈ Gim merupakan sebuah permainan yang berfungsi sebagai salah satu bentuk hiburan
- ◈ Sekarang ini, banyak sekali permainan yang bisa dilakukan melalui gawai, misalnya gim video yang tersedia pada *game console* atau yang tersedia di toko aplikasi pada ponsel pintar.
- ◈ Kemudahan dan kepraktisan dalam memainkan gim membuat banyak masyarakat tertarik untuk menggunakannya untuk mengusir rasa bosan atau menghabiskan waktu senggang. Oleh karena itu, banyak sekali pengembang (*game developer*) yang memproduksi berbagai jenis gim video yang dijual secara masif kepada khalayak umum.



# Tugas 1

## (Deskripsi data dan file)

---



# Deskripsi Data

## Deskripsi data:

- ❖ Data ini digunakan sebagai acuan analisis terhadap lebih dari 16500 jenis *video games* Sebelum *cleansing*

| Format file | Dimensi file (brs x klm) | Ukuran data | Ukuran file |
|-------------|--------------------------|-------------|-------------|
| csv         | 16719 x 16               | 220788      | 1618040     |

## Proses Pencarian Ukuran Data sebelum *Data Cleansing*

```
#ukuran dimensi csv (tanpa data kosong)
data.size - (data.isnull().sum().sum())
```

✓ 0.1s

220788

## Proses Pencarian Ukuran File (dalam *bytes*) sebelum *Data Cleansing*

```
import os
#mencari informasi ukuran file csv
print(os.path.getsize("video_games_sales - Edit.csv"), " bytes")
```

✓ 0.1s

1618040 bytes

## Proses Pencarian Jumlah Baris dan Kolom

```
#Jumlah baris
● print("Ada ",len(data)," baris")
#Jumlah kolom
print("Ada",len(data.columns)," kolom")
```

✓ 0.6s

Ada 16719 baris

Ada 16 kolom



# Deskripsi Data

Setelah *cleansing*

| Format file | Dimensi file (brs x klm) | Ukuran data | Ukuran file |
|-------------|--------------------------|-------------|-------------|
| csv         | 6825 x 16                | 109200      | 791986      |

Proses Pencarian Ukuran Data setelah *Data Cleansing*

```
#ukuran dimensi csv setelah cleaning
data.size
```

✓ 0.7s

109200

Proses Pencarian Ukuran File (dalam *bytes*) setelah *Data Cleansing*

```
import os
#mencari informasi ukuran file csv
os.path.getsize("video_games_sales.csv")
```

✓ 0.1s

791986

Proses Pencarian Jumlah Baris dan Kolom

```
#Jumlah baris
print("Ada "+str(len(data))+ " baris")
#Jumlah kolom
print("Ada",len(data.columns)," kolom")
```

✓ 0.9s

Ada 6825 baris

Ada 16 kolom



# Tugas 2

## Karakteristik data



# Kolom, Keterangan, dan Jenis Data

| Kolom           | Keterangan  | Jenis Data               |
|-----------------|---|--------------------------|
| Name            | Nama dari video games   | Categorical - nominal    |
| Platform        | Konsol dari video games   | Categorical - nominal    |
| Year_of_Release | Tahun rilis video games   | Quantitative - continues |
| Publisher       | Pihak yang menerbitkan video games  | Categorical - nominal    |
| Genre           | Jenis permainan video games   | Categorical - nominal    |
| NA_Sales        | Penjualan video games   | Quantitative - discrete  |
| EU_Sales        | Penjualan video games di Europe   | Quantitative - discrete  |
| JP_Sales        | Penjualan video games di Jepang   | Quantitative - discrete  |
| Other_Sales     | Penjualan video games di seluruh negara kecuali North America, Europe, dan Jepang | Quantitative - discrete  |
| Global_Sales    | Penjualan video games di seluruh dunia  | Quantitative - discrete  |
| Critic_Score    | Nilai kritik dari video games   | Categorical - ordinal    |
| Critic_Count    | Jumlah penilai dari informasi Critic_Score  | Quantitative - discrete  |
| User_Score      | Penilaian dari user terhadap video games  | Categorical - ordinal    |
| User_Count      | Jumlah pengguna dari video games  | Quantitative - discrete  |
| Developer       | Perusahaan yang membuat video games   | Categorical - nominal    |
| Rating          | Penilaian keseluruhan video games   | Categorical - ordinal    |

Kode Pemrograman untuk mencari kolom “name”

```
data[["Name"]]
```

✓ 0.6s

|       |                               |
|-------|-------------------------------|
| 0     | Wii Sports                    |
| 1     | Super Mario Bros.             |
| 2     | Mario Kart Wii                |
| 3     | Wii Sports Resort             |
| 4     | Pokemon Red/Pokemon Blue      |
|       | ...                           |
| 16714 | Samurai Warriors: Sanada Maru |
| 16715 | LMA Manager 2007              |
| 16716 | Haitaka no Psychedelica       |
| 16717 | Spirits & Spells              |
| 16718 | Winning Post 8 2016           |

Kode Pemrograman untuk mencari kolom “platform”

```
data["Platform"].value_counts()
```

✓ 0.1s

|      |      |
|------|------|
| PS2  | 1140 |
| X360 | 858  |
| PS3  | 769  |
| PC   | 651  |
| XB   | 565  |
| Wii  | 479  |
| DS   | 464  |
| PSP  | 390  |
| GC   | 348  |
| PS4  | 239  |
| GBA  | 237  |
| XOne | 159  |
| 3DS  | 155  |
| PS   | 150  |
| PSV  | 118  |
| WiiU | 89   |
| DC   | 14   |

Kode Pemrograman untuk mencari kolom “year of release”

```
print("Range:",data1["Year_of_Release"].min()," - ",data1["Year_of_Release"].max())
```

✓ 0.1s

Range: 1985.0 - 2016.0

## Kode Pemrograman untuk mencari kolom “publisher”

```
data["Publisher"].value_counts()
✓ 0.1s
```

|                             |     |
|-----------------------------|-----|
| Electronic Arts             | 944 |
| Ubisoft                     | 496 |
| Activision                  | 492 |
| Sony Computer Entertainment | 316 |
| THQ                         | 307 |
| ...                         |     |
| DSI Games                   | 1   |
| Sunsoft                     | 1   |
| 03 Entertainment            | 1   |
| Tetris Online               | 1   |
| NewKidCo                    | 1   |

## Kode Pemrograman untuk mencari kolom “genre”

```
data["Genre"]
✓ 0.2s
```

|       |              |
|-------|--------------|
| 0     | Sports       |
| 1     | Platform     |
| 2     | Racing       |
| 3     | Sports       |
| 4     | Role-Playing |
| ...   |              |
| 16714 | Action       |
| 16715 | Sports       |
| 16716 | Adventure    |
| 16717 | Platform     |
| 16718 | Simulation   |

## Kode Pemrograman untuk mencari kolom “NA\_Sales”

```
print("Range:",data1["NA_Sales"].min()," - ",data1["NA_Sales"].max())
✓ 0.9s
```

Range: 0.0 - 41.36

## Kode Pemrograman untuk mencari kolom “JP\_Sales”

```
print("Range:",data1["JP_Sales"].min()," - ",data1["JP_Sales"].max())
✓ 0.2s
```

Range: 0.0 - 10.22

## Kode Pemrograman untuk mencari kolom “EU\_Sales”

```
print("Range:",data1["EU_Sales"].min()," - ",data1["EU_Sales"].max())
✓ 0.5s
```

Range: 0.0 - 28.96

## Kode Pemrograman untuk mencari kolom “Other\_Sales”

```
print("Range:",data1["Other_Sales"].min()," - ",data1["Other_Sales"].max())
✓ 0.2s
```

Range: 0.0 - 10.57



## Kode Pemrograman untuk mencari kolom “Global\_Sales”

```
print("Range:",data1["Global_Sales"].min()," - ",data1["Global_Sales"].max())
```

✓ 0.2s

Range: 0.01 - 82.53

## Kode Pemrograman untuk mencari kolom “Critic\_Score”

```
print("Range:",data1["Critic_Score"].min()," - ",data1["Critic_Score"].max())
```

✓ 0.2s

Range: 13.0 - 98.0

## Kode Pemrograman untuk mencari kolom “Critic\_Count”

```
print("Range:",data1["Critic_Count"].min()," - ",data1["Critic_Count"].max())
```

✓ 0.2s

Range: 3.0 - 113.0

## Kode Pemrograman untuk mencari kolom “User\_Score”

```
print("Range:",data1["User_Score"].min()," - ",data1["User_Score"].max())
```

✓ 0.1s

Range: 0.5 - 9.6

## Kode Pemrograman untuk mencari kolom “User\_Count”

```
print("Range:",data1["User_Count"].min()," - ",data1["User_Count"].max())
```

✓ 0.2s

Range: 4.0 - 10665.0

## Kode Pemrograman untuk mencari kolom “Developer”

```
data["Developer"].value_counts()
```

✓ 0.2s

|                                    |     |
|------------------------------------|-----|
| EA Canada                          | 149 |
| EA Sports                          | 142 |
| Capcom                             | 126 |
| Ubisoft                            | 103 |
| Konami                             | 95  |
| ...                                |     |
| Sanzaru Games, Sanzaru Games, Inc. | 1   |
| SCEA, Think and Feel               | 1   |
| Ubisoft Annecy                     | 1   |
| Omega Force, Koei Canada           | 1   |
| Atomic Games                       | 1   |

## Kode Pemrograman untuk mencari kolom “Rating”

```
data["Rating"].value_counts()
```

✓ 0.1s

|      |      |
|------|------|
| T    | 2377 |
| E    | 2082 |
| M    | 1433 |
| E10+ | 930  |
| AO   | 1    |
| K-A  | 1    |
| RP   | 1    |

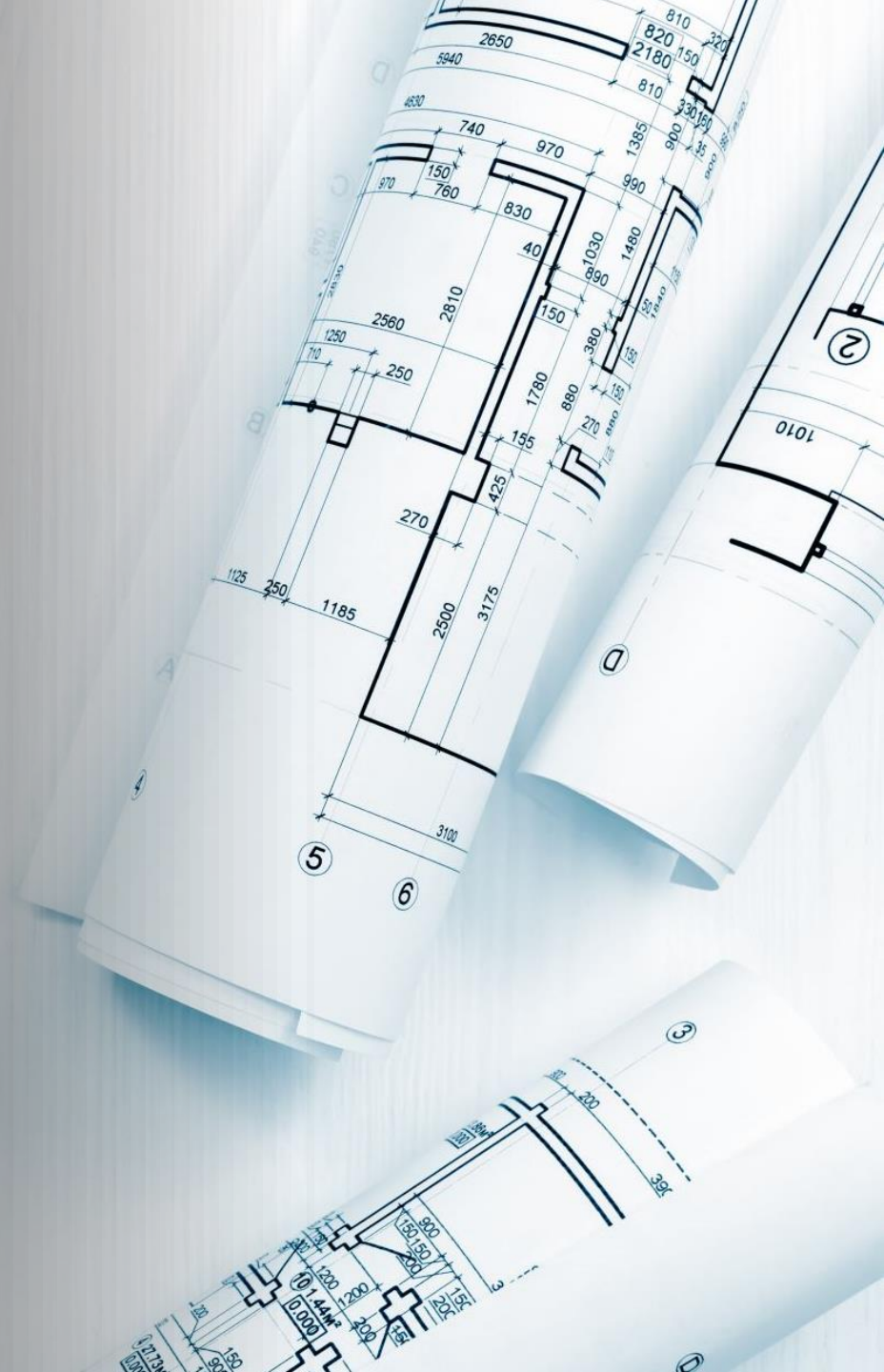
# Kolom dan karakteristik data

| Kolom           | Karakteristik  |
|-----------------|--|
| Name            | Wii Sports, Super Mario Bros., Mario Cart Wii, Wii Sports Resort, Pokemon Red/Pokemon Blue, ..., Samurai Warriors: Sanada Maru, LMA Manager 2007, Haitaka no Psychedelica, Spirits & Spells, Winning Post 8 2016<br><br><b>Catatan:</b> Data kolom “Name” tidak bisa ditampilkan semua karena berjumlah sangat banyak. |
| Platform        | PS2, DS, PS3, Wii, X360, PSP, PS, PC, XB, GBA, GC, 3DS, PSV, PS4 N64, XOne, SNES, SAT, WiiU, 2600, NES, GB, DC, GEN, NG, SCD, WS, 3DO, TG16, GG, PCFX  |
| Year_of_Release | 1985 - 2016  |
| Publisher       | Electronic Arts, Ubisoft, Activision, Sony Computer Entertainment, THQ, ..., DSI Games, Sunsoft, O3 Entertainment, Tetris Online, NewKidCo<br><br><b>Catatan:</b> Data kolom “Publisher” tidak bisa ditampilkan semua karena berjumlah sangat banyak.  |
| Genre           | Action, Sports, Misc, Role-Playing, Shooter, Adventure, Racing, Platform, Simulation, Fighting, Strategy, Puzzle   |
| NA_Sales        | 0 – 41,36 juta   |
| EU_Sales        | 0 – 28,96 juta   |
| JP_Sales        | 0 – 10,22 juta   |
| Other_Sales     | 0 – 10,57 juta   |
| Global_Sales    | 0 – 82,53 juta   |
| Critic_Score    | 13 – 98%   |
| Critic_Count    | 3 – 113 orang  |
| User_Score      | 0.5 – 9.6%   |
| User_Count      | 4 – 10665 orang  |
| Developer       | EA Canada, EA Sports, Capcom, Ubisoft, Konami, ..., Sanzaru Games, Inc., SCEA, Think and Feel, Ubisoft Annecy, Omega Force, Koei Canada, Atomic Games<br><br><b>Catatan:</b> Data kolom “Developer” tidak bisa ditampilkan semua karena berjumlah sangat banyak.   |
| Rating          | T, E, M, E10+, AO, K-A, RP   |



# Tugas 3 Statistik

---





| Menampilkan 5 Data Teratas dari Keseluruhan Data |                          |          |                 |              |           |          |          |          |             |              |
|--|--------------------------|----------|-----------------|--------------|-----------|----------|----------|----------|-------------|--------------|
| <pre>#mencetak 5 data pertama data.head(5)</pre> |                          |          |                 |              |           |          |          |          |             |              |
|  |                          |          |                 |              |           |          |          |          |             |              |
|  | Name                     | Platform | Year_of_Release | Genre        | Publisher | NA_Sales | EU_Sales | JP_Sales | Other_Sales | Global_Sales |
| 0  | Wii Sports               | Wii      | 2006.0          | Sports       | Nintendo  | 41.36    | 28.96    | 3.77     | 8.45        | 82.53        |
| 1  | Super Mario Bros.        | NES      | 1985.0          | Platform     | Nintendo  | 29.08    | 3.58     | 6.81     | 0.77        | 40.24        |
| 2  | Mario Kart Wii           | Wii      | 2008.0          | Racing       | Nintendo  | 15.68    | 12.76    | 3.79     | 3.29        | 35.52        |
| 3  | Wii Sports Resort        | Wii      | 2009.0          | Sports       | Nintendo  | 15.61    | 10.93    | 3.28     | 2.95        | 32.77        |
| 4  | Pokemon Red/Pokemon Blue | GB       | 1996.0          | Role-Playing | Nintendo  | 11.27    | 8.89     | 10.22    | 1.00        | 31.37        |

| Menampilkan 3 Game dengan Penjualan Global Tertinggi   |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|
| <pre>#mencari data game dengan penjualan global tertinggi global_sales = data.sort_values("Global_Sales", ascending=False) #mencetak 3 data pertama global_sales.head(3)</pre> |  |  |  |  |  |  |  |  |  |  |

|   | Name              | Platform | Year_of_Release | Genre    | Publisher | NA_Sales | EU_Sales | JP_Sales | Other_Sales | Global_Sales |
|---|-------------------|----------|-----------------|----------|-----------|----------|----------|----------|-------------|--------------|
| 0 | Wii Sports        | Wii      | 2006.0          | Sports   | Nintendo  | 41.36    | 28.96    | 3.77     | 8.45        | 82.53        |
| 1 | Super Mario Bros. | NES      | 1985.0          | Platform | Nintendo  | 29.08    | 3.58     | 6.81     | 0.77        | 40.24        |
| 2 | Mario Kart Wii    | Wii      | 2008.0          | Racing   | Nintendo  | 15.68    | 12.76    | 3.79     | 3.29        | 35.52        |

## Menampilkan 3 Game dengan Tahun Rilis Terlama

```
#mencari data tahun rilis terlama
year_of_release = data.sort_values("Year_of_Release", ascending=True)
#mencetak 3 data pertama
year_of_release.head(3)
```

|      | Name            | Platform | Year_of_Release | Genre   | Publisher  | NA_Sales | EU_Sales | JP_Sales | Other_Sales | Global_Sales |
|------|-----------------|----------|-----------------|---------|------------|----------|----------|----------|-------------|--------------|
| 262  | Asteroids       | 2600     | 1980.0          | Shooter | Atari      | 4.00     | 0.26     | 0.0      | 0.05        | 4.31         |
| 5360 | Freeway         | 2600     | 1980.0          | Action  | Activision | 0.32     | 0.02     | 0.0      | 0.00        | 0.34         |
| 546  | Missile Command | 2600     | 1980.0          | Shooter | Atari      | 2.56     | 0.17     | 0.0      | 0.03        | 2.76         |

## Sampel 3 Data Kolom “Name”

```
#mencetak 3 data pertama dari kolom Name
data["Name"].head(3)
```

```
0      Wii Sports
1  Super Mario Bros.
2    Mario Kart Wii
Name: Name, dtype: object
```

## 1. Rata-rata

```
#mencari rata-rata  
data["Global_Sales"].mean()
```

0.5335426759974459

## 2. Standar Deviasi

```
#mencari standar deviasi  
data["Global_Sales"].std()
```

1.5479351247177218

## 3. Persentil 10%

```
#mencari persentil 10%  
data["Global_Sales"].quantile(0.10)
```

0.02

## 4. Persentil 25%

```
#mencari persentil 25%  
data["Global_Sales"].quantile(0.25)
```

0.06

## 5. Persentil 50%

```
#mencari persentil 50%  
data["Global_Sales"].quantile(0.50)
```

0.17

## 6. Persentil 75%

```
#mencari persentil 75%  
data["Global_Sales"].quantile(0.75)
```

0.47

## 7. Persentil 90%

```
#mencari persentil 90%  
data["Global_Sales"].quantile(0.9)
```

1.2



## 8. Nilai Maksimum

```
#mencari nilai maksimum  
data["Global_Sales"].max()
```

```
82.53
```

## 9. Nilai Minimum

```
#mencari nilai minimum  
data["Global_Sales"].min()
```

```
0.01
```

## 10. Distribusi Frekuensi

```
#mencari data publisher  
publisher = data["Publisher"].value_counts()  
publisher.head(5)
```

```
Electronic Arts      1356  
Activision           985  
Namco Bandai Games   939  
Ubisoft              933  
Konami Digital Entertainment  834  
Name: Publisher, dtype: int64
```



# VISUALISASI DATA

---

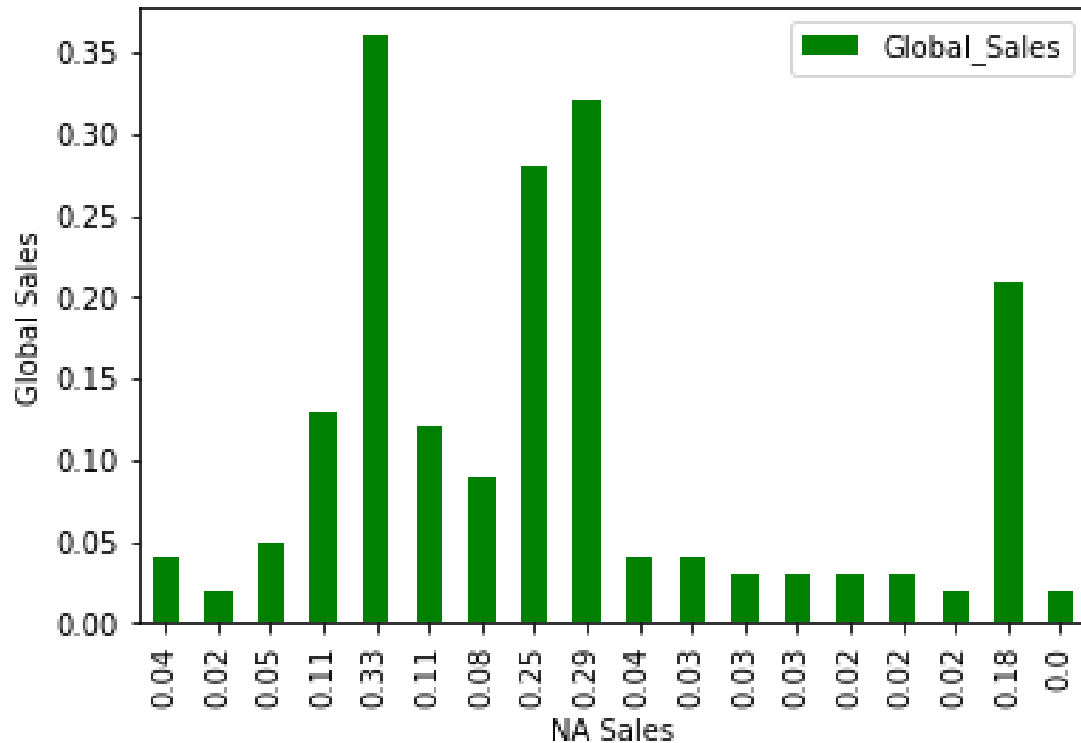


## 1. Perbandingan kategori

- Perbandingan penjualan di Amerika Utara dengan penjualan global berdasarkan Zushi Games

```
import matplotlib.pyplot as plt
#mencari data Zushi Games
data_zhushi_games = data.loc[data["Publisher"] == "Zushi Games"].sort_values("Year_of_Release", ascending=True)
#melakukan plotting
data_zhushi_games.plot(kind="bar", x = "NA_Sales", y = "Global_Sales", title = " Perbandingan Antara Penjualan di Amerika Utara dengan Penjualan Global", color=
plt.xlabel("NA Sales")
plt.ylabel("Global Sales")
plt.show()
#Penggunaan warna hijau mengindikasikan penjualan yang cenderung fluktuatif
```

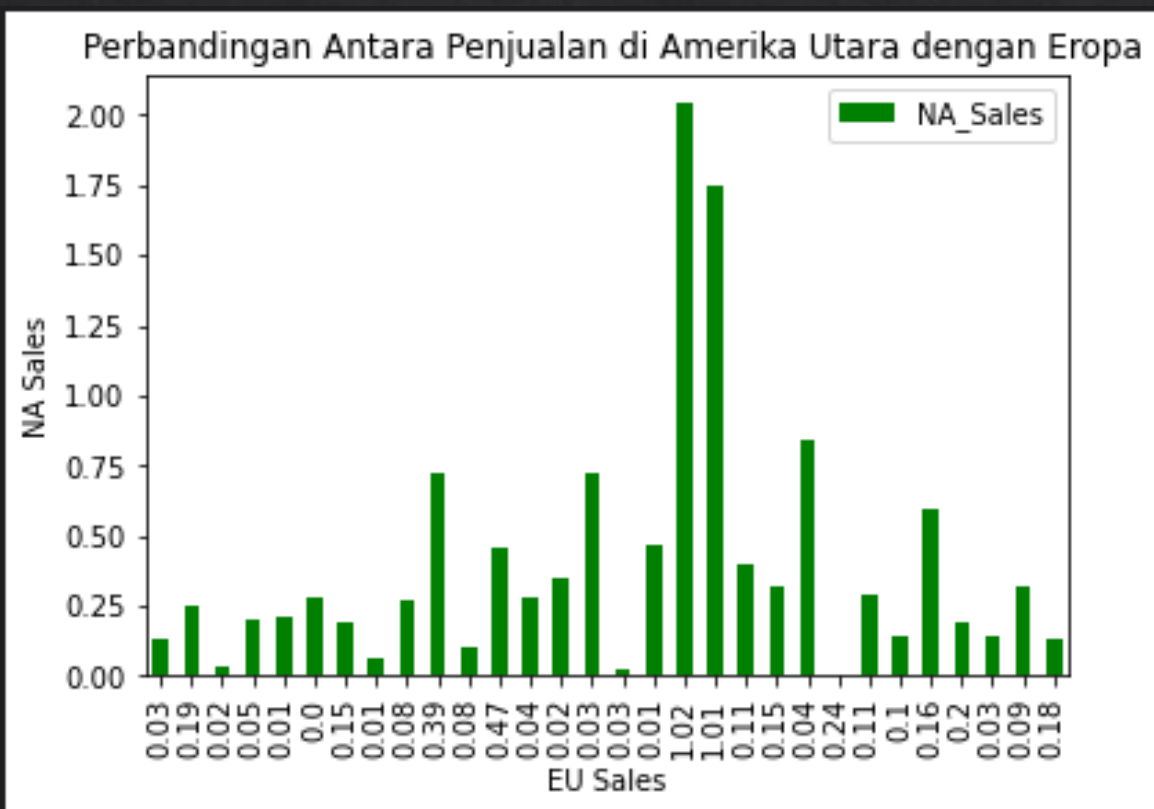
Perbandingan Antara Penjualan di Amerika Utara dengan Penjualan Global





- Perbandingan penjualan games Warner Bros. Interactive Entertainment Amerika Utara dengan Eropa

```
#mencari data Warner Bros. Interactive Entertainment
data_Warner_Bros_Interactive_Entertainment = data.loc[data["Publisher"] == "Warner Bros. Interactive Entertainment"].sort_values("Year_of_Release", ascending=True)
#menlakukan plotting
data_Warner_Bros_Interactive_Entertainment[:30].plot(kind="bar", x = "EU_Sales", y = "NA_Sales", title = " Perbandingan Antara Penjualan di Amerika Utara dengan Eropa")
plt.xlabel("EU Sales")
plt.ylabel("NA Sales")
plt.show
#Penggunaan warna hijau mengindikasikan penjualan yang cenderung fluktuatif
```

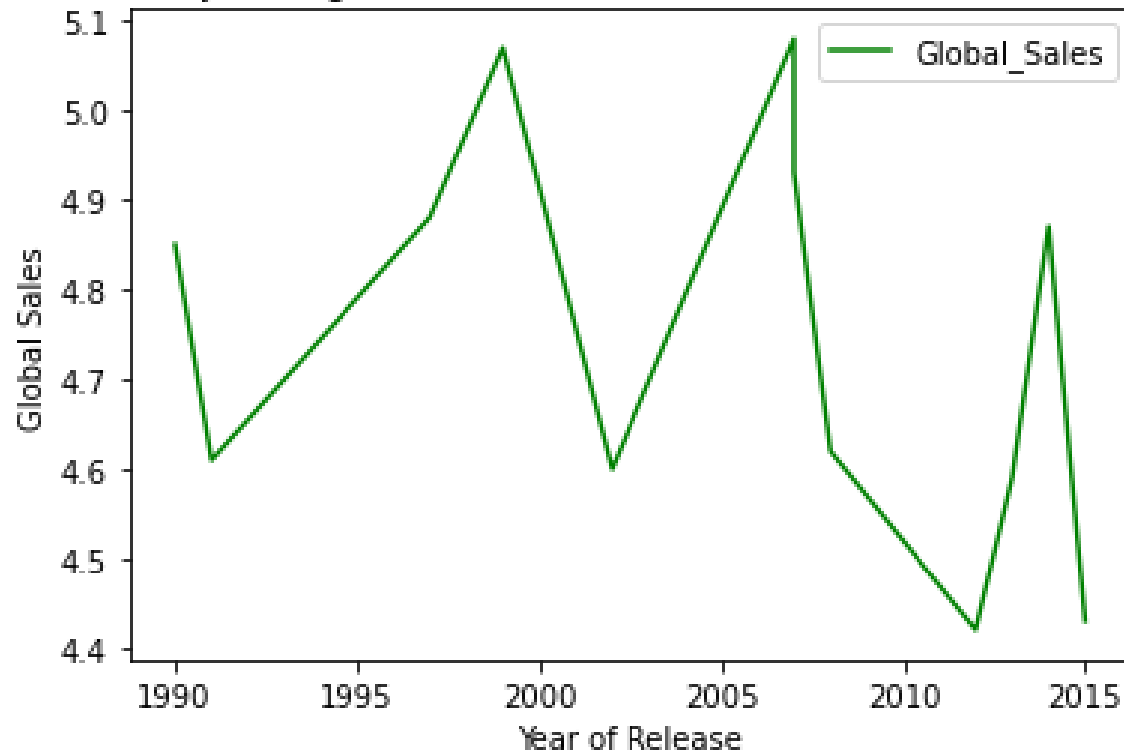


## 2. Penampilan perubahan terhadap waktu

- Grafik penjualan global dari Nintendo dari tahun 1990-2015

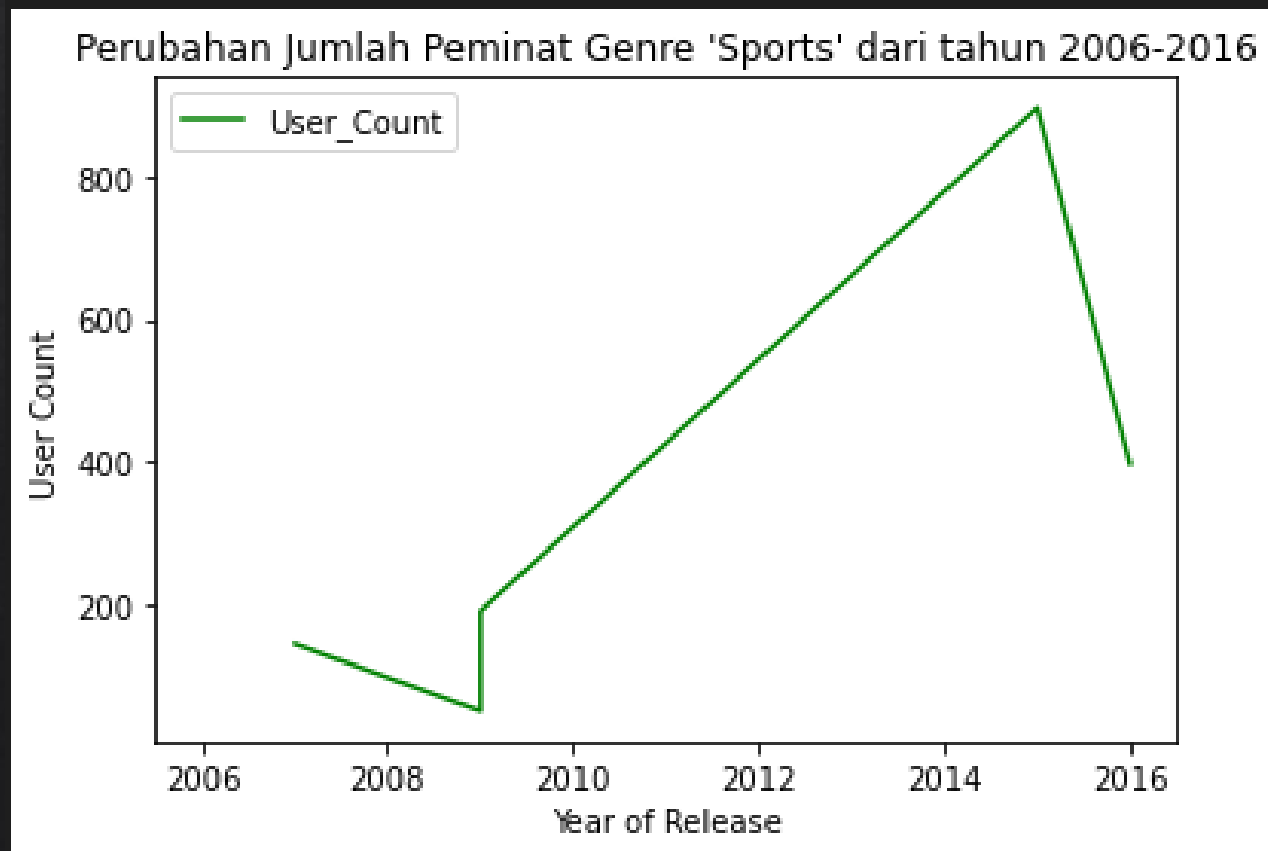
```
#mencari data Nintendo
data_time_series_nintendo = data[200:250].loc[data["Publisher"]=="Nintendo"].sort_values("Year_of_Release", ascending=True)
#melakukan plotting
data_time_series_nintendo.plot(kind="line", x = "Year_of_Release", y = "Global_Sales", title="Perubahan Penjualan game Nintendo di Amerika Utara dari Tahun 1990-2015")
plt.xlabel("Year of Release")
plt.ylabel("Global Sales")
plt.show()
```

Perubahan Penjualan game Nintendo di Amerika Utara dari Tahun 1990-2015



- Grafik perubahan jumlah peminat genre 'sports' dari tahun 2006-2016

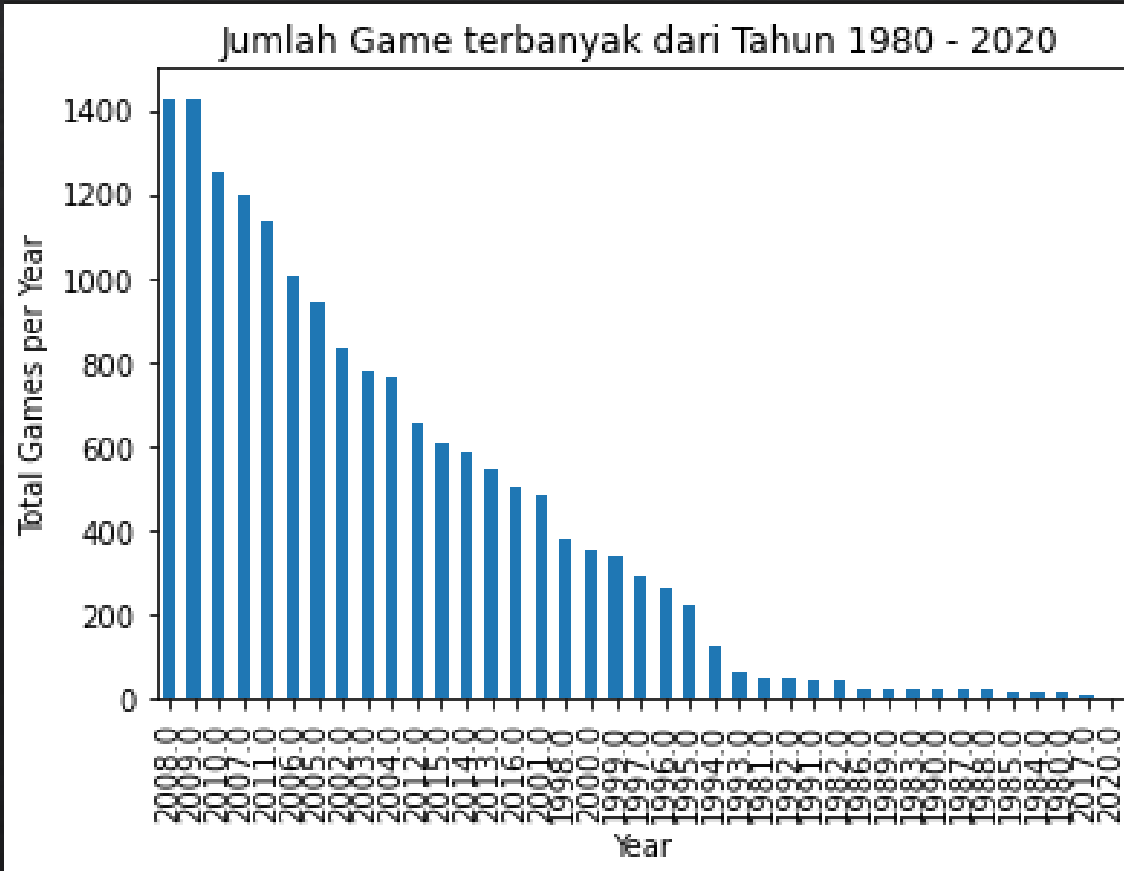
```
#mencari data dengan genre sports
data_time_series_nintendo = data[:100].loc[data["Genre"]=="Sports"].sort_values("Year_of_Release", ascending=False)
#melakukan plotting
data_time_series_nintendo.plot(kind="line", x = "Year_of_Release", y = "User_Count", title="Perubahan Jumlah Peminat Genre 'Sports' dari tahun 2006-2016",
plt.xlabel("Year of Release")
plt.ylabel("User Count")
plt.show()
```



### 3. Penampilan hierarki dan hubungan keseluruhan bagian

- Jumlah game terbanyak dari tahun 1980-2020

```
#mencari data dari total game yang dirilis tiap tahun
data_total_game = data["Year_of_Release"]
#melakukan plotting
data_total_game.value_counts().plot(kind="bar",stacked = True, title="Jumlah Game terbanyak dari Tahun 1980 - 2020")
plt.xlabel("Year")
plt.ylabel("Total Games per Year")
plt.show()
```





- Jumlah game terbanyak dari tahun 1980-2020

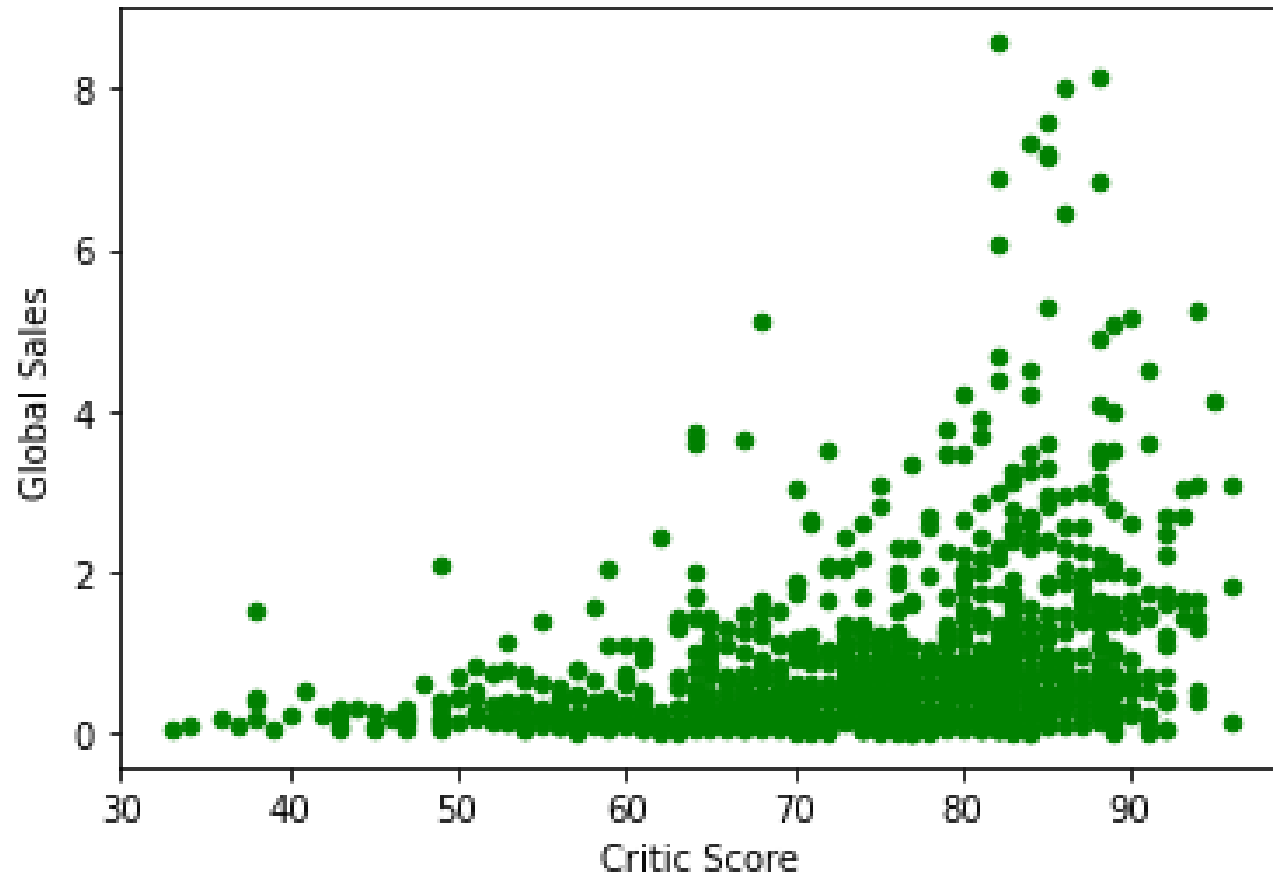
```
#mencari data dari total genre games dari tahun 1980-2020
data_total_game = data["Genre"]
#melakukan plotting
data_total_game.value_counts().plot(kind="pie", title="Genre games terbanyak dari Tahun 1980 - 2020")
plt.show()
```



## 4. Plotting Relationships

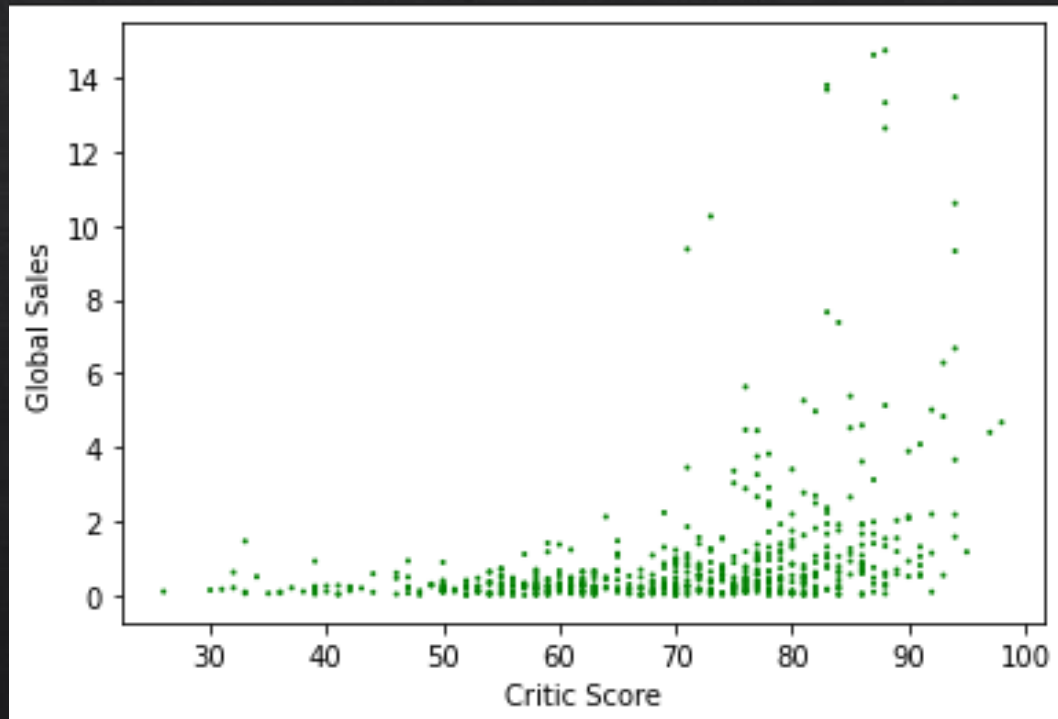
- Scatter plot dari penjualan Global Economic Arts terhadap year of release

```
#mencari data Electronic Arts
data_electronic_arts = data.loc[data["Publisher"]=="Electronic Arts"].sort_values("Year_of_Release", ascending=True)
#melakukan plotting
data_electronic_arts.plot(kind="scatter", x = "Critic_Score", y = "Global_Sales", color="green")
plt.xlabel("Critic Score")
plt.ylabel("Global Sales")
plt.show()
```



- Bubble plot dari penjualan global activision terhadap year of release, dengan sizes merupakan frekuensi video games dari publisher Activision

```
#mencari data Activision
data_electronic_arts = data.loc[data["Publisher"]=="Activision"].sort_values("Year_of_Release", ascending=True)
#melakukan plotting
data_electronic_arts.plot(kind="scatter", sizes = data_electronic_arts.value_counts() , x = "Critic_Score", y = "Global_Sales",
plt.xlabel("Critic Score")
plt.ylabel("Global Sales")
plt.show())
```





# Tugas 5

## Korelasi

---





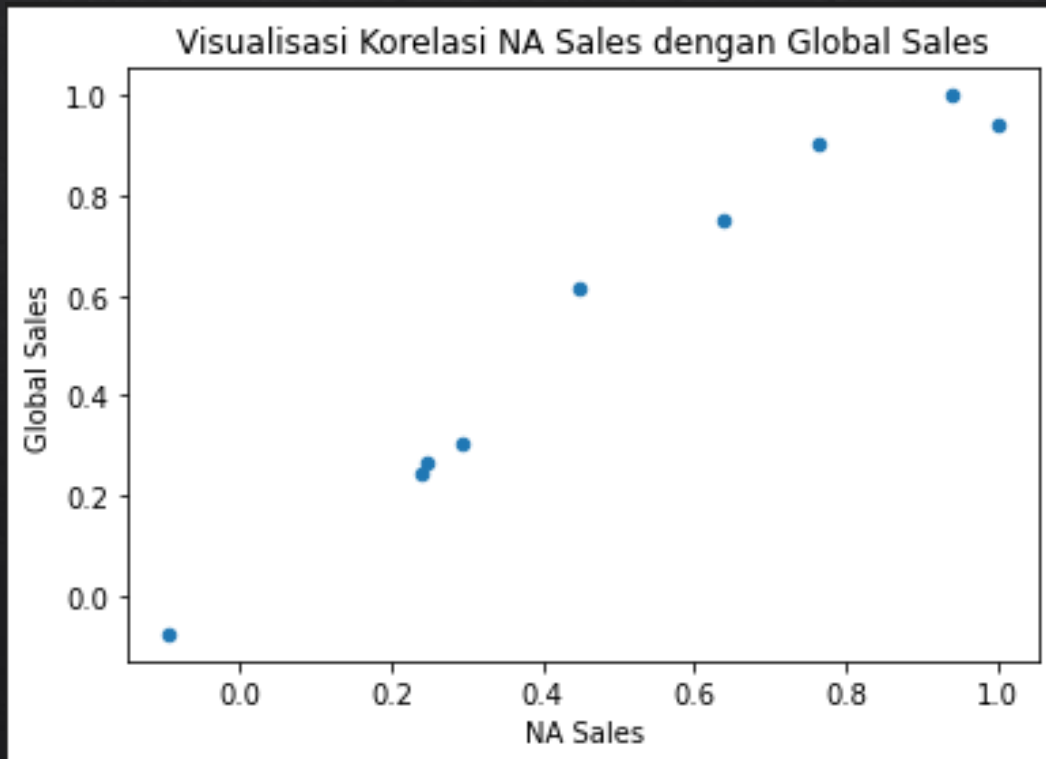
Korelasi antara semua atribut kuantitatif yang ada dalam data

```
print("Korelasi antara semua atribut/data kuantitatif adalah")  
#menampilkan korelasi  
data.corr(method="pearson")
```

|                 | Year_of_Release | NA_Sales  | EU_Sales | JP_Sales  | Other_Sales | Global_Sales | Critic_Score | Critic_Count | User_Count |
|-----------------|-----------------|-----------|----------|-----------|-------------|--------------|--------------|--------------|------------|
| Year_of_Release | 1.000000        | -0.092562 | 0.003842 | -0.168386 | 0.037700    | -0.076433    | 0.011411     | 0.223407     | 0.175339   |
| NA_Sales        | -0.092562       | 1.000000  | 0.765336 | 0.449598  | 0.638654    | 0.941010     | 0.240755     | 0.295413     | 0.246429   |
| EU_Sales        | 0.003842        | 0.765336  | 1.000000 | 0.435068  | 0.722796    | 0.901239     | 0.220752     | 0.277533     | 0.283360   |
| JP_Sales        | -0.168386       | 0.449598  | 0.435068 | 1.000000  | 0.291096    | 0.612300     | 0.152593     | 0.180219     | 0.075638   |
| Other_Sales     | 0.037700        | 0.638654  | 0.722796 | 0.291096  | 1.000000    | 0.749242     | 0.198554     | 0.251639     | 0.238982   |
| Global_Sales    | -0.076433       | 0.941010  | 0.901239 | 0.612300  | 0.749242    | 1.000000     | 0.245471     | 0.303571     | 0.265012   |
| Critic_Score    | 0.011411        | 0.240755  | 0.220752 | 0.152593  | 0.198554    | 0.245471     | 1.000000     | 0.425504     | 0.264376   |
| Critic_Count    | 0.223407        | 0.295413  | 0.277533 | 0.180219  | 0.251639    | 0.303571     | 0.425504     | 1.000000     | 0.362334   |
| User_Count      | 0.175339        | 0.246429  | 0.283360 | 0.075638  | 0.238982    | 0.265012     | 0.264376     | 0.362334     | 1.000000   |

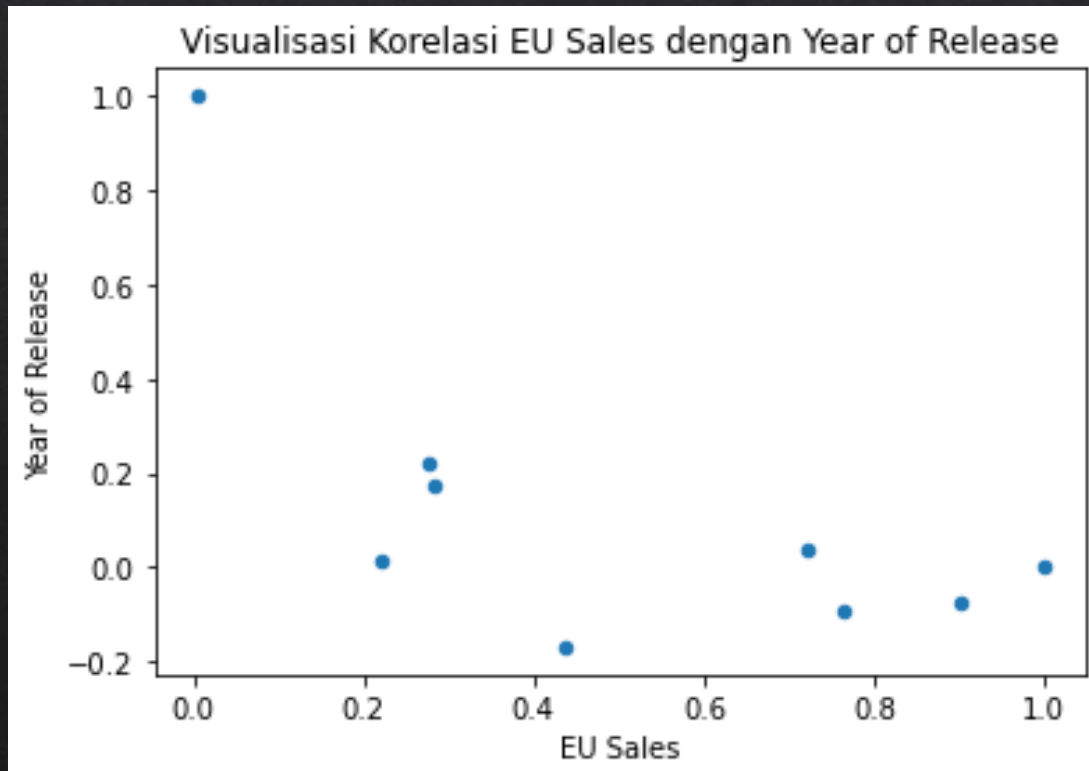
## Visualisasi korelasi antara penjualan Amerika Utara dengan penjualan global

```
#mencari data korelasi
correlation = data.corr(method="pearson")
#melakukan plotting
correlation.plot(kind="scatter", x="NA_Sales", y="Global_Sales", title="Visualisasi Korelasi NA Sales dengan Global Sales")
plt.xlabel("NA Sales")
plt.ylabel("Global Sales")
plt.show()
```



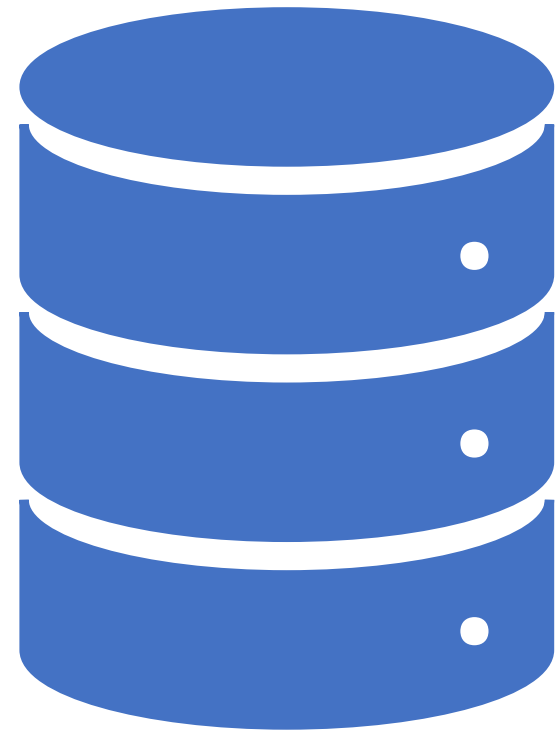
## Visualisasi korelasi antara penjualan Eropa dengan year of release

```
#mencari data korelasi
correlation = data.corr(method="pearson")
#melakukan plotting
correlation.plot(kind="scatter", x="EU_Sales", y="Year_of_Release", title="Visualisasi Korelasi EU Sales dengan Year of Release")
plt.xlabel("EU Sales")
plt.ylabel("Year of Release")
plt.show()
```



# Tugas 6

## Data cleansing





## 1. Data mentah

- Banyak data kosong pada dataframe

```
# Menghitung jumlah data bernilai Null atau kosong  
print("Data kosong : ", data.isnull().sum().sum())
```

✓ 5.6s

Data kosong : 46716

- Banyak data yang tidak logis pada dataframe

```
# menghitung jumlah data tidak logis  
not_logical_data = data.loc[  
    (data["Year_of_Release"] < 0) |  
    (data["NA_Sales"] < 0) |  
    (data["EU_Sales"] < 0) |  
    (data["JP_Sales"] < 0) |  
    (data["Other_Sales"] < 0) |  
    (data["Global_Sales"] < 0) |  
    (data["Critic_Score"] < 0) |  
    (data["Critic_Count"] < 0) |  
    (data["User_Count"] < 0)  
]  
|  
print('Data yang tidak logis :', len(not_logical_data))
```

✓ 0.5s

Data yang tidak logis : 0

## 2. Pembersihan Dataframe

- Menggunakan fungsi dropna()

```
#membersihkan data kosong  
data.dropna(inplace=True)
```

- Hasil dataframe baru:

```
# Menghitung jumlah data bernilai Null atau kosong  
print("Data kosong : ", data.isnull().sum().sum())
```

✓ 0.1s

Data kosong : 0

# KESIMPULAN

- data yang digunakan untuk menganalisis data merupakan data *video games* dengan penjualan terbanyak di dunia hingga tahun 2020, dengan
  - format csv,
  - dimensi data 16.720 baris dan 16 kolom, serta
  - ukuran 1.581 KB
  - Sumber <https://www.kaggle.com/gregorut/videogamesales>.
- Karakteristik data yang diambil terdiri dari *Categorical-nominal*, *Categorical-ordinal*, *Quantitative-continues*, dan *Quantitative-discrete*.
- Dari hasil analisis data didapatkan rata-rata, standar deviasi, persentile (10%, 25%, 50%, 75%, 90%), nilai maksimum dan minimum serta distribusi frekuensi nilai pada data.
- Dari hasil visualisasi data, dapat diketahui perbandingan dari setiap kategori, perbandingan terhadap waktu, serta hierarki dan hubungan keseluruhan bagian. Kami juga mendapatkan korelasi antar data sehingga dapat diketahui apakah data-data berkorelasi lurus, terbalik, ataupun tidak berkorelasi.
- dikarenakan banyaknya data yang kosong dalam dataset kami, maka dilakukan datacleansing supaya informasi yang didapatkan tidak salah.
- Dalam suatu penelitian, perlu dilakukan analisis dan visualisasi data karena banyaknya data yang harus diolah.



THANK YOU

