Disney movies and Box Office Success

1. The dataset

Walt Disney Studios is the foundation on which The Walt Disney Company was built. The Studios has produced more than 600 films since their debut film, Snow White and the Seven Dwarfs in 1937. While many of its films were big hits, some of them were not. In this notebook, we will explore a dataset of Disney movies and analyze what contributes to the success of Disney movies.

In [3]:
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import calendar
gross = pd.read_csv("https://raw.githubusercontent.com/gayathriravi2111997/project-try/main/disney_movies.csv", parse_dates=["release_date"])
gross
```

Out[3]:

	movie_title	release_date	genre	mpaa_rating	total_gross	inflation_adjusted_gross
0	Snow White and the Seven Dwarfs	1937-12-21	Musical	G	184925485	5228953251
1	Pinocchio	1940-02-09	Adventure	G	84300000	2188229052
2	Fantasia	1940-11-13	Musical	G	83320000	2187090808
3	Song of the South	1946-11-12	Adventure	G	65000000	1078510579
4	Cinderella	1950-02-15	Drama	G	85000000	920608730
...	...	...	...	...	...	...
574	The Light Between Oceans	2016-09-02	Drama	PG-13	12545979	12545979
575	Queen of Katwe	2016-09-23	Drama	PG	8874389	8874389
576	Doctor Strange	2016-11-04	Adventure	PG-13	232532923	232532923
577	Moana	2016-11-23	Adventure	PG	246082029	246082029
578	Rogue One: A Star Wars Story	2016-12-16	Adventure	PG-13	529483936	529483936

579 rows × 6 columns

In [5]:
```python
len(gross['movie_title'])
```

Out[5]:
```
579
```

In [8]:
```python
pd.to_datetime(gross['release_date'],format='%Y-%m-%d').dt.year.min(),pd.to_datetime(gross['release_date'],format='%Y-%m-%d').dt.year.max()
```

Out[8]:
```
(1937, 2016)
```

We are going to explore 579 movies of Disney released between 1937 -2016 .

Quality Check

In [9]:
```python
gross.isna().sum()/gross.shape[0]
```

Out[9]:
```
movie_title                 0.000000
release_date                0.000000
genre                       0.029361
mpaa_rating                 0.096718
total_gross                 0.000000
inflation_adjusted_gross    0.000000
dtype: float64
```

Columns genre and mpaa rating have null values but they are less than 10 % and will not have much impact.

In [11]:
```python
gross.duplicated().sum()
```

Out[11]:
```
0
```

There are no duplicates.

In [12]:
```python
gross['movie_title'].value_counts().loc[lambda x:x>1]
```

Out[12]:
```
The Jungle Book    3
Freaky Friday      2
Cinderella         2
Bad Company        2
101 Dalmatians     2
Name: movie_title, dtype: int64
```

There are 5 movie's which have multiple entries. We need to be careful while summarizing.

Create Features

In [13]:
```python
gross['release_year']=pd.to_datetime(gross['release_date']).dt.year
gross['release_month']=pd.to_datetime(gross['release_date']).dt.month.apply(lambda x:calendar.month_abbr[x])
```

In [14]:
```python
gross.head()
```

Out[14]:

	movie_title	release_date	genre	mpaa_rating	total_gross	inflation_adjusted_gross	release_year	release_month
0	Snow White and the Seven Dwarfs	1937-12-21	Musical	G	184925485	5228953251	1937	Dec
1	Pinocchio	1940-02-09	Adventure	G	84300000	2188229052	1940	Feb
2	Fantasia	1940-11-13	Musical	G	83320000	2187090808	1940	Nov
3	Song of the South	1946-11-12	Adventure	G	65000000	1078510579	1946	Nov
4	Cinderella	1950-02-15	Drama	G	85000000	920608730	1950	Feb

1. Top ten movies at the box office

Let's started by exploring the data. We will check which are the 10 Disney movies that have earned the most at the box office. We can do this by sorting movies by their inflation-adjusted gross (we will call it adjusted gross from this point onward).

In [16]:
```python
gross.sort_values(by="inflation_adjusted_gross", ascending=False).head(10)
```

Out[16]:

	movie_title	release_date	genre	mpaa_rating	total_gross	inflation_adjusted_gross	release_year	release_month
0	Snow White and the Seven Dwarfs	1937-12-21	Musical	G	184925485	5228953251	1937	Dec
1	Pinocchio	1940-02-09	Adventure	G	84300000	2188229052	1940	Feb
2	Fantasia	1940-11-13	Musical	G	83320000	2187090808	1940	Nov
8	101 Dalmatians	1961-01-25	Comedy	G	153000000	1362870985	1961	Jan
6	Lady and the Tramp	1955-06-22	Drama	G	93600000	1236035515	1955	Jun
3	Song of the South	1946-11-12	Adventure	G	65000000	1078510579	1946	Nov
564	Star Wars Ep. VII: The Force Awakens	2015-12-18	Adventure	PG-13	936662225	936662225	2015	Dec
4	Cinderella	1950-02-15	Drama	G	85000000	920608730	1950	Feb
13	The Jungle Book	1967-10-18	Musical	Not Rated	141843000	789612346	1967	Oct
179	The Lion King	1994-06-15	Adventure	G	422780140	761640898	1994	Jun

1. Movie genre trend

From the top 10 movies above, it seems that some genres are more popular than others. So, we will check which genres are growing stronger in popularity. To do this, we will group movies by genre and then by year to see the adjusted gross of each genre in each year.

In [17]:
```python
gross['release_year'] = pd.DatetimeIndex(gross['release_date']).year
group = gross.groupby(['genre', 'release_year']).mean()
genre_yearly = group.reset_index()
genre_yearly.head(10)
```

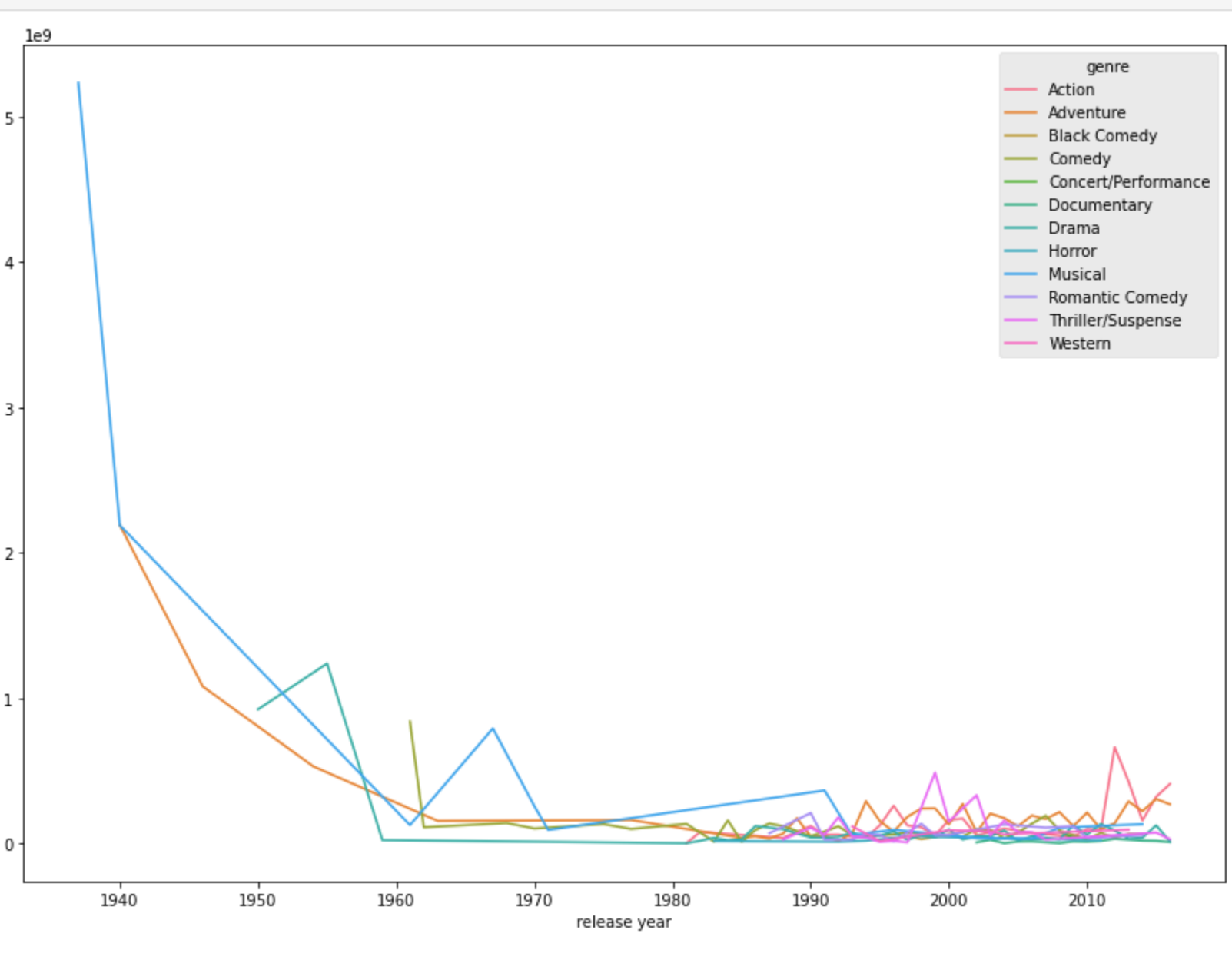
Out[17]:

	genre	release_year	total_gross	inflation_adjusted_gross
0	Action	1981	0.0	0.0
1	Action	1982	26918576.0	77184895.0
2	Action	1988	17577696.0	36053517.0
3	Action	1990	59249588.5	118358772.0
4	Action	1991	28924936.5	57918572.5
5	Action	1992	29028000.0	58965304.0
6	Action	1993	21943553.5	44682157.0
7	Action	1994	19180582.0	39545796.0
8	Action	1995	63037553.5	122162426.5
9	Action	1996	135281096.0	257755262.5

1. Visualize the genre popularity trend

We will make a plot out of these means of groups to better see how box office revenues have changed over time.

In [18]:
```python
import matplotlib.pyplot as plt
import seaborn as sns
fig, ax = plt.subplots(figsize=(14,10))
plt.style.use('ggplot')
sns.lineplot(x="release_year", y="inflation_adjusted_gross",
             data=genre_yearly, hue='genre')
plt.xlabel("release year")
plt.ylabel("inflation_adjusted_gross")
plt.show()
```



1. Data transformation

The line plot supports our belief that some genres are growing faster in popularity than others. For Disney movies, Action and Adventure genres are growing the fastest. Next, we will build a linear regression model to understand the relationship between genre and box office gross. Since linear regression requires numerical variables and the genre variable is a categorical variable, we'll use a technique called one-hot encoding to convert the categorical variables to numerical. This technique transforms each category value into a new column and assigns a 1 or 0 to the column. For this dataset, there will be 11 dummy variables, one for each genre except the action genre which we will use as a baseline. For example, if a movie is an adventure movie, like The Lion King, the adventure variable will be 1 and other dummy variables will be 0. Since the action genre is our baseline, if a movie is an action movie, such as The Avengers, all dummy variables will be 0.

In [19]:
```python
genre_dummies =  pd.get_dummies(gross['genre'], drop_first=True)
genre_dummies.head()
```

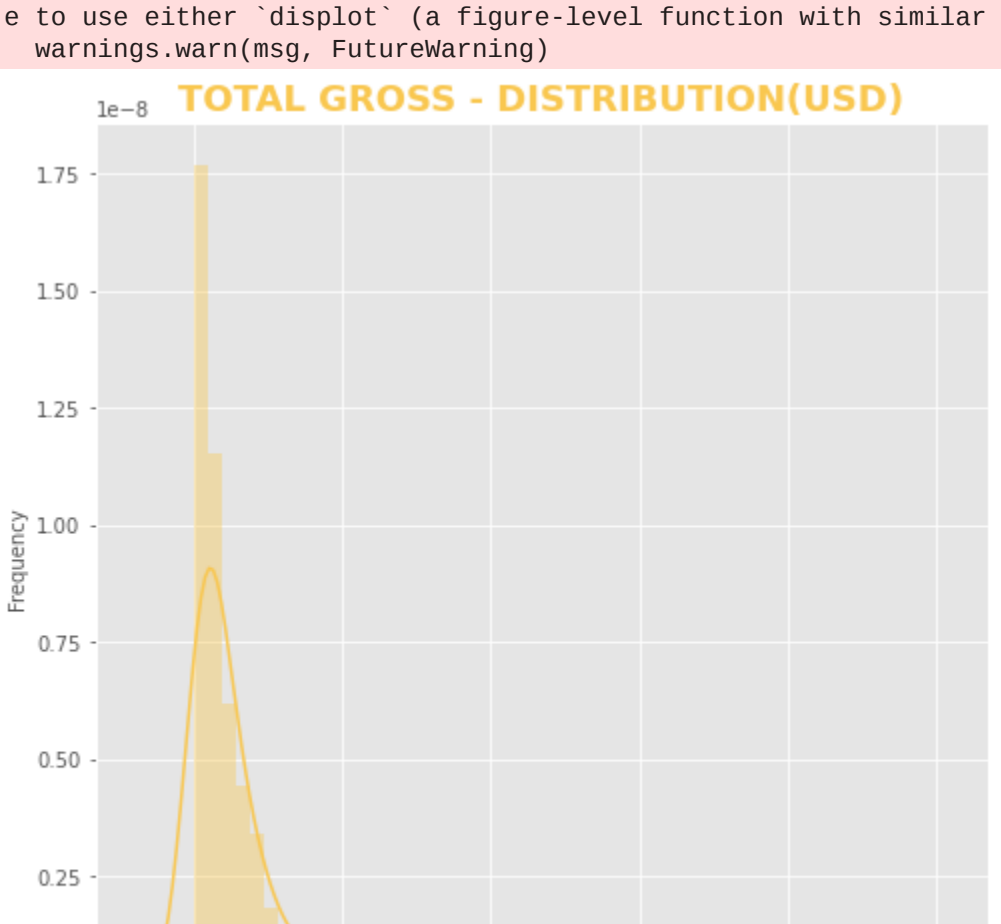
Out[19]:

	Adventure	Black Comedy	Comedy	Concert/Performance	Documentary	Drama	Horror	Musical	Romantic Comedy	Thriller/Suspense	Western
0	0	0	0	0	0	0	0	1	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	1	0	0	0
3	1	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	1	0	0	0	0	0

Gross Distribution

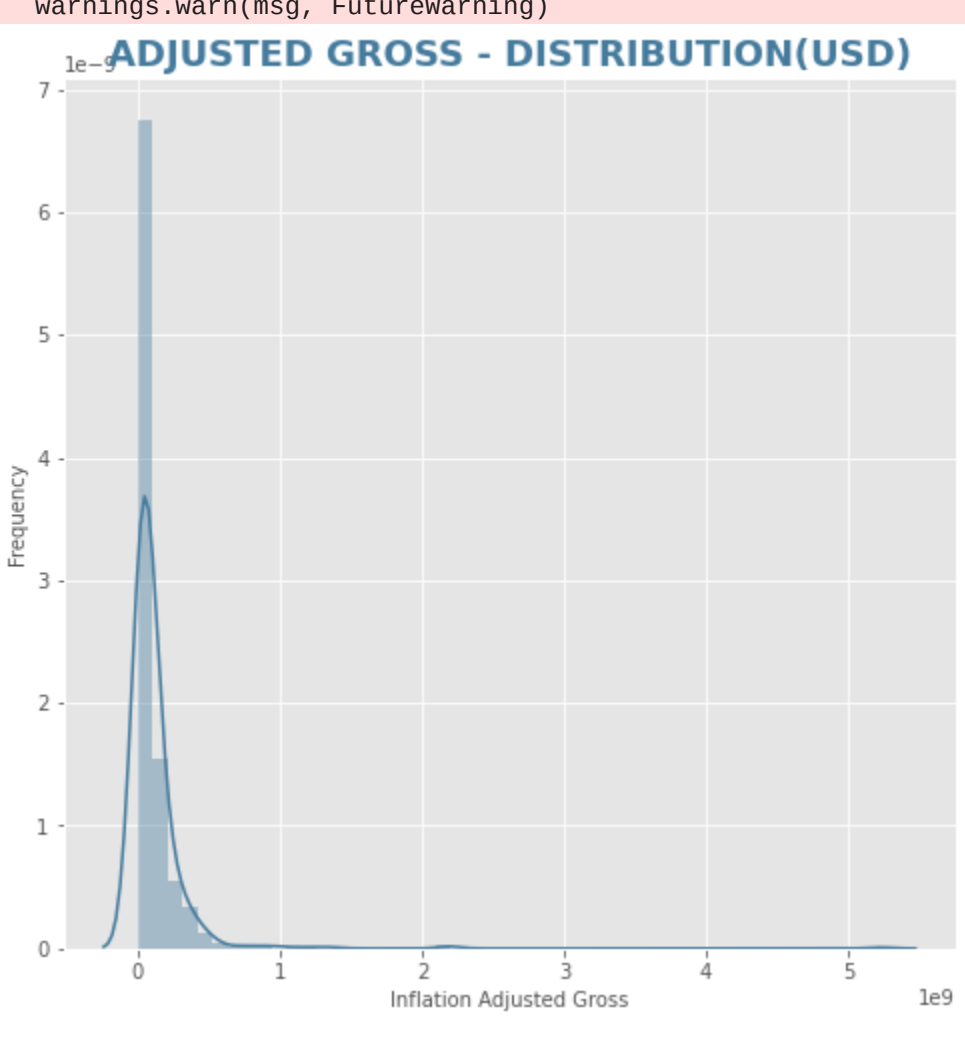
In [21]:
```python
plt.figure(figsize=(8,8))
sns.distplot(gross['total_gross'],color='#f9c74f')
plt.title('TOTAL GROSS - DISTRIBUTION(USD)',fontsize=18,color='#f9c74f',fontweight='bold')
plt.xlabel('Total Gross',fontsize=10)
plt.ylabel('Frequency',fontsize=10)
plt.show()
```

```
C:\Users\SRIRAM\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your cod
e to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



In [23]:
```python
plt.figure(figsize=(8,8))
sns.distplot(gross['inflation_adjusted_gross'],color='#457b9d')
plt.title('ADJUSTED GROSS - DISTRIBUTION(USD)',fontsize=18,color='#457b9d',fontweight='bold')
plt.xlabel('Inflation Adjusted Gross',fontsize=10)
plt.ylabel('Frequency',fontsize=10)
plt.show()
```

```
C:\Users\SRIRAM\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your cod
e to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



In [24]:
```python
gross.groupby('movie_title')['inflation_adjusted_gross'].sum().sort_values(ascending=False)[:5]
```

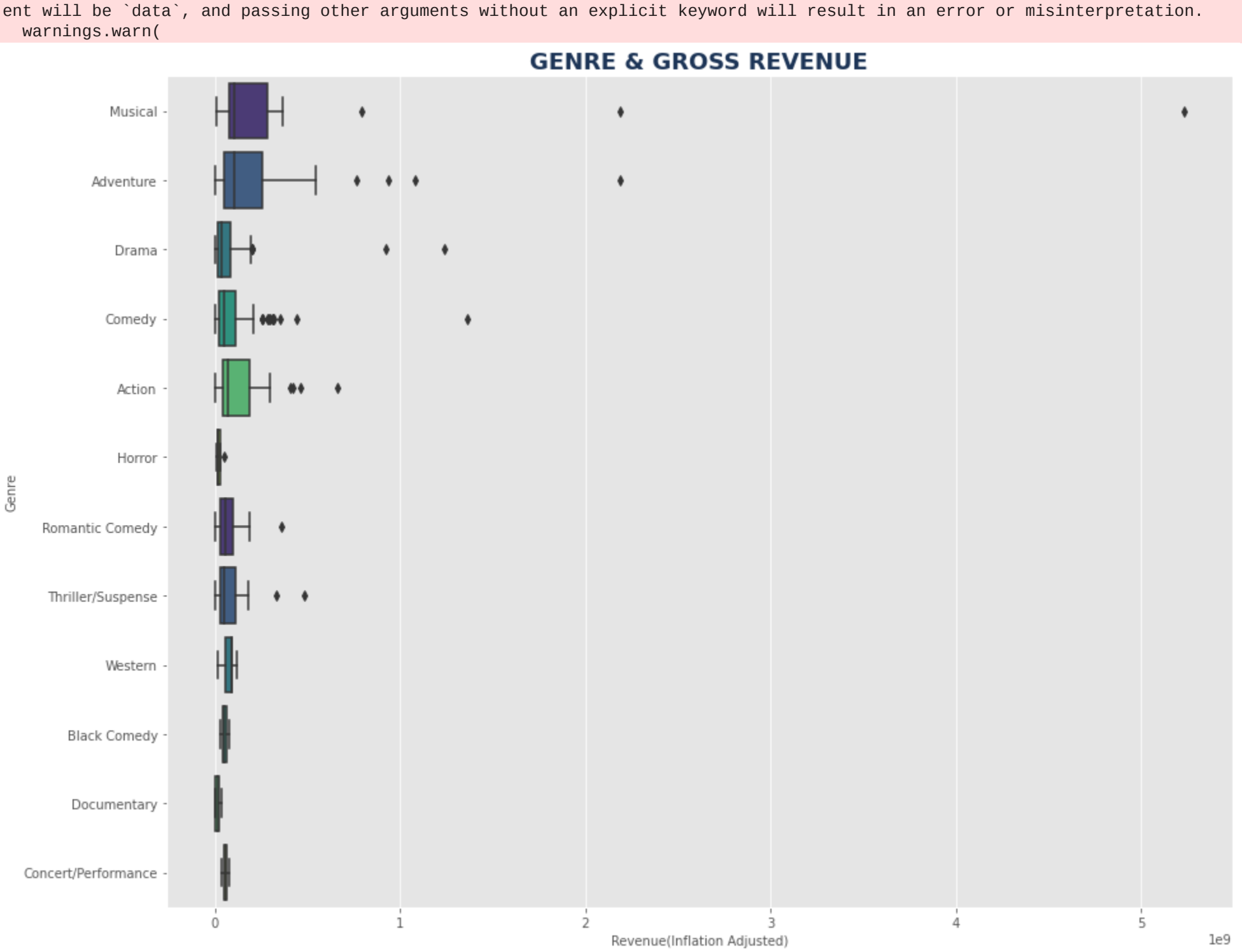
Out[24]:
```
movie_title
Snow White and the Seven Dwarfs    5228953251
Pinocchio                          2188229052
Fantasia                           2187090808
101 Dalmatians                     1621599883
The Jungle Book                    1242543790
Name: inflation_adjusted_gross, dtype: int64
```

Genre

Lets check the distribution of revenue by genre,

In [27]:
```python
plt.figure(figsize=(15,12))
sns.boxplot(gross['inflation_adjusted_gross'],gross['genre'],palette=sns.color_palette('viridis'))
plt.title('GENRE & GROSS REVENUE',fontsize=18,fontweight='bold',color='#1d3557')
plt.ylabel('Genre',fontsize=10)
plt.xlabel('Revenue(Inflation Adjusted)',fontsize=10)
plt.show()
```

```
C:\Users\SRIRAM\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variables as keyword args: x, y. From version 0.12, the only valid positional argum
ent will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

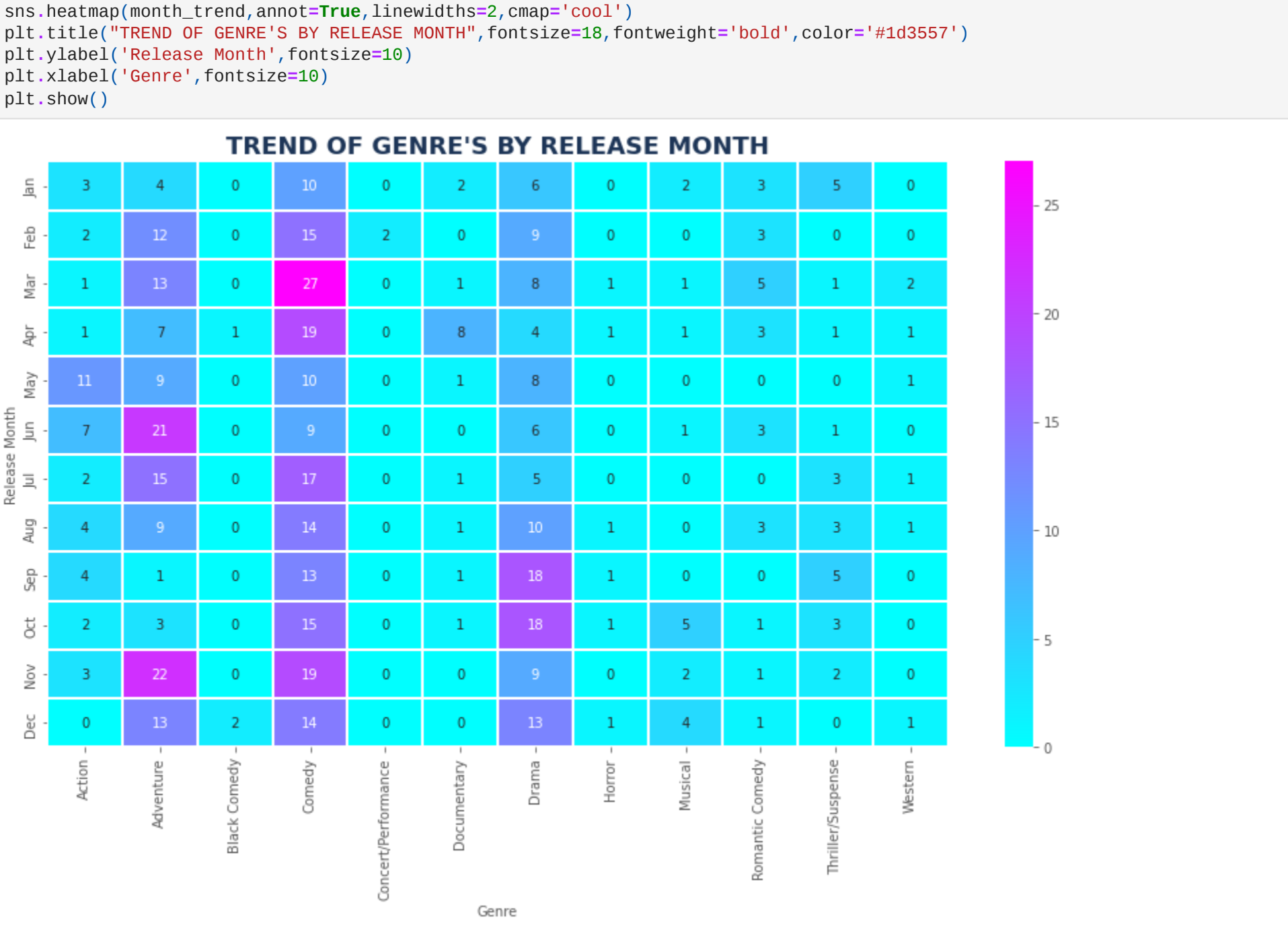


In [ ]:
```
A movie could have be categorized into multiple genres.But in this dataset, we have one genre tagged to every movie.
The below interpretation might change, if the data changes.

* Disney has released more movies in Comedy genre.Yet the median revenue from the genre is less and has few outliers.
* There are extreme outliers for musical category though they are less compared to other genres.
* The median revenue for Musical and Adventure genre are closer. Similarly for Drama,Comedy and Action which are heavily
dominated by outliers.
```

In [28]:
```python
month_order=['Jan','Feb','Mar','Apr','May','Jun','Jul','Aug','Sep','Oct','Nov','Dec']
month_trend=gross.groupby('genre')['release_month'].value_counts().unstack().fillna(0)[month_order].T
```

In [29]:
```python
plt.figure(figsize=(15,8))
sns.heatmap(month_trend,annot=True,linewidths=2,cmap='cool')
plt.title("TREND OF GENRE'S BY RELEASE MONTH",fontsize=18,fontweight='bold',color='#1d3557')
plt.ylabel('Release Month',fontsize=10)
plt.xlabel('Genre',fontsize=10)
plt.show()
```



1. The genre effect

Now that we have dummy variables, we can build a linear regression model to predict the adjusted gross using these dummy variables. From the regression model, we can check the effect of each genre by looking at its coefficient given in units of box office gross dollars. We will focus on the impact of action and adventure genres here. (Note that the intercept and the first coefficient values represent the effect of action and adventure genres respectively). We expect that movies like the Lion King or Star Wars would perform better for box office.

In [30]:
```python
from sklearn.linear_model import LinearRegression
regr = LinearRegression()
X = genre_dummies
y = gross['inflation_adjusted_gross']
regr.fit(X, y)
action =  regr.intercept_
adventure = regr.coef_[[0]][0]
print((action, adventure))
```

```
(102921757.36842175, 87475654.70909745)
```

In [ ]:
```
7. Conclusion
The confidence intervals from the bootstrap method for the intercept and coefficient do not contain the value zero,
as we have already seen that lower and upper bounds of both confidence intervals are positive.
These tell us that it is likely that the adjusted gross is significantly correlated with the action and adventure genres.
From the results of the bootstrap analysis and the trend plot we have done earlier, we could say that Disney movies with plots
that fit into the action and adventure genre, according to our data, tend to do better in terms of adjusted gross than other
genres.
```