DATA AND ARTIFICIAL INTELLIGENCE

CYBER SHUJAA PROGRAM

WEEK 2 ASSIGNMENT

NAME: PAULINE KUNGU

PROJECT: DATA WRANGLING PROJECT

DATE: MAY 26, 2025

Table of Contents

DATA AND ARTIFICIAL INTELLIGENCE 1

CYBER SHUJAA PROGRAM 1

WEEK 2 ASSIGNMENT 1

One: Introduction 3

1.1 Overview 3

1.2 Objectives 3

Two: Task Completion Steps 3

2.1 Data Loading and Exploration 3

.... 6

2.2 Data Cleaning 7

2.3 Data Structuring and Transformation 8

2.4 Data Enrichment 9

2.5 Data Validation 10

2.6 Data Exportation 11

Three: Conclusion 13

Link to Notebook 13

One: Introduction

1.1 Overview

Data wrangling is the process of converting raw data into desired and usable format, essentially preparing the data for mode training and analysis. This systematic approach ensures that the data scientist can extract meaningful insight and build robust machine learning models from clean, well-structured datasets. This project focuses on cleaning and preparing the Netflix Movies and TV Shows dataset from kaggle.

1.2 Objectives

The objectives of the assignment are to:

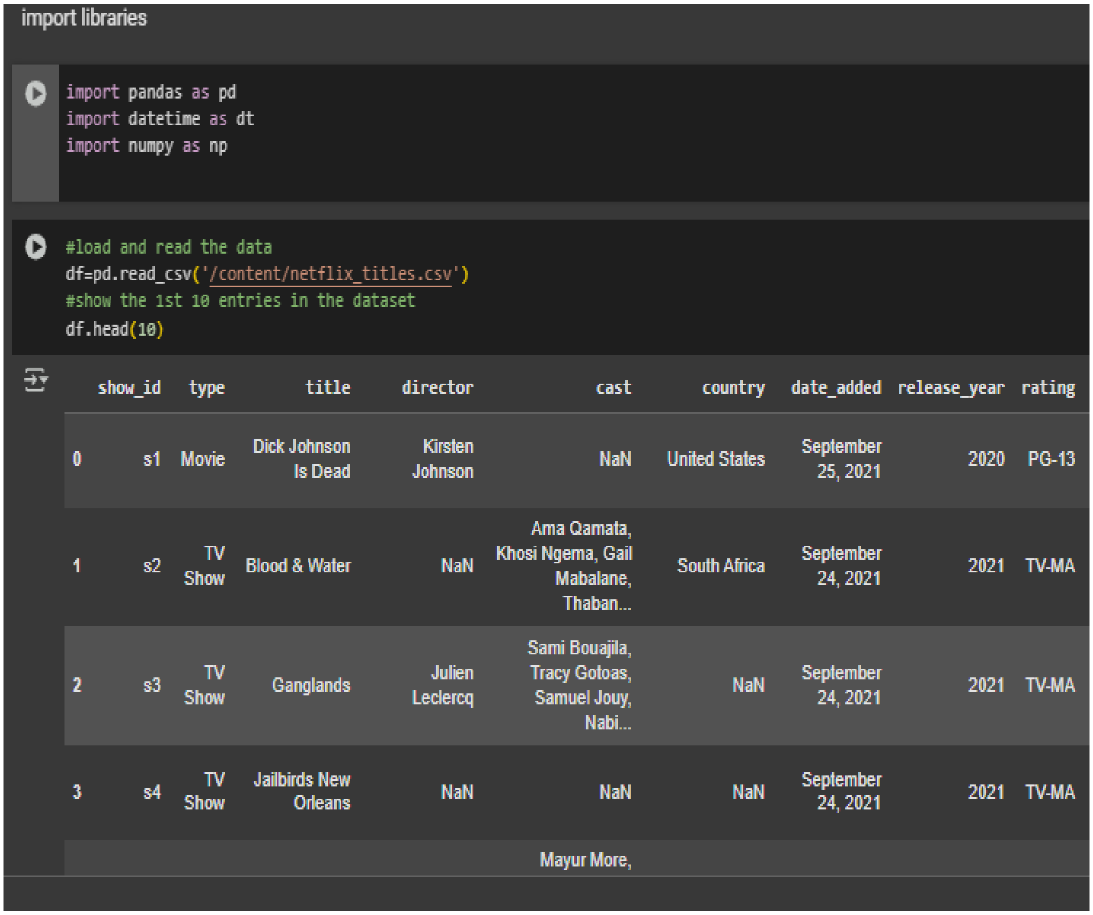
1. Load the Netflix dataset from a CSV file and explore its structure using pandas.
2. Perform data discovery to assess data types, missing values, and quality issues.
3. Clean the dataset by handling duplicates, missing values, and formatting inconsistencies.
4. Transform and enrich the dataset using techniques like filtering, sorting, grouping, and feature extraction.
5. Validate the final dataset by checking consistency, completeness, and logical accuracy.
6. Export the final cleaned dataset to a .csv file ready for analysis or visualization.

Two: Task Completion Steps

2.1 Data Loading and Exploration

Using the pandas python library functions and attributes, load the Netflix dataset which I downloaded from kaggle as CSV file and perform comprehensive exploration. Some key information found are:

- dataset contains 8807 entries(rows) and 12 columns
- there were missing value
- data types include objects and integer

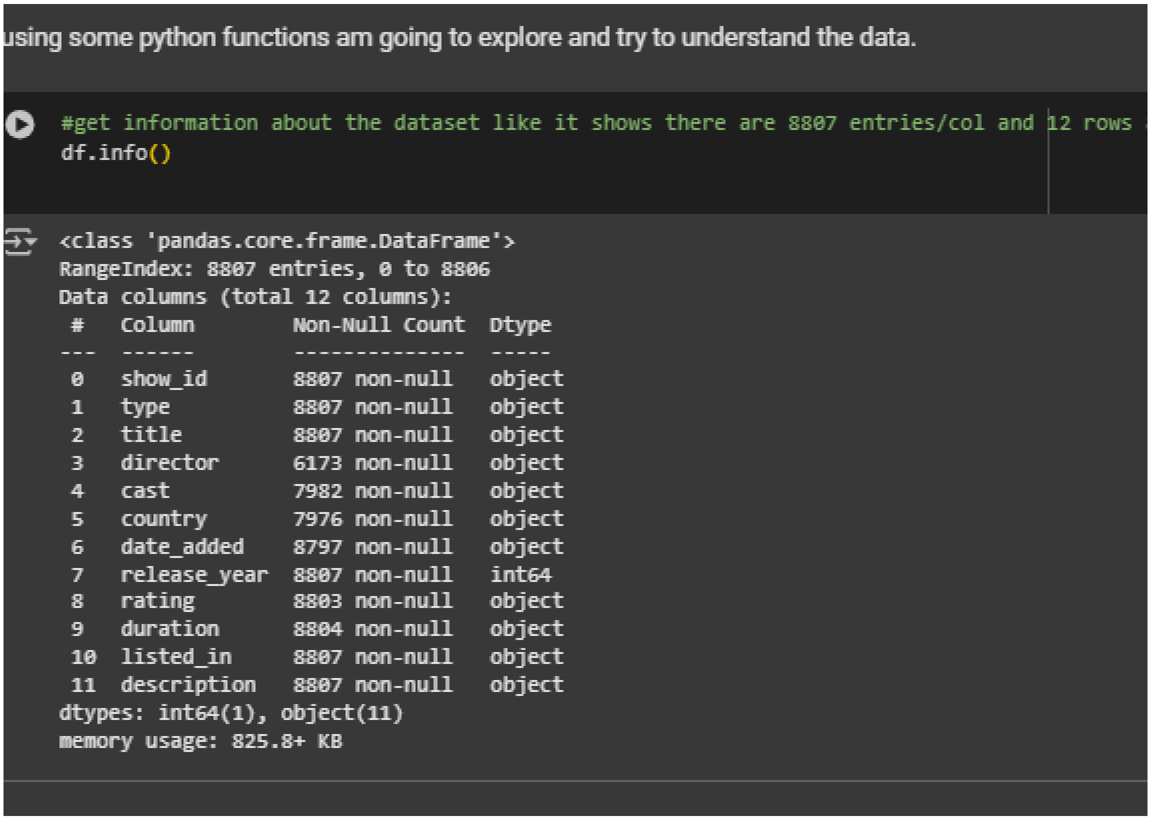


import libraries

```
import pandas as pd
import datetime as dt
import numpy as np
```

```
#load and read the data
df=pd.read_csv('/content/netflix_titles.csv')
#show the 1st 10 entries in the dataset
df.head(10)
```

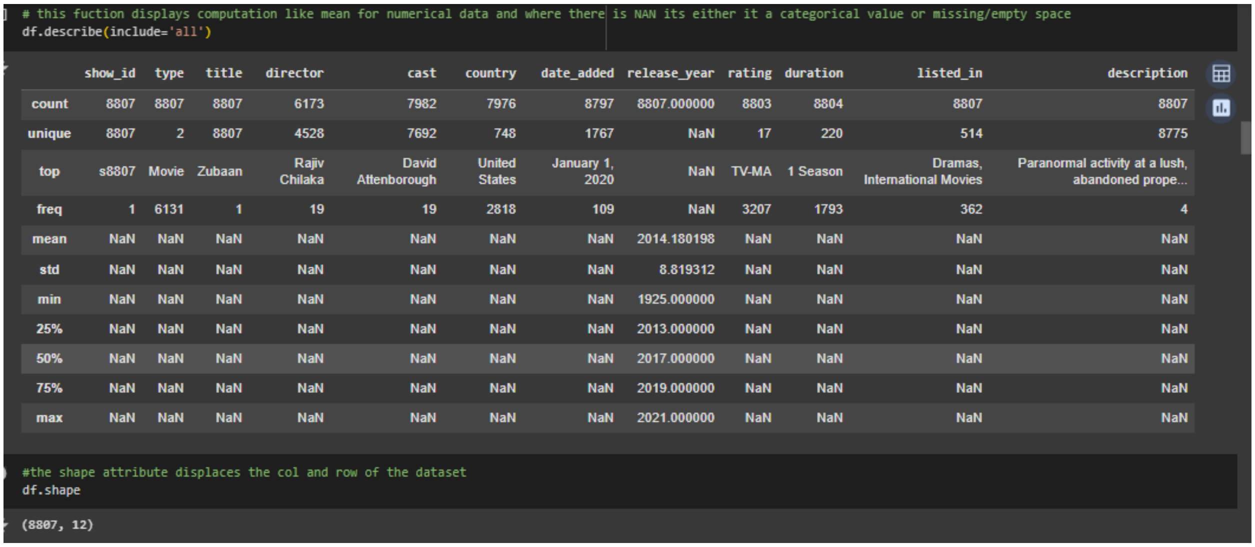
	show_id	type	title	director	cast	country	date_added	release_year	rating
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA
					Mayur More,				



using some python functions am going to explore and try to understand the data.

```
#get information about the dataset like it shows there are 8807 entries/col and 12 rows
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8807 entries, 0 to 8806
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   show_id       8807 non-null   object
 1   type          8807 non-null   object
 2   title         8807 non-null   object
 3   director      6173 non-null   object
 4   cast          7982 non-null   object
 5   country       7976 non-null   object
 6   date_added    8797 non-null   object
 7   release_year  8807 non-null   int64 
 8   rating        8803 non-null   object
 9   duration      8804 non-null   object
 10  listed_in     8807 non-null   object
 11  description   8807 non-null   object
dtypes: int64(1), object(11)
memory usage: 825.8+ KB
```

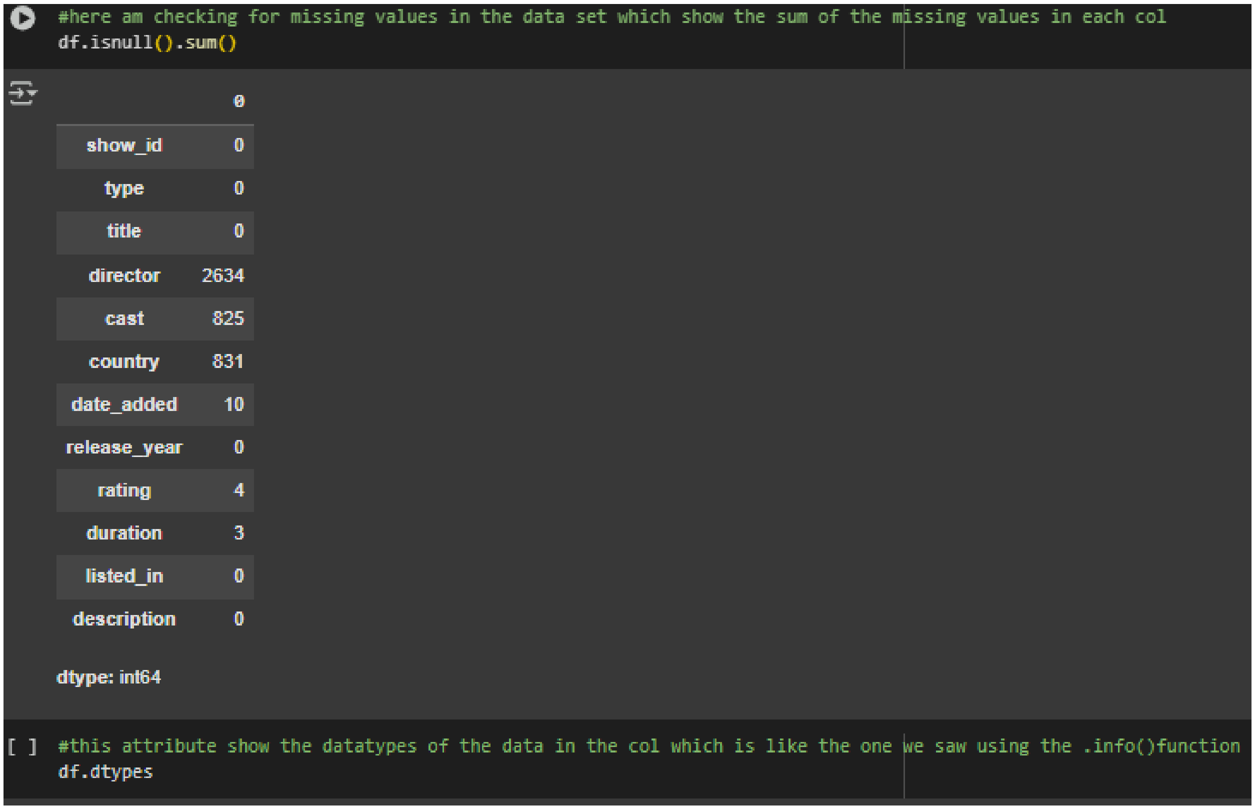


```
# this fuction displays computation like mean for numerical data and where there is NAN its either it a categorical value or missing/empty space
df.describe(include='all')
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
count	8807	8807	8807	6173	7982	7976	8797	8807.000000	8803	8804	8807	8807
unique	8807	2	8807	4528	7692	748	1767	NaN	17	220	514	8775
top	s8807	Movie	Zubaan	Rajiv Chilaka	David Attenborough	United States	January 1, 2020	NaN	TV-MA	1 Season	Dramas, International Movies	Paranormal activity at a lush, abandoned prope...
freq	1	6131	1	19	19	2818	109	NaN	3207	1793	362	4
mean	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2014.180198	NaN	NaN	NaN	NaN
std	NaN	NaN	NaN	NaN	NaN	NaN	NaN	8.819312	NaN	NaN	NaN	NaN
min	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1925.000000	NaN	NaN	NaN	NaN
25%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2013.000000	NaN	NaN	NaN	NaN
50%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2017.000000	NaN	NaN	NaN	NaN
75%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2019.000000	NaN	NaN	NaN	NaN
max	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2021.000000	NaN	NaN	NaN	NaN

```
#the shape attribute displaces the col and row of the dataset
df.shape
```

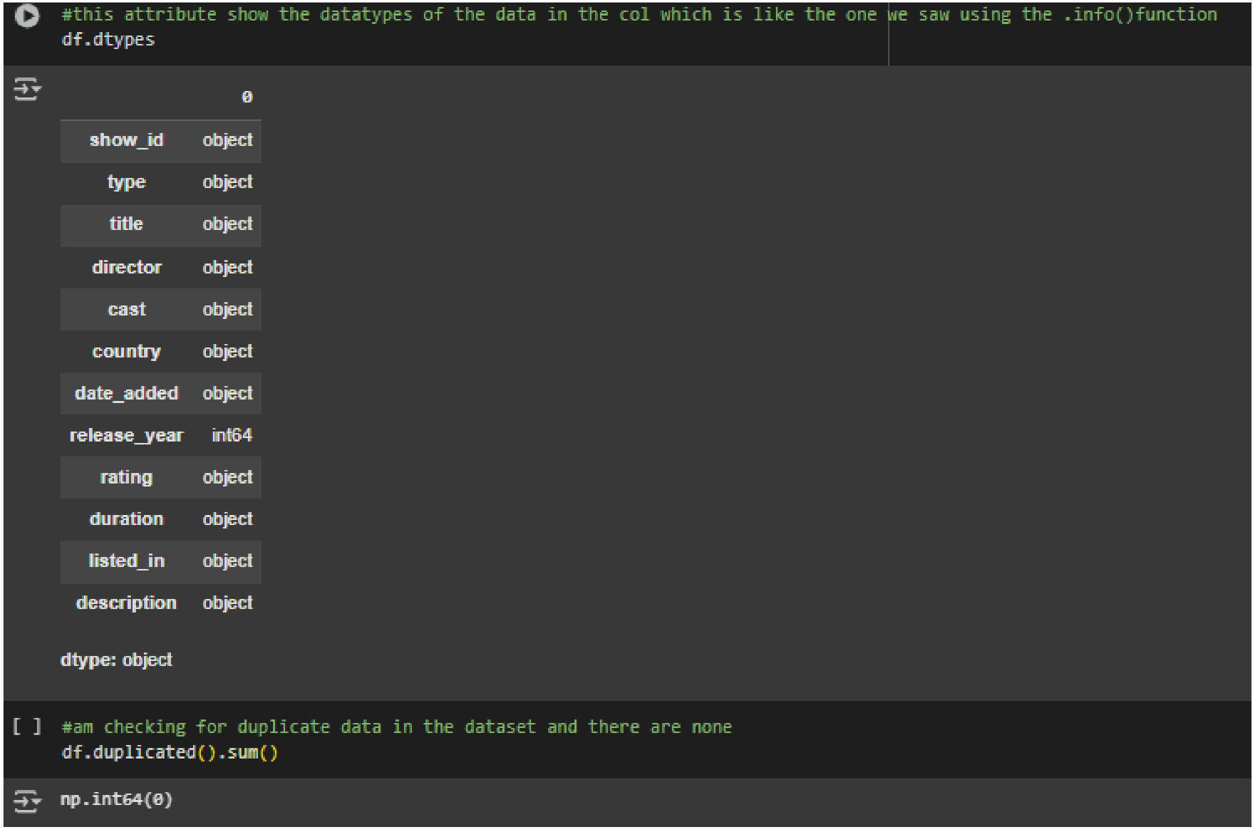
```
(8807, 12)
```



```
#here am checking for missing values in the data set which show the sum of the missing values in each col
df.isnull().sum()
```

```
                 0
     show_id     0
        type     0
       title     0
    director  2634
        cast   825
     country   831
  date_added    10
release_year     0
      rating     4
    duration     3
   listed_in     0
 description     0

dtype: int64
```

```
[ ] #this attribute show the datatypes of the data in the col which is like the one we saw using the .info()function
    df.dtypes
```



```
#this attribute show the datatypes of the data in the col which is like the one we saw using the .info()function
df.dtypes
```

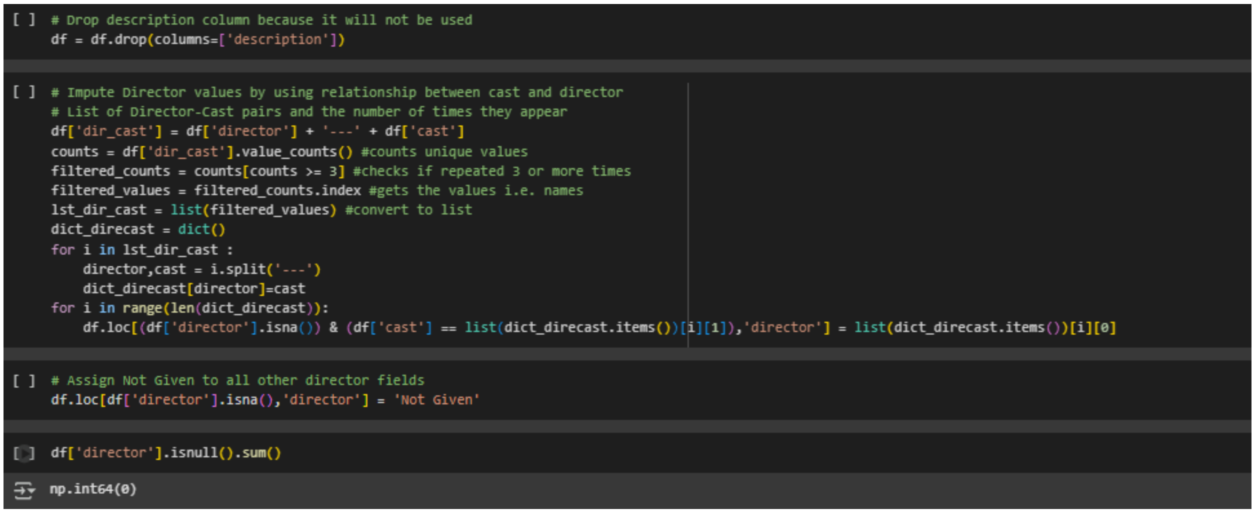
```
                   0
     show_id  object
        type  object
       title  object
    director  object
        cast  object
     country  object
  date_added  object
release_year   int64
      rating  object
    duration  object
   listed_in  object
 description  object

dtype: object
```

```
[ ] #am checking for duplicate data in the dataset and there are none
    df.duplicated().sum()
```

```
np.int64(0)
```

2.2 Data Cleaning

Clean the dataset to remove duplicates, missing values and inconsistencies. The dataset had no duplicates but had a number of missing values in the director, cast, country, date_added, rating and duration columns. Using the relationship between the director and cast I was able to fill in some of directors missing values and replaced the rest with “not given”. I did the same for country and directors while I droped some columns like description because it will not be used, dir_cast, show_id which I think might mislead during model training to presume feature importances and other missing values that were not that many. Alternatively you can use mean, medium and mode to fill the missing values.



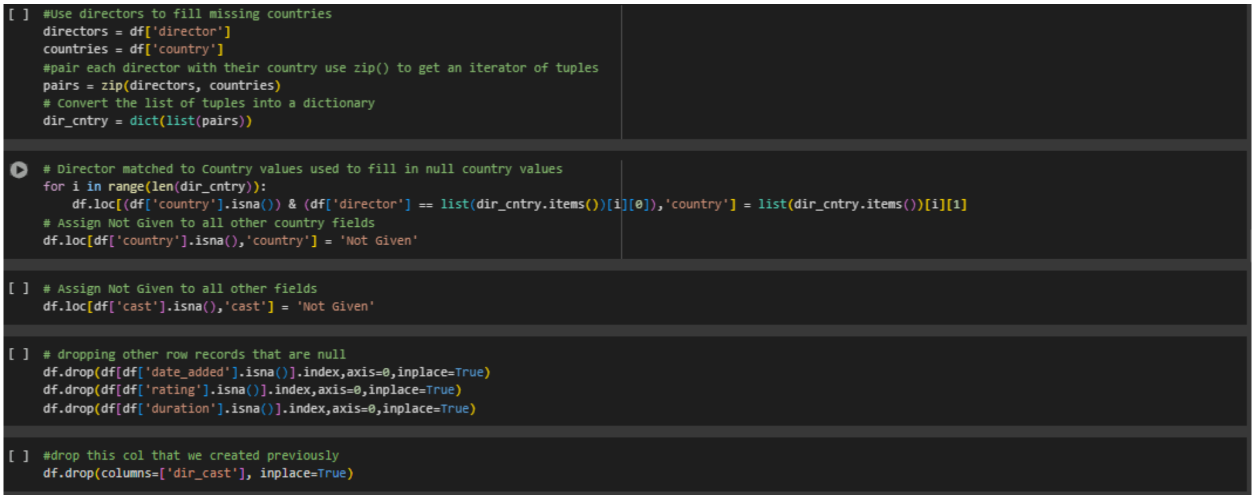
```
# Drop description column because it will not be used
df = df.drop(columns=['description'])
```

```
# Impute Director values by using relationship between cast and director
# List of Director-Cast pairs and the number of times they appear
df['dir_cast'] = df['director'] + '---' + df['cast']
counts = df['dir_cast'].value_counts() #counts unique values
filtered_counts = counts[counts >= 3] #checks if repeated 3 or more times
filtered_values = filtered_counts.index #gets the values i.e. names
lst_dir_cast = list(filtered_values) #convert to list
dict_direcast = dict()
for i in lst_dir_cast :
    director,cast = i.split('---')
    dict_direcast[director]=cast
for i in range(len(dict_direcast)):
    df.loc[(df['director'].isna()) & (df['cast'] == list(dict_direcast.items())[i][1]),'director'] = list(dict_direcast.items())[i][0]
```

```
# Assign Not Given to all other director fields
df.loc[df['director'].isna(),'director'] = 'Not Given'
```

```
df['director'].isnull().sum()
```

```
np.int64(0)
```



```
#Use directors to fill missing countries
directors = df['director']
countries = df['country']
#pair each director with their country use zip() to get an iterator of tuples
pairs = zip(directors, countries)
# Convert the list of tuples into a dictionary
dir_cntry = dict(list(pairs))
```

```
# Director matched to Country values used to fill in null country values
for i in range(len(dir_cntry)):
    df.loc[(df['country'].isna()) & (df['director'] == list(dir_cntry.items())[i][0]),'country'] = list(dir_cntry.items())[i][1]
# Assign Not Given to all other country fields
df.loc[df['country'].isna(),'country'] = 'Not Given'
```

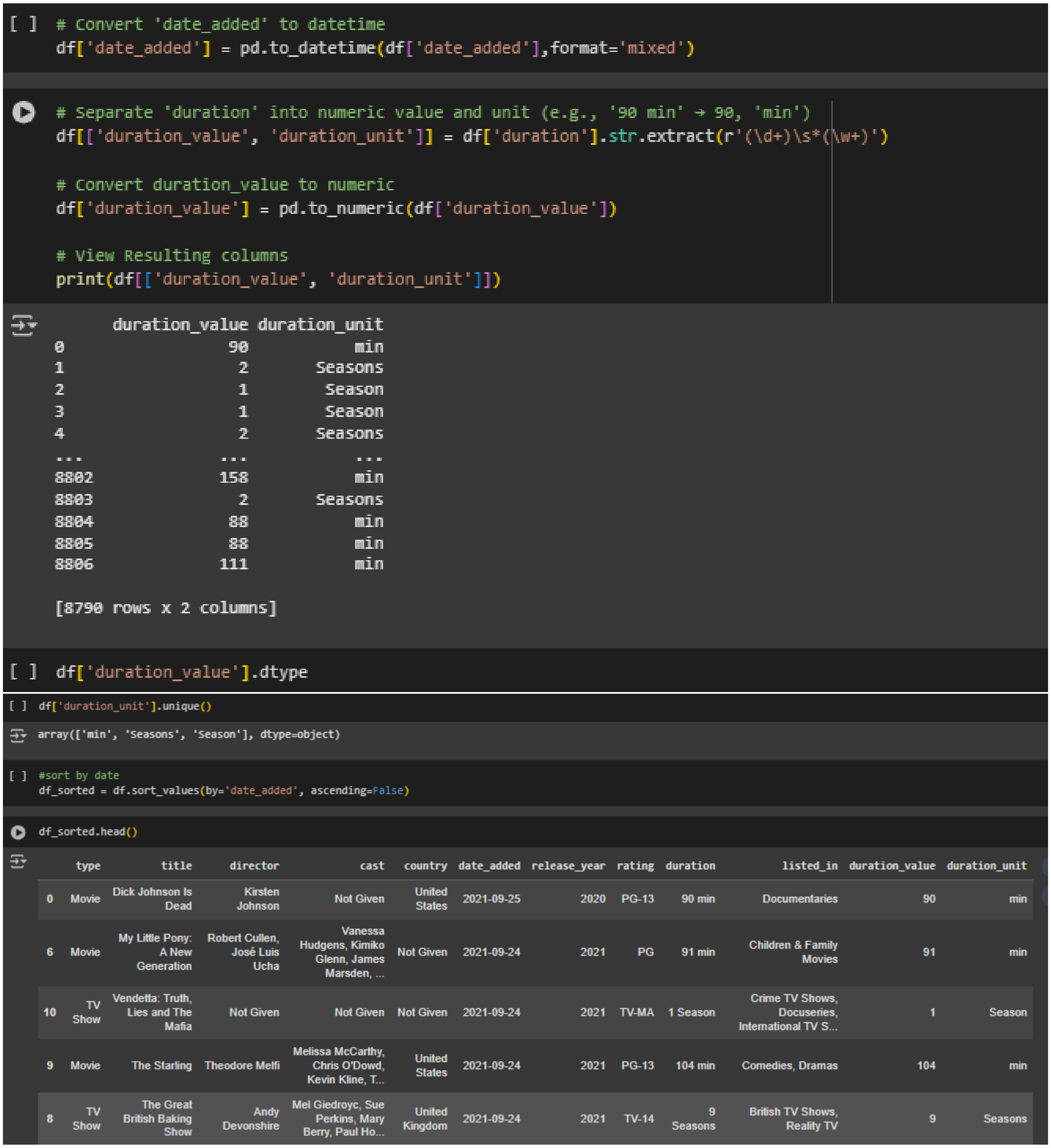
```
# Assign Not Given to all other fields
df.loc[df['cast'].isna(),'cast'] = 'Not Given'
```

```
# dropping other row records that are null
df.drop(df[df['date_added'].isna()].index,axis=0,inplace=True)
df.drop(df[df['rating'].isna()].index,axis=0,inplace=True)
df.drop(df[df['duration'].isna()].index,axis=0,inplace=True)
```

```
#drop this col that we created previously
df.drop(columns=['dir_cast'], inplace=True)
```

2.3 Data Structuring and Transformation

Here structure data with proper data types and extract meaningful components e.g. converting the date column to datetime with techniques like normalization, standardization and feature extraction



```
[ ]  # Convert 'date_added' to datetime
     df['date_added'] = pd.to_datetime(df['date_added'],format='mixed')
```

```
# Separate 'duration' into numeric value and unit (e.g., '90 min' → 90, 'min')
df[['duration_value', 'duration_unit']] = df['duration'].str.extract(r'(\d+)\s*(\w+)')

# Convert duration_value to numeric
df['duration_value'] = pd.to_numeric(df['duration_value'])

# View Resulting columns
print(df[['duration_value', 'duration_unit']])
```

```
      duration_value duration_unit
0                 90           min
1                  2       Seasons
2                  1        Season
3                  1        Season
4                  2       Seasons
...              ...           ...
8802             158           min
8803               2       Seasons
8804              88           min
8805              88           min
8806             111           min

[8790 rows x 2 columns]
```

```
[ ]  df['duration_value'].dtype
```

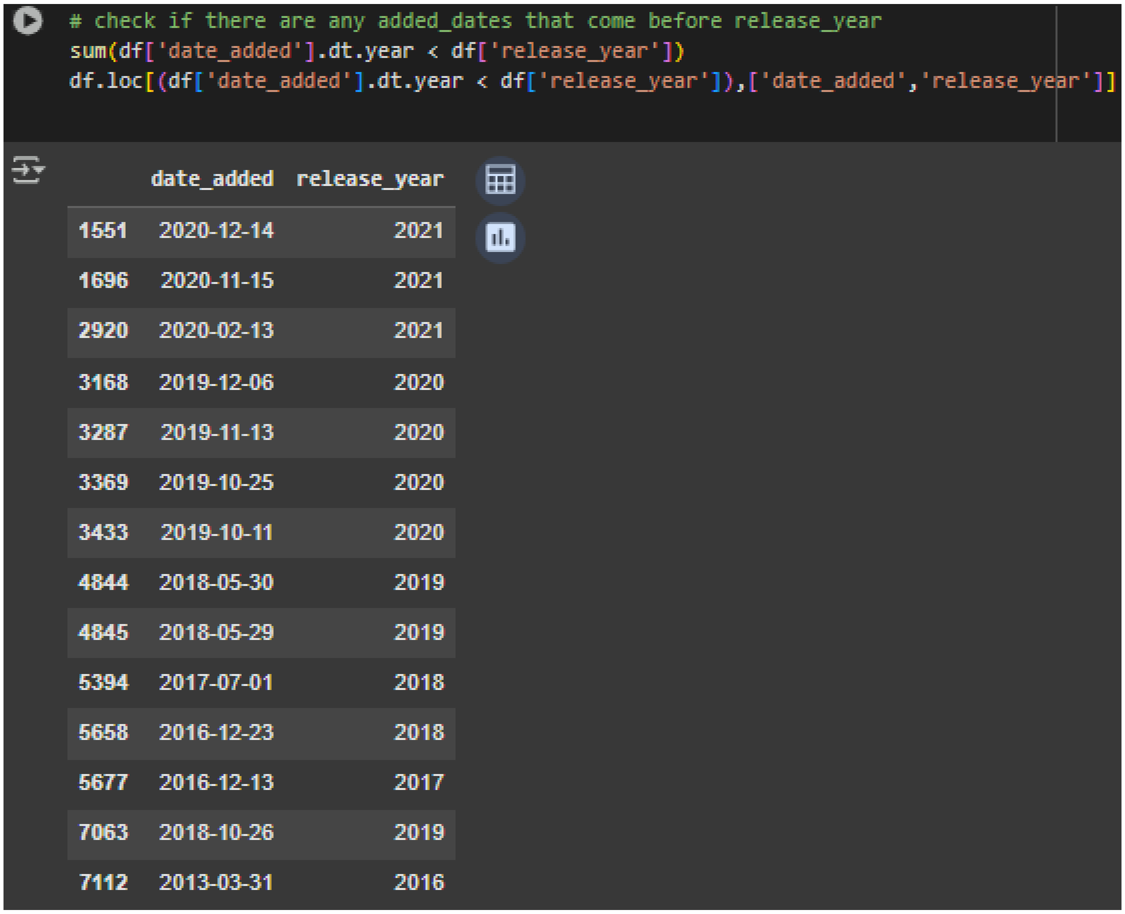
```
[ ]  df['duration_unit'].unique()
```

```
array(['min', 'Seasons', 'Season'], dtype=object)
```

```
[ ]  #sort by date
     df_sorted = df.sort_values(by='date_added', ascending=False)
```

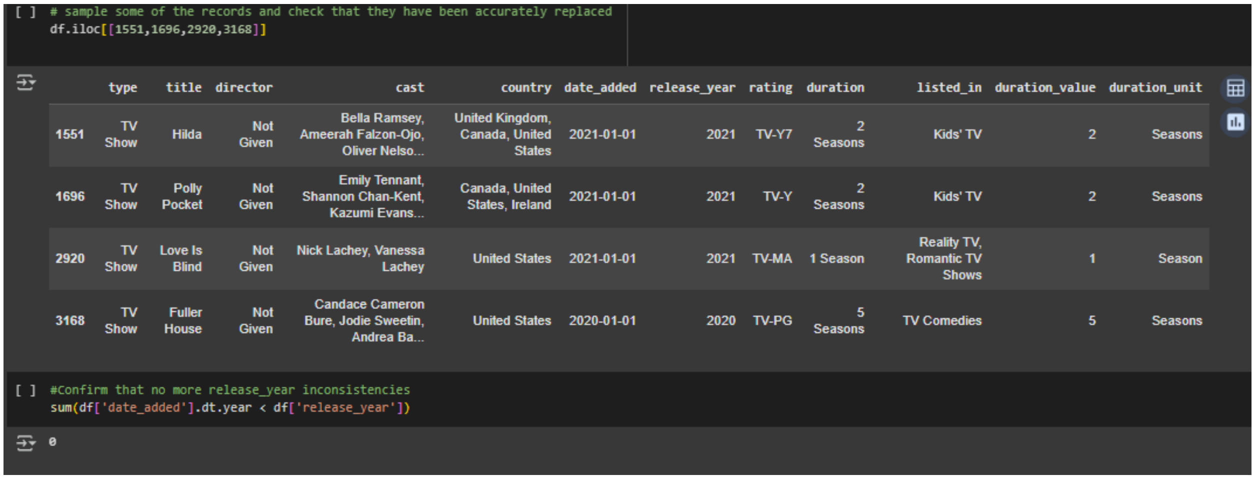
```
df_sorted.head()
```

	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	duration_value	duration_unit
0	Movie	Dick Johnson Is Dead	Kirsten Johnson	Not Given	United States	2021-09-25	2020	PG-13	90 min	Documentaries	90	min
6	Movie	My Little Pony: A New Generation	Robert Cullen, José Luis Ucha	Vanessa Hudgens, Kimiko Glenn, James Marsden, ...	Not Given	2021-09-24	2021	PG	91 min	Children & Family Movies	91	min
10	TV Show	Vendetta: Truth, Lies and The Mafia	Not Given	Not Given	Not Given	2021-09-24	2021	TV-MA	1 Season	Crime TV Shows, Docuseries, International TV S...	1	Season
9	Movie	The Starling	Theodore Melfi	Melissa McCarthy, Chris O'Dowd, Kevin Kline, T...	United States	2021-09-24	2021	PG-13	104 min	Comedies, Dramas	104	min
8	TV Show	The Great British Baking Show	Andy Devonshire	Mel Giedroyc, Sue Perkins, Mary Berry, Paul Ho...	United Kingdom	2021-09-24	2021	TV-14	9 Seasons	British TV Shows, Reality TV	9	Seasons



```python
# check if there are any added_dates that come before release_year
sum(df['date_added'].dt.year < df['release_year'])
df.loc[(df['date_added'].dt.year < df['release_year']),['date_added','release_year']]
```

	date_added	release_year
1551	2020-12-14	2021
1696	2020-11-15	2021
2920	2020-02-13	2021
3168	2019-12-06	2020
3287	2019-11-13	2020
3369	2019-10-25	2020
3433	2019-10-11	2020
4844	2018-05-30	2019
4845	2018-05-29	2019
5394	2017-07-01	2018
5658	2016-12-23	2018
5677	2016-12-13	2017
7063	2018-10-26	2019
7112	2013-03-31	2016



```python
# sample some of the records and check that they have been accurately replaced
df.iloc[[1551,1696,2920,3168]]
```

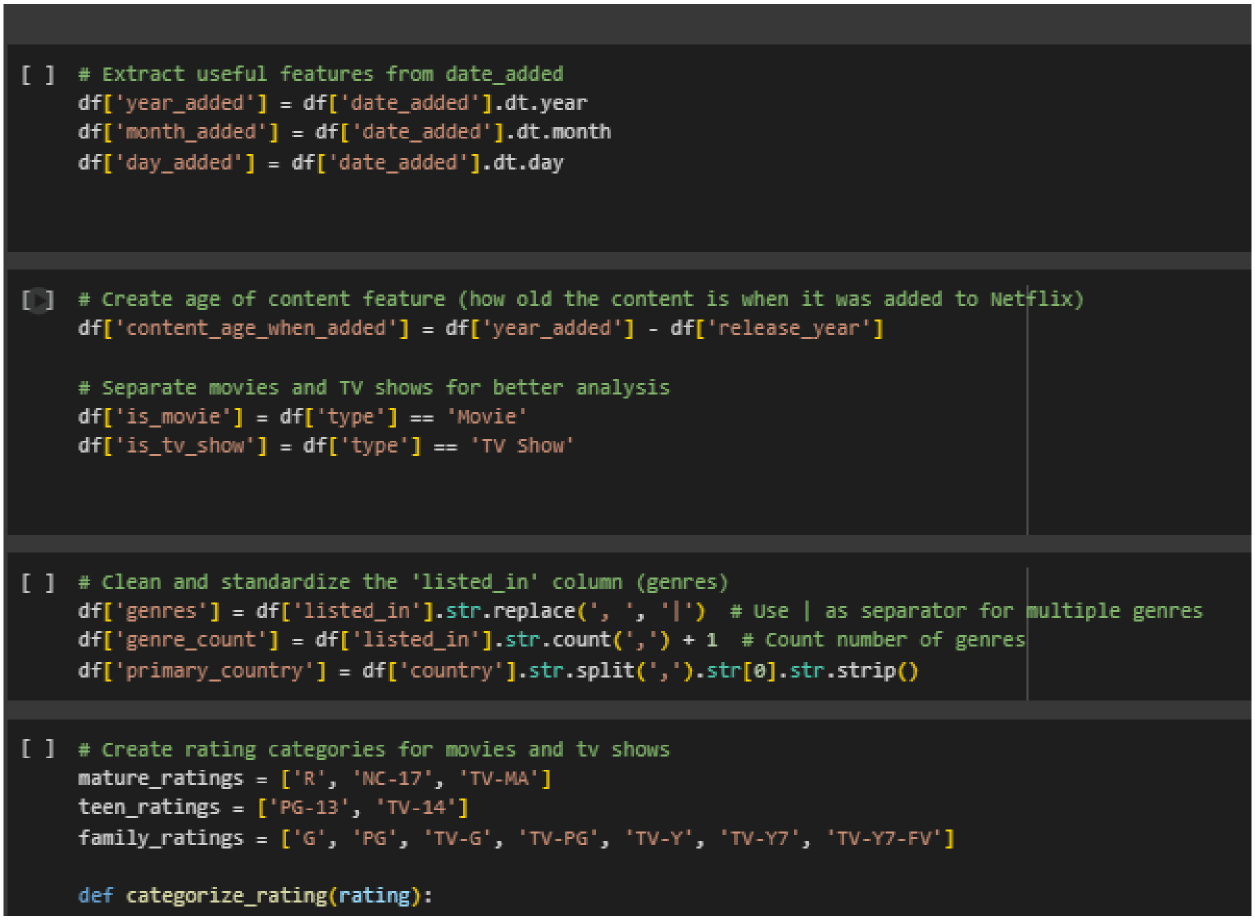
	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	duration_value	duration_unit
1551	TV Show	Hilda	Not Given	Bella Ramsey, Ameerah Falzon-Ojo, Oliver Nelso...	United Kingdom, Canada, United States	2021-01-01	2021	TV-Y7	2 Seasons	Kids' TV	2	Seasons
1696	TV Show	Polly Pocket	Not Given	Emily Tennant, Shannon Chan-Kent, Kazumi Evans...	Canada, United States, Ireland	2021-01-01	2021	TV-Y	2 Seasons	Kids' TV	2	Seasons
2920	TV Show	Love Is Blind	Not Given	Nick Lachey, Vanessa Lachey	United States	2021-01-01	2021	TV-MA	1 Season	Reality TV, Romantic TV Shows	1	Season
3168	TV Show	Fuller House	Not Given	Candace Cameron Bure, Jodie Sweetin, Andrea Ba...	United States	2020-01-01	2020	TV-PG	5 Seasons	TV Comedies	5	Seasons

```python
#Confirm that no more release_year inconsistencies
sum(df['date_added'].dt.year < df['release_year'])
```

```
0
```

2.4 Data Enrichment

Enhancing the dataset with features like day_added using feature engineering and other techniques:



```
[ ]  # Extract useful features from date_added
     df['year_added'] = df['date_added'].dt.year
     df['month_added'] = df['date_added'].dt.month
     df['day_added'] = df['date_added'].dt.day
```

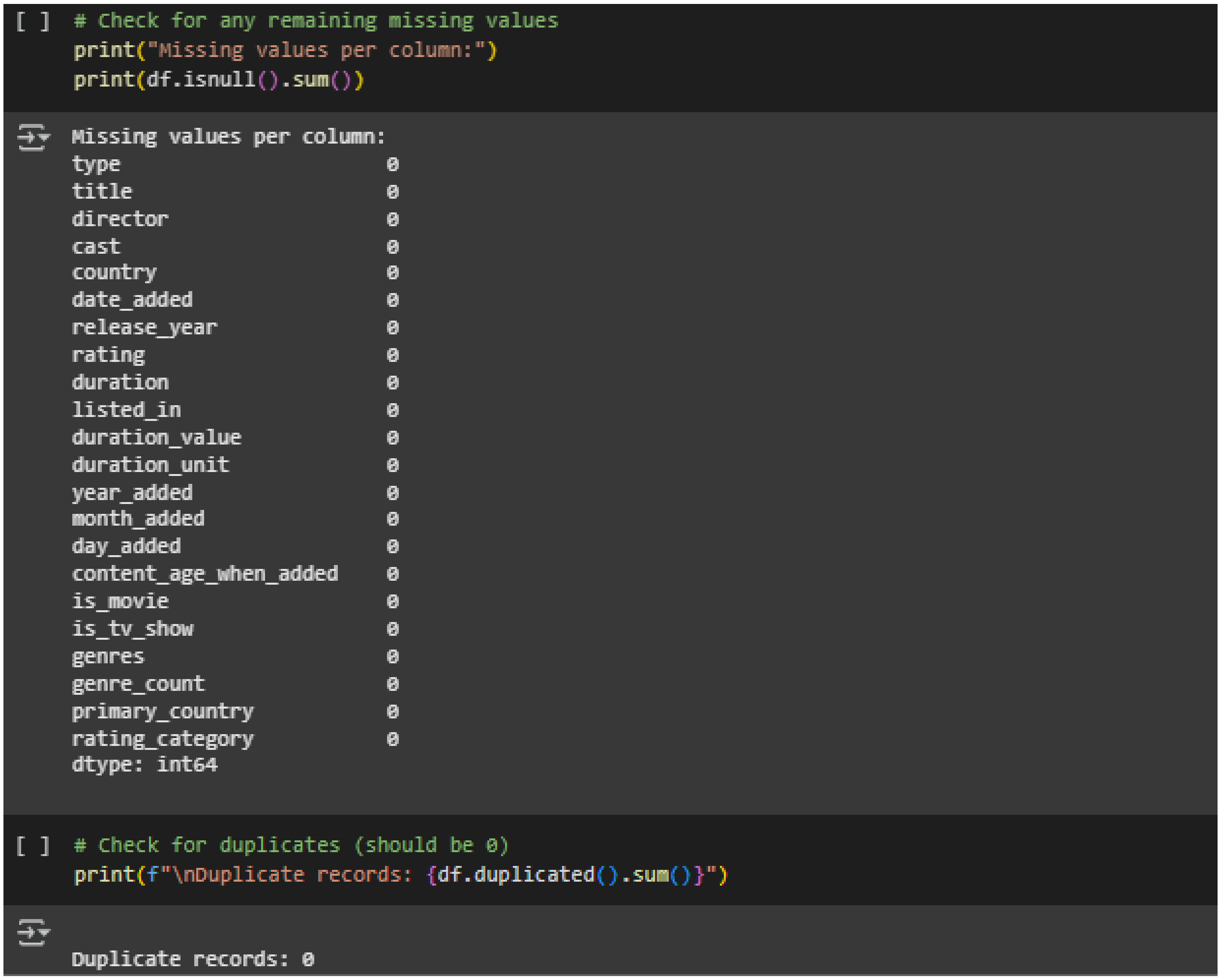
```
[ ]  # Create age of content feature (how old the content is when it was added to Netflix)
     df['content_age_when_added'] = df['year_added'] - df['release_year']

     # Separate movies and TV shows for better analysis
     df['is_movie'] = df['type'] == 'Movie'
     df['is_tv_show'] = df['type'] == 'TV Show'
```

```
[ ]  # Clean and standardize the 'listed_in' column (genres)
     df['genres'] = df['listed_in'].str.replace(', ', '|')  # Use | as separator for multiple genres
     df['genre_count'] = df['listed_in'].str.count(',') + 1  # Count number of genres
     df['primary_country'] = df['country'].str.split(',').str[0].str.strip()
```

```
[ ]  # Create rating categories for movies and tv shows
     mature_ratings = ['R', 'NC-17', 'TV-MA']
     teen_ratings = ['PG-13', 'TV-14']
     family_ratings = ['G', 'PG', 'TV-G', 'TV-PG', 'TV-Y', 'TV-Y7', 'TV-Y7-FV']

     def categorize_rating(rating):
```

2.5 Data Validation

This step included implementing comprehensive validation procedure to ensure data accuracy, completeness and logical consistency.



```
# Check for any remaining missing values
print("Missing values per column:")
print(df.isnull().sum())
```

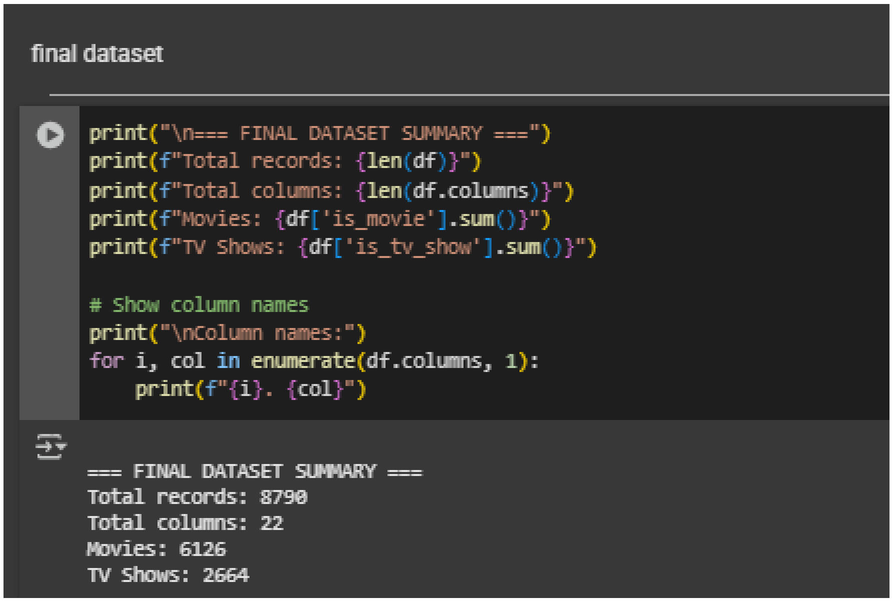
```
Missing values per column:
type                       0
title                      0
director                   0
cast                       0
country                    0
date_added                 0
release_year               0
rating                     0
duration                   0
listed_in                  0
duration_value             0
duration_unit              0
year_added                 0
month_added                0
day_added                  0
content_age_when_added     0
is_movie                   0
is_tv_show                 0
genres                     0
genre_count                0
primary_country            0
rating_category            0
dtype: int64
```

```
# Check for duplicates (should be 0)
print(f"\nDuplicate records: {df.duplicated().sum()}")
```

```

Duplicate records: 0
```

2.6 Data Exportation

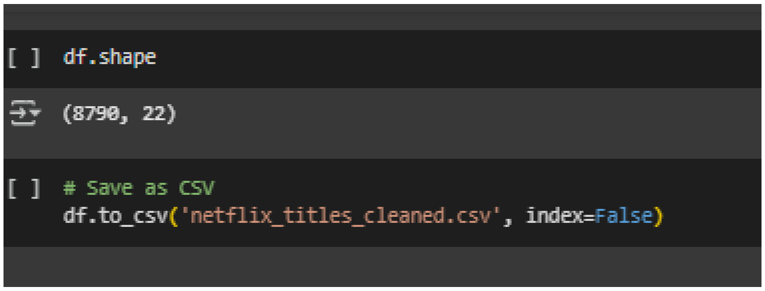
Exported the cleaned data in CSV format which can be used for further analysis.



final dataset

```
print("\n=== FINAL DATASET SUMMARY ===")
print(f"Total records: {len(df)}")
print(f"Total columns: {len(df.columns)}")
print(f"Movies: {df['is_movie'].sum()}")
print(f"TV Shows: {df['is_tv_show'].sum()}")

# Show column names
print("\nColumn names:")
for i, col in enumerate(df.columns, 1):
    print(f"{i}. {col}")
```

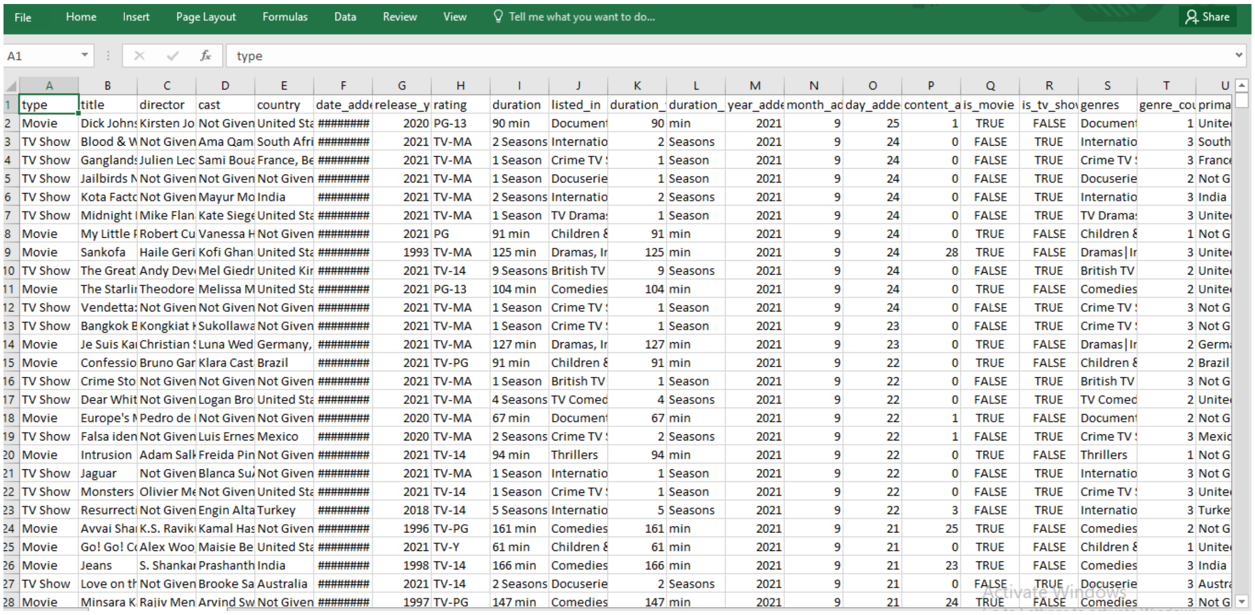
```
=== FINAL DATASET SUMMARY ===
Total records: 8790
Total columns: 22
Movies: 6126
TV Shows: 2664
```



```
[ ] df.shape
```

```
(8790, 22)
```

```
[ ] # Save as CSV
    df.to_csv('netflix_titles_cleaned.csv', index=False)
```



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	type	title	director	cast	country	date_adde	release_y	rating	duration	listed_in	duration_	duration_	year_adde	month_ad	day_adde	content_a	is_movie	is_tv_shov	genres	genre_cou	prima
2	Movie	Dick Johns	Kirsten Jo	Not Given	United Sta	#########	2020	PG-13	90 min	Document	90	min	2021	9	25	1	TRUE	FALSE	Document	1	Unite
3	TV Show	Blood & W	Not Given	Ama Qam	South Afri	#########	2021	TV-MA	2 Seasons	Internatio	2	Seasons	2021	9	24	0	FALSE	TRUE	Internatio	3	South
4	TV Show	Ganglands	Julien Lec	Sami Boua	France, Be	#########	2021	TV-MA	1 Season	Crime TV	1	Season	2021	9	24	0	FALSE	TRUE	Crime TV	3	France
5	TV Show	Jailbirds N	Not Given	Not Given	Not Given	#########	2021	TV-MA	1 Season	Docuserie	1	Season	2021	9	24	0	FALSE	TRUE	Docuserie	2	Not G
6	TV Show	Kota Factc	Not Given	Mayur Mo	India	#########	2021	TV-MA	2 Seasons	Internatio	2	Seasons	2021	9	24	0	FALSE	TRUE	Internatio	3	India
7	TV Show	Midnight I	Mike Flan	Kate Siege	United Sta	#########	2021	TV-MA	1 Season	TV Dramas	1	Season	2021	9	24	0	FALSE	TRUE	TV Dramas	3	Unite
8	Movie	My Little F	Robert Cu	Vanessa H	Not Given	#########	2021	PG	91 min	Children &	91	min	2021	9	24	0	TRUE	FALSE	Children &	1	Not G
9	Movie	Sankofa	Haile Geri	Kofi Ghan	United Sta	#########	1993	TV-MA	125 min	Dramas, Ir	125	min	2021	9	24	28	TRUE	FALSE	Dramas|Ir	3	Unite
10	TV Show	The Great	Andy Dev	Mel Giedr	United Kir	#########	2021	TV-14	9 Seasons	British TV	9	Seasons	2021	9	24	0	FALSE	TRUE	British TV	2	Unite
11	Movie	The Starlir	Theodore	Melissa M	United Sta	#########	2021	PG-13	104 min	Comedies	104	min	2021	9	24	0	TRUE	FALSE	Comedies	2	Unite
12	TV Show	Vendetta:	Not Given	Not Given	Not Given	#########	2021	TV-MA	1 Season	Crime TV	1	Season	2021	9	24	0	FALSE	TRUE	Crime TV	3	Not G
13	TV Show	Bangkok E	Kongkiat I	Sukollawa	Not Given	#########	2021	TV-MA	1 Season	Crime TV	1	Season	2021	9	23	0	FALSE	TRUE	Crime TV	3	Not G
14	Movie	Je Suis Kar	Christian S	Luna Wed	Germany,	#########	2021	TV-MA	127 min	Dramas, Ir	127	min	2021	9	23	0	TRUE	FALSE	Dramas|Ir	2	Germa
15	Movie	Confessio	Bruno Gar	Klara Cast	Brazil	#########	2021	TV-PG	91 min	Children &	91	min	2021	9	22	0	TRUE	FALSE	Children &	2	Brazil
16	TV Show	Crime Sto	Not Given	Not Given	Not Given	#########	2021	TV-MA	1 Season	British TV	1	Season	2021	9	22	0	FALSE	TRUE	British TV	3	Not G
17	TV Show	Dear Whit	Not Given	Logan Bro	United Sta	#########	2021	TV-MA	4 Seasons	TV Comed	4	Seasons	2021	9	22	0	FALSE	TRUE	TV Comed	2	Unite
18	Movie	Europe's M	Pedro de I	Not Given	Not Given	#########	2020	TV-MA	67 min	Document	67	min	2021	9	22	1	TRUE	FALSE	Document	2	Not G
19	TV Show	Falsa iden	Not Given	Luis Ernes	Mexico	#########	2020	TV-MA	2 Seasons	Crime TV	2	Seasons	2021	9	22	1	FALSE	TRUE	Crime TV	3	Mexic
20	Movie	Intrusion	Adam Salk	Freida Pin	Not Given	#########	2021	TV-14	94 min	Thrillers	94	min	2021	9	22	0	TRUE	FALSE	Thrillers	1	Not G
21	TV Show	Jaguar	Not Given	Blanca Su	Not Given	#########	2021	TV-MA	1 Season	Internatio	1	Season	2021	9	22	0	FALSE	TRUE	Internatio	3	Not G
22	TV Show	Monsters	Olivier Me	Not Given	United Sta	#########	2021	TV-14	1 Season	Crime TV	1	Season	2021	9	22	0	FALSE	TRUE	Crime TV	3	Unite
23	TV Show	Resurrecti	Not Given	Engin Alta	Turkey	#########	2018	TV-14	5 Seasons	Internatio	5	Seasons	2021	9	22	3	FALSE	TRUE	Internatio	3	Turke
24	Movie	Avvai Shar	K.S. Ravik	Kamal Has	Not Given	#########	1996	TV-PG	161 min	Comedies	161	min	2021	9	21	25	TRUE	FALSE	Comedies	2	Not G
25	Movie	Go! Go! C	Alex Woo	Maisie Be	United Sta	#########	2021	TV-Y	61 min	Children &	61	min	2021	9	21	0	TRUE	FALSE	Children &	1	Unite
26	Movie	Jeans	S. Shankar	Prashanth	India	#########	1998	TV-14	166 min	Comedies	166	min	2021	9	21	23	TRUE	FALSE	Comedies	3	India
27	TV Show	Love on th	Not Given	Brooke Sa	Australia	#########	2021	TV-14	2 Seasons	Docuserie	2	Seasons	2021	9	21	0	FALSE	TRUE	Docuserie	3	Austra
28	Movie	Minsara K	Rajiv Men	Arvind Sw	Not Given	#########	1997	TV-PG	147 min	Comedies	147	min	2021	9	21	24	TRUE	FALSE	Comedies	3	Not G

Three: Conclusion

In this data wrangling project I was able to transform the Netflix dataset that is ready for analysis and machine leaning applications. Through a systematic six-step approach, the project addressed critical data quality issues while maintaining data integrity and usability.

This project demonstrates the critical importance of thorough data preparation in the data science workflow, establishing solid groundwork for subsequent analytical endeavors and model development initiatives.

Link to Notebook

Link: https://colab.research.google.com/drive/1N7f2XnGnLdg8Rjkj3iUrLBDVjE4wsfEN?usp=sharing