

Need to use a different data set

data is in CSV formatt, but the category name is given in a different json format

3. cheking for the most liked category

```
In [1]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
```

```
In [2]: %matplotlib inline
```

```
In [3]: import os      # Imported the OS to handle files
```

```
In [5]: files = os.listdir(r'C:\Users\PC\Desktop\Data Analitys project\youtube comment\additional_data')  # File path where datasets are store
```

```
In [6]: files      # All the files in the directory
```

```
Out[6]: ['CAvideos.csv',  
        'CA_category_id.json',  
        'DEvideos.csv',  
        'DE_category_id.json',  
        'FRvideos.csv',  
        'FR_category_id.json',  
        'GBvideos.csv',  
        'GB_category_id.json',  
        'INvideos.csv',  
        'IN_category_id.json',  
        'JPvideos.csv',  
        'JP_category_id.json',  
        'KRvideos.csv',  
        'KR_category_id.json',  
        'MXvideos.csv',  
        'MX_category_id.json',  
        'RUvideos.csv',  
        'RU_category_id.json',  
        'USvideos.csv',  
        'US_category_id.json']
```

```
In [13]: files_csv = []      # Making the list of all the csv files  
  
for i in files:  
    if '.csv' in i:  
        files_csv.append(i)
```

```
In [14]: files_csv      # All the CSV files are now under a different list
```

```
Out[14]: ['CAvideos.csv',  
        'DEvideos.csv',  
        'FRvideos.csv',  
        'GBvideos.csv',  
        'INvideos.csv',  
        'JPvideos.csv',  
        'KRvideos.csv',  
        'MXvideos.csv',  
        'RUvideos.csv',  
        'USvideos.csv']
```

<https://docs.python.org/3/library/codecs.html> (<https://docs.python.org/3/library/codecs.html>) <https://docs.python.org/3/library/codecs.html#standard-encodings> `#standard-encodings%C2%B6`

```
In [16]: full_df = pd.DataFrame()                                # Creating a dataset out of all the CSV files
path=r'C:\Users\PC\Desktop\Data Analitys project\youtube comment\additional_data'
for i in files_csv:
    temp_df = pd.read_csv(path + '/' + i, encoding = 'iso-8859-1', error_bad_lines=False)
    full_df = pd.concat([full_df, temp_df], ignore_index=True)
```

```
C:\Users\PC\AppData\Local\Temp\ipykernel_8800\957244681.py:4: FutureWarning: The error_bad_lines argument has been deprecated and will be removed in a future version. Use on_bad_lines in the future.
```

```
temp_df = pd.read_csv(path + '/' + i, encoding = 'iso-8859-1', error_bad_lines=False)
C:\Users\PC\AppData\Local\Temp\ipykernel_8800\957244681.py:4: FutureWarning: The error_bad_lines argument has been deprecated and will be removed in a future version. Use on_bad_lines in the future.
```

```
temp_df = pd.read_csv(path + '/' + i, encoding = 'iso-8859-1', error_bad_lines=False)
C:\Users\PC\AppData\Local\Temp\ipykernel_8800\957244681.py:4: FutureWarning: The error_bad_lines argument has been deprecated and will be removed in a future version. Use on_bad_lines in the future.
```

```
temp_df = pd.read_csv(path + '/' + i, encoding = 'iso-8859-1', error_bad_lines=False)
C:\Users\PC\AppData\Local\Temp\ipykernel_8800\957244681.py:4: FutureWarning: The error_bad_lines argument has been deprecated and will be removed in a future version. Use on_bad_lines in the future.
```

```
temp_df = pd.read_csv(path + '/' + i, encoding = 'iso-8859-1', error_bad_lines=False)
C:\Users\PC\AppData\Local\Temp\ipykernel_8800\957244681.py:4: FutureWarning: The error_bad_lines argument has been deprecated and will be removed in a future version. Use on_bad_lines in the future.
```

```
temp_df = pd.read_csv(path + '/' + i, encoding = 'iso-8859-1', error_bad_lines=False)
C:\Users\PC\AppData\Local\Temp\ipykernel_8800\957244681.py:4: FutureWarning: The error_bad_lines argument has been deprecated and will be removed in a future version. Use on_bad_lines in the future.
```

```
temp_df = pd.read_csv(path + '/' + i, encoding = 'iso-8859-1', error_bad_lines=False)
C:\Users\PC\AppData\Local\Temp\ipykernel_8800\957244681.py:4: FutureWarning: The error_bad_lines argument has been deprecated and will be removed in a future version. Use on_bad_lines in the future.
```

```
temp_df = pd.read_csv(path + '/' + i, encoding = 'iso-8859-1', error_bad_lines=False)
C:\Users\PC\AppData\Local\Temp\ipykernel_8800\957244681.py:4: FutureWarning: The error_bad_lines argument has been deprecated and will be removed in a future version. Use on_bad_lines in the future.
```

```
temp_df = pd.read_csv(path + '/' + i, encoding = 'iso-8859-1', error_bad_lines=False)
C:\Users\PC\AppData\Local\Temp\ipykernel_8800\957244681.py:4: FutureWarning: The error_bad_lines argument has been deprecated and will be removed in a future version. Use on_bad_lines in the future.
```

```
temp_df = pd.read_csv(path + '/' + i, encoding = 'iso-8859-1', error_bad_lines=False)
C:\Users\PC\AppData\Local\Temp\ipykernel_8800\957244681.py:4: FutureWarning: The error_bad_lines argument has been deprecated and will be removed in a future version. Use on_bad_lines in the future.
```

be removed in a future version. Use `on_bad_lines` in the future.

```
temp_df = pd.read_csv(path + '/' + i, encoding = 'iso-8859-1', error_bad_lines=False)
```

```
In [17]: full_df          # Concatinating all the datasets together
```

Out[17]:

	video_id	trending_date	title	channel_title	category_id	publish_time	tags	views	likes	dis
0	n1WpP7iowLc	17.14.11	Eminem - Walk On Water (Audio) ft. Beyonc�	EminemVEVO	10	2017-11-10T17:00:03.000Z	Eminem "Walk "On "Water "Aftermath/Shady/In...	17158579	787425	4k
1	0dBIkQ4Mz1M	17.14.11	PLUSH - Bad Unboxing Fan Mail	iDubbbzTV	23	2017-11-13T17:00:00.000Z	plush "bad unboxing "unboxing "fan mail "id...	1014651	127794	10k
2	5qpjK5DgCt4	17.14.11	Racist Superman Rudy Mancuso, King Bach & Le...	Rudy Mancuso	23	2017-11-12T19:05:24.000Z	racist superman "rudy "mancuso "king "bach"...	3191434	146035	10k
3	d380meD0W0M	17.14.11	I Dare You: GOING BALD!?	nigahiga	24	2017-11-12T18:01:41.000Z	ryan "higa "higatv "nigahiga "i dare you "...	2095828	132239	10k
4	2Vv-BfVoq4g	17.14.11	Ed Sheeran - Perfect (Official Music Video)	Ed Sheeran	10	2017-11-09T11:04:14.000Z	edsheeran "ed sheeran "acoustic "live "cove...	33523622	1634130	20k
...
375937	BZt0qjTWNhw	18.14.06	The Cat Who Caught the Laser	AaronsAnimals	15	2018-05-18T13:00:04.000Z	aarons animals "aarons "animals "cat "cats"...	1685609	38160	10k
375938	1h7KV2sjUWY	18.14.06	True Facts : Ant Mutualism	zefrank1	22	2018-05-18T01:00:06.000Z	[none]	1064798	60008	10k
375939	D6Oy4LfoqsU	18.14.06	I GAVE SAFIYA NYGAARD A PERFECT HAIR MAKEOVER ...	Brad Mondo	24	2018-05-18T17:34:22.000Z	I gave safiya nygaard a perfect hair makeover ...	1066451	48068	10k
375940	oV0zkMe1K8s	18.14.06	How Black Panther Should Have Ended	How It Should Have Ended	1	2018-05-17T17:00:04.000Z	Black Panther "HISHE "Marvel "Infinity War"...	5660813	192957	10k

	video_id	trending_date	title	channel_title	category_id	publish_time	tags	views	likes	disl
375941	ooyjaVdt-jA	18.14.06	Official Call of Duty®: Black Ops 4 Mult...	Call of Duty	20	2018-05-17T17:09:38.000Z	call of duty "cod" "activision" "Black Ops 4"	10306119	357079	211

375942 rows × 16 columns

```
In [48]: full_df['category_id'].unique()           # Checking for all the unique categories
```

```
Out[48]: array([10, 23, 24, 25, 22, 26,  1, 28, 20, 17, 29, 15, 19,  2, 27, 43, 30,
                44], dtype=int64)
```

```
In [27]: json_df = pd.read_json(r'C:\Users\PC\Desktop\Data Analitys project\youtube comment\additional_data\US_category_id.json')
```

```
In [42]: json_df['items'][0]['snippet']['title']    # Using this logic to extract all the category names
```

```
Out[42]: 'Film & Animation'
```

```
In [43]: category_id = {}
          for i in json_df['items']:
              category_id[int(i['id'])] = i['snippet']['title']           # Using a Dictinory to pair the category id and the category name in
```

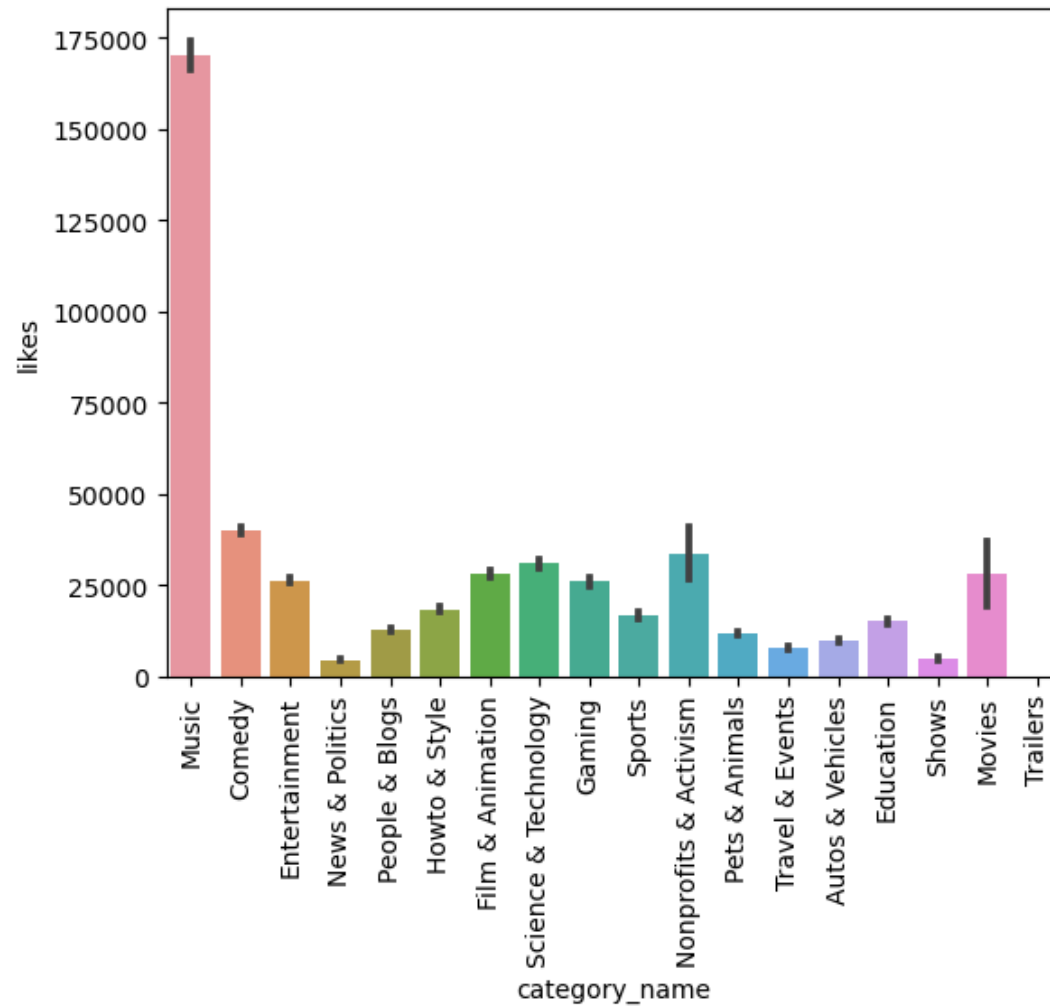
```
In [44]: category_id      # Here are the category id's and there names
```

```
Out[44]: {1: 'Film & Animation',
          2: 'Autos & Vehicles',
          10: 'Music',
          15: 'Pets & Animals',
          17: 'Sports',
          18: 'Short Movies',
          19: 'Travel & Events',
          20: 'Gaming',
          21: 'Videoblogging',
          22: 'People & Blogs',
          23: 'Comedy',
          24: 'Entertainment',
          25: 'News & Politics',
          26: 'Howto & Style',
          27: 'Education',
          28: 'Science & Technology',
          29: 'Nonprofits & Activism',
          30: 'Movies',
          31: 'Anime/Animation',
          32: 'Action/Adventure',
          33: 'Classics',
          34: 'Comedy',
          35: 'Documentary',
          36: 'Drama',
          37: 'Family',
          38: 'Foreign',
          39: 'Horror',
          40: 'Sci-Fi/Fantasy',
          41: 'Thriller',
          42: 'Shorts',
          43: 'Shows',
          44: 'Trailers'}
```

```
In [50]: full_df['category_name'] = full_df['category_id'].map(category_id)
```

```
In [63]: sns.barplot(data = full_df, x='category_name', y='likes')
plt.xticks(rotation='vertical')
```

```
Out[63]: (array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
                17]),
          [Text(0, 0, 'Music'),
           Text(1, 0, 'Comedy'),
           Text(2, 0, 'Entertainment'),
           Text(3, 0, 'News & Politics'),
           Text(4, 0, 'People & Blogs'),
           Text(5, 0, 'Howto & Style'),
           Text(6, 0, 'Film & Animation'),
           Text(7, 0, 'Science & Technology'),
           Text(8, 0, 'Gaming'),
           Text(9, 0, 'Sports'),
           Text(10, 0, 'Nonprofits & Activism'),
           Text(11, 0, 'Pets & Animals'),
           Text(12, 0, 'Travel & Events'),
           Text(13, 0, 'Autos & Vehicles'),
           Text(14, 0, 'Education'),
           Text(15, 0, 'Shows'),
           Text(16, 0, 'Movies'),
           Text(17, 0, 'Trailers')])
```



Here is the most liked category's, as per the above graph Music is the most liked Category

4. Find out whether audience is engaged or not

In order to find whether the audience are engaged or not let's find out the like and dislike percentage from the total views

In [66]: `full_df.head(2)`

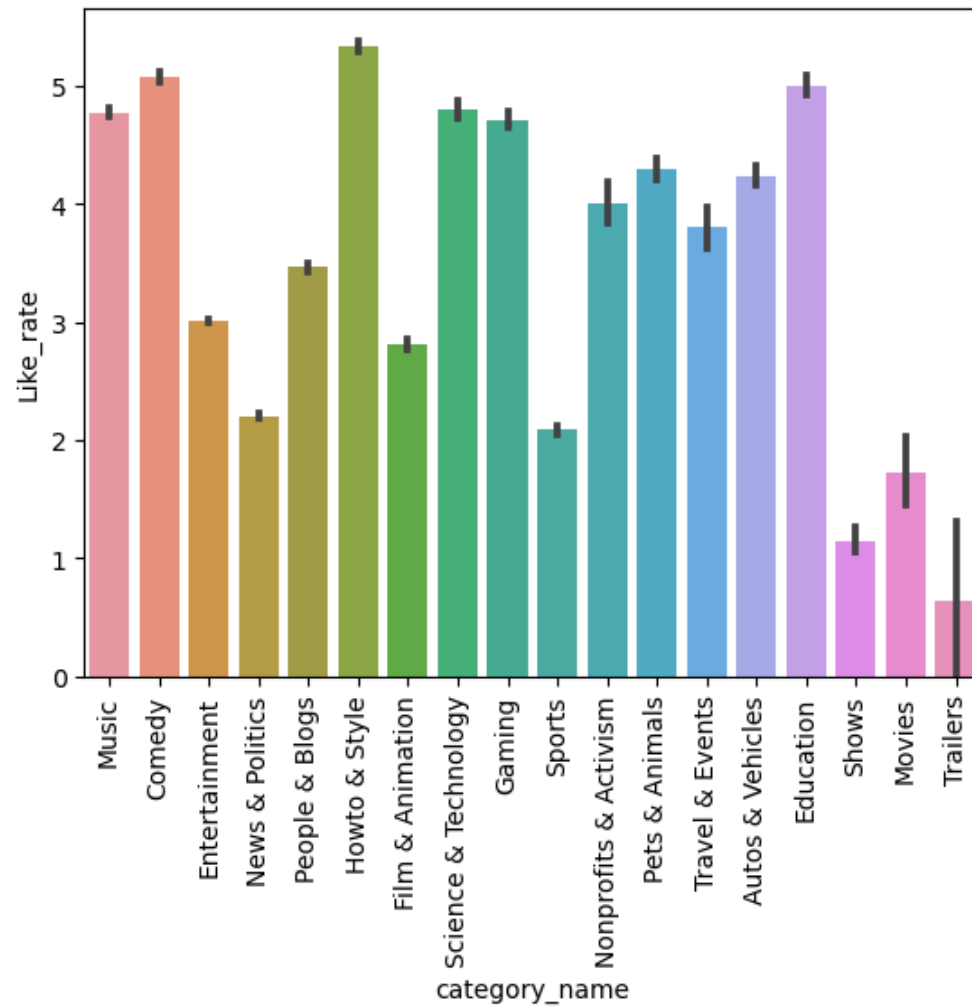
Out[66]:

video_id	trending_date	title	channel_title	category_id	publish_time	tags	views	likes	dislikes	comment
n1WpP7iowLc	17.14.11	Eminem - Walk On Water (Audio) ft. Beyonc�	EminemVEVO	10	2017-11-10T17:00:03.000Z	Eminem "Walk "On "Water "Aftermath/Shady/In...	17158579	787425	43420	
0dBlkQ4Mz1M	17.14.11	PLUSH - Bad Unboxing Fan Mail	iDubbbzTV	23	2017-11-13T17:00:00.000Z	plush "bad unboxing "unboxing "fan mail "id...	1014651	127794	1688	

In [67]: `full_df['Like_rate'] = (full_df['likes']/full_df['views'])*100`
`full_df['dislikes_rate'] = (full_df['dislikes']/full_df['views'])*100`
`full_df['comment_rate'] = (full_df['comment_count']/full_df['views'])*100`

```
In [72]: sns.barplot(x='category_name', y='Like_rate' , data = full_df)
plt.xticks(rotation = "vertical")
```

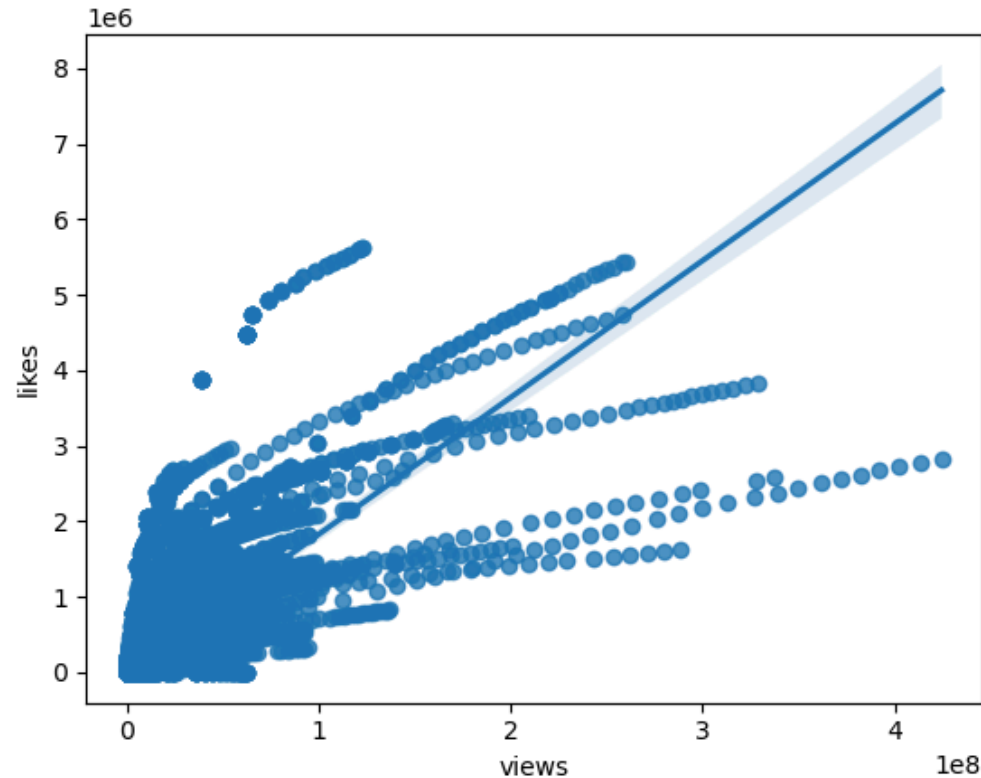
```
Out[72]: (array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
                17]),
          [Text(0, 0, 'Music'),
           Text(1, 0, 'Comedy'),
           Text(2, 0, 'Entertainment'),
           Text(3, 0, 'News & Politics'),
           Text(4, 0, 'People & Blogs'),
           Text(5, 0, 'Howto & Style'),
           Text(6, 0, 'Film & Animation'),
           Text(7, 0, 'Science & Technology'),
           Text(8, 0, 'Gaming'),
           Text(9, 0, 'Sports'),
           Text(10, 0, 'Nonprofits & Activism'),
           Text(11, 0, 'Pets & Animals'),
           Text(12, 0, 'Travel & Events'),
           Text(13, 0, 'Autos & Vehicles'),
           Text(14, 0, 'Education'),
           Text(15, 0, 'Shows'),
           Text(16, 0, 'Movies'),
           Text(17, 0, 'Trailers')])
```



5. Find out whether there is a relationship between likes and views

```
In [84]: sns.regplot(data = full_df, x="views", y="likes")
```

```
Out[84]: <Axes: xlabel='views', ylabel='likes'>
```



As per the above regression plot we can see that the relation between likes and views are linearly proportional

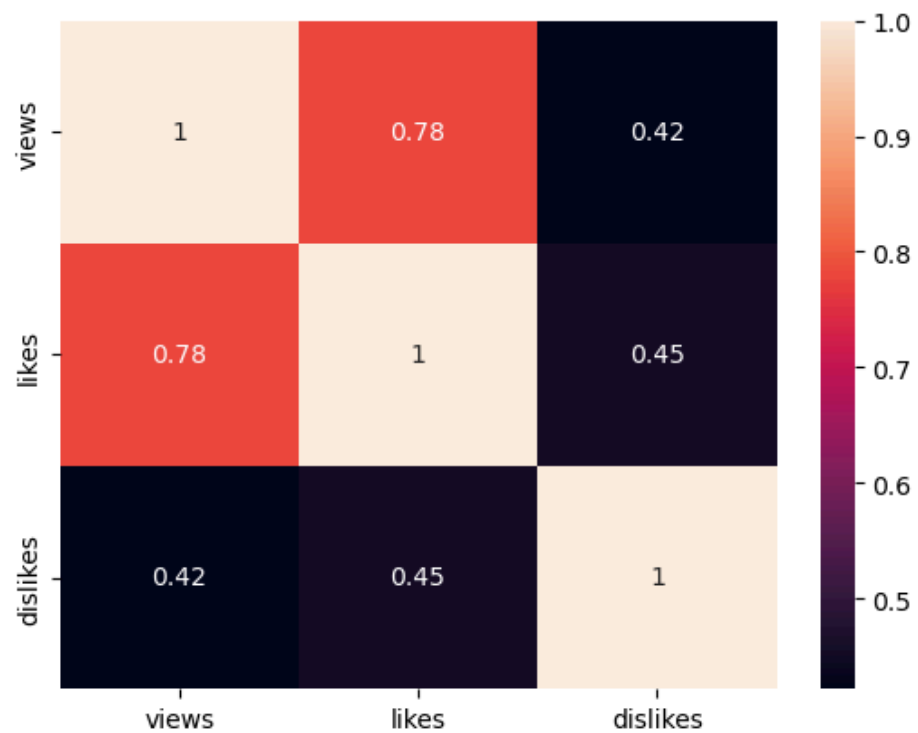
```
In [86]: full_df.columns
```

```
Out[86]: Index(['video_id', 'trending_date', 'title', 'channel_title', 'category_id',  
              'publish_time', 'tags', 'views', 'likes', 'dislikes', 'comment_count',  
              'thumbnail_link', 'comments_disabled', 'ratings_disabled',  
              'video_error_or_removed', 'description', 'category_name', 'Like_rate',  
              'dislikes_rate', 'comment_rate'],  
              dtype='object')
```



```
In [94]: sns.heatmap(data = full_df[['views', 'likes', 'dislikes']].corr(), annot = True)
```

Out[94]: <Axes: >



6. Which channels have the largest number of trending videos?

In [97]:

full_df.head(5)

Out[97]:

	video_id	trending_date	title	channel_title	category_id	publish_time	tags	views	likes	dislikes	co
0	n1WpP7iowLc	17.14.11	Eminem - Walk On Water (Audio) ft. Beyonc�	EminemVEVO	10	2017-11-10T17:00:03.000Z	Eminem "Walk "On "Water "Aftermath/Shady/In...	17158579	787425	43420	
1	0dBikQ4Mz1M	17.14.11	PLUSH - Bad Unboxing Fan Mail	iDubbbzTV	23	2017-11-13T17:00:00.000Z	plush "bad unboxing "unboxing "fan mail "id...	1014651	127794	1688	
2	5qpjK5DgCt4	17.14.11	Racist Superman Rudy Mancuso, King Bach & Le...	Rudy Mancuso	23	2017-11-12T19:05:24.000Z	racist superman "rudy "mancuso "king "bach"...	3191434	146035	5339	
3	d380meD0W0M	17.14.11	I Dare You: GOING BALD!?	nigahiga	24	2017-11-12T18:01:41.000Z	ryan "higa "higatv "nigahiga "i dare you "...	2095828	132239	1989	
4	2Vv-BfVoq4g	17.14.11	Ed Sheeran - Perfect (Official Music Video)	Ed Sheeran	10	2017-11-09T11:04:14.000Z	edsheeran "ed sheeran "acoustic "live "cove...	33523622	1634130	21082	

```
In [107]: full_df['channel_title'].value_counts().reset_index()
```

Out[107]:

	index	channel_title	
0	The Late Show with Stephen Colbert	984	
1	WWE	804	
2	Late Night with Seth Meyers	773	
3	VikatanTV	763	
4	TheEllenShow	743	
...	
37819	DFC Orrivals	1	
37820	haiblubbblubb	1	
37821	SOYER	1	
37822	GOLD CLAN	1	
37823	Herr Zymny	1	

37824 rows × 2 columns

```
In [108]: import plotly.express as px
```

```
In [109]: px.bar(data_frame=cdf[0:20] , x='channel_title' , y='total_videos')
```

