



dibimbing.id

# FINAL REPORT

*Louis Oktovianus*

DSF40 – Data Science

# Tugas

- Carilah 1 artikel yang belum optimal dan ingin dioptimalkan pada blog, lalu tentukan usulan perbaikan pada artikel tersebut!
- Tulis ulang artikelnnya dengan bahasamu sendiri, mulai dari Judul, Headings, hingga Body Text, minimal 500 kata, tanpa bantuan AI!

LOUIS OKTOVIANUS  
FINAL REPORT 2025

# Wordpress

Link Wordpress:

<https://louisoktovianus31.wordpress.com/2025/06/23/panduan-lengkap-menulis-artikel-seo-friendly-untuk-blogger-pemula-2025/>

LOUIS OKTOVIANUS  
FINAL REPORT

engkap-menulis-artikel-seo-friendly-untuk-blogger-pemula-2025/

Design a site like this with WordPress.com [Get started](#)

## Panduan Lengkap Menulis Artikel SEO Friendly untuk Blogger Pemula (2025)

June 23, 2025



Pernah merasa artikel blogmu sudah bagus tapi tidak muncul di halaman pertama Google? Kamu tidak sendiri. Banyak blogger pemula mengalami hal yang sama karena belum memahami cara menulis artikel SEO friendly yang efektif. Artikel ini akan membahas langkah-langkah praktis untuk membuat konten blog yang ramah mesin pencari, mudah dipahami pembaca, dan tentunya meningkatkan peluang tampil di hasil pencarian Google.

### 1. Tentukan Kata Kunci Utama

Langkah pertama dalam menulis artikel SEO friendly adalah menentukan kata kunci utama yang relevan dengan topik yang ingin kamu bahas. Kata kunci ini adalah frasa yang sering dicari pengguna.

[Comment](#) [Reblog](#)

# Tugas Introduction to Python

- ✓ Buatlah sebuah program yang:
    1. Meminta input dari pengguna:
      - Nama mahasiswa
      - NIM mahasiswa
      - Nilai ujian (dalam angka, 0–100)
    2. Cek dan tampilkan tipe data dari setiap input.
    3. Gunakan if-elif-else untuk menentukan kategori nilai, dengan kriteria sebagai berikut:

Nilai	Kategori
85 – 100	A (Sangat Baik)
75 – 84	B (Baik)
60 – 74	C (Cukup)
40 – 59	D (Kurang)
< 40	E (Sangat Kurang)

```
==== Ekspektasi Output Jawaban : =====
Masukkan nama mahasiswa: Sarah
Masukkan NIM: 12345678
Masukkan nilai ujian (0-100): 81

Nama: Sarah (type: <class 'str'>)
NIM: 12345678 (type: <class 'str'>)
Nilai: 81 (type: <class 'int'>

Hasil Evaluasi:
Mahasiswa: Sarah (NIM: 12345678)
Nilai Ujian: 81
Kategori Nilai: B (Baik)
```

# LOUIS OKTOVIANUS

## FINAL REPORT 2025

# Jawaban

```
# Meminta input dari pengguna
nama = input("Masukkan nama mahasiswa: ")
nim = input("Masukkan NIM: ")
nilai_input = input("Masukkan nilai ujian (0-100): ")

# Cek dan tampilkan tipe data dari setiap input
print(f"\nNama: {nama} (type: {type(nama)})")
print(f"NIM: {nim} (type: {type(nim)})")

# Pastikan nilai ujian bertipe integer
try:
    nilai = int(nilai_input)
    print(f"Nilai: {nilai} (type: {type(nilai)})")

    # Menentukan kategori nilai
    if 85 <= nilai <= 100:
        kategori = "A (Sangat Baik)"
    elif 75 <= nilai <= 84:
        kategori = "B (Baik)"
    elif 60 <= nilai <= 74:
        kategori = "C (Cukup)"
    elif 40 <= nilai <= 59:
        kategori = "D (Kurang)"
    elif 0 <= nilai < 40:
        kategori = "E (Sangat Kurang)"
    else:
        kategori = "Nilai tidak valid (harus 0-100)"

    # Menampilkan hasil evaluasi
    print("\nHasil Evaluasi:")
    print(f"Mahasiswa: {nama} (NIM: {nim})")
    print(f"Nilai Ujian: {nilai}")
    print(f"Kategori Nilai: {kategori}")

except ValueError:
    print("Nilai ujian harus berupa angka antara 0-100.")
```

## INTRO TO PYTHON

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Dibimbing\\_template\\_introduction\\_python.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Dibimbing_template_introduction_python.ipynb)

# Tugas EDA

Rincian Tugas:

1. Carilah sebuah dataset (dari kaggle ataupun website lainnya)
2. Lakukan Exploratory Data Analysis pada dataset tersebut.
  - EDA yang dilakukan WAJIB melakukan Handling Missing Value dan Duplicate
  - Proses EDA yang lain bersifat opsional (e.g. *handling outlier, encoding*)
3. Buat portfolio dalam bentuk Power Point
4. Upload di LinkedIn dan tag dibimbing.id

L O U I S   O K T O V I A N U S

FINAL REPORT 2025

# DataSet

ANAND SHAW · UPDATED 6 MONTHS AGO

578

Code Download

## Netflix Movies and TV Shows

Listings of movies and tv shows on Netflix

Data Card Code (48) Discussion (2)

### About Dataset

**Don't forget to hit the upvote!**

About this Dataset: Netflix is one of the most popular media and video streaming platforms. They have over 8k+ movies or tv shows available on their platform, as of mid-2024, they have over 282 million Subscribers globally. This tabular dataset consists of listings of all the movies and tv shows available on Netflix, along with details such as - cast, directors, ratings, release year, duration more.

**Columns and Descriptions:**

- **show\_id:** Unique identifier for each show (s1, s2).
- **type:** Specifies whether the title is a "Movie" or "TV Show".
- **title:** The name of the Netflix title.
- **director:** The director of the title
- **cast:** The main actors involved in the title.
- **country:** The country where the title was produced.

**Usability** 10.00

**License** CC0: Public Domain

**Expected update frequency** Never

**Tags**

- Arts and Entertainment
- Movies and TV Shows
- Data Visualization
- Classification
- Exploratory Data Analysis

E D A

LINK : <https://www.kaggle.com/datasets/anandshaw2001/netflix-movies-and-tv-shows>

# Jawaban

```
# 1. Import libraries
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

EDA

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
# 2. Load dataset  
df = pd.read_csv('/content/netflix_titles.csv')
```

EDA

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
# 3. Lihat 5 data teratas  
df.head()
```

E D A

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
# 4. Cek informasi umum  
df.info()
```

E D A

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
# 5. Cek missing values
print("Jumlah missing value per kolom:")
print(df.isnull().sum())
```

E D A

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
# ===== Mengatasi Missing Value pada Dataset Netflix =====

# 1. Kolom 'director', 'cast', dan 'country' bertipe teks, dan tidak selalu wajib diisi
#   Solusi: Isi dengan 'Unknown' agar tetap informatif tanpa menghapus baris
df['director'] = df['director'].fillna('Unknown')
df['cast'] = df['cast'].fillna('Unknown')
df['country'] = df['country'].fillna('Unknown')

# 2. Kolom 'date_added' menyimpan tanggal kapan tayangan ditambahkan ke Netflix
#   Solusi: Diisi dengan nilai modus (tanggal yang paling sering muncul)
df['date_added'] = df['date_added'].fillna(df['date_added'].mode()[0])

# 3. Kolom 'rating' menyimpan klasifikasi umur (seperti TV-MA, PG, dll)
#   Solusi: Karena rating bersifat kategorikal, diisi dengan modus
df['rating'] = df['rating'].fillna(df['rating'].mode()[0])

# 4. Kolom 'duration' bisa berupa "90 min" (untuk film) atau "1 Season" (untuk TV Show)
#   Solusi: Diisi dengan modus agar tetap relevan dan tidak bias
df['duration'] = df['duration'].fillna(df['duration'].mode()[0])
```

E D A

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
# Cek ulang untuk memastikan semua sudah di-handle  
print("Cek missing value setelah pembersihan:")  
print(df.isnull().sum())
```

E D A

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
# 6. Cek duplicate
print("\nJumlah data duplikat:")
print(df.duplicated().sum())
print("")
print(df.duplicated())
```

E D A

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
# 7. Drop duplicate  
df = df.drop_duplicates()
```

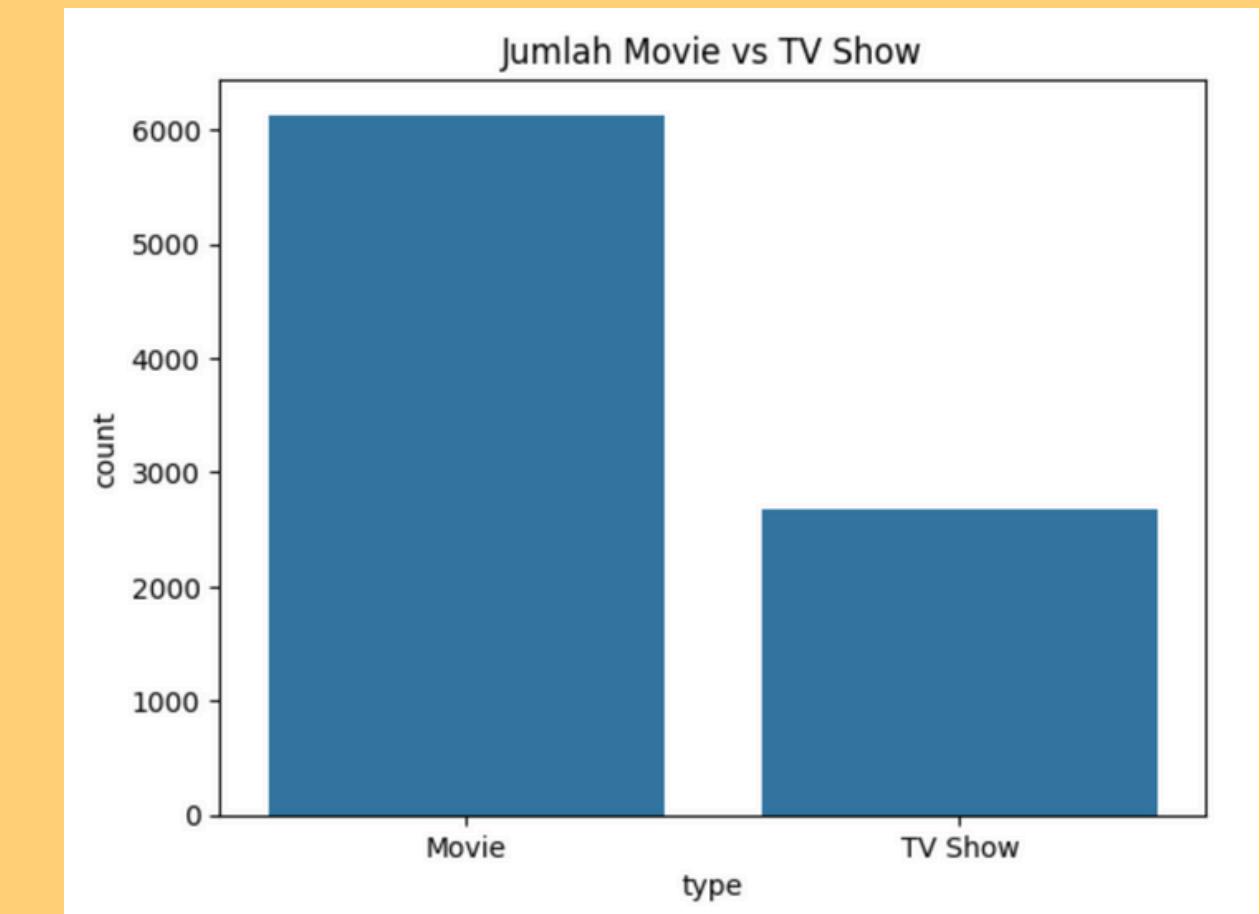
E D A

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
# 8. Visualisasi jumlah konten berdasarkan jenis  
sns.countplot(data=df, x='type')  
plt.title('Jumlah Movie vs TV Show')  
plt.show()
```



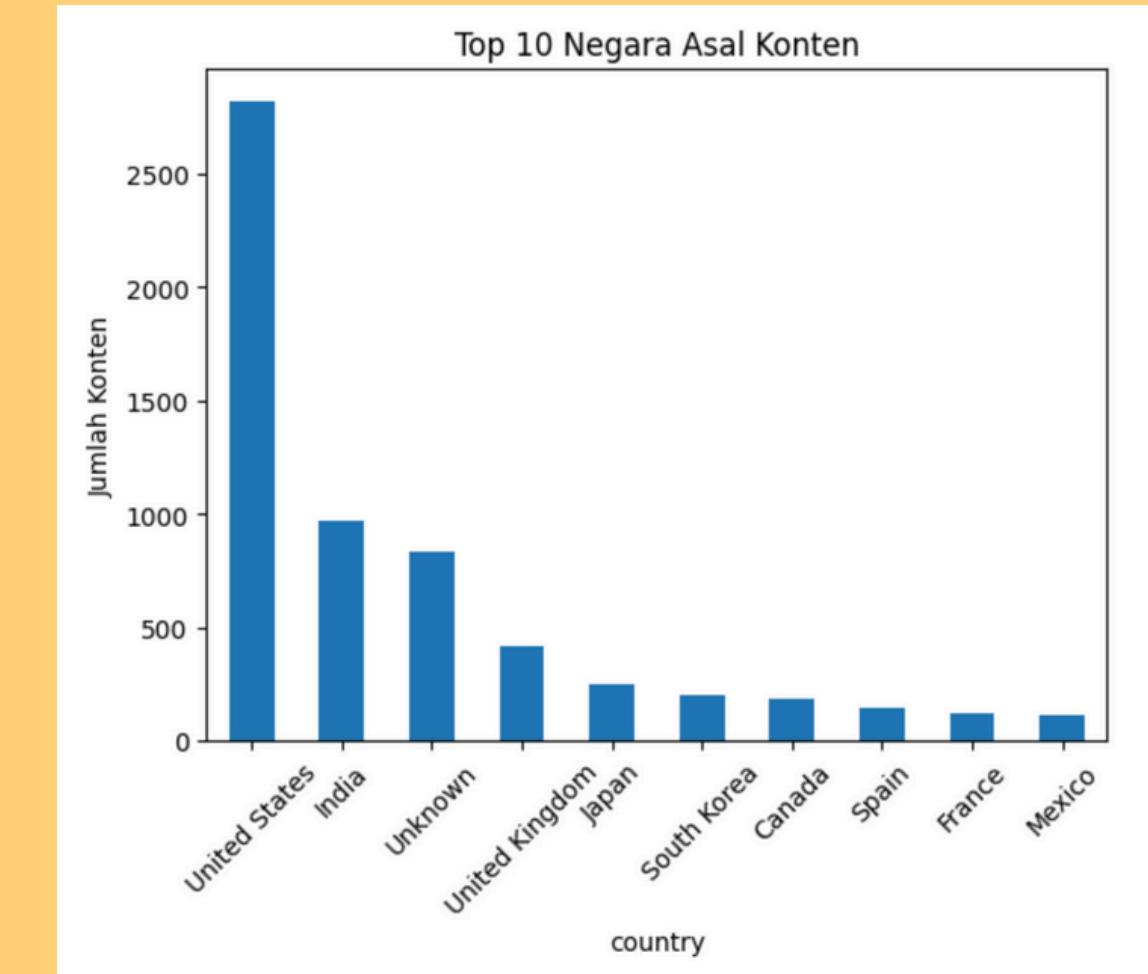
E D A

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
# 9. Visualisasi top 10 negara dengan konten terbanyak
top_countries = df['country'].value_counts().head(10)
top_countries.plot(kind='bar')
plt.title('Top 10 Negara Asal Konten')
plt.ylabel('Jumlah Konten')
plt.xticks(rotation=45)
plt.show()
```



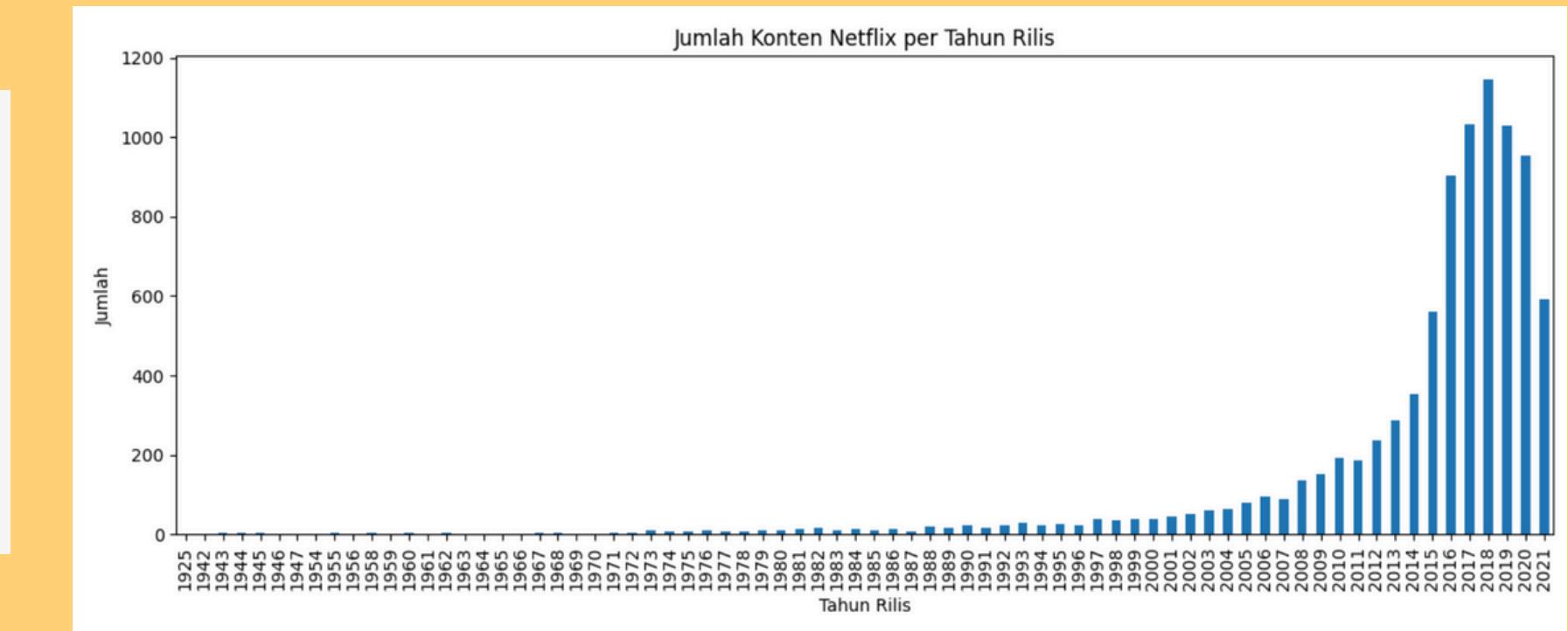
EDA

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
# 10. Jumlah konten tiap tahun rilis
plt.figure(figsize=(12,5))
df['release_year'].value_counts().sort_index().plot(kind='bar')
plt.title('Jumlah Konten Netflix per Tahun Rilis')
plt.xlabel('Tahun Rilis')
plt.ylabel('Jumlah')
plt.tight_layout()
plt.show()
```



EDA

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
#11. Distribusi Genre – Visualisasi Top 10 Genre

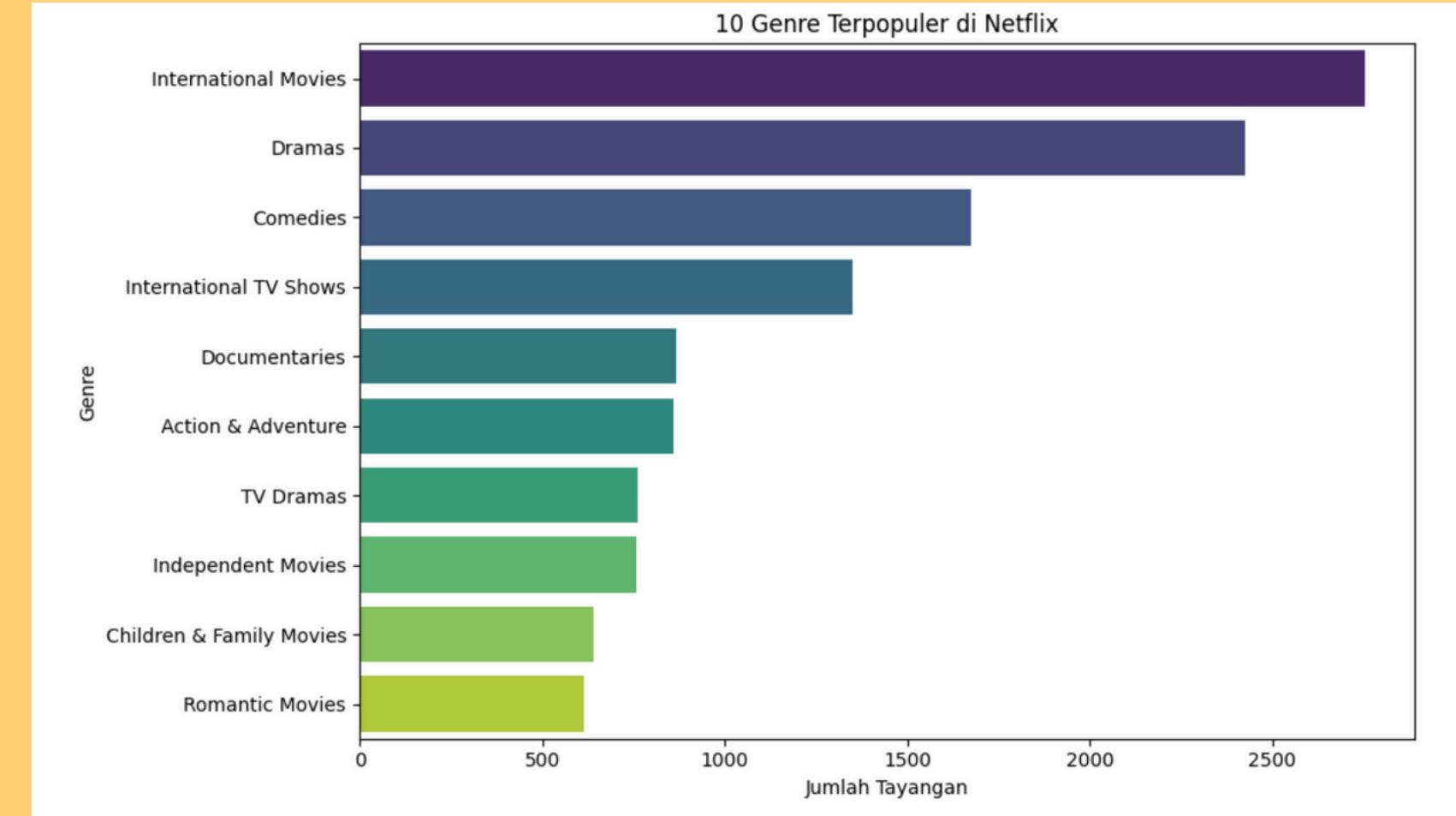
from collections import Counter
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

# Pecah string genre jadi list
all_genres = df['listed_in'].dropna().str.split(',', ',')

# Hitung total genre
genre_count = Counter([genre for sublist in all_genres for genre in sublist])

# Ubah ke DataFrame
genre_df = pd.DataFrame(genre_count.items(), columns=['Genre', 'Jumlah']).sort_values(by='Jumlah', ascending=False)

# Barplot top 10 genre (fix warning)
plt.figure(figsize=(10, 6))
sns.barplot(
    data=genre_df.head(10),
    x='Jumlah',
    y='Genre',
    hue=None,
    legend=False,
    palette='viridis'
)
plt.title('10 Genre Terpopuler di Netflix')
plt.xlabel('Jumlah Tayangan')
plt.ylabel('Genre')
plt.tight_layout()
plt.show()
```



EDA

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
#12. Word Cloud untuk Judul Film / TV Show

from wordcloud import WordCloud

# Gabungkan semua judul menjadi satu string
text = ' '.join(df['title'].dropna())

# Buat word cloud
wordcloud = WordCloud(width=1000, height=500, background_color='black', colormap='Set2').generate(text)

# Tampilkan word cloud
plt.figure(figsize=(12, 6))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.title('Kata yang Sering Muncul di Judul Netflix')
plt.show()
```



EDA

## LINK :

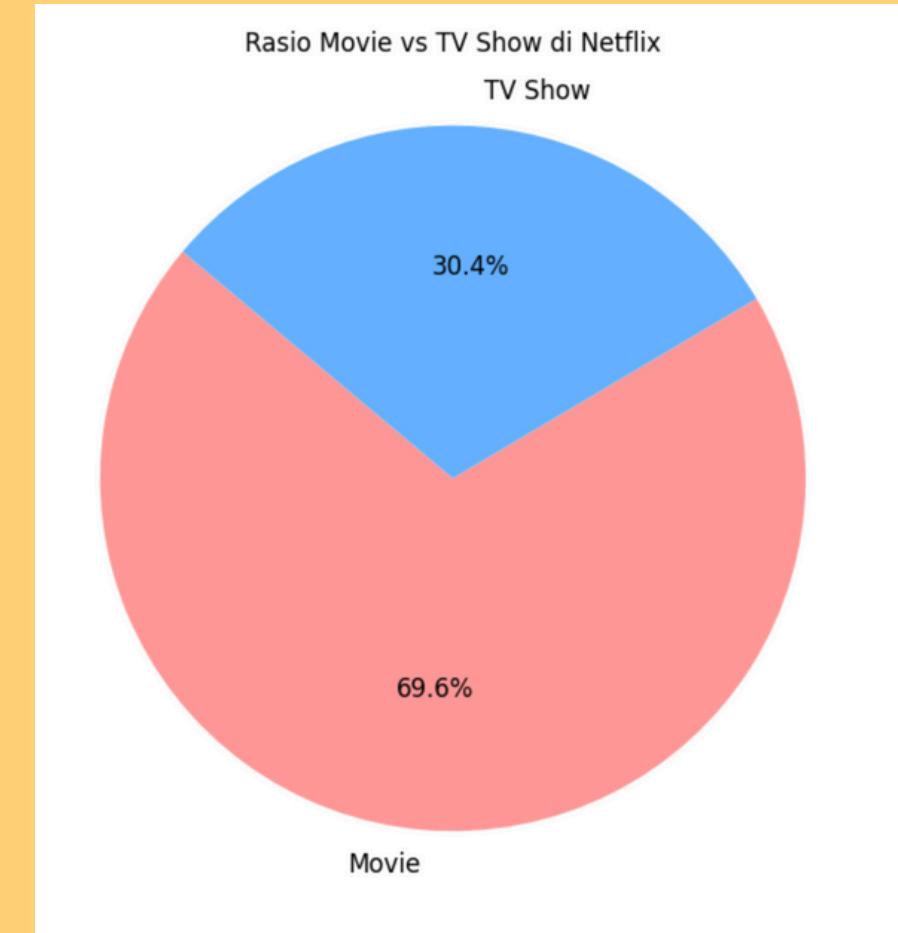
[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)

# Jawaban

```
#13. Rasio Movie vs TV Show – Pie Chart

# Ambil data
labels = df['type'].value_counts().index
sizes = df['type'].value_counts().values
colors = ['#ff9999', '#66b3ff']

# Pie chart
plt.figure(figsize=(6, 6))
plt.pie(sizes, labels=labels, autopct='%.1f%%', colors=colors, startangle=140, textprops={'fontsize': 12})
plt.title('Rasio Movie vs TV Show di Netflix', pad=20)
plt.axis('equal') # Biar pie-nya lingkaran sempurna
plt.tight_layout()
plt.show()
```



E D A

LINK :

[https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing\\_Final\\_DSF\\_EDA.ipynb](https://github.com/LouisOktovianus31/DSF40-DataScience/blob/main/Final%20Assignment/Dibimbing_Final_DSF_EDA.ipynb)