Dataset About "YouTube Top 5000 Channels"

About this file:

This dataset explores the YouTube Top 5000 channels.This analysis examines the dataset of the top 5000 YouTube channels, focusing on the key performance indicators (KPIs) that drive success on the platform. The dataset likely includes metrics such as:

- **Channel Name:** The name of the YouTube channel.
- **Subscribers:** Total number of subscribers for each channel, representing its audience size.
- **Total Views:** The cumulative number of views across all videos of the channel.
- **Video Count:** The total number of videos published by the channel.
- **Category:** The content niche or category (e.g., entertainment, education, gaming, etc.).
- **Country:** The geographic origin of the channel.

- The data is available as a CSV file.We are going to analyze this dataset using the Pandas DataFrame

1.Data Collection

1.1 Import Libraries

In [65]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [66]:
```
# Importing CSV file
data = pd.read_csv('top-5000-youtube-channels.csv')
data
```

Out[66]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views
0	1st	A++	Zee TV	82757	18752951	20869786591
1	2nd	A++	T-Series	12661	61196302	47548839843
2	3rd	A++	Cocomelon - Nursery Rhymes	373	19238251	9793305082
3	4th	A++	SET India	27323	31180559	22675948293
4	5th	A++	WWE	36756	32852346	26273668433
...	...	...	...	...	...	...
4995	4,996th	B+	Uras Benlioğlu	706	2072942	441202795
4996	4,997th	B+	HI-TECH MUSIC LTD	797	1055091	377331722
4997	4,998th	B+	Mastersaint	110	3265735	311758426
4998	4,999th	B+	Bruce McIntosh	3475	32990	14563764
4999	5,000th	B+	SehatAQUA	254	21172	73312511

5000 rows × 6 columns

1.2 Checking Duplicates

In [67]:
```
duplicates = data.duplicated().sum()
print(f"Number of duplicate rows: {duplicates}")
```

```
Number of duplicate rows: 0
```

2. Data Exploration

2.1 .head()

It shows the first N rows in the data(by default,N=5).

In [68]:
```
data.head()
```

Out[68]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views
0	1st	A++	Zee TV	82757	18752951	20869786591
1	2nd	A++	T-Series	12661	61196302	47548839843
2	3rd	A++	Cocomelon - Nursery Rhymes	373	19238251	9793305082
3	4th	A++	SET India	27323	31180559	22675948293
4	5th	A++	WWE	36756	32852346	26273668433

2.2 .shape

It shows the total no.of rows and no.of columns of the dataframes

In [69]:
```
data.shape
```

Out[69]:
```
(5000, 6)
```

2.3 .index

The attribute provides the index of the dataframe

In [70]:
```
data.index
```

Out[70]:
```
RangeIndex(start=0, stop=5000, step=1)
```

2.4 .columns

It shows the name of the column

In [71]:
```
data.columns
```

Out[71]:
```
Index(['Rank', 'Grade', 'Channel name', 'Video Uploads', 'Subscribers',
       'Video views'],
      dtype='object')
```

2.5 .dtypes

It shows the data-type of each column

In [72]:
```
data.dtypes
```

Out[72]:
```
Rank             object
Grade            object
Channel name     object
Video Uploads    object
Subscribers      object
Video views       int64
dtype: object
```

2.6 .Info()

It is used to view the DataFrame Information

In [73]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5000 entries, 0 to 4999
Data columns (total 6 columns):
 #   Column         Non-Null Count  Dtype 
---  ------         --------------  ----- 
 0   Rank           5000 non-null   object
 1   Grade          5000 non-null   object
 2   Channel name   5000 non-null   object
 3   Video Uploads  5000 non-null   object
 4   Subscribers    5000 non-null   object
 5   Video views    5000 non-null   int64 
dtypes: int64(1), object(5)
memory usage: 234.5+ KB
```

2.7 .describe()

It is used to summarize the statistics

In [74]:
```
data.describe()
```

Out[74]:

	Video views
count	5.000000e+03
mean	1.071449e+09
std	2.003844e+09
min	7.500000e+01
25%	1.862329e+08
50%	4.820548e+08
75%	1.124368e+09
max	4.754884e+10

2.8 .isnull().sum()

It is used to identify the missing values

In [75]:
```
missing_values = data.isnull().sum()
print(missing_values)
```

```
Rank            0
Grade           0
Channel name    0
Video Uploads   0
Subscribers     0
Video views     0
dtype: int64
```

3.Data Cleaning

3.1.Replace '--' to NaN

In [76]:
```
data.head(20)
```

Out[76]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views
0	1st	A++	Zee TV	82757	18752951	20869786591
1	2nd	A++	T-Series	12661	61196302	47548839843
2	3rd	A++	Cocomelon - Nursery Rhymes	373	19238251	9793305082
3	4th	A++	SET India	27323	31180559	22675948293
4	5th	A++	WWE	36756	32852346	26273668433
5	6th	A++	Movieclips	30243	17149705	16618094724
6	7th	A++	netd müzik	8500	11373567	23898730764
7	8th	A++	ABS-CBN Entertainment	100147	12149206	17202609850
8	9th	A++	Ryan ToysReview	1140	16082927	24518098041
9	10th	A++	Zee Marathi	74607	2841811	2591830307
10	11th	A+	5-Minute Crafts	2085	33492951	8587520379
11	12th	A+	Canal KondZilla	822	39409726	19291034467
12	13th	A+	Like Nastya Vlog	150	7662886	2540099931
13	14th	A+	Ozuna	50	18824912	8727783225
14	15th	A+	Wave Music	16119	15899764	10989179147
15	16th	A+	Ch3Thailand	49239	11569723	9388600275
16	17th	A+	WORLDSTARHIPHOP	4778	15830098	11102158475
17	18th	A+	Vlad and Nikita	53	--	1428274554
18	19th	A+	Badabun	3060	23603062	5860444053
19	20th	A+	WorkpointOfficial	24287	17687229	14022189654

In [77]:
```
data = data.replace('--' , np.nan,regex=True)
```

In [78]:
```
data.head(20)
```

Out[78]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views
0	1st	A++	Zee TV	82757	18752951	20869786591
1	2nd	A++	T-Series	12661	61196302	47548839843
2	3rd	A++	Cocomelon - Nursery Rhymes	373	19238251	9793305082
3	4th	A++	SET India	27323	31180559	22675948293
4	5th	A++	WWE	36756	32852346	26273668433
5	6th	A++	Movieclips	30243	17149705	16618094724
6	7th	A++	netd müzik	8500	11373567	23898730764
7	8th	A++	ABS-CBN Entertainment	100147	12149206	17202609850
8	9th	A++	Ryan ToysReview	1140	16082927	24518098041
9	10th	A++	Zee Marathi	74607	2841811	2591830307
10	11th	A+	5-Minute Crafts	2085	33492951	8587520379
11	12th	A+	Canal KondZilla	822	39409726	19291034467
12	13th	A+	Like Nastya Vlog	150	7662886	2540099931
13	14th	A+	Ozuna	50	18824912	8727783225
14	15th	A+	Wave Music	16119	15899764	10989179147
15	16th	A+	Ch3Thailand	49239	11569723	9388600275
16	17th	A+	WORLDSTARHIPHOP	4778	15830098	11102158475
17	18th	A+	Vlad and Nikita	53	NaN	1428274554
18	19th	A+	Badabun	3060	23603062	5860444053
19	20th	A+	WorkpointOfficial	24287	17687229	14022189654

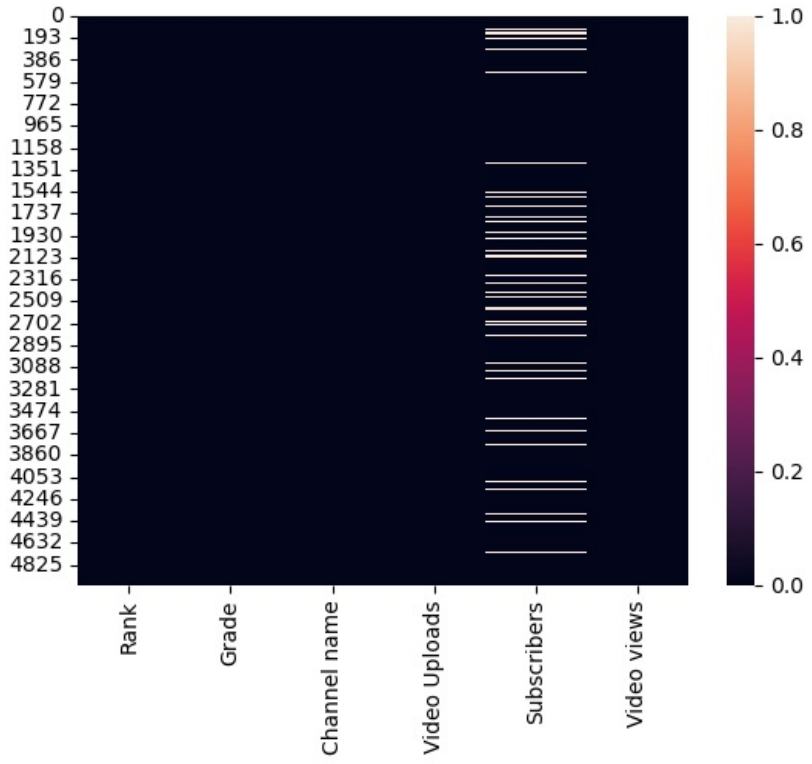
3.2 Find the percentage of missing values

In [79]:
```
percentage_missing = data.isna().sum() *100/len(data)
percentage_missing
```

Out[79]:
```
Rank             0.00
Grade            0.00
Channel name     0.00
Video Uploads    0.12
Subscribers      7.74
Video views      0.00
dtype: float64
```

3.3. visualize the null values

In [80]:
```
sns.heatmap(data.isna());
```



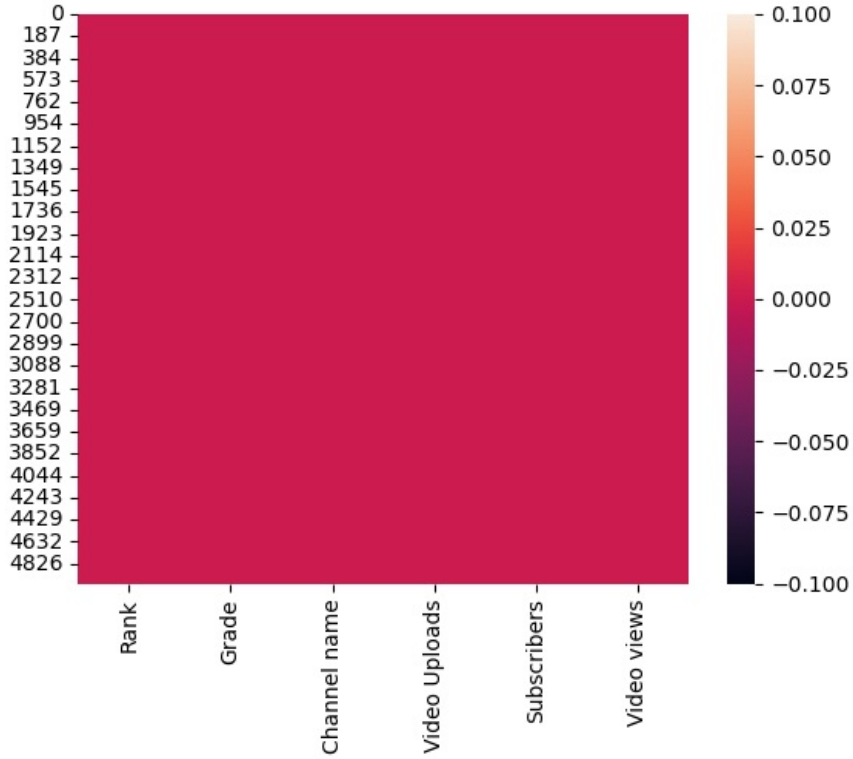
3.4 Drop the missing values

In [81]:
```
data.dropna(inplace=True)
```

In [82]:
```
data.isna().sum()
```

Out[82]:
```
Rank            0
Grade           0
Channel name    0
Video Uploads   0
Subscribers     0
Video views     0
dtype: int64
```

In [83]:
```
sns.heatmap(data.isna());
```



3.5 Data cleaning [Rank column]

In [84]:
```
data.head()
```

Out[84]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views
0	1st	A++	Zee TV	82757	18752951	20869786591
1	2nd	A++	T-Series	12661	61196302	47548839843
2	3rd	A++	Cocomelon - Nursery Rhymes	373	19238251	9793305082
3	4th	A++	SET India	27323	31180559	22675948293
4	5th	A++	WWE	36756	32852346	26273668433

In [85]:
```
data.tail()
```

Out[85]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views
4995	4,996th	B+	Uras Benlioğlu	706	2072942	441202795
4996	4,997th	B+	HI-TECH MUSIC LTD	797	1055091	377331722
4997	4,998th	B+	Mastersaint	110	3265735	311758426
4998	4,999th	B+	Bruce McIntosh	3475	32990	14563764
4999	5,000th	B+	SehatAQUA	254	21172	73312511

In [86]:
```
data.dtypes
```

Out[86]:
```
Rank            object
Grade           object
Channel name    object
Video Uploads   object
Subscribers     object
Video views      int64
dtype: object
```

3.5.1 Remove last 2 characters from Rank Column

In [87]:
```
data['Rank'].dtype
```

Out[87]:
```
dtype('O')
```

In [88]:
```
data['Rank'] = data['Rank'].str[0:-2]
```

In [89]:
```
data.head()
```

Out[89]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views
0	1	A++	Zee TV	82757	18752951	20869786591
1	2	A++	T-Series	12661	61196302	47548839843
2	3	A++	Cocomelon - Nursery Rhymes	373	19238251	9793305082
3	4	A++	SET India	27323	31180559	22675948293
4	5	A++	WWE	36756	32852346	26273668433

In [90]:
```
data.tail()
```

Out[90]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views
4995	4,996	B+	Uras Benlioğlu	706	2072942	441202795
4996	4,997	B+	HI-TECH MUSIC LTD	797	1055091	377331722
4997	4,998	B+	Mastersaint	110	3265735	311758426
4998	4,999	B+	Bruce McIntosh	3475	32990	14563764
4999	5,000	B+	SehatAQUA	254	21172	73312511

3.5.2 Remove comma (,) from Rank Column

In [91]:
```
data['Rank'] = data['Rank'].str.replace(',','')
```

In [92]:
```
data.tail()
```

Out[92]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views
4995	4996	B+	Uras Benlioğlu	706	2072942	441202795
4996	4997	B+	HI-TECH MUSIC LTD	797	1055091	377331722
4997	4998	B+	Mastersaint	110	3265735	311758426
4998	4999	B+	Bruce McIntosh	3475	32990	14563764
4999	5000	B+	SehatAQUA	254	21172	73312511

3.5.3 Convert Rank datatype into Integer

In [93]:
```
data['Rank'].dtype
```

Out[93]:
```
dtype('O')
```

In [94]:
```
data['Rank'] = data['Rank'].astype('int')
```

In [95]:
```
data['Rank'].dtype
```

Out[95]:
```
dtype('int32')
```

3.6 Data Cleaning[Video Upload & Subscribers]

In [96]:
```
data.dtypes
```

Out[96]:
```
Rank             int32
Grade           object
Channel name    object
Video Uploads   object
Subscribers     object
Video views      int64
dtype: object
```

3.6.1 convert Video Upload and Subscribers "**datatype** into **Integer**"

In [97]:
```
data['Video Uploads'] =data['Video Uploads'].astype('int')
data['Subscribers'] = data['Subscribers'].astype('int')
```

In [98]:
```
data.dtypes
```

Out[98]:
```
Rank             int32
Grade           object
Channel name    object
Video Uploads    int32
Subscribers      int32
Video views      int64
dtype: object
```

3.7 Data Cleaning [Grade Column]

In [99]:
```
data.head()
```

Out[99]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views
0	1	A++	Zee TV	82757	18752951	20869786591
1	2	A++	T-Series	12661	61196302	47548839843
2	3	A++	Cocomelon - Nursery Rhymes	373	19238251	9793305082
3	4	A++	SET India	27323	31180559	22675948293
4	5	A++	WWE	36756	32852346	26273668433

In [100…
```
data.tail()
```

Out[100]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views
4995	4996	B+	Uras Benlioğlu	706	2072942	441202795
4996	4997	B+	HI-TECH MUSIC LTD	797	1055091	377331722
4997	4998	B+	Mastersaint	110	3265735	311758426
4998	4999	B+	Bruce McIntosh	3475	32990	14563764
4999	5000	B+	SehatAQUA	254	21172	73312511

In [101…
```
data['Grade'].unique()
```

Out[101]:
```
array(['A++ ', 'A+ ', 'A ', 'A- ', 'B+ '], dtype=object)
```

In [102…
```
data['Grade']=data['Grade'].map({'A++ ':5,'A+ ':4,'A ':3,'A- ':2,'B+ ':1});
```

3.8 Find Average views for each channel

In [103…
```
data.head()
```

Out[103]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views
0	1	5	Zee TV	82757	18752951	20869786591
1	2	5	T-Series	12661	61196302	47548839843
2	3	5	Cocomelon - Nursery Rhymes	373	19238251	9793305082
3	4	5	SET India	27323	31180559	22675948293
4	5	5	WWE	36756	32852346	26273668433

In [104…
```
data['Avg_Views'] = data['Video views'] / data['Video Uploads']
```

In [105…
```
data.head()
```

Out[105]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views	Avg_Views
0	1	5	Zee TV	82757	18752951	20869786591	2.521815e+05
1	2	5	T-Series	12661	61196302	47548839843	3.755536e+06
2	3	5	Cocomelon - Nursery Rhymes	373	19238251	9793305082	2.625551e+07
3	4	5	SET India	27323	31180559	22675948293	8.299216e+05
4	5	5	WWE	36756	32852346	26273668433	7.148130e+05

3.9 Find top 5 channels with maximum number of video uploads

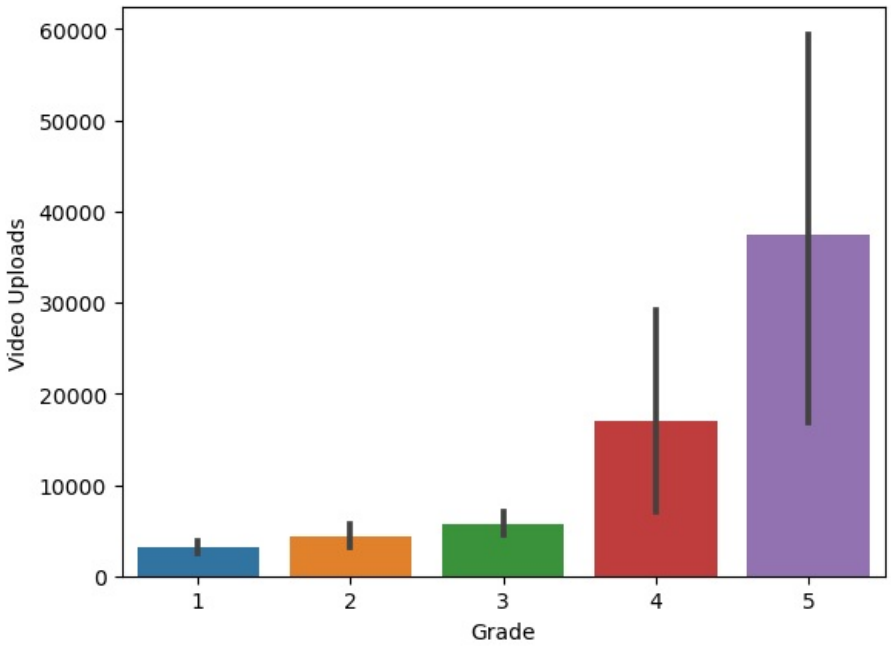
In [106…
```
data.sort_values(by='Video Uploads', ascending=False).head()
```

Out[106]:

	Rank	Grade	Channel name	Video Uploads	Subscribers	Video views	Avg_Views
3453	3454	1	AP Archive	422326	746325	548619569	1299.042846
1149	1150	2	YTN NEWS	355996	820108	1640347646	4607.769879
2223	2224	1	SBS Drama	335521	1418619	1565758044	4666.646928
323	324	3	GMA News	269065	2599175	2786949164	10357.902975
2956	2957	1	MLB	267649	1434206	1329206392	4966.229622

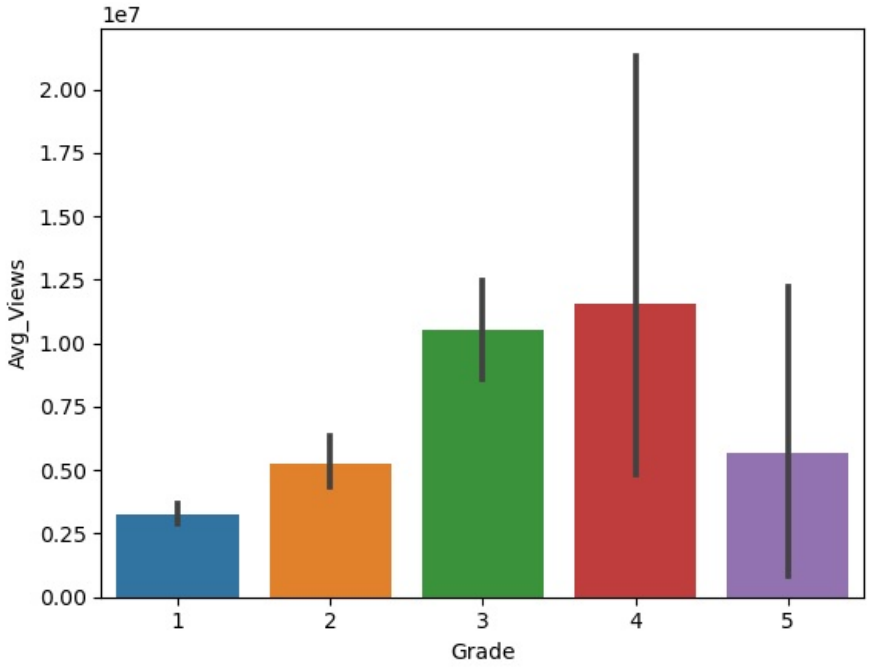
4 . Which Grade has a maximum number of video Uploads?

In [107…
```
sns.barplot(x='Grade', y='Video Uploads', data=data);
```



5. Which grade has the Highest Average Views?

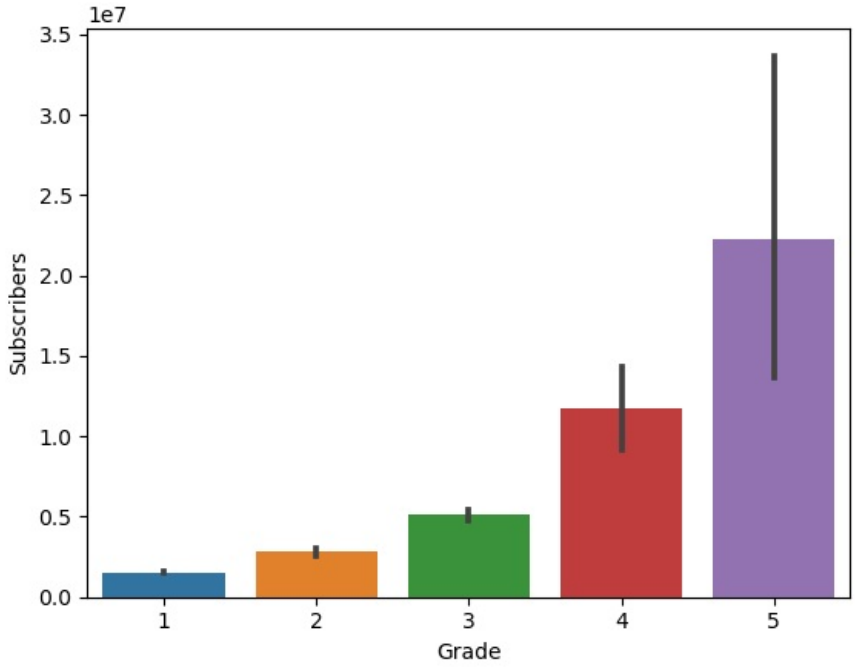
In [108…
```
sns.barplot(data=data, x='Grade', y='Avg_Views');
```



6. Which Grade has the Highest Number of Subscribers?

In [109…
```
sns.barplot(data=data, x='Grade', y='Subscribers')
```

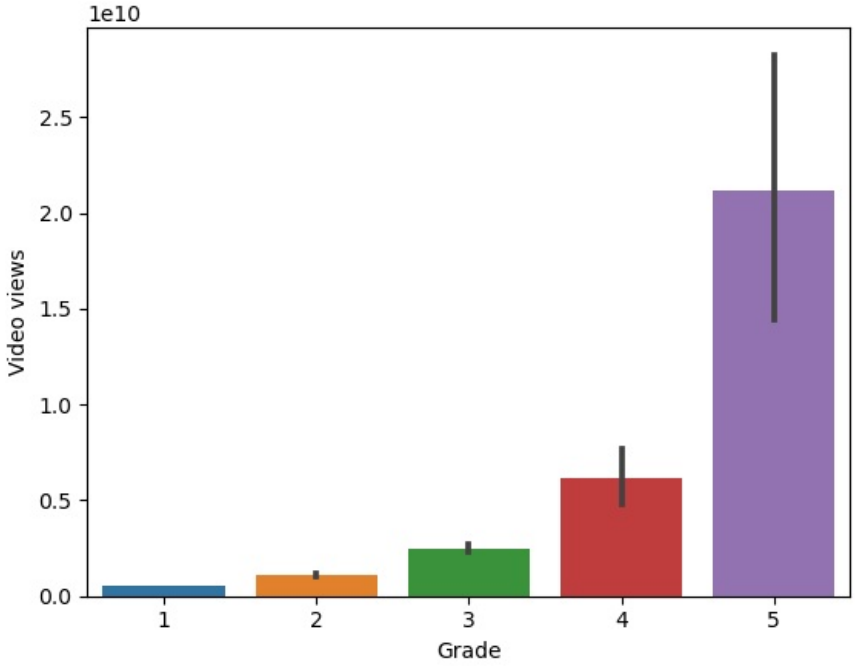
Out[109]:
```
<Axes: xlabel='Grade', ylabel='Subscribers'>
```



7.Which Grade has the Highest Video Views?

In [111…
```
sns.barplot(data=data, x='Grade', y='Video views')
```

Out[111]:
```
<Axes: xlabel='Grade', ylabel='Video views'>
```



Thank You

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js