In [82]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

import warnings
warnings.filterwarnings("ignore")
```

In [43]:

```
movies = pd.read_csv("C:\\Users\\jaydeepb751\\Documents\\PwC\\CERTIFICATION & TRAINING\\Training 23-24\\EDA-Python\\
```

In [106]:

```
movies = pd.read_csv('Movie-Ratings.csv')
```

In [47]:

```
movies.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 559 entries, 0 to 558
Data columns (total 6 columns):
 #   Column                     Non-Null Count  Dtype 
---  ------                     --------------  ----- 
 0   Film                       559 non-null    object
 1   Genre                      559 non-null    object
 2   Rotten Tomatoes Ratings %  559 non-null    int64 
 3   Audience Ratings %         559 non-null    int64 
 4   Budget (million $)         559 non-null    int64 
 5   Year of release            559 non-null    int64 
dtypes: int64(4), object(2)
memory usage: 26.3+ KB
```

In [48]:

```
movies.head()
```

Out[48]:

	Film	Genre	Rotten Tomatoes Ratings %	Audience Ratings %	Budget (million $)	Year of release
0	(500) Days of Summer	Comedy	87	81	8	2009
1	10,000 B.C.	Adventure	9	44	105	2008
2	12 Rounds	Action	30	52	20	2009
3	127 Hours	Adventure	93	84	18	2010
4	17 Again	Comedy	55	70	20	2009

In [107]:

```
movies.columns = ['Film','Genre','CR','AR','Budget($M)','Year']
```

In [50]:

```
movies.head()
```

Out[50]:

	Film	Genre	CR	AR	Budget($M)	Year
0	(500) Days of Summer	Comedy	87	81	8	2009
1	10,000 B.C.	Adventure	9	44	105	2008
2	12 Rounds	Action	30	52	20	2009
3	127 Hours	Adventure	93	84	18	2010
4	17 Again	Comedy	55	70	20	2009

In [51]:

```
movies.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 559 entries, 0 to 558
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   Film        559 non-null    object
 1   Genre       559 non-null    object
 2   CR          559 non-null    int64 
 3   AR          559 non-null    int64 
 4   Budget($M)  559 non-null    int64 
 5   Year        559 non-null    int64 
dtypes: int64(4), object(2)
memory usage: 26.3+ KB
```

In [52]:

```
movies.describe()
```

Out[52]:

	CR	AR	Budget($M)	Year
count	559.000000	559.000000	559.000000	559.000000
mean	47.309481	58.744186	50.236136	2009.152057
std	26.413091	16.826887	48.731817	1.362632
min	0.000000	0.000000	0.000000	2007.000000
25%	25.000000	47.000000	20.000000	2008.000000
50%	46.000000	58.000000	35.000000	2009.000000
75%	70.000000	72.000000	65.000000	2010.000000
max	97.000000	96.000000	300.000000	2011.000000

In [54]:

```
movies['Year'] = movies['Year'].astype(str)
```

In [55]:

```
movies.describe()
```

Out[55]:

	CR	AR	Budget($M)
count	559.000000	559.000000	559.000000
mean	47.309481	58.744186	50.236136
std	26.413091	16.826887	48.731817
min	0.000000	0.000000	0.000000
25%	25.000000	47.000000	20.000000
50%	46.000000	58.000000	35.000000
75%	70.000000	72.000000	65.000000
max	97.000000	96.000000	300.000000

In [56]:

```
movies['Genre'].value_counts()
```

Out[56]:

```
Genre
Comedy       172
Action       154
Drama        101
Horror        49
Thriller      36
Adventure     29
Romance       18
Name: count, dtype: int64
```

In [68]:

```
movies.groupby(['Genre'])['AR'].median()
```

Out[68]:

```
Genre
Horror       48.0
Comedy       56.0
Action       57.5
Adventure    61.0
Romance      65.0
Drama        66.0
Thriller     68.5
Name: AR, dtype: float64
```

In [70]:

```
movies.sort_values(by=['Budget($M)','AR'],ascending=False)
```

Out[70]:

	Film	Genre	CR	AR	Budget($M)	Year
304	Pirates of the Caribbean: At World's End	Action	45	74	300	2007
360	Spider-Man 3	Action	61	54	258	2007
167	Harry Potter and the Half-Blood Prince	Adventure	83	75	250	2009
303	Pirates of the Caribbean: On Stranger Tides	Action	34	61	250	2011
33	Avatar	Action	83	92	237	2009
...	...	...	...	...	...	...
474	The Spy Next Door	Action	13	46	0	2010
539	When in Rome	Comedy	15	44	0	2010
356	Soul Men	Comedy	45	42	0	2008
154	Greenberg	Comedy	75	40	0	2010
185	I'm Still Here	Comedy	52	38	0	2010

559 rows × 6 columns

In [69]:

```
movies.groupby(['Genre'])['AR'].median().sort_values(ascending=False)
```

Out[69]:

```
Genre
Thriller     68.5
Drama        66.0
Romance      65.0
Adventure    61.0
Action       57.5
Comedy       56.0
Horror       48.0
Name: AR, dtype: float64
```

In [74]:

```
movies.groupby(['Genre']).aggregate({'CR':'mean','AR':'median','Budget($M)':'sum'})
```

Out[74]:

	CR	AR	Budget($M)
Genre			
Action	44.402597	57.5	13033
Adventure	53.103448	61.0	2363
Comedy	44.918605	56.0	6211
Drama	56.475248	66.0	2813
Horror	34.571429	48.0	1062
Romance	45.388889	65.0	632
Thriller	59.083333	68.5	1968

In [73]:

```
ıovies.groupby(['Genre']).aggregate({'CR':'mean','AR':'median','Budget($M)':'sum'}).sort_values(by=['Budget($M)'],asc
```

Out[73]:

	CR	AR	Budget($M)
Genre			
Action	44.402597	57.5	13033
Comedy	44.918605	56.0	6211
Drama	56.475248	66.0	2813
Adventure	53.103448	61.0	2363
Thriller	59.083333	68.5	1968
Horror	34.571429	48.0	1062
Romance	45.388889	65.0	632

In [59]:

```
movies.corr(method = 'pearson' , min_periods = 1, numeric_only = True)
```

Out[59]:

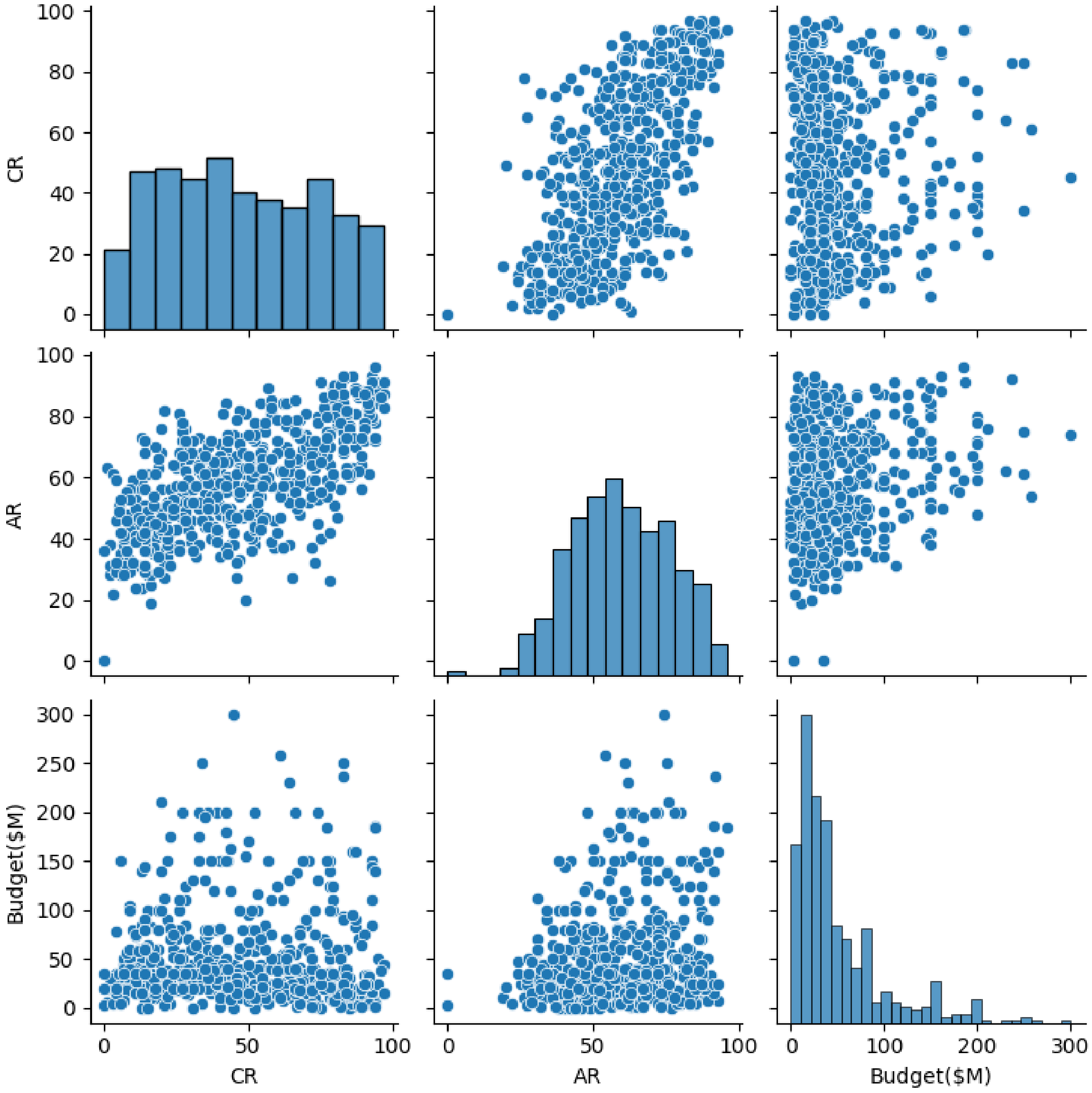
	CR	AR	Budget($M)
CR	1.000000	0.654803	0.014071
AR	0.654803	1.000000	0.191108
Budget($M)	0.014071	0.191108	1.000000

In [60]:

```
sns.pairplot(movies)
```

Out[60]:

```
<seaborn.axisgrid.PairGrid at 0x228bd519310>
```



In [80]:

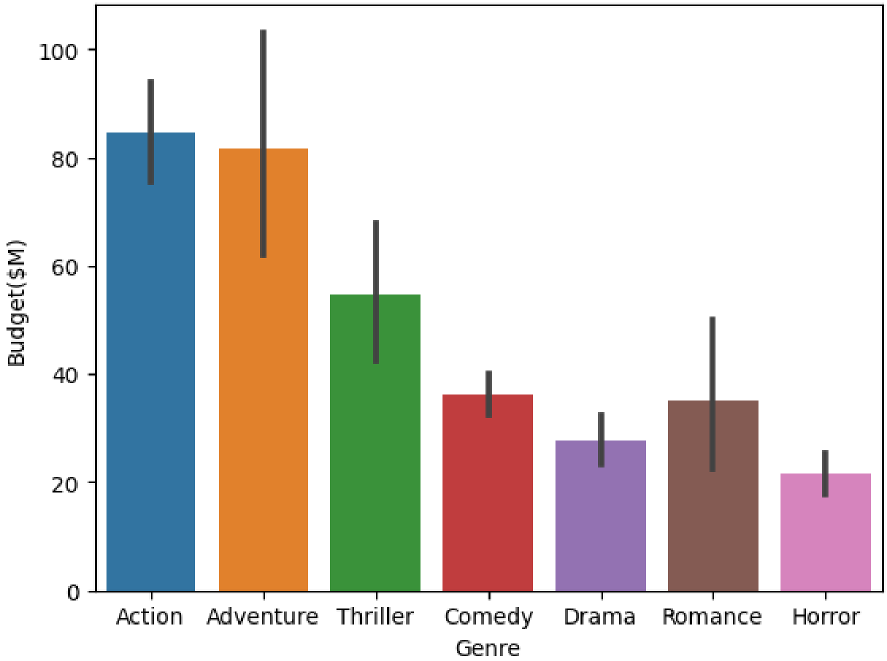
```
import matplotlib.pyplot as plt
```

In [108]:

```
movies=movies.sort_values(by=['Budget($M)'],ascending=False)
sns.barplot(x='Genre', y='Budget($M)', data=movies)
```

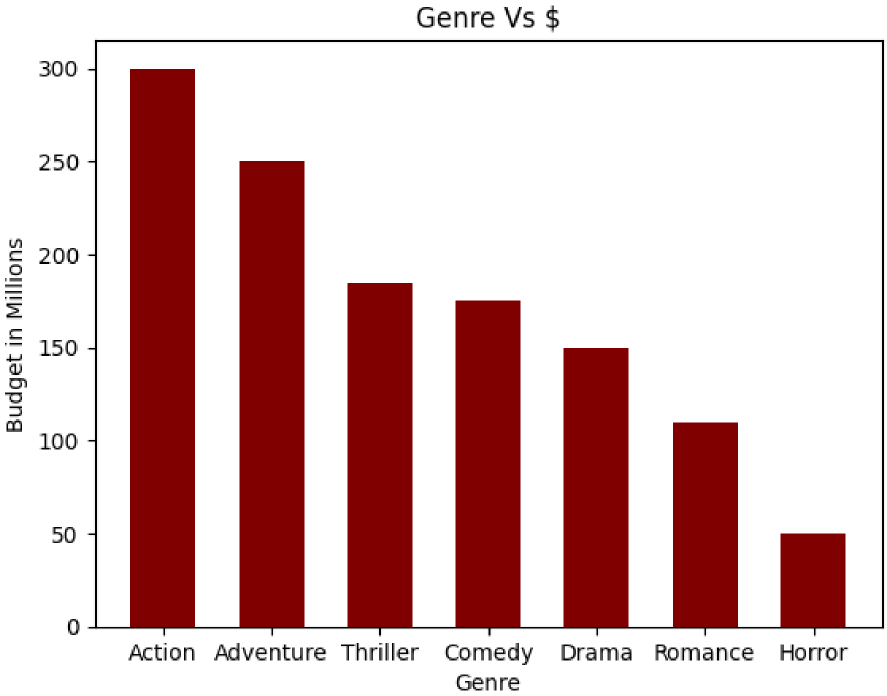
Out[108]:

```
<Axes: xlabel='Genre', ylabel='Budget($M)'>
```



In [109]:

```
    plt.bar(movies['Genre'], movies['Budget($M)'], color ='maroon', width = 0.6)
    plt.xlabel("Genre")
    plt.ylabel("Budget in Millions")
    plt.title("Genre Vs $")
    plt.show()
```

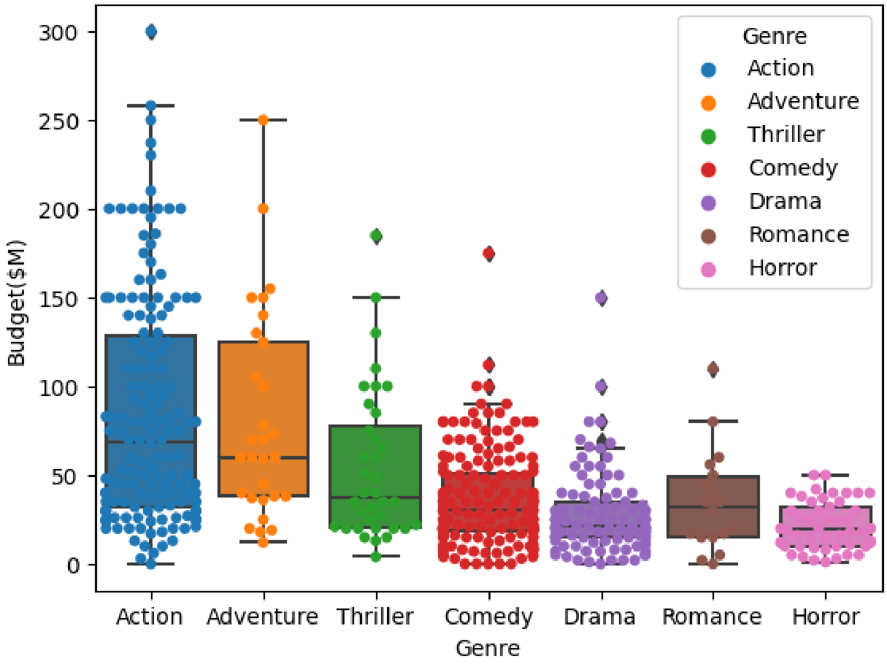


In [110]:

```
ax = sns.swarmplot(x='Genre',y='Budget($M)', data = movies, hue = 'Genre')
sns.boxplot(x='Genre',y='Budget($M)', data = movies, ax=ax)
```

Out[110]:

```
<Axes: xlabel='Genre', ylabel='Budget($M)'>
```



In [ ]: