[3]:
```
#Find the missing values :

for col in df.columns:
    pct_missing = np.mean(df[col].isnull())
    print('{} - {}% '.format(col,pct_missing))
```

```
name - 0.0%
rating - 0.010041731872717789%
genre - 0.0%
year - 0.0%
released - 0.00026082420448617 63%
score - 0.00039123630672926 45%
votes - 0.00039123630672926 45%
director - 0.0%
writer - 0.00039123630672926 45%
star - 0.00013041210224308815%
country - 0.00039123630672926 45%
budget - 0.2831246739697444%
gross - 0.02464788732394366%
company - 0.002217005738132499%
runtime - 0.0005216484089723526%
```

[4]:
```
df.isnull().sum()
```

[4]:
```
name          0
rating       77
genre         0
year          0
released      2
score         3
votes         3
```

[22]:
```
# Replace NaNs with 0
df['budget'] = df['budget'].fillna(0)
df['gross'] = df['gross'].fillna(0)

# Now convert to integers
df['budget'] = df['budget'].astype('int64')
df['gross'] = df['gross'].astype('int64')
```

[23]:
```
df
```

[23]:

| | name | rating | genre | year | released | score | votes | director | writer | star | country | budget | g |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | The Shining | R | Drama | 1980 | June 13, 1980 (United States) | 8.4 | 927000.0 | Stanley Kubrick | Stephen King | Jack Nicholson | United Kingdom | 19000000 | 4699 |
| **1** | The Blue Lagoon | R | Adventure | 1980 | July 2, 1980 (United States) | 5.8 | 65000.0 | Randal Kleiser | Henry De Vere Stacpoole | Brooke Shields | United States | 4500000 | 5885 |
| **2** | Star Wars: Episode V - The Empire Strikes Back | PG | Action | 1980 | June 20, 1980 (United States) | 8.7 | 1200000.0 | Irvin Kershner | Leigh Brackett | Mark Hamill | United States | 18000000 | 53837 |
| **3** | Airplane! | PG | Comedy | 1980 | July 2, 1980 (United States) | 7.7 | 221000.0 | Jim Abrahams | Jim Abrahams | Robert Hays | United States | 3500000 | 8345 |

[19]:
```
# we are going to create a new column called year matching with released to create correct year column
```

[19]:
```
# we are going to create a new column called year matching with released to create correct year column
df['yearcorrect']=df['released'].astype(str)
df
```

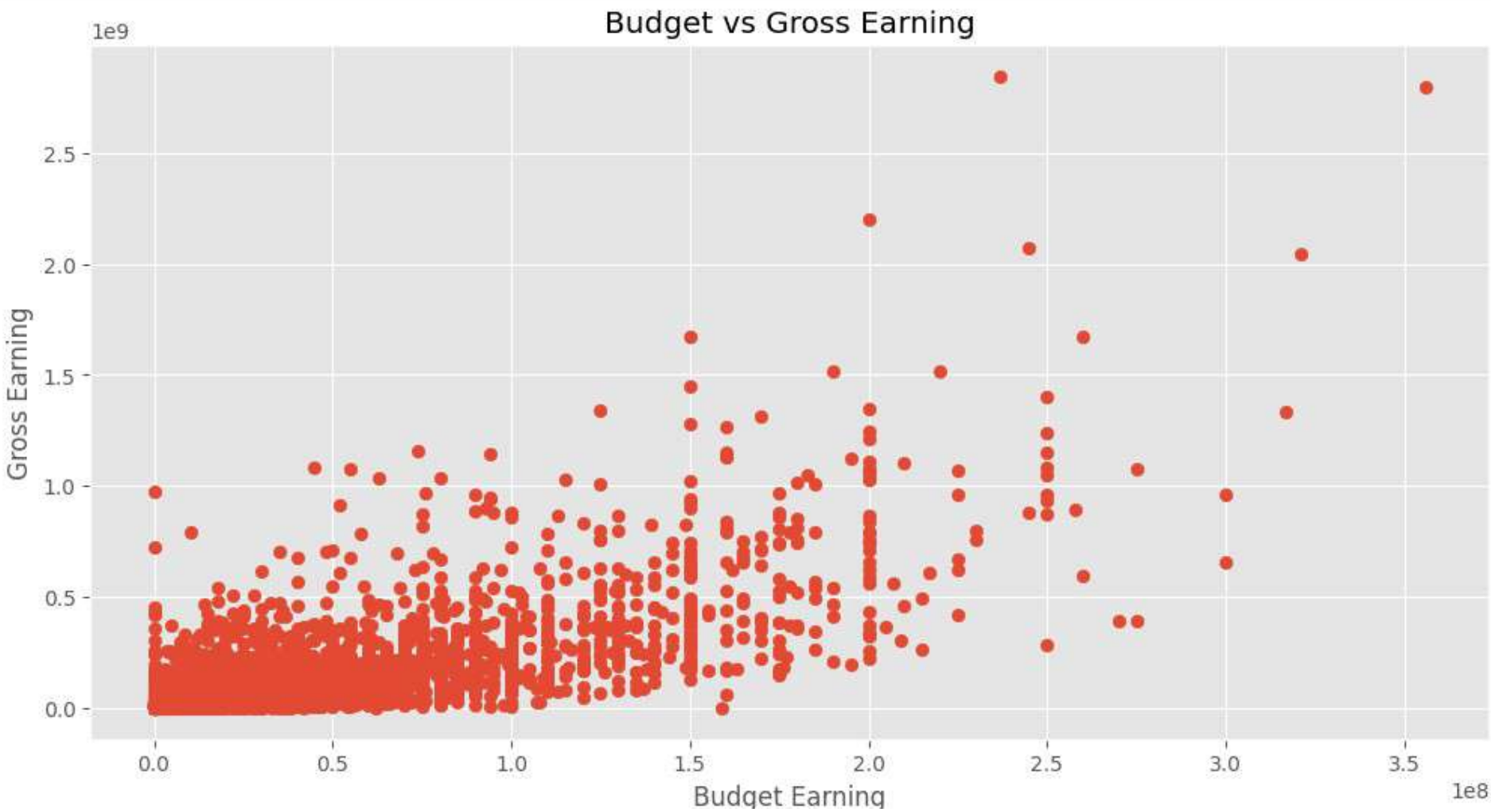
[19]:

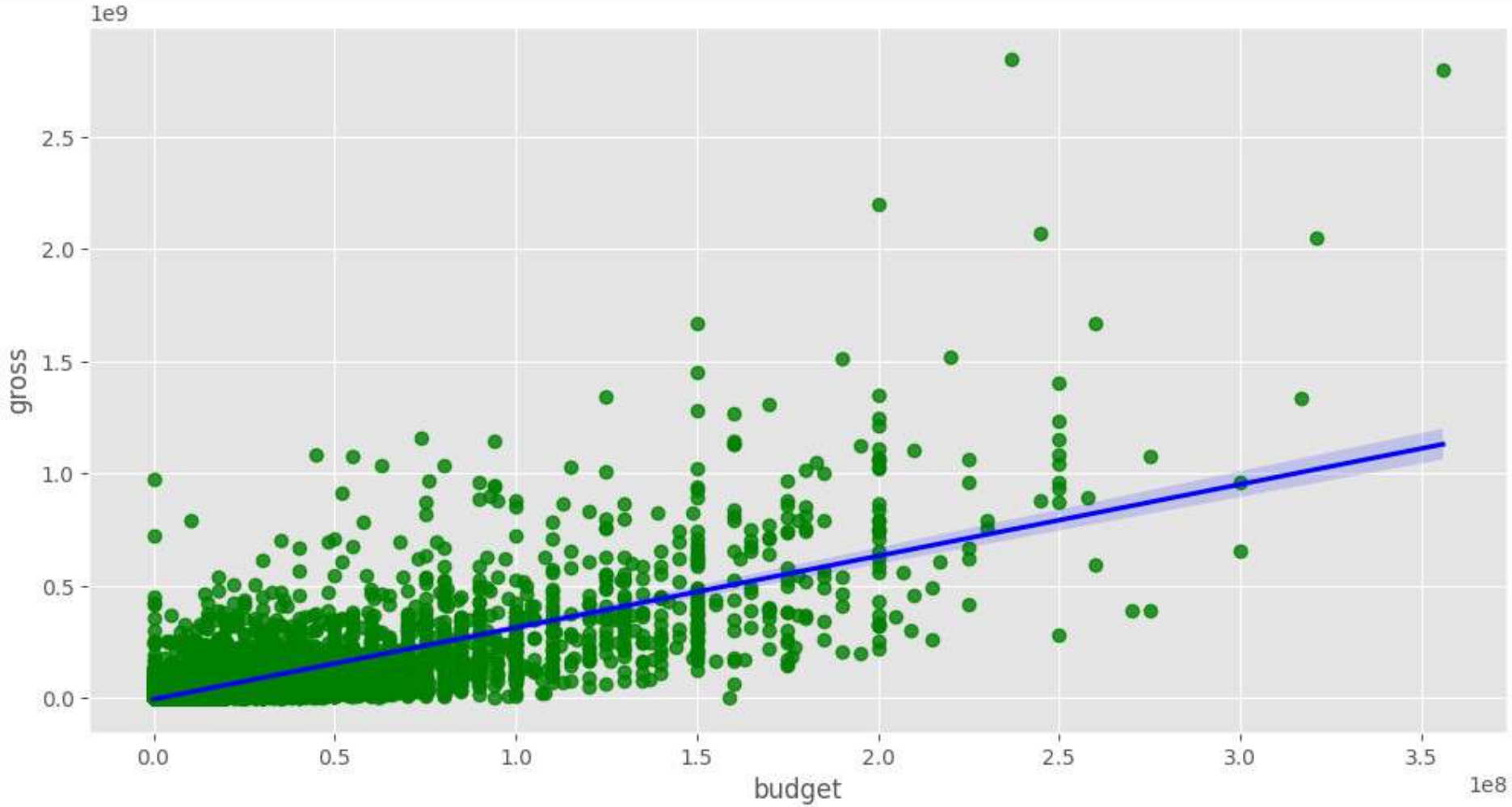
| | name | rating | genre | year | released | score | votes | director | writer | star | country | budget | gross | company | runtime | yearcorrect |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **5445** | 533 | 5 | 0 | 2009 | 696 | 7.8 | 1100000.0 | 1155 | 1778 | 2334 | 55 | 237000000 | 2847246203 | 2253 | 162.0 | 696 |
| **7445** | 535 | 5 | 0 | 2019 | 183 | 8.4 | 903000.0 | 162 | 743 | 2241 | 55 | 356000000 | 2797501328 | 1606 | 181.0 | 183 |
| **3045** | 6896 | 5 | 6 | 1997 | 704 | 7.8 | 1100000.0 | 1155 | 1778 | 1595 | 55 | 200000000 | 2201647264 | 2253 | 194.0 | 704 |
| **6663** | 5144 | 5 | 0 | 2015 | 698 | 7.8 | 876000.0 | 1125 | 2550 | 524 | 55 | 245000000 | 2069521700 | 1540 | 138.0 | 698 |
| **7244** | 536 | 5 | 0 | 2018 | 192 | 8.4 | 897000.0 | 162 | 743 | 2241 | 55 | 321000000 | 2048359754 | 1606 | 149.0 | 192 |
| **7480** | 6194 | 4 | 2 | 2019 | 1488 | 6.9 | 222000.0 | 1455 | 1919 | 676 | 55 | 260000000 | 1670727580 | 2316 | 118.0 | 1488 |
| **6653** | 2969 | 5 | 0 | 2015 | 1704 | 7.0 | 593000.0 | 517 | 3568 | 437 | 55 | 150000000 | 1670516444 | 2281 | 124.0 | 1704 |
| **6043** | 5502 | 5 | 0 | 2012 | 2472 | 8.0 | 1300000.0 | 1517 | 2314 | 2241 | 55 | 220000000 | 1518815515 | 1606 | 143.0 | 2472 |
| **6646** | 2145 | 5 | 0 | 2015 | 221 | 7.1 | 370000.0 | 1189 | 706 | 2721 | 55 | 190000000 | 1515341399 | 2281 | 137.0 | 221 |

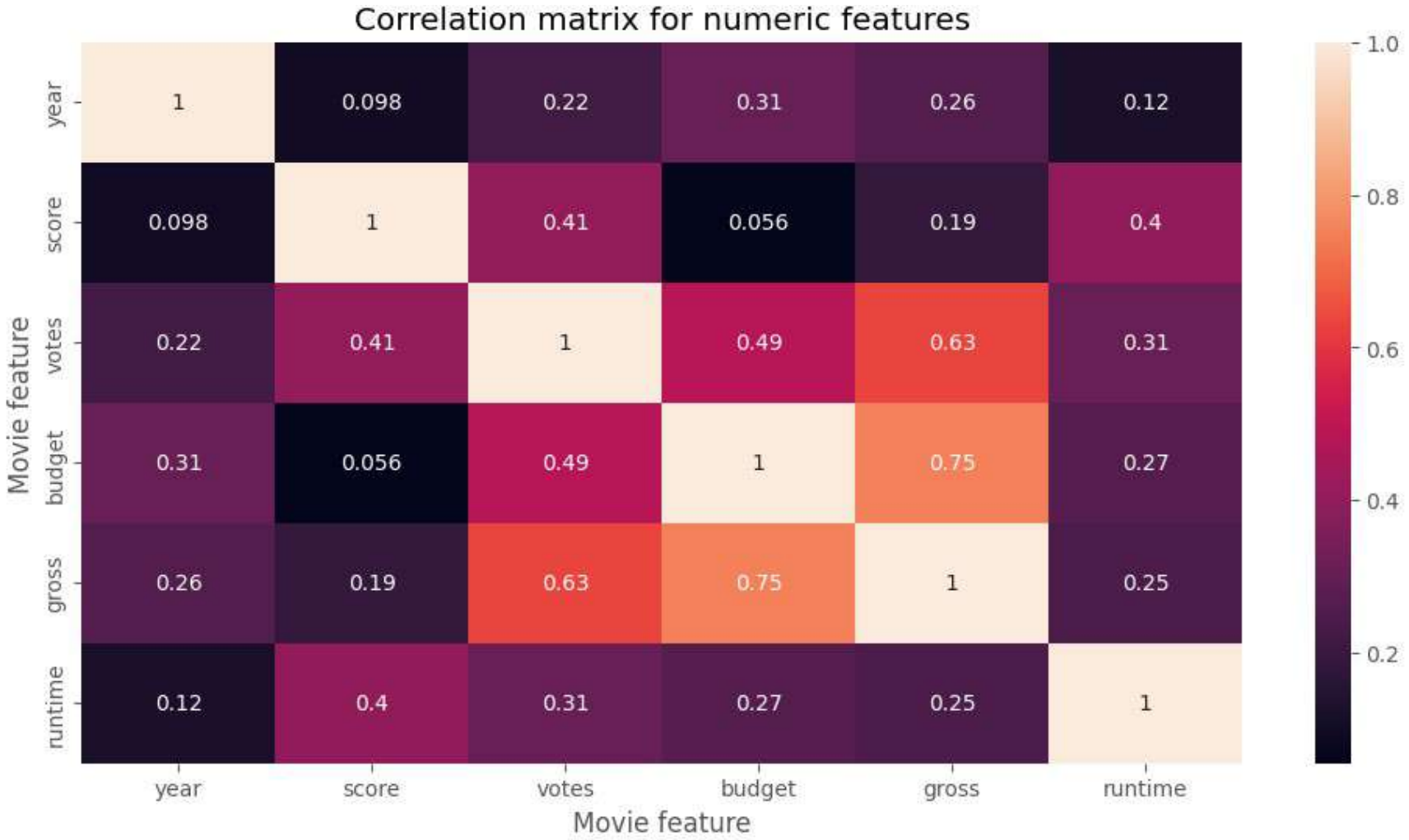
[18]:
```
#Now we are going to sort the values based on gross
df=df.sort_values(by=['gross'], inplace=False, ascending=False)
df
```

[18]:

| | name | rating | genre | year | released | score | votes | director | writer | star | country | budget | gross | company | runtime | yearcorrect |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **5445** | 533 | 5 | 0 | 2009 | 696 | 7.8 | 1100000.0 | 1155 | 1778 | 2334 | 55 | 237000000 | 2847246203 | 2253 | 162.0 | 696 |


Budget vs Gross Earning
1e9
2.5
2.0
1.5
1.0
0.5
0.0
Gross Earning
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
1e8
Budget Earning


1e9
2.5
2.0
1.5
1.0
0.5
0.0
gross
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
1e8
budget


Correlation matrix for numeric features
Movie feature
year
score
votes
budget
gross
runtime
1
0.098
0.22
0.31
0.26
0.12
0.098
1
0.41
0.056
0.19
0.4
0.22
0.41
1
0.49
0.63
0.31
0.31
0.056
0.49
1
0.75
0.27
0.26
0.19
0.63
0.75
1
0.25
0.12
0.4
0.31
0.27
0.25
1
Movie feature
1.0
0.8
0.6
0.4
0.2

[17]:
```
df_numerized=df
for col_name in df_numerized.columns:
    if(df_numerized[col_name].dtype == 'object'):
        df_numerized[col_name] = df_numerized[col_name].astype('category')
        df_numerized[col_name]=df_numerized[col_name].cat.codes

df_numerized
```

[17]:

| | name | rating | genre | year | released | score | votes | director | writer | star | country | budget | gross | company | runtime | yearcorrect |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **5445** | 533 | 5 | 0 | 2009 | 696 | 7.8 | 1100000.0 | 1155 | 1778 | 2334 | 55 | 237000000 | 2847246203 | 2253 | 162.0 | 696 |
| **7445** | 535 | 5 | 0 | 2019 | 183 | 8.4 | 903000.0 | 162 | 743 | 2241 | 55 | 356000000 | 2797501328 | 1606 | 181.0 | 183 |
| **3045** | 6896 | 5 | 6 | 1997 | 704 | 7.8 | 1100000.0 | 1155 | 1778 | 1595 | 55 | 200000000 | 2201647264 | 2253 | 194.0 | 704 |
| **6663** | 5144 | 5 | 0 | 2015 | 698 | 7.8 | 876000.0 | 1125 | 2550 | 524 | 55 | 245000000 | 2069521700 | 1540 | 138.0 | 698 |
| **7244** | 536 | 5 | 0 | 2018 | 192 | 8.4 | 897000.0 | 162 | 743 | 2241 | 55 | 321000000 | 2048359754 | 1606 | 149.0 | 192 |
| **7480** | 6194 | 4 | 2 | 2019 | 1488 | 6.9 | 222000.0 | 1455 | 1919 | 676 | 55 | 260000000 | 1670727580 | 2316 | 118.0 | 1488 |
| **6653** | 2969 | 5 | 0 | 2015 | 1704 | 7.0 | 593000.0 | 517 | 3568 | 437 | 55 | 150000000 | 1670516444 | 2281 | 124.0 | 1704 |
| **6043** | 5502 | 5 | 0 | 2012 | 2472 | 8.0 | 1300000.0 | 1517 | 2314 | 2241 | 55 | 220000000 | 1518815515 | 1606 | 143.0 | 2472 |
| **6646** | 2145 | 5 | 0 | 2015 | 221 | 7.1 | 370000.0 | 1189 | 706 | 2721 | 55 | 190000000 | 1515341399 | 2281 | 137.0 | 221 |

[21]:
```
#Read data
df=pd.read_csv(r'C:\Users\DELL\Desktop\python_data\movies.csv')
df.head()
```

[21]:

| | name | rating | genre | year | released | score | votes | director | writer | star | country | budget | gross | company | runtime |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | The Shining | R | Drama | 1980 | June 13, 1980 (United States) | 8.4 | 927000.0 | Stanley Kubrick | Stephen King | Jack Nicholson | United Kingdom | 19000000.0 | 46998772.0 | Warner Bros. | 146.0 |
| **1** | The Blue Lagoon | R | Adventure | 1980 | July 2, 1980 (United States) | 5.8 | 65000.0 | Randal Kleiser | Henry De Vere Stacpoole | Brooke Shields | United States | 4500000.0 | 58853106.0 | Columbia Pictures | 104.0 |
| **2** | Star Wars: Episode V - The Empire Strikes Back | PG | Action | 1980 | June 20, 1980 (United States) | 8.7 | 1200000.0 | Irvin Kershner | Leigh Brackett | Mark Hamill | United States | 18000000.0 | 538375067.0 | Lucasfilm | 124.0 |
| **3** | Airplane! | PG | Comedy | 1980 | July 2, 1980 (United States) | 7.7 | 221000.0 | Jim Abrahams | Jim Abrahams | Robert Hays | United States | 3500000.0 | 83453539.0 | Paramount Pictures | 88.0 |
| **4** | Caddyshack | R | Comedy | 1980 | July 25, 1980 (United States) | 7.3 | 108000.0 | Harold Ramis | Brian Doyle-Murray | Chevy Chase | United States | 6000000.0 | 39846344.0 | Orion Pictures | 98.0 |