YOUTUBE-ANALYSIS-API-PROJECT

Analysed Data :-

> Top Subscribed Channel.
>
> Top Subscribed Channel.
>
> Least Subscribed Channel.
>
> Graph View of Channels by Subscribers.
>
> Graph View of Channels by Views.
>
> Graph View of Channels by Total_videos.
>
> Channel Details of Subscribers ,Views and Total_videos.
>
> Top Channel analysis.
>
> Top 10 Viewed Videos.
>
> Least 10 viewed videos.
>
> Year wise Maximum Views.
>
> Maxmium Likes Video Title.
>
> Minimum likes Video Title.
>
> Year wise Maximum Viewed Video Title.
>
> Month wise Views count.
>
> Month wise Maximum Likes by Year base .
>
> Total week days wise Maximum Views.
>
> Year wise Maximum WeekDays Views.
>
> Month wise Maximum WeekDays Views.
>
> Single day data of Views , Likes and Comments.
>
> Top comments Video Datails.
>
> Least comments Video Details.

In [1]:
```
# pip install google-api-python-client      # used for access the api key accept
```

In [2]:
```
#   imported Libraries

from googleapiclient.discovery import build
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [3]:
```
api_key = 'AIzaSyDL52G31l-UV-d90kkUxlX1KeRzzUxhyes'       #  access key from Google developer console
channel_ids = ["UC7Q7pl0z0MrdayvmAnchlJQ",# mortal
               "UCqNH56x9g4QYVpzmWTzqVYg",  # dynamo gaming
               "UC5c9VlYTSvBSCaoMu_GI6gQ",#total gaming
               "UCNoiWmvkDUL0q-6ECxNFH0Q", # jonathan gaming
               "UCYxMATvBqKQx7utYcYK3waA", # scout
               "UC0IWRLai-BAwci_e9MylNGw", # carryislive
               "UCQfBN7Ut5TiZJAhMo_Rgfyw", # snax gaming
               "UCNawD-EOzpfjDs5CBs5mBiQ",# mavi
               "UCfbS6xFS5pKesagI6xIQOyw", #  goblin
               "UC6GIR5W1Bm5Moc3kzKV-nhw" ]# krutika plays

youtube = build('youtube', 'v3', developerKey=api_key)
```

In [4]:
```
def channels(youtube, channel_ids):                                  # reference taken from youtube data API
    all_data = []
    request = youtube.channels().list(
                part='snippet,contentDetails,statistics',
```

```
                id=','.join(channel_ids))
    response = request.execute()

    for i in range(len(response['items'])):                                    # All data convert from json and u
        data = dict(Channel_name = response['items'][i]['snippet']['title'],
                    Subscribers = response['items'][i]['statistics']['subscriberCount'],
                    Views = response['items'][i]['statistics']['viewCount'],
                    Total_videos = response['items'][i]['statistics']['videoCount'],
                    playlist_id = response['items'][i]['contentDetails']['relatedPlaylists']['uploads'])
        all_data.append(data)

    return all_data
```

In [5]:
```
channel_statistics = channels(youtube, channel_ids)   # stored function in new variable
```

In [6]:
```
channel_data = pd.DataFrame(channel_statistics)   # converted to dataframe
```

In [7]:
```
channel_data     # fetched data
```

Out[7]:

	Channel_name	Subscribers	Views	Total_videos	playlist_id
0	CarryisLive	11600000	1450676239	947	UU0IWRLai-BAwci_e9MylNGw
1	JONATHAN GAMING	4900000	461416540	392	UUNoiWmvkDUL0q-6ECxNFH0Q
2	Goblin	369000	34933939	324	UUfbS6xFS5pKesagI6xIQOyw
3	Krutika Plays	629000	126199779	444	UU6GIR5W1Bm5Moc3kzKV-nhw
4	MAVI	1260000	242664924	683	UUNawD-EOzpfjDs5CBs5mBiQ
5	MortaL	6980000	1198491210	1662	UU7Q7pl0z0MrdayvmAnchlJQ
6	sc0ut	4670000	577290406	1091	UUYxMATvBqKQx7utYcYK3waA
7	Total Gaming	34300000	5302731146	1138	UU5c9VlYTSvBSCaoMu_GI6gQ
8	Dynamo Gaming	10000000	1175092765	2008	UUqNH56x9g4QYVpzmWTzqVYg
9	Snax Gaming	1410000	153723026	434	UUQfBN7Ut5TiZJAhMo_Rgfyw

In [8]:
```
channel_data.info()    # fetched Datatype
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   Channel_name  10 non-null     object
 1   Subscribers   10 non-null     object
 2   Views         10 non-null     object
 3   Total_videos  10 non-null     object
 4   playlist_id   10 non-null     object
dtypes: object(5)
memory usage: 528.0+ bytes
```

In [9]:
```
 # converted date from object type to numerical type

channel_data["Subscribers"]=pd.to_numeric(channel_data["Subscribers"])   #  Subscribers column
channel_data["Views"]=pd.to_numeric(channel_data["Views"])    #Views column
channel_data["Total_videos"]=pd.to_numeric(channel_data["Total_videos"])   #Total_videos column
```

In [10]:
```
channel_data.info()   # fetched data type
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   Channel_name  10 non-null     object
 1   Subscribers   10 non-null     int64 
 2   Views         10 non-null     int64 
 3   Total_videos  10 non-null     int64 
 4   playlist_id   10 non-null     object
dtypes: int64(3), object(2)
memory usage: 528.0+ bytes
```

In [11]:
```
print(channel_data.isnull().sum())    #checked null values
```

```
Channel_name    0
Subscribers     0
Views           0
Total_videos    0
playlist_id     0
dtype: int64
```

* Top Subscribed Channel

In [12]:
```
Subscribed_channel_top=channel_data.sort_values(by="Subscribers",ascending=False)  # sorted values
```

In [12]:
```
Subscribed_channel_top=channel_data.sort_values(by= 'Subscribers' ,ascending=False)  # sorted values
Subscribed_channel_top.head(1).transpose() # fetched top data
```

Out[12]:

	7
Channel_name	Total Gaming
Subscribers	34300000
Views	5302731146
Total_videos	1138
playlist_id	UU5c9VlYTSvBSCaoMu_GI6gQ

* Least Subscribed Channel

In [13]:
```
Subscribed_channel_least=channel_data.sort_values(by="Subscribers",ascending=True)  # sorted values
Subscribed_channel_least.head(1).transpose()   # fetched top data
```

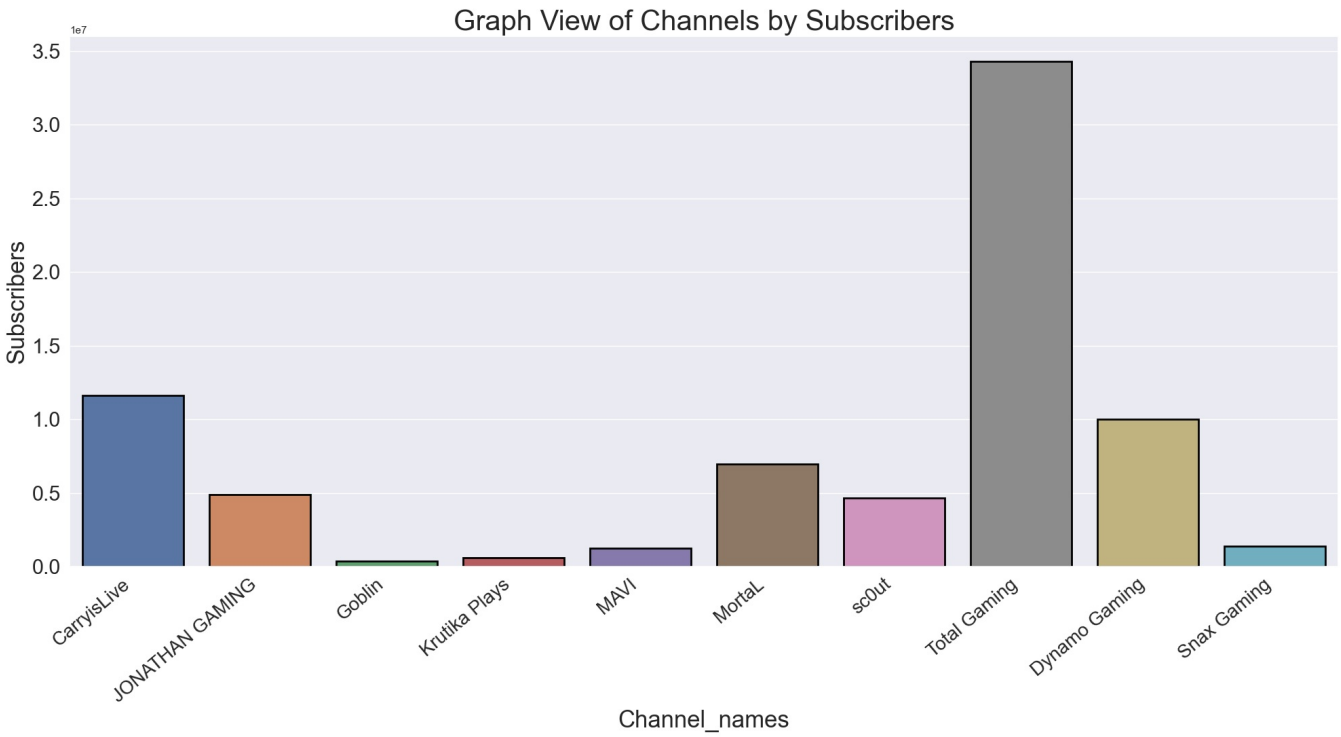
Out[13]:

	2
Channel_name	Goblin
Subscribers	369000
Views	34933939
Total_videos	324
playlist_id	UUfbS6xFS5pKesagI6xIQOyw

Graph View of Channels by Subscribers

In [14]:
```
# creating graph for Subscribers observation

plt.figure(figsize=(20,11))
sns.set(rc={'figure.figsize':(18,8)})

Subscribers = sns.barplot(x='Channel_name', y='Subscribers', data=channel_data,edgecolor="black",linewidth=2) #

plt.title("Graph View of Channels by Subscribers",fontsize=30)
plt.xticks(fontsize=20)
plt.ylabel("Subscribers",fontsize=25)
plt.xlabel("Channel_names",fontsize=25)
plt.yticks(fontsize=22)
plt.xticks(rotation=40,horizontalalignment="right",fontsize=20)
plt.tight_layout()

Subscribed_channel=channel_data
print(Subscribed_channel[["Channel_name","Subscribers"]])   # fetched Channel_name and Subscribers data
```

```
      Channel_name  Subscribers
0      CarryisLive     11600000
1  JONATHAN GAMING      4900000
2           Goblin       369000
3    Krutika Plays       629000
4             MAVI      1260000
5           MortaL      6980000
6            sc0ut      4670000
7     Total Gaming     34300000
8    Dynamo Gaming     10000000
9      Snax Gaming      1410000
```

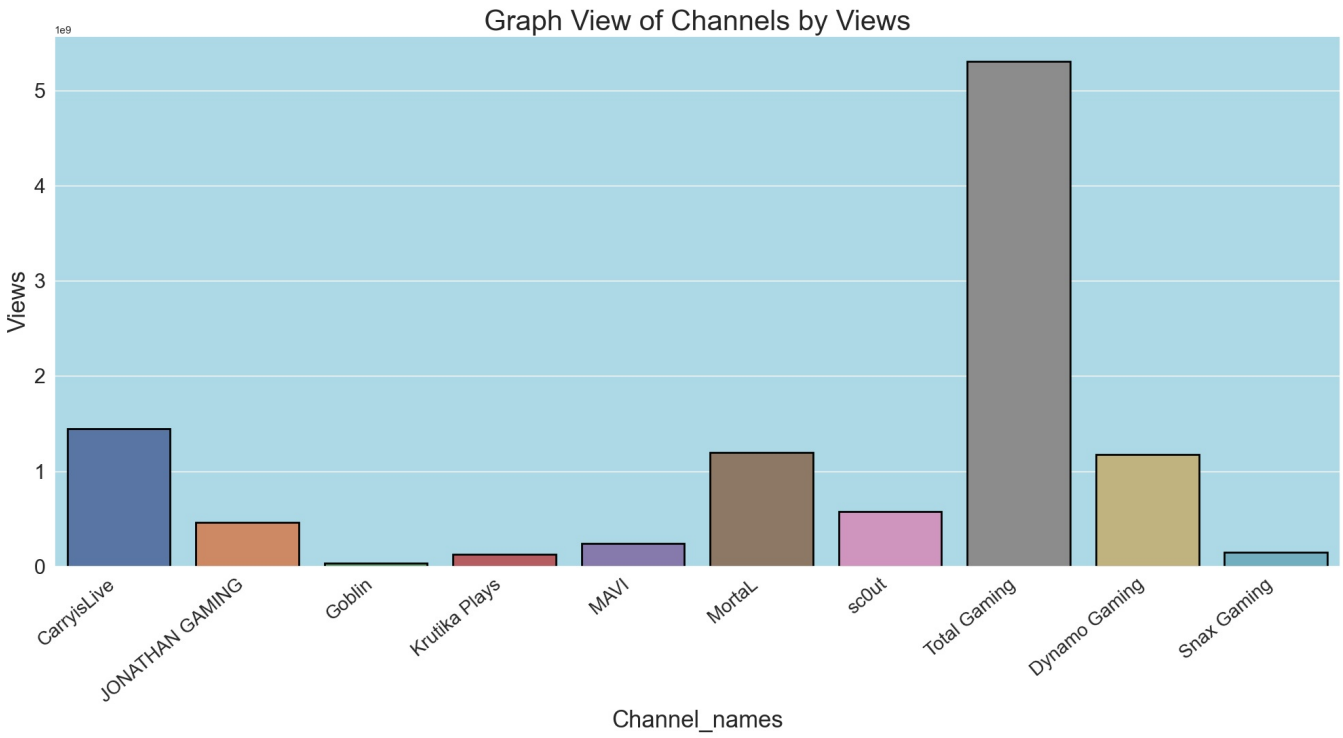


Graph View of Channels by Views

In [15]:
```
# creating graph for views observation

plt.figure(figsize=(20,11))
bg=plt.gca()
bg.set_facecolor("lightblue")
sns.set(rc={'figure.figsize':(18,8)})

Views_column = sns.barplot(x='Channel_name', y='Views', data=channel_data,edgecolor="black",linewidth=2) #used

plt.title("Graph View of Channels by Views",fontsize=30)
plt.xticks(fontsize=20)
plt.ylabel("Views",fontsize=25)
plt.xlabel("Channel_names",fontsize=25)
plt.yticks(fontsize=22)
plt.xticks(rotation=40,horizontalalignment="right",fontsize=20)
plt.tight_layout()

View_channel=channel_data
View_channel[["Channel_name","Views"]]   # fetched Channel_name and Views data
```

Out[15]:

	Channel_name	Views
0	CarryisLive	1450676239
1	JONATHAN GAMING	461416540
2	Goblin	34933939
3	Krutika Plays	126199779
4	MAVI	242664924
5	MortaL	1198491210
6	sc0ut	577290406
7	Total Gaming	5302731146
8	Dynamo Gaming	1175092765
9	Snax Gaming	153723026



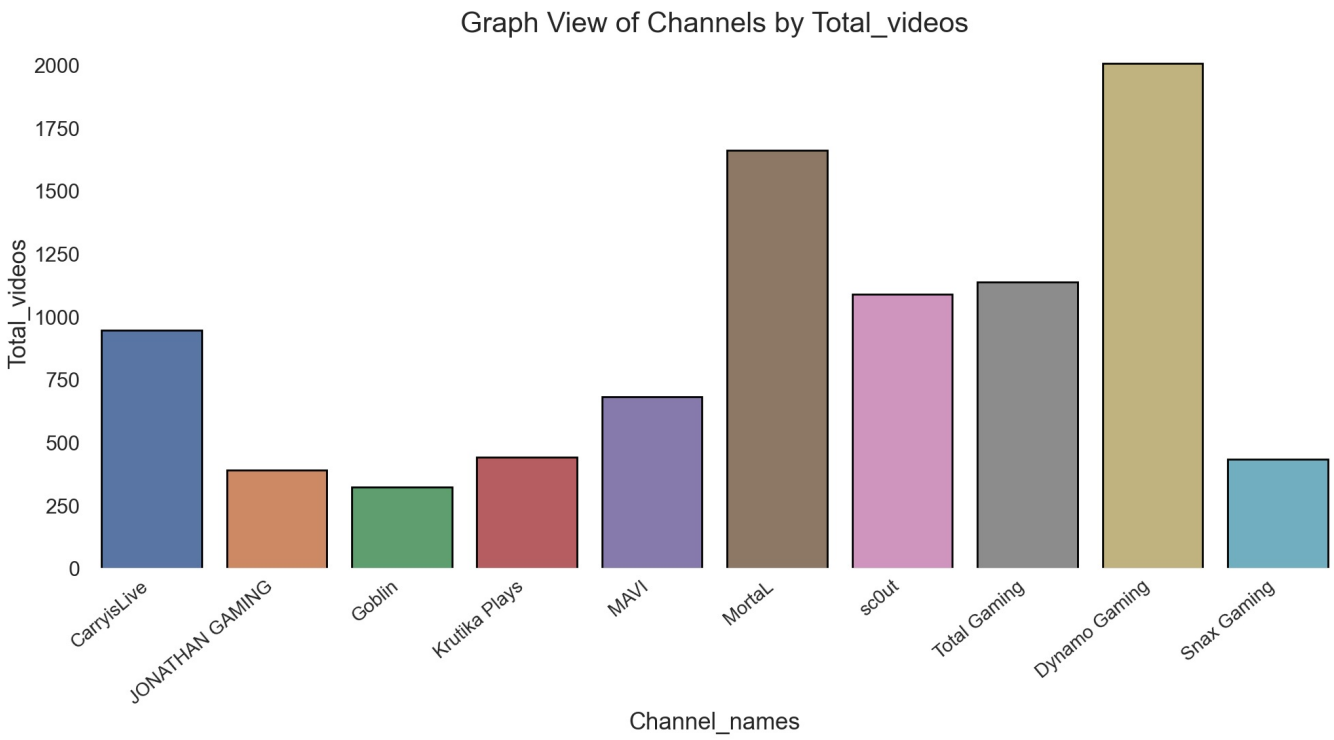
Graph View of Channels by Total_videos

In [16]:

```
# creating graph for Total_videos observation

plt.figure(figsize=(20,11))
bg=plt.gca()
bg.set_facecolor("white")
sns.set(rc={'figure.figsize':(18,8)})

Total_videos = sns.barplot(x='Channel_name', y='Total_videos', data=channel_data,edgecolor="black",linewidth=2)

plt.title("Graph View of Channels by Total_videos",fontsize=30)
plt.xticks(fontsize=20)
plt.ylabel("Total_videos",fontsize=25)
plt.xlabel("Channel_names",fontsize=25)
plt.yticks(fontsize=22)
plt.xticks(rotation=40,horizontalalignment="right",fontsize=20)
plt.tight_layout()

Total_videos_channel=channel_data
Total_videos_channel[["Channel_name","Total_videos"]]  # fetched Channel_name and Total_videos data
```

Out[16]:

	Channel_name	Total_videos
0	CarryisLive	947
1	JONATHAN GAMING	392
2	Goblin	324
3	Krutika Plays	444
4	MAVI	683
5	MortaL	1662
6	sc0ut	1091
7	Total Gaming	1138
8	Dynamo Gaming	2008
9	Snax Gaming	434



Channel Details of Subscribers ,Views and Total_videos

In [32]:
```
channel_name=(input("Enter the Channel_name "))#taking input

updated=channel_data[channel_data["Channel_name"]==channel_name]     #pulling data

Category=["Subscribers","Views","Total_videos"]   # selected targeted columns
Quantity=[]                            # created blank list

Subscribers=updated["Subscribers"].max()
Quantity.append(Subscribers)                       # append data into Quantity

Views=updated["Views"].max()
Quantity.append(Views)                   # append data into Quantity

Total_videos=updated["Total_videos"].max()
Quantity.append(Total_videos)             # append data into Quantity

                  # fetching data
# print("Subscribers COUNT : ",(updated["Subscribers"]/3)*100)
# print("\n")
# print("Views COUNT : ",(updated["Views"]/3)*100)
# print("\n")
# print("Total_videos COUNT :",(updated["Total_videos"]/3)*100)


                     #creating graph
plt.figure(figsize=(10,6))
my_explode=[0.3,2.0,2.2]

plt.pie(Quantity,labels=Category,shadow=True,autopct='%1.2f%%',
        explode=my_explode,wedgeprops={"edgecolor":"k"})
plt.xticks(fontsize=30)
plt.tight_layout()
plt.show()
```

```
Enter the Channel_name Total Gaming
```

Top Channel analysis

In [33]:
```
playlist_id = channel_data.loc[channel_data['Channel_name']=='Total Gaming', 'playlist_id'].iloc[0]  # for play
```

In [34]:
```
def get_video_ids(youtube, playlist_id):     # created function for playlist_id details

                                                                        #reference taken from youtube d

    request = youtube.playlistItems().list(
                part='contentDetails',
                playlistId = playlist_id,
                maxResults = 50)
    response = request.execute()

    video_ids = []                             # created blank list

    for i in range(len(response['items'])):
        video_ids.append(response['items'][i]['contentDetails']['videoId'])  #slicing

    next_page_token = response.get('nextPageToken')        #pulling data and stored in next_page_token
    more_pages = True

    while more_pages:
        if next_page_token is None:
            more_pages = False
        else:
            request = youtube.playlistItems().list(
                        part='contentDetails',
                        playlistId = playlist_id,
                        maxResults = 50,
                        pageToken = next_page_token)
            response = request.execute()

            for i in range(len(response['items'])):
                video_ids.append(response['items'][i]['contentDetails']['videoId'])

            next_page_token = response.get('nextPageToken')

    return video_ids
```

In [35]:
```
video_ids = get_video_ids(youtube, playlist_id)    #stored function parameter in video_ids variable
```

In [36]:
```
def get_video_details(youtube, video_ids):           # created function for video details
    all_video_stats = []

    for i in range(0, len(video_ids), 50):
        request = youtube.videos().list(
                    part='snippet,statistics',
                    id=','.join(video_ids[i:i+50]))
        response = request.execute()

        for video in response['items']:
            video_stats = dict(Title = video['snippet']['title'],
                               Published_date = video['snippet']['publishedAt'],
                               Views = video['statistics']['viewCount'],
                               Likes = video['statistics']['likeCount'],
                               Comments = video['statistics']['commentCount']
                               )
            all_video_stats.append(video_stats)

    return all_video_stats
```

In [37]:
```
video_details = get_video_details(youtube, video_ids)   #stored function parameter in video_details variable
```

In [38]:
```
video_data = pd.DataFrame(video_details)   # created data frame
video_data.info()      # fetched  data types
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1138 entries, 0 to 1137
Data columns (total 5 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   Title           1138 non-null   object
 1   Published_date  1138 non-null   object
 2   Views           1138 non-null   object
 3   Likes           1138 non-null   object
 4   Comments        1138 non-null   object
dtypes: object(5)
memory usage: 44.6+ KB
```

In [39]:
```
# converted data to numeric data type from object

video_data['Published_date'] = pd.to_datetime(video_data['Published_date']).dt.date
video_data['Views'] = pd.to_numeric(video_data['Views'])
video_data['Likes'] = pd.to_numeric(video_data['Likes'])
video_data['Views'] = pd.to_numeric(video_data['Views'])
video_data
```

Out[39]:

	Title	Published_date	Views	Likes	Comments
0	HACKER KI TARAH MARA☹ SOLO VS SQUAD OP GAMEPLA...	2023-01-20	787112	84066	4466
1	Uncle Ko Horror Train Se Bacha Liya | Choo Cho...	2023-01-18	574945	87442	811
2	END OF BLUE CRIMINAL (3 vs 6 GAMEPLAY) | GAREN...	2023-01-18	631686	63451	4080
3	SECRET CHARACTER SKILL IN FREE FIRE | GARENA F...	2023-01-16	1262268	223129	2323
4	CHOO CHOO CHARLES KA BAAP AJJUBHAI (HORROR GAM...	2023-01-13	548326	56768	5707
...	...	...	...	...	...
1133	Free Fire Hack, Meet Hacker Afridi, Diamond Ha...	2019-07-24	3195285	219246	4435
1134	I Meet Wall Hacker In Free Fire, Speed Hack, C...	2019-07-22	9546506	442639	6612
1135	26 Kill in Squad Match Grand Master - Total Ga...	2019-07-09	326288	26841	565
1136	8 Year Old Indian Best Free Fire Player - Gar...	2019-06-22	2621685	171317	3603
1137	Free Fire : EPIC WIN! Total 24 Kills SQUAD Mat...	2018-12-02	2426651	210481	17479

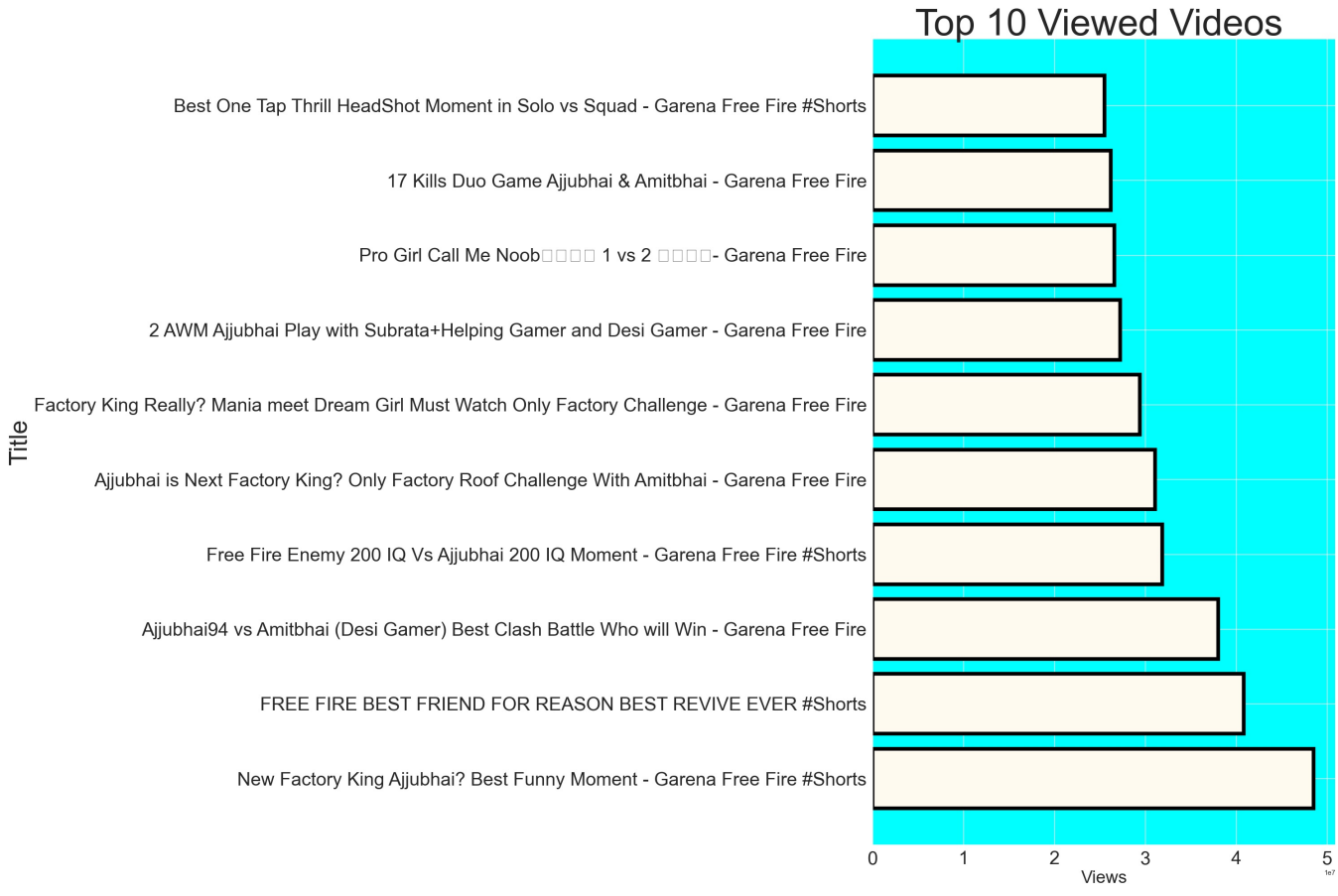
1138 rows × 5 columns

Top 10 Viewed Videos

In [75]:
```
topvideo_viewed=video_data.sort_values(by="Views",ascending=False)  #sorting data
top_viewed=topvideo_viewed.head(10)     # fetching top 10 data

print(top_viewed[["Title","Views"]].sort_values(by="Views",ascending=False))

                                # creating graph
plt.figure(figsize=(30,20))

bg=plt.gca()
bg.set_facecolor("aqua")

plt.barh(top_viewed["Title"],top_viewed["Views"],color="floralwhite",edgecolor="black",linewidth=6)  # used bar
                                                                                                      # matplotlib

plt.title("Top 10 Viewed Videos ",fontsize=60)
plt.xticks(fontsize=30)
plt.ylabel("Title",fontsize=40)
plt.xlabel("Views",fontsize=28)
plt.yticks(fontsize=30)
plt.tight_layout()
plt.show()
```

```
                                                 Title     Views
532   New Factory King Ajjubhai? Best Funny Moment -...  48497741
411   FREE FIRE BEST FRIEND FOR REASON BEST REVIVE E...  40835858
968   Ajjubhai94 vs Amitbhai (Desi Gamer) Best Clash...  38017717
363   Free Fire Enemy 200 IQ Vs Ajjubhai 200 IQ Mome...  31840950
752   Ajjubhai is Next Factory King? Only Factory Ro...  31052857
537   Factory King Really? Mania meet Dream Girl Mus...  29405563
980   2 AWM Ajjubhai Play with Subrata+Helping Gamer...  27211653
750   Pro Girl Call Me Noob☹आजा 1 vs 2 में- Garena ...  26565022
1062  17 Kills Duo Game Ajjubhai & Amitbhai - Garena...  26207060
487   Best One Tap Thrill HeadShot Moment in Solo vs...  25509076
```

```
C:\Users\Ajith Pauldurai\AppData\Local\Temp\ipykernel_9028\1667948043.py:20: UserWarning: Glyph 128545 (\N{POUTING FACE}) missing from current font.
  plt.tight_layout()
C:\Users\Ajith Pauldurai\AppData\Local\Temp\ipykernel_9028\1667948043.py:20: UserWarning: Glyph 2310 (\N{DEVANAGARI LETTER AA}) missing from current font.
  plt.tight_layout()
C:\Users\Ajith Pauldurai\AppData\Local\Temp\ipykernel_9028\1667948043.py:20: UserWarning: Matplotlib currently does not support Devanagari natively.
  plt.tight_layout()
C:\Users\Ajith Pauldurai\AppData\Local\Temp\ipykernel_9028\1667948043.py:20: UserWarning: Glyph 2332 (\N{DEVANAGARI LETTER JA}) missing from current font.
  plt.tight_layout()
C:\Users\Ajith Pauldurai\AppData\Local\Temp\ipykernel_9028\1667948043.py:20: UserWarning: Glyph 2366 (\N{DEVANAGARI VOWEL SIGN AA}) missing from current font.
  plt.tight_layout()
C:\Users\Ajith Pauldurai\AppData\Local\Temp\ipykernel_9028\1667948043.py:20: UserWarning: Glyph 2350 (\N{DEVANAGARI LETTER MA}) missing from current font.
  plt.tight_layout()
C:\Users\Ajith Pauldurai\AppData\Local\Temp\ipykernel_9028\1667948043.py:20: UserWarning: Glyph 2375 (\N{DEVANAGARI VOWEL SIGN E}) missing from current font.
  plt.tight_layout()
C:\Users\Ajith Pauldurai\AppData\Local\Temp\ipykernel_9028\1667948043.py:20: UserWarning: Glyph 2306 (\N{DEVANAGARI SIGN ANUSVARA}) missing from current font.
  plt.tight_layout()
C:\Users\Ajith Pauldurai\AppData\Local\Temp\ipykernel_9028\1667948043.py:20: UserWarning: Glyph 128293 (\N{FIRE}) missing from current font.
  plt.tight_layout()
D:\ajithkumar\anaconda\lib\site-packages\IPython\core\pylabtools.py:151: UserWarning: Glyph 128545 (\N{POUTING FACE}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
D:\ajithkumar\anaconda\lib\site-packages\IPython\core\pylabtools.py:151: UserWarning: Glyph 2310 (\N{DEVANAGARI LETTER AA}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
D:\ajithkumar\anaconda\lib\site-packages\IPython\core\pylabtools.py:151: UserWarning: Matplotlib currently does not support Devanagari natively.
  fig.canvas.print_figure(bytes_io, **kw)
D:\ajithkumar\anaconda\lib\site-packages\IPython\core\pylabtools.py:151: UserWarning: Glyph 2332 (\N{DEVANAGARI LETTER JA}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
D:\ajithkumar\anaconda\lib\site-packages\IPython\core\pylabtools.py:151: UserWarning: Glyph 2366 (\N{DEVANAGARI VOWEL SIGN AA}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
D:\ajithkumar\anaconda\lib\site-packages\IPython\core\pylabtools.py:151: UserWarning: Glyph 2350 (\N{DEVANAGARI LETTER MA}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
D:\ajithkumar\anaconda\lib\site-packages\IPython\core\pylabtools.py:151: UserWarning: Glyph 2375 (\N{DEVANAGARI VOWEL SIGN E}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
D:\ajithkumar\anaconda\lib\site-packages\IPython\core\pylabtools.py:151: UserWarning: Glyph 2306 (\N{DEVANAGARI SIGN ANUSVARA}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
D:\ajithkumar\anaconda\lib\site-packages\IPython\core\pylabtools.py:151: UserWarning: Glyph 128293 (\N{FIRE}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
```



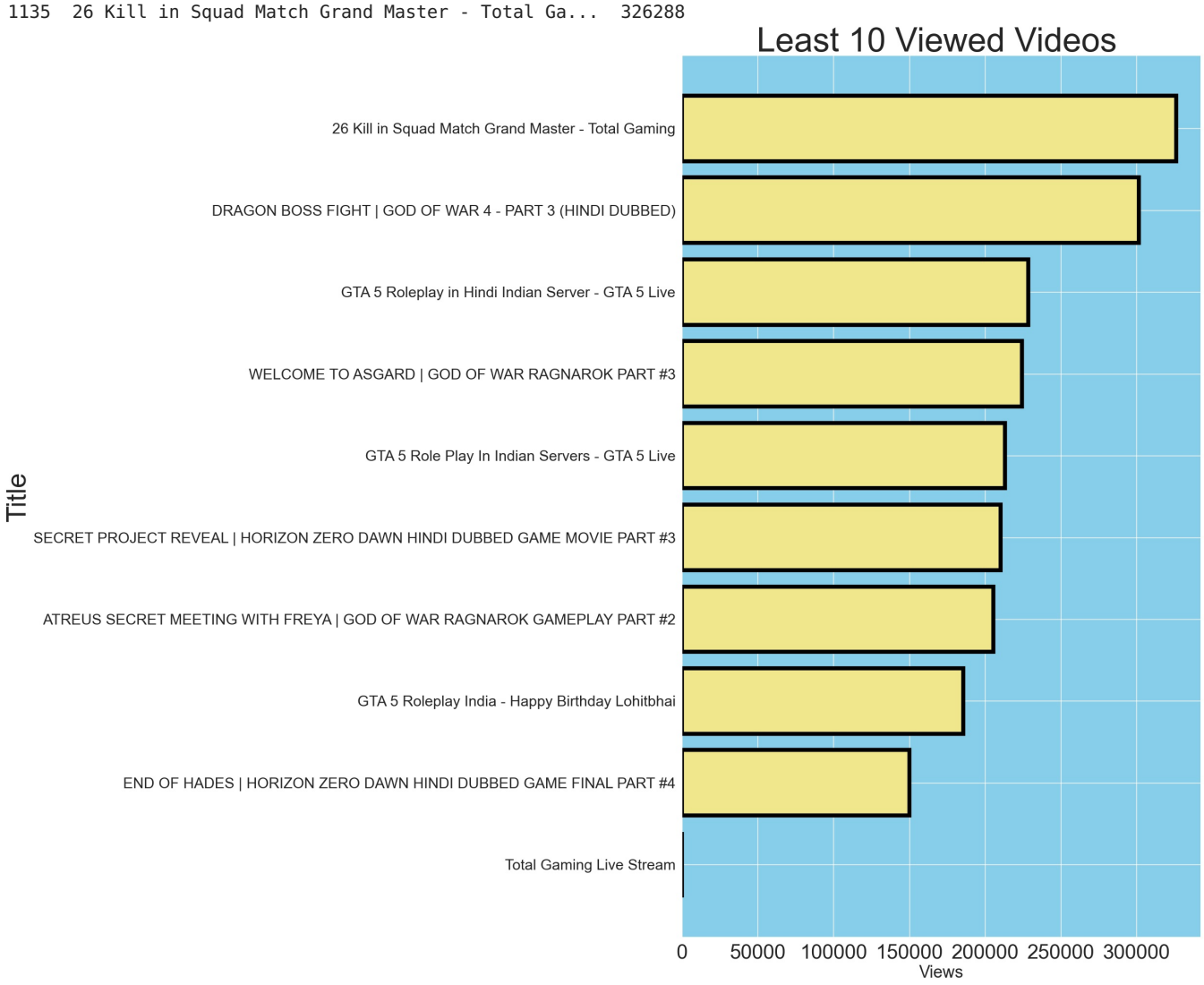
Least 10 viewed videos

In [76]:

```
Leastvideo_viewed=video_data.sort_values(by="Views",ascending=True)      # sorting data
Least_view=Leastvideo_viewed.head(10)  # fetched 10 value

print(Least_view[["Title","Views"]].sort_values(by="Views",ascending=True))

                                #creating graph
plt.figure(figsize=(25,20))
bg=plt.gca()
bg.set_facecolor("skyblue")

plt.barh(Least_view["Title"],Least_view["Views"],color="khaki",edgecolor="black",linewidth=6)
```

```
plt.title("Least 10 Viewed Videos ",fontsize=50)
plt.xticks(fontsize=30)
plt.ylabel("Title",fontsize=40)
plt.xlabel("Views",fontsize=25)
plt.yticks(fontsize=22)

plt.tight_layout()
plt.show()
```

```
                                                  Title   Views
895                             Total Gaming Live Stream       0
23    END OF HADES | HORIZON ZERO DAWN HINDI DUBBED ...  150260
1117    GTA 5 Roleplay India - Happy Birthday Lohitbhai  185759
33    ATREUS SECRET MEETING WITH FREYA | GOD OF WAR ...  205646
30    SECRET PROJECT REVEAL | HORIZON ZERO DAWN HIND...  210449
1115     GTA 5 Role Play In Indian Servers - GTA 5 Live  213319
32      WELCOME TO ASGARD | GOD OF WAR RAGNAROK PART #3  224458
1104  GTA 5 Roleplay in Hindi Indian Server - GTA 5 ...  228526
13    DRAGON BOSS FIGHT | GOD OF WAR 4 - PART 3 (HIN...  301593
1135  26 Kill in Squad Match Grand Master - Total Ga...  326288
```



In [46]:
```
            #formatting in date time format
video_data["Published_date"]=pd.to_datetime(video_data["Published_date"])
video_data["Years"]=video_data["Published_date"].dt.year
video_data["Months"]=video_data["Published_date"].dt.month              #extract data in year , months and week
video_data["Days"]=video_data["Published_date"].dt.day
video_data["WeekDays"]=video_data["Published_date"].dt.day_name()
```

In [48]:
```
video_data      # fetched data
```

Out[48]:

	Title	Published_date	Views	Likes	Comments	Years	Months	Days	WeekDays
0	HACKER KI TARAH MARA☹ SOLO VS SQUAD OP GAMEPLA...	2023-01-20	787112	84066	4466	2023	1	20	Friday
1	Uncle Ko Horror Train Se Bacha Liya | Choo Cho...	2023-01-18	574945	87442	811	2023	1	18	Wednesday
2	END OF BLUE CRIMINAL (3 vs 6 GAMEPLAY) | GAREN...	2023-01-18	631686	63451	4080	2023	1	18	Wednesday
3	SECRET CHARACTER SKILL IN FREE FIRE | GARENA F...	2023-01-16	1262268	223129	2323	2023	1	16	Monday
4	CHOO CHOO CHARLES KA BAAP AJJUBHAI (HORROR GAM...	2023-01-13	548326	56768	5707	2023	1	13	Friday
...	...	...	...	...	...	...	...	...	...
1133	Free Fire Hack, Meet Hacker Afridi, Diamond Ha...	2019-07-24	3195285	219246	4435	2019	7	24	Wednesday
1134	I Meet Wall Hacker In Free Fire, Speed Hack, C...	2019-07-22	9546506	442639	6612	2019	7	22	Monday
1135	26 Kill in Squad Match Grand Master - Total Ga...	2019-07-09	326288	26841	565	2019	7	9	Tuesday
1136	8 Year Old Indian Best Free Fire Player - Gar...	2019-06-22	2621685	171317	3603	2019	6	22	Saturday
1137	Free Fire : EPIC WIN! Total 24 Kills SQUAD Mat...	2018-12-02	2426651	210481	17479	2018	12	2	Sunday

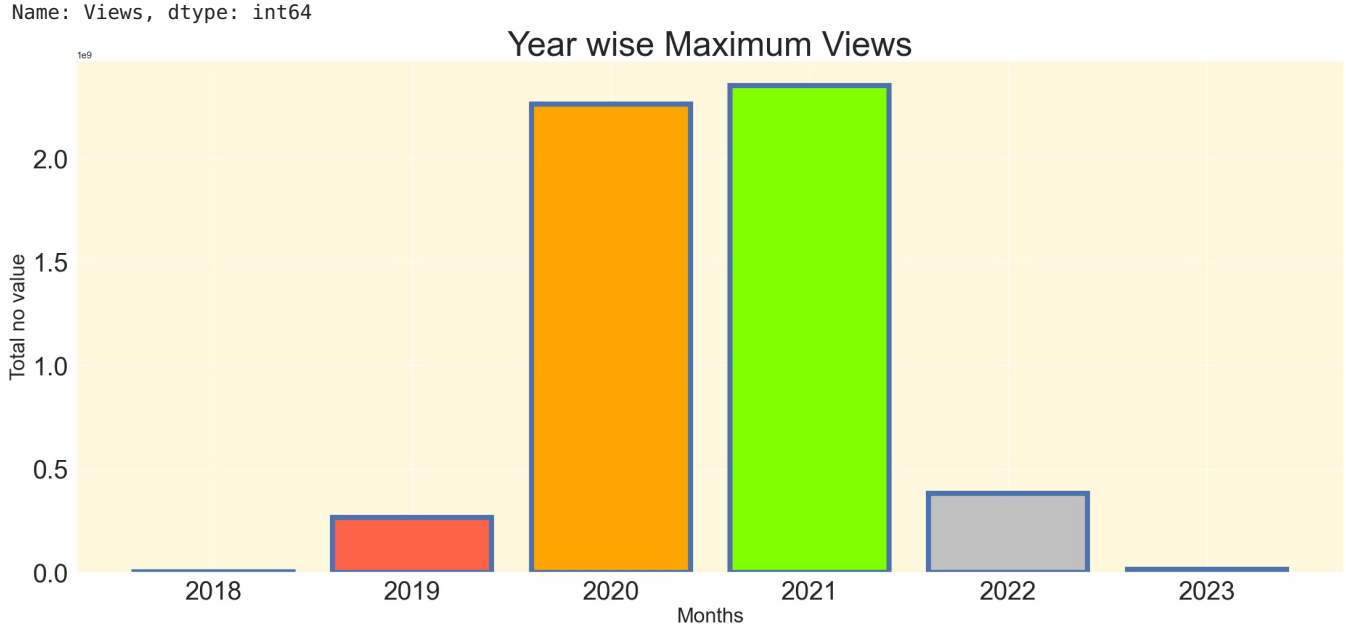
1138 rows × 9 columns

Year wise Maximum Views

In [49]:
```
yearwise=video_data.groupby("Years")
```

In [52]:
```
view=yearwise["Views"].sum()                # total count
view_updated=view.sort_values()   #sorting
updated=view_updated.index      #pulling index and stored into updated
year_wise_viewed=pd.Series(updated)   # created pandas Series and stored in year_wise_viewed

    # creating graph for Year wise maximum view

plt.figure(figsize=(22,10))
bg=plt.gca()
bg.set_facecolor("cornsilk")

plt.bar(year_wise_viewed,view_updated,color=["chartreuse","cyan","tomato","silver","orange",],edgecolor="b",lin

plt.title("Year wise Maximum Views",fontsize=40)
plt.xlabel("Months",fontsize=24)
plt.ylabel("Total no value",fontsize=24)
plt.xticks(fontsize=30)
plt.yticks(fontsize=30)
plt.tight_layout()

print("Details : ",view_updated)
```

```
Details :  Years
2018       2426651
2023      14973145
2019     266455070
2022     383234894
2020    2264000634
2021    2353465167
Name: Views, dtype: int64
```



Maxmium Likes Video Title

In [54]:
```
max_Likes=video_data.sort_values(by="Likes",ascending=True) # sorting
max_Likes[["Title","Likes"]].tail(1).transpose()   # transposed
```

Out[54]:

	411
Title	FREE FIRE BEST FRIEND FOR REASON BEST REVIVE E...
Likes	3164352

Minimum likes Video Title

In [55]:
```
min_Likes=video_data.sort_values(by="Likes",ascending=True)      # sorting
min_Likes[["Title","Likes"]].head(1).transpose()      # transposed
```

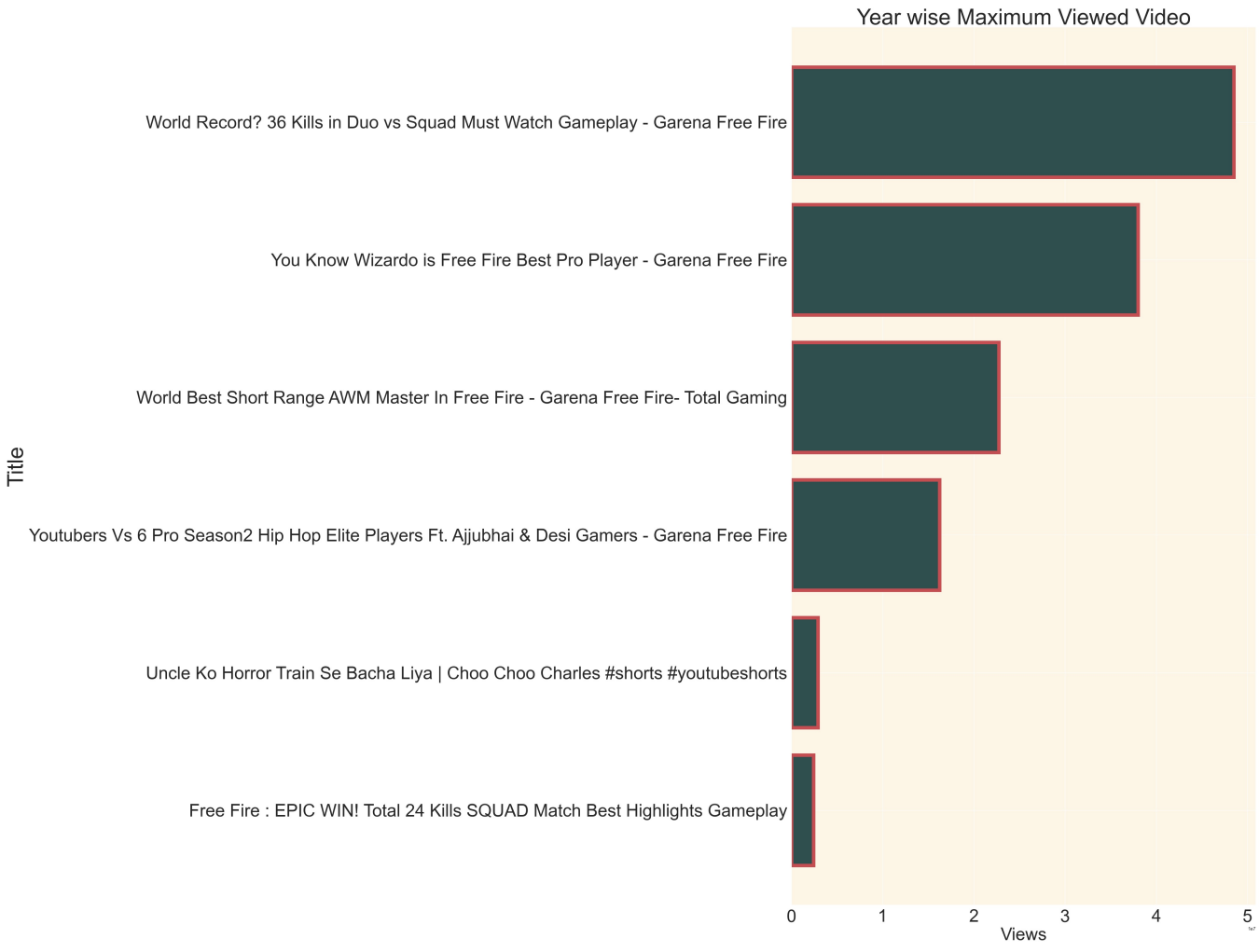
Out[55]:

	895
Title	Total Gaming Live Stream
Likes	224

Year wise Maximum Viewed Video Title

In [57]:
```
views=video_data.groupby("Years")        #grouping Years column

Year_wise_viewed_video=views[["Title","Views"]].max()
Year_wise_viewed_video=Year_wise_viewed_video.sort_values(by="Views",ascending=True)    # sorting
x_axis=Year_wise_viewed_video.index
x_axis=pd.Series(x_axis)
y_axis=Year_wise_viewed_video["Views"]
print(Year_wise_viewed_video)
                                  # creating graph
plt.figure(figsize=(40,30))
bg=plt.gca()
bg.set_facecolor("oldlace")
plt.barh(Year_wise_viewed_video["Title"],y_axis,color="darkslategrey",edgecolor="r",linewidth=8)
plt.title("Year wise Maximum Viewed Video",fontsize=50)
plt.xticks(fontsize=40)
plt.ylabel("Title",fontsize=50)
plt.xlabel("Views",fontsize=40)
plt.yticks(fontsize=40)
plt.tight_layout()
plt.show()
```

```
                                                Title     Views
Years
2018   Free Fire : EPIC WIN! Total 24 Kills SQUAD Mat...   2426651
2023   Uncle Ko Horror Train Se Bacha Liya | Choo Cho...   2912037
2022   Youtubers Vs 6 Pro Season2 Hip Hop Elite Playe...  16235725
2019   World Best Short Range AWM Master In Free Fire...  22746821
2020   You Know Wizardo is Free Fire Best Pro Player ...  38017717
2021   World Record? 36 Kills in Duo vs Squad Must Wa...  48497741
```

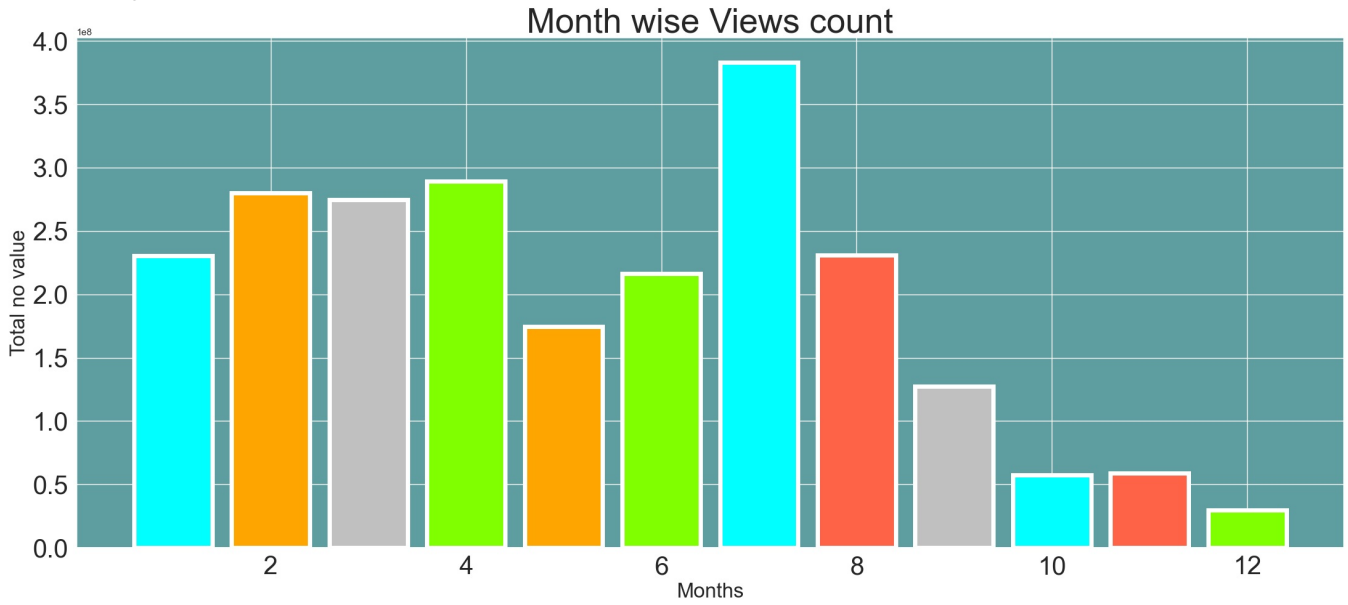


Month wise Views count

In [77]:
```
year=int(input("Enter the year "))
data=video_data[video_data["Years"]==year]
update=data.groupby("Months")
view=update["Views"].sum()
month_wise=view.sort_values()
updated_index=b.index
Month_wise_view=pd.Series(updated_index)

plt.figure(figsize=(22,10))
bg=plt.gca()
bg.set_facecolor("cadetblue")
plt.bar(Month_wise_view,month_wise,color=["chartreuse","cyan","tomato","silver","orange",],edgecolor="white",li
plt.title("Month wise Views count",fontsize=40)
```

```
plt.xlabel("Months",fontsize=24)
plt.ylabel("Total no value",fontsize=24)
plt.xticks(fontsize=30)
plt.yticks(fontsize=30)
plt.tight_layout()
```

```
Enter the year 2021
```

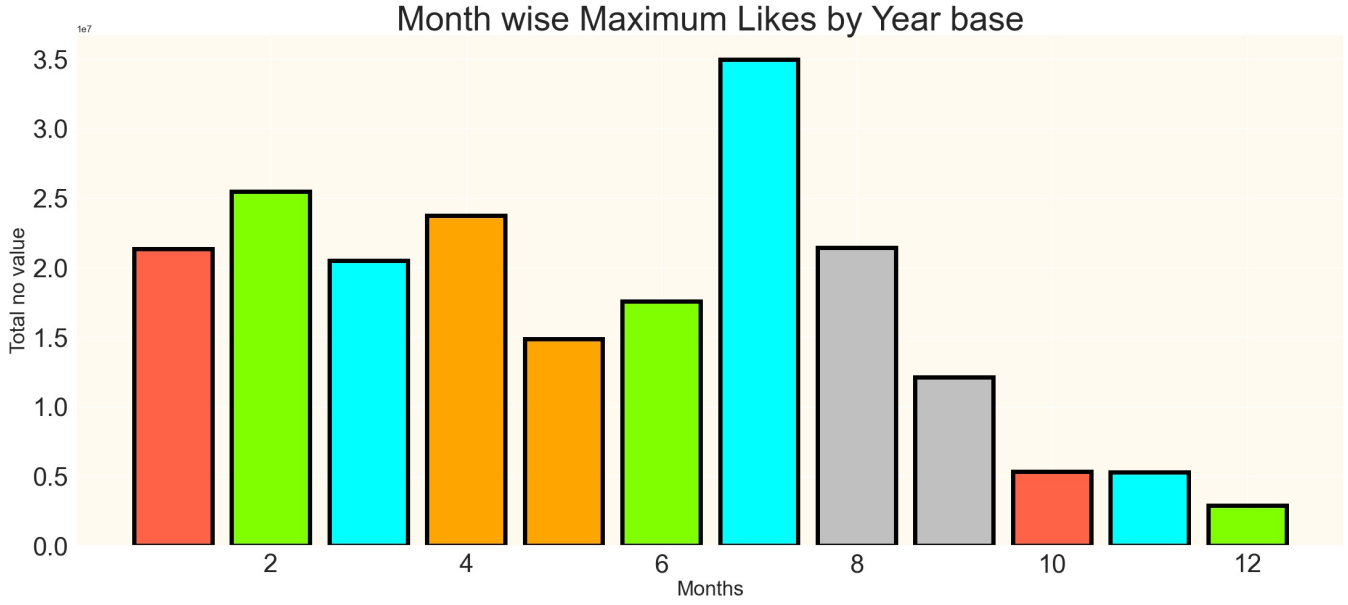


Month wise Maximum Likes by Year base

In [63]:

```
year=int(input("Enter the year "))                  # input taken
data=video_data[video_data["Years"]==year]       # compare input
update=data.groupby("Months")  # grouping method
updated=update["Likes"].sum()
month_wise=updated.sort_values() # sorting
month_wise_1=month_wise.index   #index value stored in month_wise_1
month_wise_likes=pd.Series(month_wise_1)
                        # created graph
plt.figure(figsize=(22,10))
bg=plt.gca()
bg.set_facecolor("floralwhite")

plt.bar(month_wise_likes,month_wise,color=["chartreuse","cyan","tomato","silver","orange",],edgecolor="black",l

plt.title("Month wise Maximum Likes by Year base",fontsize=40)
plt.xlabel("Months",fontsize=24)
plt.ylabel("Total no value",fontsize=24)
plt.xticks(fontsize=30)
plt.yticks(fontsize=30)
plt.tight_layout()
```

```
Enter the year 2021
```

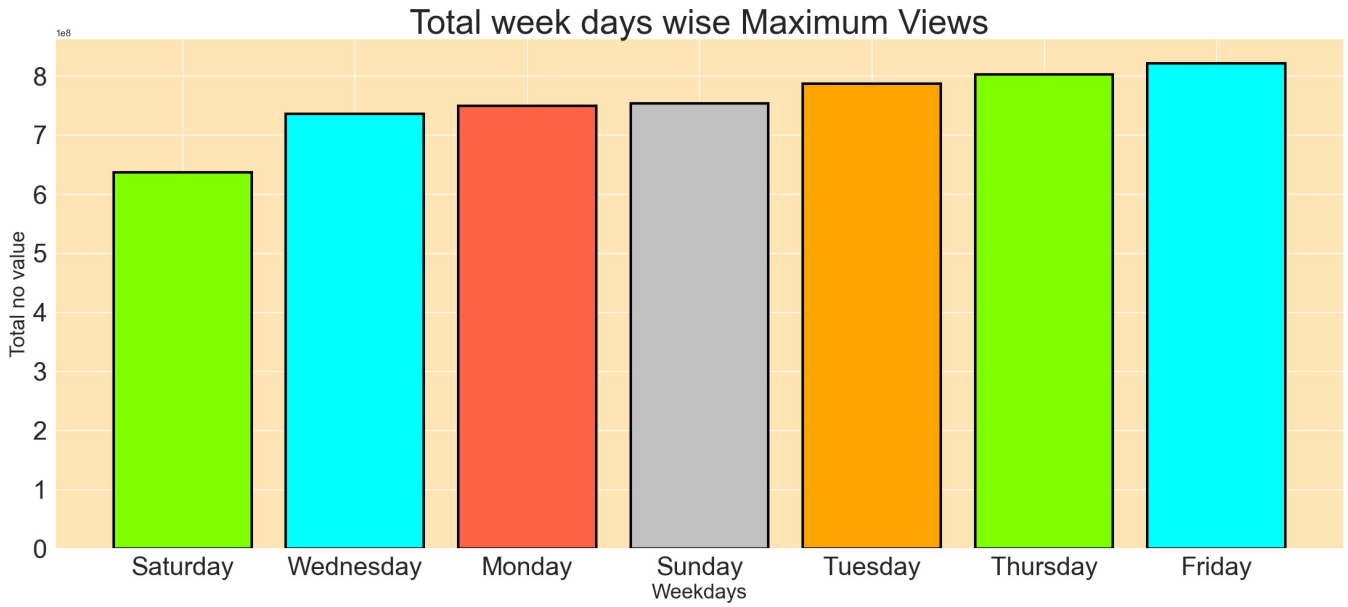


Total week days wise Maximum Views

In [69]:

```
update=video_data.groupby("WeekDays")        # grouping
```

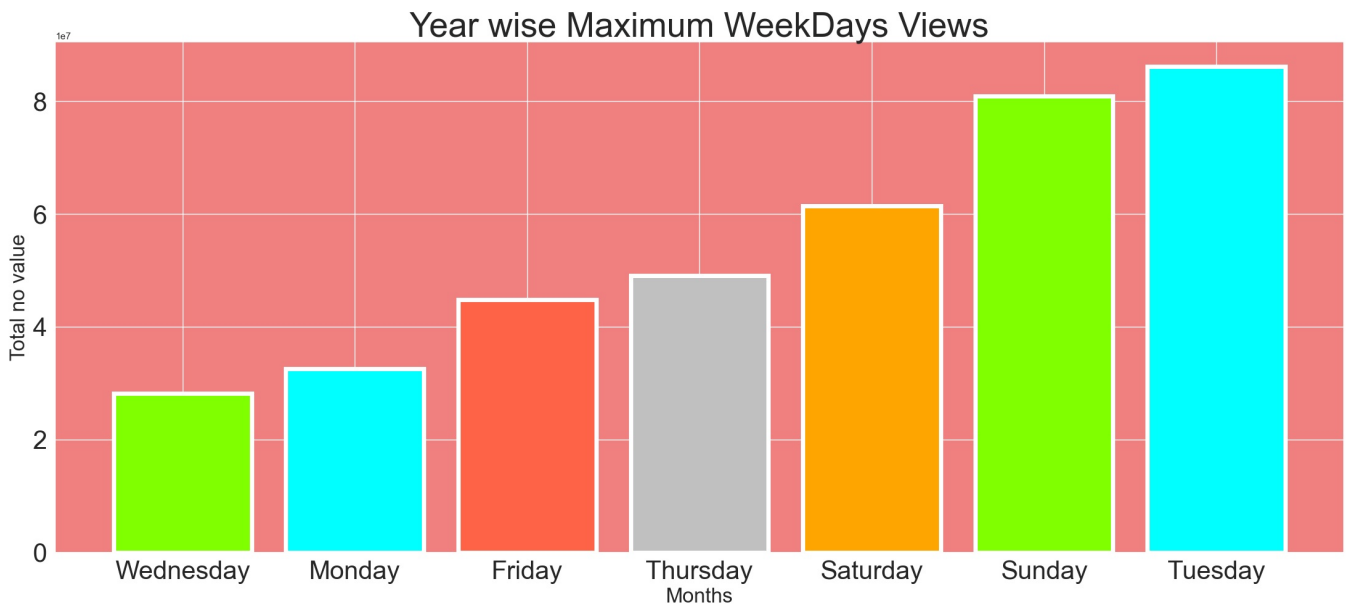
In [68]:
```
update=video_data.groupby("WeekDays")          # grouping
update1=update["Views"].sum()
update1=update1.sort_values()          #sorting
x_axis=update1.index       # pulling index
x_axis=pd.Series(x_axis) # created pandas series

                # created graph
plt.figure(figsize=(22,10))
bg=plt.gca()
bg.set_facecolor("moccasin")

plt.bar(x_axis,update1,color=["chartreuse","cyan","tomato","silver","orange",],edgecolor="black",linewidth=3)

plt.title("Total week days wise Maximum Views",fontsize=40)
plt.xlabel("Weekdays",fontsize=24)
plt.ylabel("Total no value",fontsize=24)
plt.xticks(fontsize=30)
plt.yticks(fontsize=30)
plt.tight_layout()
```



Year wise Maximum WeekDays Views

In [70]:
```
year=int(input("Enter the year "))              # ask for input

data=video_data[video_data["Years"]==year]  # comparing input
update=data.groupby("WeekDays")    #grouping
update1=update["Views"].sum()
update1=update1.sort_values()
x_axis=update1.index
x_axis=pd.Series(x_axis)

                                # creating graph
plt.figure(figsize=(22,10))
bg=plt.gca()
bg.set_facecolor("lightcoral")

plt.bar(x_axis,update1,color=["chartreuse","cyan","tomato","silver","orange",],edgecolor="white",linewidth=5)

plt.title("Year wise Maximum WeekDays Views",fontsize=40)
plt.xlabel("Months",fontsize=24)
plt.ylabel("Total no value",fontsize=24)
plt.xticks(fontsize=30)
plt.yticks(fontsize=30)
plt.tight_layout()
```

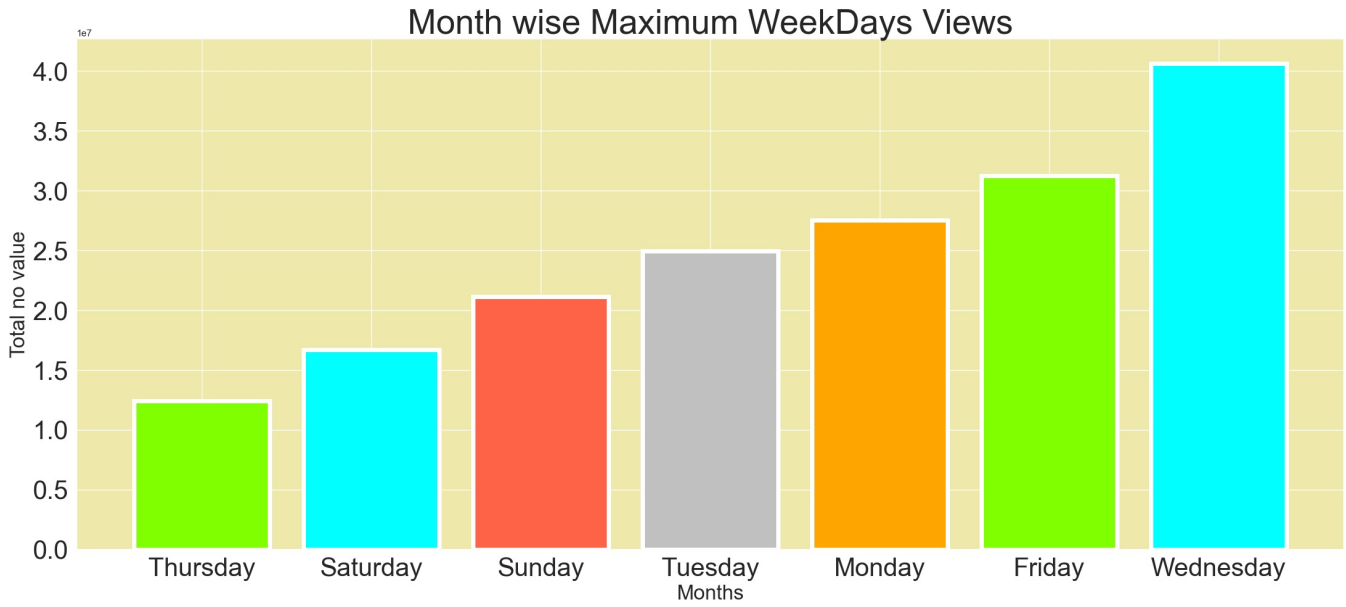
```
Enter the year 2022
```



Month wise Maximum WeekDays Views

In [71]:
```
# input taken
year=int(input("Enter the year "))
mon=int(input("Enter the month "))
                                    # comparing
data=video_data[(video_data["Years"]==year) & (video_data["Months"]==mon)]
update=data.groupby("WeekDays")             # grouping
update1=update["Views"].sum()
update1=update1.sort_values()
x_axis=update1.index             #   pulling index
x_axis=pd.Series(x_axis)

                           # creating graph
plt.figure(figsize=(22,10))
bg=plt.gca()
bg.set_facecolor("palegoldenrod")
plt.bar(x_axis,update1,color=["chartreuse","cyan","tomato","silver","orange",],edgecolor="white",linewidth=5)
plt.title("Month wise Maximum WeekDays Views",fontsize=40)
plt.xlabel("Months",fontsize=24)
plt.ylabel("Total no value",fontsize=24)
plt.xticks(fontsize=30)
plt.yticks(fontsize=30)
plt.tight_layout()
```

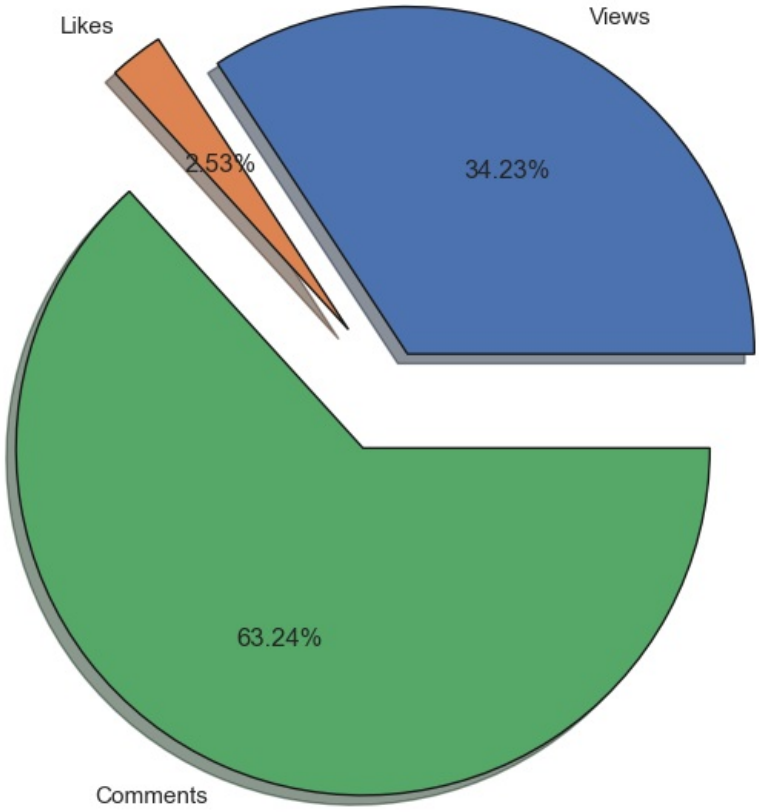
```
Enter the year 2021
Enter the month 5
```



Single day data of Views , Likes and Comments

In [72]:
```
#taking input
year=int(input("Enter the year "))
month=int(input("Enter the month number "))
date=int(input("Enter the date"))


data=video_data[(video_data["Years"]==year) & (video_data["Months"]==month) & (video_data["Days"]==date)]  #pul
Category=["Views","Likes","Comments"]
Quantity=[]                                          #empty list created
Views=data["Views"].mean()
Quantity.append(Views)
Likes=data["Likes"].mean()
Quantity.append(Likes)
Comments=data["Comments"].mean()
Quantity.append(Comments)
                                                     #creating graph

plt.figure(figsize=(18,6))
my_explode=[0.1,0.2,0.2]
plt.pie(Quantity,labels=Category,shadow=True,autopct='%1.2f%%',
        explode=my_explode,wedgeprops={"edgecolor":"k"})
plt.tight_layout()
plt.show()
```

```
Enter the year 2021
Enter the month number 4
Enter the date13
```



Top comments Video Datails

In [73]:
```
data=video_data.sort_values(by="Comments",ascending=False)
data.head(1).transpose()
```

Out[73]:

	652
Title	Solo vs Squad Best AWM, Mp40, SCAR and SHOTGUN...
Published_date	2021-01-06 00:00:00
Views	2833689
Likes	309071
Comments	9996
Years	2021
Months	1
Days	6
WeekDays	Wednesday

Least comments Video Details

In [74]:
```
data=video_data.sort_values(by="Comments",ascending=False)
data.tail(1).transpose()
```

Out[74]:

	895
Title	Total Gaming Live Stream
Published_date	2020-08-08 00:00:00
Views	0
Likes	224
Comments	0
Years	2020
Months	8
Days	8
WeekDays	Saturday

In [ ]: