Project: IMBD movies data analysis

Table of Contents

Introduction

In this project I will analyze the IMDB movies dataset, The dataset contains 21 columns each column has data that belong to movies

The questions I would like to answer:

1. The relation between popularty and revenue?
2. The relation between release year and vote?
3. Number of movies for each genre?

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

Data Wrangling

General Properties

In [2]:
```
df = pd.read_csv("tmdb-movies.csv")
```

reading csv file.

In [3]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10866 entries, 0 to 10865
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   id                    10866 non-null  int64
 1   imdb_id               10856 non-null  object
 2   popularity            10866 non-null  float64
 3   budget                10866 non-null  int64
 4   revenue               10866 non-null  int64
 5   original_title        10866 non-null  object
 6   cast                  10790 non-null  object
 7   homepage              2936 non-null   object
 8   director              10822 non-null  object
 9   tagline               8042 non-null   object
 10  keywords              9373 non-null   object
 11  overview              10862 non-null  object
 12  runtime               10866 non-null  int64
 13  genres                10843 non-null  object
 14  production_companies  9836 non-null   object
 15  release_date          10866 non-null  object
 16  vote_count            10866 non-null  int64
 17  vote_average          10866 non-null  float64
 18  release_year          10866 non-null  int64
 19  budget_adj            10866 non-null  float64
 20  revenue_adj           10866 non-null  float64
dtypes: float64(4), int64(6), object(11)
memory usage: 1.7+ MB
```

In [4]:
```
df.shape
```

Out[4]: (10866, 21)

In [5]:
```
df.describe()
```

Out[5]:

	id	popularity	budget	revenue	runtime	vote_count	vote_average	release_year	budget_a
count	10866.000000	10866.000000	1.086600e+04	1.086600e+04	10866.000000	10866.000000	10866.000000	10866.000000	1.086600e+
mean	66064.177434	0.646441	1.462570e+07	3.982332e+07	102.070863	217.389748	5.974922	2001.322658	1.755104e+
std	92130.136561	1.000185	3.091321e+07	1.170035e+08	31.381405	575.619058	0.935142	12.812941	3.430616e+
min	5.000000	0.000065	0.000000e+00	0.000000e+00	0.000000	10.000000	1.500000	1960.000000	0.000000e+
25%	10596.250000	0.207583	0.000000e+00	0.000000e+00	90.000000	17.000000	5.400000	1995.000000	0.000000e+
50%	20669.000000	0.383856	0.000000e+00	0.000000e+00	99.000000	38.000000	6.000000	2006.000000	0.000000e+
75%	75610.000000	0.713817	1.500000e+07	2.400000e+07	111.000000	145.750000	6.600000	2011.000000	2.085325e+
max	417859.000000	32.985763	4.250000e+08	2.781506e+09	900.000000	9767.000000	9.200000	2015.000000	4.250000e+

In [6]:
```
df.head()
```

Out[6]:

	id	imdb_id	popularity	budget	revenue	original_title	cast	homepage	c
0	135397	tt0369610	32.985763	150000000	1513528810	Jurassic World	Chris Pratt|Bryce Dallas Howard|Irrfan Khan|Vi...	http://www.jurassicworld.com/	Tre
1	76341	tt1392190	28.419936	150000000	378436354	Mad Max: Fury Road	Tom Hardy|Charlize Theron|Hugh Keays-Byrne|Nic...	http://www.madmaxmovie.com/	
2	262500	tt2908446	13.112507	110000000	295238201	Insurgent	Shailene Woodley|Theo James|Kate Winslet|Ansel...	http://www.thedivergentseries.movie/#insurgent	Sch
3	140607	tt2488496	11.173104	200000000	2068178225	Star Wars: The Force Awakens	Harrison Ford|Mark Hamill|Carrie Fisher|Adam D...	http://www.starwars.com/films/star-wars-episod...	
4	168259	tt2820852	9.335014	190000000	1506249360	Furious 7	Vin Diesel|Paul Walker|Jason Statham|Michelle ...	http://www.furious7.com/	

5 rows × 21 columns

check the columns and data type of each column and mising value for each row.

Data Cleaning: remove unnessery columns and fix the nan value

In [7]:
```
unnecessary_col =np.array(["homepage",
                   "tagline",
                   "overview",
                   "cast",
                   "release_date",
                   "production_companies",
                   "imdb_id",
                  "vote_count",
                  "runtime",
                  "original_title",
                  "director",
                  "budget_adj",
                  "revenue_adj"])
df.drop(unnecessary_col,axis=1,inplace=True)
```

I will not use these columns to answer my questions.

In [8]:
```
df.duplicated().sum()
```

Out[8]: 1

There one duplicate row in this dataset.

In [9]:
```
df.drop_duplicates(inplace=True)
```

Remove the duplicate rows.

In [10]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 10865 entries, 0 to 10865
Data columns (total 8 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   id            10865 non-null  int64
 1   popularity    10865 non-null  float64
 2   budget        10865 non-null  int64
 3   revenue       10865 non-null  int64
 4   keywords      9372 non-null   object
 5   genres        10842 non-null  object
 6   vote_average  10865 non-null  float64
 7   release_year  10865 non-null  int64
dtypes: float64(2), int64(4), object(2)
memory usage: 763.9+ KB
```

In [11]:
```
genres_null = df[df["genres"].isnull() == True]
genres_null.head()
```

Out[11]:

	id	popularity	budget	revenue	keywords	genres	vote_average	release_year
424	363869	0.244648	0	0	NaN	NaN	6.1	2015
620	361043	0.129696	0	0	NaN	NaN	5.0	2015
997	287663	0.330431	0	0	NaN	NaN	6.8	2014
1712	21634	0.302095	0	0	NaN	NaN	7.4	2009
1897	40534	0.020701	0	0	duringcreditsstinger	NaN	7.0	2009

I see the value 0 in both the Budget and Revenue columns, I will make query to find this mistake then I will fix it.

In [12]:
```
df.drop(genres_null.index,axis=0,inplace=True)
```

Remove rows with massing value in **genres** column.

In [13]:
```
budget_revenue_zero = df.query("budget == 0 and revenue == 0")
budget_revenue_zero.shape
```

Out[13]: (4679, 8)

In [14]:
```
df.drop(budget_revenue_zero.index,axis=0,inplace=True)
```

Remove rows if **budget** and **revenue** value are 0.

In [15]:
```
df.drop("keywords",axis=1,inplace=True)
```

Remove **keywords** column because they are a lot of missing value.

Unfortunately, I had a question about this column.

In [16]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 6163 entries, 0 to 10865
Data columns (total 7 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   id            6163 non-null   int64
 1   popularity    6163 non-null   float64
 2   budget        6163 non-null   int64
 3   revenue       6163 non-null   int64
 4   genres        6163 non-null   object
 5   vote_average  6163 non-null   float64
 6   release_year  6163 non-null   int64
dtypes: float64(2), int64(4), object(1)
memory usage: 385.2+ KB
```

Now the dataset are claen.

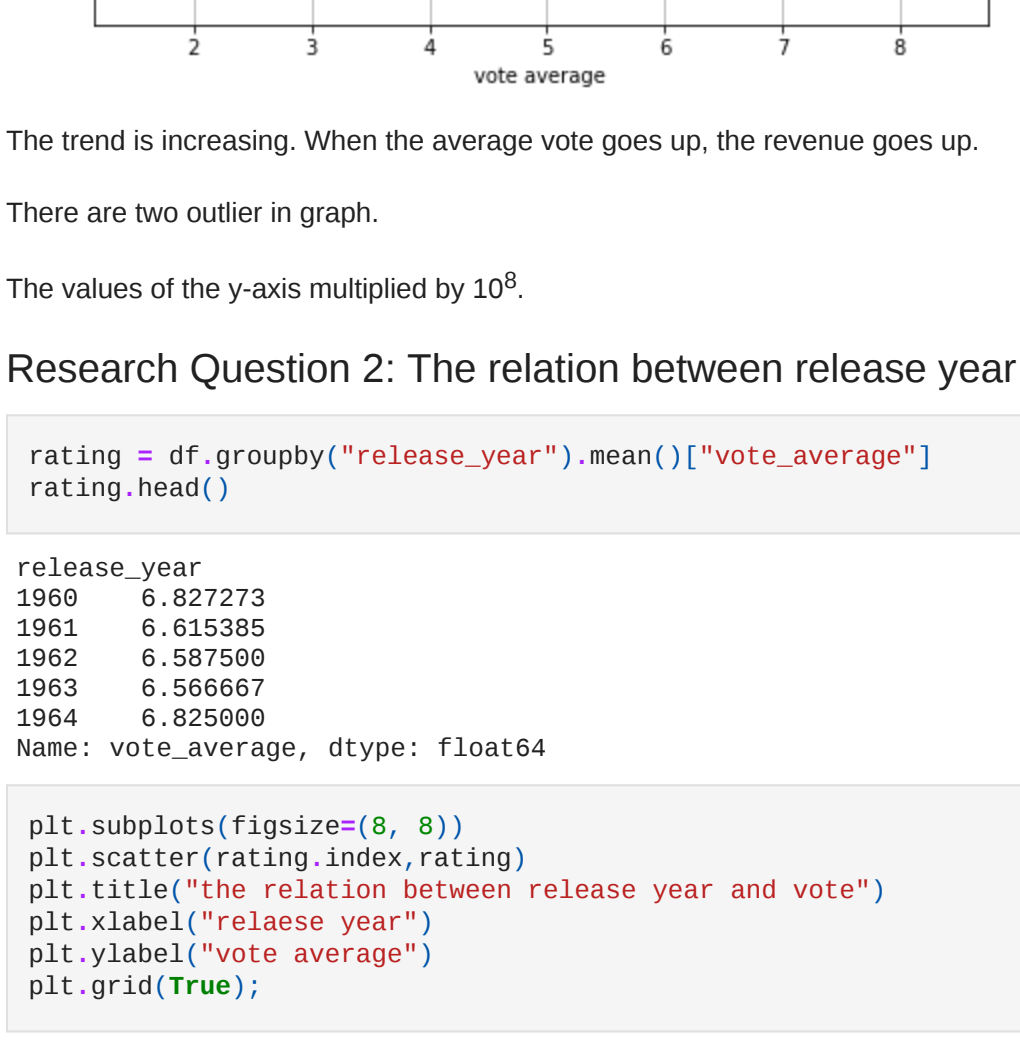
Exploratory Data Analysis

Research Question 1: The relation between voting average and revenue?

In [17]:
```
revenue = df.groupby("vote_average").mean()["revenue"]
revenue
```

Out[17]:
```
vote_average
1.5    0.000000e+00
2.1    9.109322e+06
2.2    7.370600e+04
2.4    0.000000e+00
2.6    0.000000e+00
           ...
8.0    1.637860e+08
8.1    2.364798e+08
8.2    1.399309e+07
8.3    2.450664e+08
8.4    1.110680e+07
Name: revenue, Length: 62, dtype: float64
```

In [18]:
```
plt.subplots(figsize=(8, 8))
plt.scatter(x=revenue.index,y=revenue)
plt.title("the relation between popularty and revenue")
plt.xlabel("vote average")
plt.ylabel("revenue")
plt.grid(True);
```



The trend is increasing. When the average vote goes up, the revenue goes up.

There are two outlier in graph.

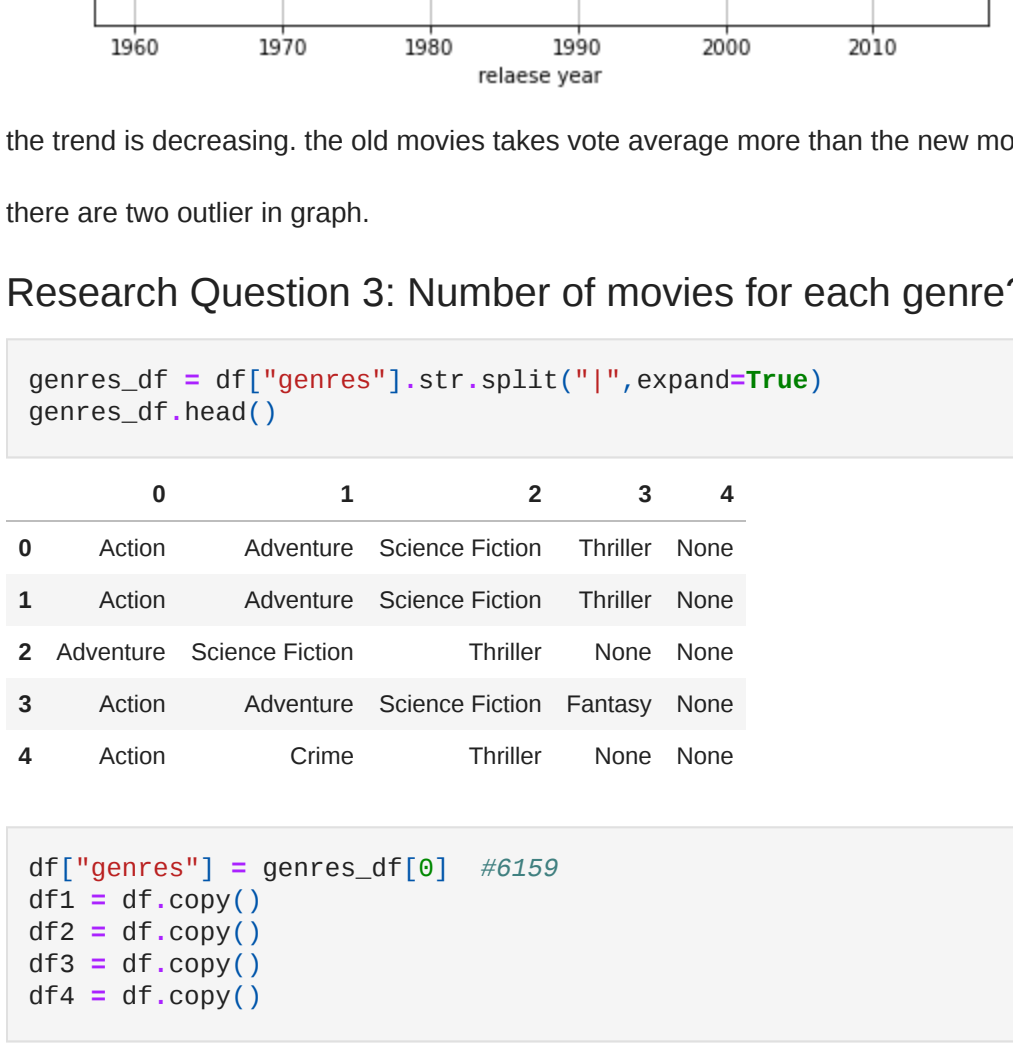
The values of the y-axis multiplied by 10^8.

Research Question 2: The relation between release year and vote average?

In [19]:
```
rating = df.groupby("release_year").mean()["vote_average"]
rating.head()
```

Out[19]:
```
release_year
1960    6.827273
1961    6.615385
1962    6.587500
1963    6.566667
1964    6.825000
Name: vote_average, dtype: float64
```

In [20]:
```
plt.subplots(figsize=(8, 8))
plt.scatter(rating.index,rating)
plt.title("the relation between release year and vote")
plt.xlabel("relaese year")
plt.ylabel("vote average")
plt.grid(True);
```



the trend is decreasing. the old movies takes vote average more than the new movies.

there are two outlier in graph.

Research Question 3: Number of movies for each genre?

In [21]:
```
genres_df = df["genres"].str.split("|",expand=True)
genres_df.head()
```

Out[21]:

	0	1	2	3	4
0	Action	Adventure	Science Fiction	Thriller	None
1	Action	Adventure	Science Fiction	Thriller	None
2	Adventure	Science Fiction	Thriller	None	None
3	Action	Adventure	Science Fiction	Fantasy	None
4	Action	Crime	Thriller	None	None

In [22]:
```
df["genres"] = genres_df[0]  #6159
df1 = df.copy()
df2 = df.copy()
df3 = df.copy()
df4 = df.copy()
```

In [23]:
```
df1["genres"] = genres_df[1]
df2["genres"] = genres_df[2]
df3["genres"] = genres_df[3]
df4["genres"] = genres_df[4]
```

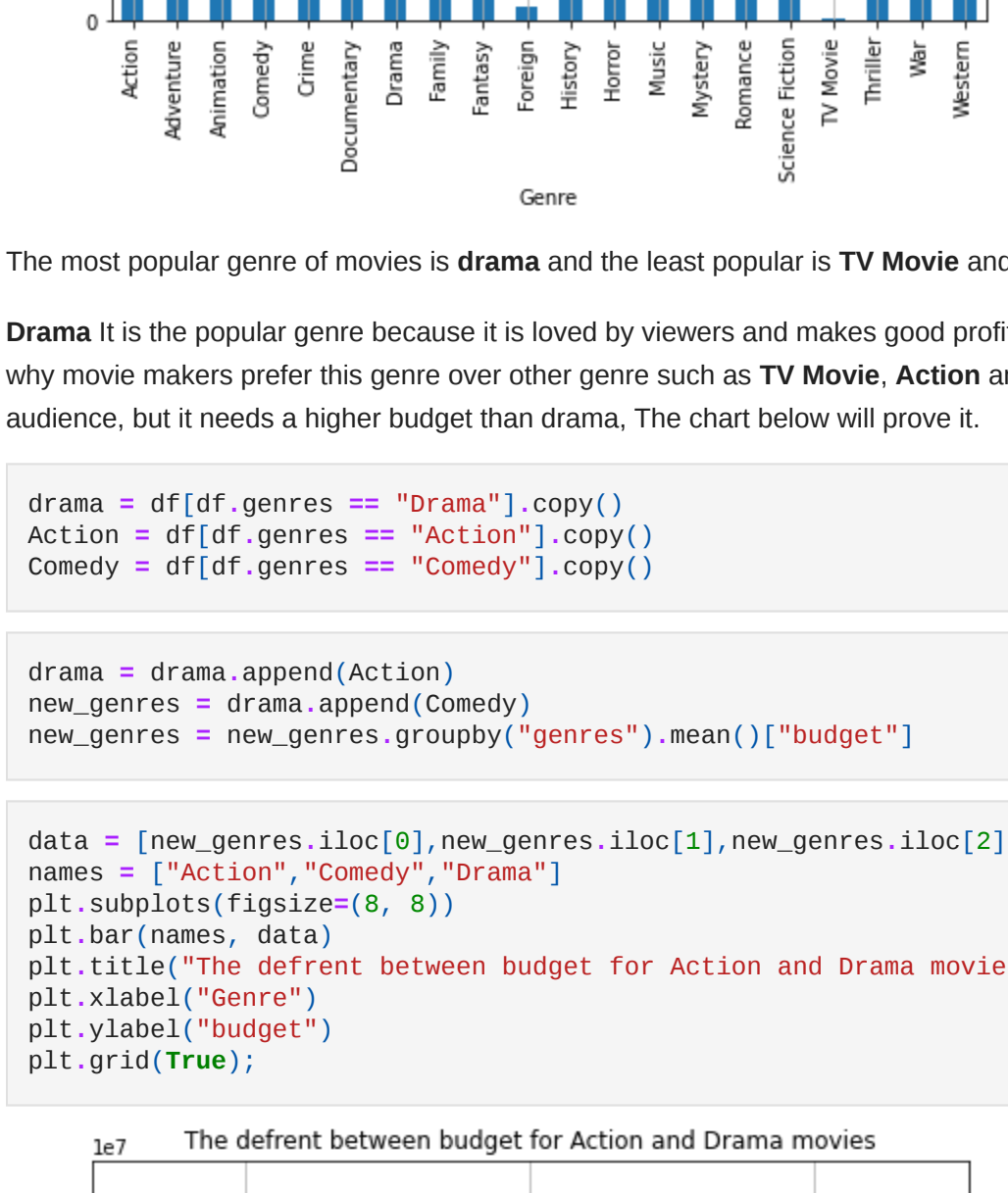
In [24]:
```
df = df.append(df1)
df = df.append(df2)
df = df.append(df3)
df = df.append(df4)
df.shape
```

Out[24]: (30815, 7)

In [25]:
```
df.dropna(axis=0,inplace=True)
df.shape
```

Out[25]: (15962, 7)

In [26]:
```
data = df.groupby("genres").count()["id"]
plt.subplots(figsize=(8, 8))
data.plot(kind="bar")
plt.title("number of movies for each genre")
plt.xlabel("Genre")
plt.ylabel("Number of movies")
plt.grid(True);
```



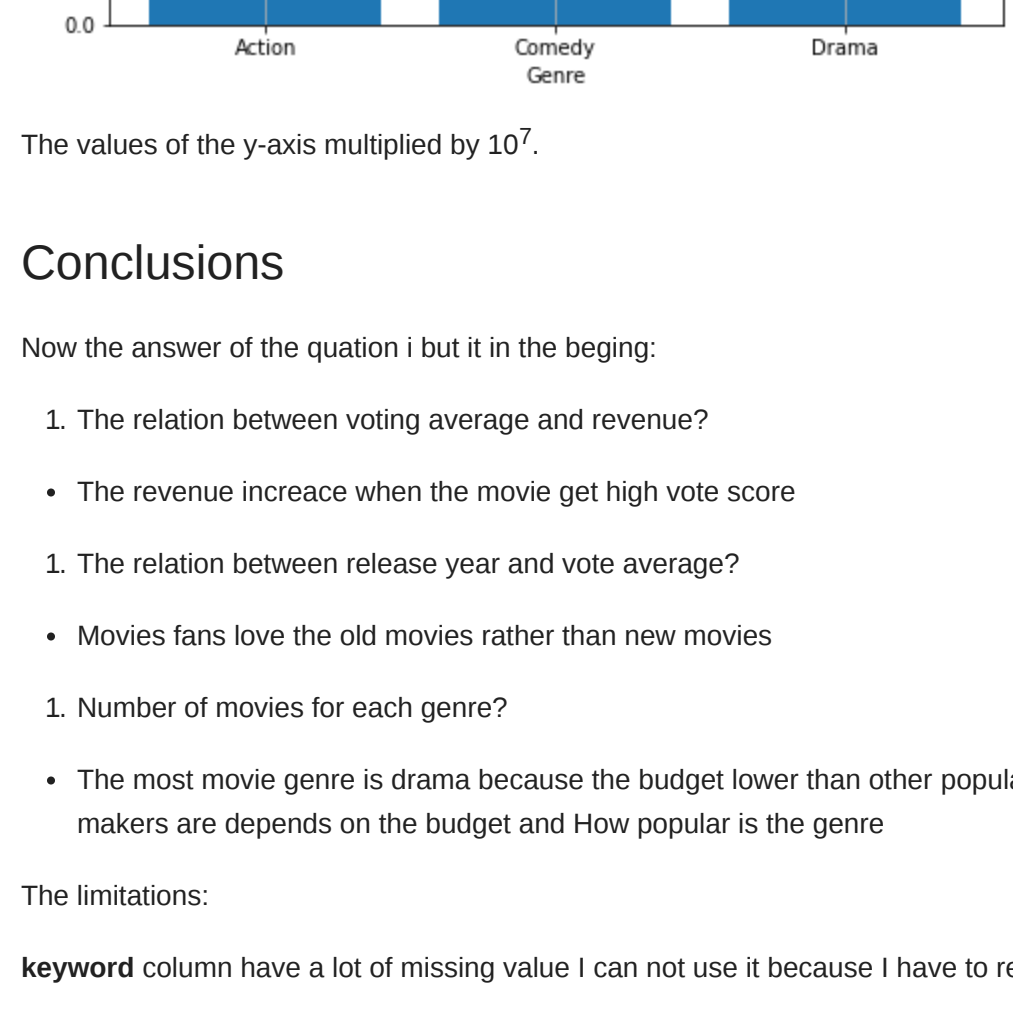
The most popular genre of movies is **drama** and the least popular is **TV Movie** and in this dataset there are 20 different genres.

Drama It is the popular genre because it is loved by viewers and makes good profits and needs a relatively small budget, and this is why movie makers prefer this genre over other genre such as **TV Movie**, **Action** and **Comedy** movies, **Action and Comedy** has an audience, but it needs a higher budget than drama, The chart below will prove it.

In [27]:
```
drama = df[df.genres == "Drama"].copy()
Action = df[df.genres == "Action"].copy()
Comedy = df[df.genres == "Comedy"].copy()
```

In [28]:
```
drama = drama.append(Action)
new_genres = drama.append(Comedy)
new_genres = new_genres.groupby("genres").mean()["budget"]
```

In [29]:
```
data = [new_genres.iloc[0],new_genres.iloc[1],new_genres.iloc[2]]
names = ["Action","Comedy","Drama"]
plt.subplots(figsize=(8, 8))
plt.bar(names, data)
plt.title("The defrent between budget for Action and Drama movies")
plt.xlabel("Genre")
plt.ylabel("budget")
plt.grid(True);
```



The values of the y-axis multiplied by 10^7.

Conclusions

Now the answer of the quation i but it in the beging:

1. The relation between voting average and revenue?

- The revenue increace when the movie get high vote score

1. The relation between release year and vote average?

- Movies fans love the old movies rather than new movies

1. Number of movies for each genre?

- The most movie genre is drama because the budget lower than other popular genre like **Comedy and action**, The movie makers are depends on the budget and How popular is the genre

The limitations:

keyword column have a lot of missing value I can not use it because I have to remove a lot of rows