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

In [2]:
```
#
df = pd.read_csv('dataset1.csv')
df
```

Out[2]:

	production_budget	worldwide_gross	original_title	start_year	runtime_minutes	genres
0	0.0	0.000000e+00	Sunghursh	2013	175.0	Action,Crime,Drama
1	425000000.0	2.776345e+09	Sunghursh	2013	175.0	Action,Crime,Drama
2	0.0	0.000000e+00	Ashad Ka Ek Din	2019	114.0	Biography,Drama
3	410600000.0	1.045664e+09	Ashad Ka Ek Din	2019	114.0	Biography,Drama
4	0.0	0.000000e+00	The Other Side of the Wind	2018	122.0	Drama
...	...	...	...	...	...	...
149526	0.0	0.000000e+00	Kuambil Lagi Hatiku	2019	123.0	Drama
149527	0.0	0.000000e+00	Rodolpho Teóphilo - O Legado de um Pioneiro	2015	NaN	Documentary
149528	0.0	0.000000e+00	Dankyavar Danka	2013	NaN	Comedy
149529	0.0	0.000000e+00	6 Gunn	2017	116.0	NaN
149530	0.0	0.000000e+00	Chico Albuquerque - Revelações	2013	NaN	Documentary

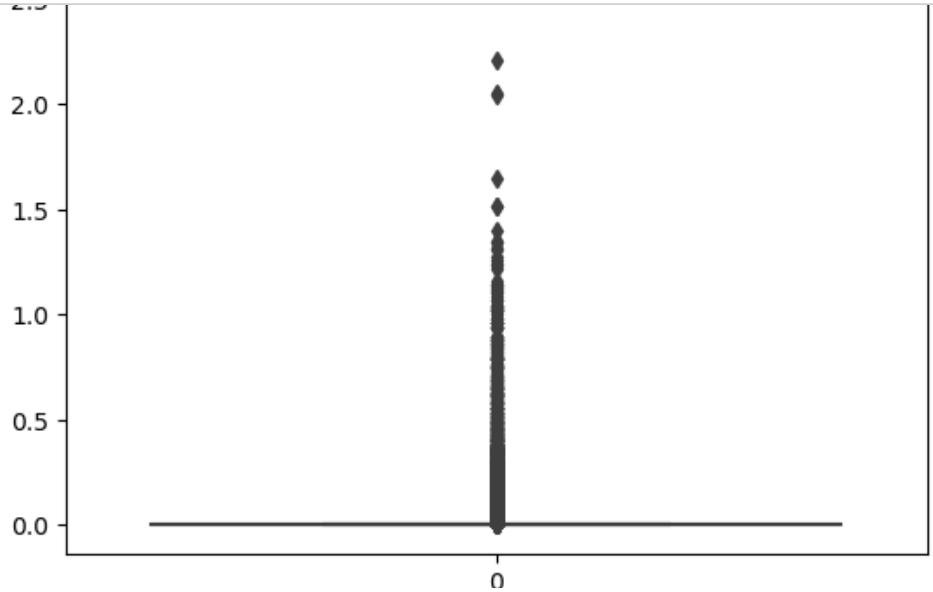
149531 rows × 6 columns

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

Out[3]:

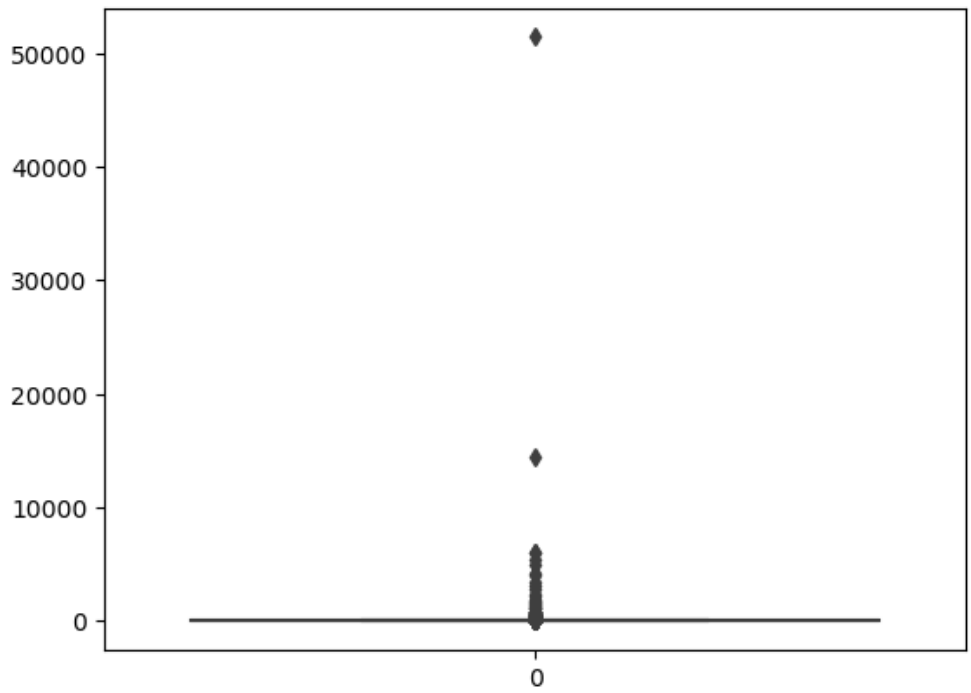
	production_budget	worldwide_gross	start_year	runtime_minutes
count	1.495310e+05	1.495310e+05	149531.000000	116373.000000
mean	1.221422e+06	3.537598e+06	2014.675907	86.224915
std	1.023131e+07	3.861793e+07	2.766890	164.993271
min	0.000000e+00	0.000000e+00	2010.000000	1.000000
25%	0.000000e+00	0.000000e+00	2012.000000	70.000000
50%	0.000000e+00	0.000000e+00	2015.000000	87.000000
75%	0.000000e+00	0.000000e+00	2017.000000	99.000000
max	4.250000e+08	2.776345e+09	2115.000000	51420.000000

In [4]:
```
# Visualizing different columns to check on outliers
sns.boxplot(df['worldwide_gross'])
```



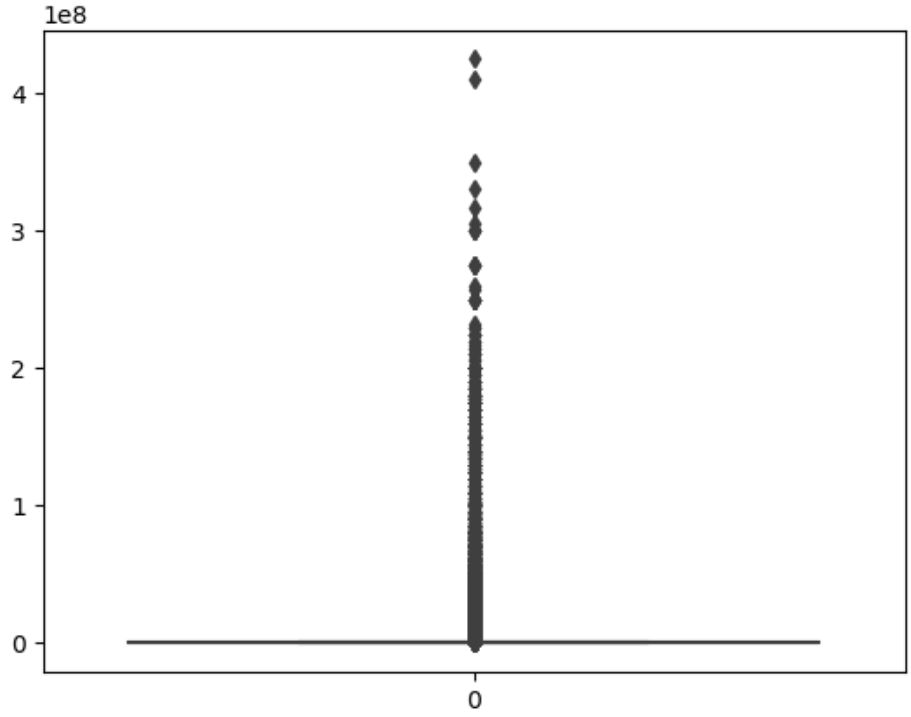
In [5]:
```
sns.boxplot(df['runtime_minutes'])
```

Out[5]: <Axes: >



In [6]:
```
sns.boxplot(df['production_budget'])
```

Out[6]: <Axes: >



In [7]:
```
# Replacing NaN by 0 for easier datatype conversion
df = df[(df['worldwide_gross'] != 0)]
df = df[(df['production_budget'] != 0)]
df
```

Out[7]:

	production_budget	worldwide_gross	original_title	start_year	runtime_minutes	gen
1	425000000.0	2.776345e+09	Sunghursh	2013	175.0	Action,Crime,Dra
3	410600000.0	1.045664e+09	Ashad Ka Ek Din	2019	114.0	Biography,Dra
5	350000000.0	1.497624e+08	The Other Side of the Wind	2018	122.0	Dra
7	330600000.0	1.403014e+09	Sabse Bada Sukh	2018	NaN	Comedy,Dra
9	317000000.0	1.316722e+09	La Telenovela Errante	2017	80.0	Comedy,Drama,Fant
...	...	...	...	...	...	
9162	7000.0	7.164400e+04	Stag Night of the Dead	2010	81.0	Action,Comedy,Ho
9163	7000.0	9.000000e+02	Mr. Nice	2010	121.0	Biography,Comedy,Cr
9165	6000.0	2.404950e+05	Marley	2012	144.0	Biography,Documentary,M
9166	5000.0	1.338000e+03	Welcome to the Rileys	2010	110.0	Dra
9168	1100.0	1.810410e+05	Sadiyaan: Boundaries Divide... Love Unites	2010	200.0	Drama,Roma

5415 rows × 6 columns

In [8]:
```
def outlier_remove(df, column):
    q1 = df[column].quantile(0.25)
    q3 = df[column].quantile(0.75)
    iqr = q3 - q1
    lower_bound = q1 - 1.5 * iqr
    upper_bound = q3 + 1.5 * iqr
    return df[(df[column] >= lower_bound) & (df[column] <= upper_bound)]
```

In [9]:
```
df = outlier_remove(df, 'worldwide_gross')
df = outlier_remove(df, 'production_budget')
df = outlier_remove(df, 'runtime_minutes')
df
```

Out[9]:

	production_budget	worldwide_gross	original_title	start_year	runtime_minutes	gen
1095	80000000.0	247023808.0	The Courier	2012	95.0	Action,Crime,Dra
1099	80000000.0	226739416.0	Pelé: Birth of a Legend	2016	107.0	Biography,Drama,S
1107	80000000.0	168311558.0	Bibliothèque Pascal	2010	105.0	Dra
1109	80000000.0	241200000.0	The Experiment	2010	96.0	Drama,Thr
1111	80000000.0	221468935.0	Stadilaista tangoa etsimässä	2010	90.0	Documentary,M
...	...	...	...	...	...	
9161	7000.0	841926.0	Pet	2016	94.0	Horror,Thr
9162	7000.0	71644.0	Stag Night of the Dead	2010	81.0	Action,Comedy,Hc
9163	7000.0	900.0	Mr. Nice	2010	121.0	Biography,Comedy,Cr
9165	6000.0	240495.0	Marley	2012	144.0	Biography,Documentary,M
9166	5000.0	1338.0	Welcome to the Rileys	2010	110.0	Dra

2177 rows × 6 columns

In [10]:
```
df.shape
```

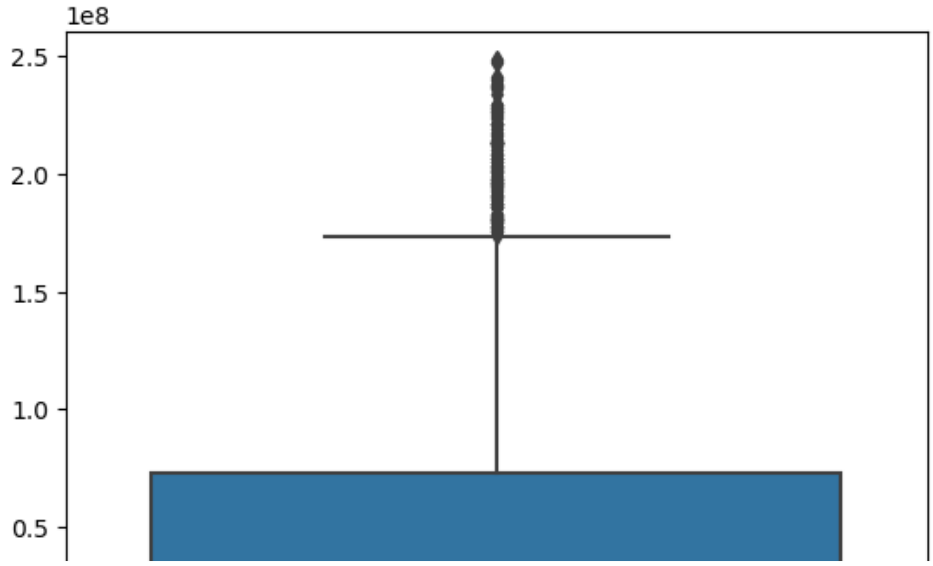
Out[10]: (2177, 6)

In [11]:
```
df.head(5)
```

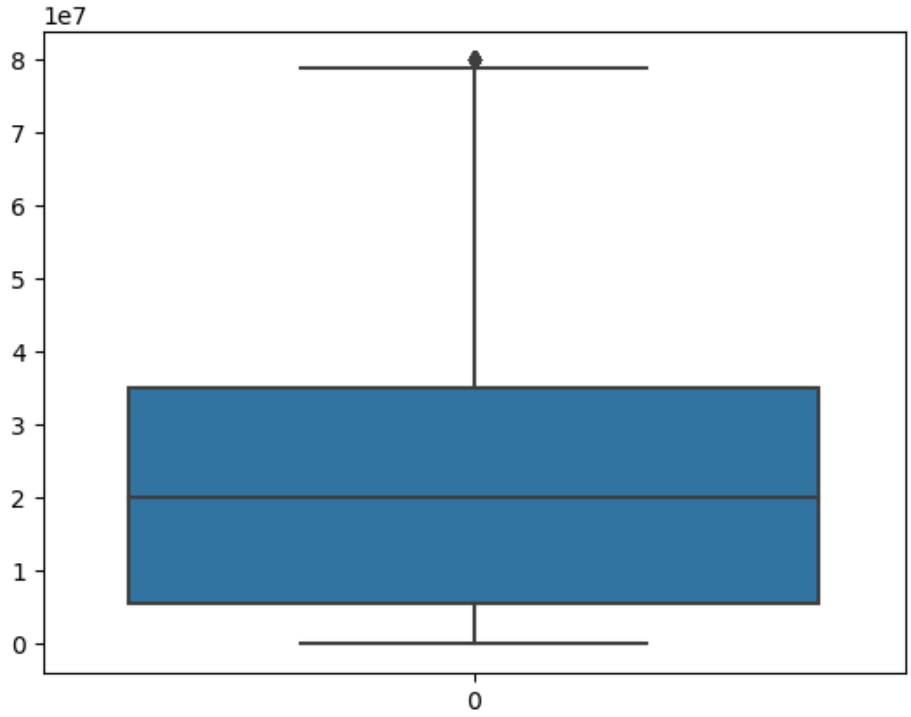
Out[11]:

	production_budget	worldwide_gross	original_title	start_year	runtime_minutes	genres
1095	80000000.0	247023808.0	The Courier	2012	95.0	Action,Crime,Drama
1099	80000000.0	226739416.0	Pelé: Birth of a Legend	2016	107.0	Biography,Drama,Sport
1107	80000000.0	168311558.0	Bibliothèque Pascal	2010	105.0	Drama
1109	80000000.0	241200000.0	The Experiment	2010	96.0	Drama,Thriller
1111	80000000.0	221468935.0	Stadilaista tangoa etsimässä	2010	90.0	Documentary,Music

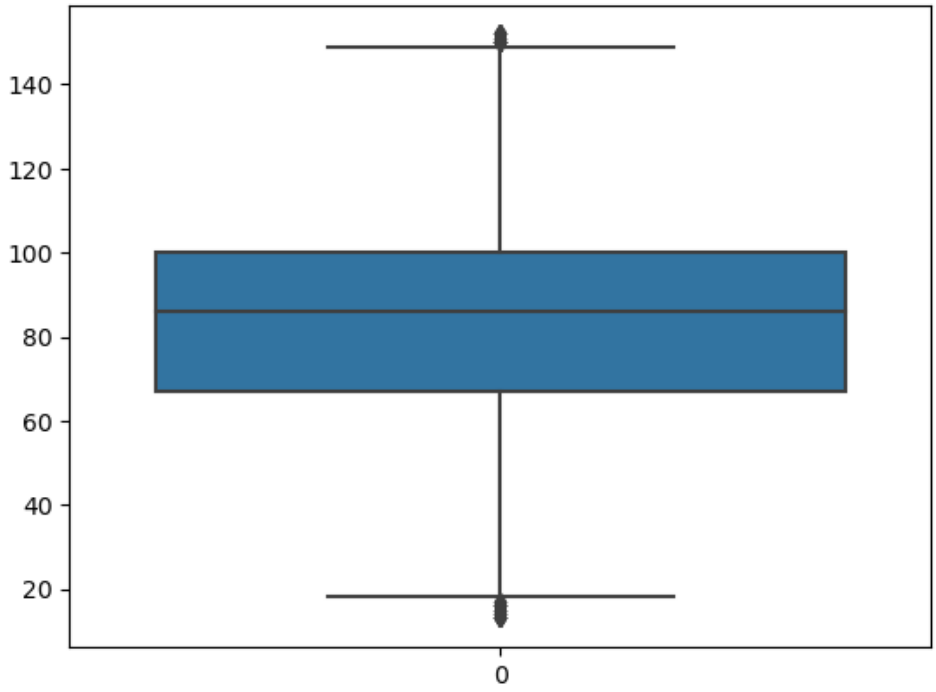
In [12]:
```
# checking visualization preview
df.reset_index(drop=True, inplace=True)
sns.boxplot(df['worldwide_gross']);
```



In [21]:
```
# checking visualization preview
sns.boxplot(df['production_budget']);
```



In [14]:
```
# checking visualization preview
sns.boxplot(df['runtime_minutes']);
```



In [15]:
```
df.describe()
```

Out[15]:

	production_budget	worldwide_gross	start_year	runtime_minutes
count	2.177000e+03	2.177000e+03	2177.000000	2177.000000
mean	2.349835e+07	4.922897e+07	2017.157097	84.491043
std	2.075417e+07	5.674658e+07	2.935761	25.241227
min	5.000000e+03	2.600000e+01	2010.000000	13.000000
25%	5.600000e+06	5.228617e+06	2016.000000	67.000000
50%	2.000000e+07	2.768874e+07	2019.000000	86.000000
75%	3.500000e+07	7.278517e+07	2019.000000	100.000000
max	8.000000e+07	2.485053e+08	2022.000000	152.000000

In [16]:
```
# Converting 'start_year' to a date format
df['start_year'] = pd.to_datetime(df['start_year'], format='%Y')

# Extracting the year
df['start_year'] = df['start_year'].dt.year
# Display the DataFrame
df
```

Out[16]:

	production_budget	worldwide_gross	original_title	start_year	runtime_minutes	gen
0	80000000.0	247023808.0	The Courier	2012	95.0	Action,Crime,Dra
1	80000000.0	226739416.0	Pelé: Birth of a Legend	2016	107.0	Biography,Drama,S
2	80000000.0	168311558.0	Bibliothèque Pascal	2010	105.0	Dra
3	80000000.0	241200000.0	The Experiment	2010	96.0	Drama,Thr
4	80000000.0	221468935.0	Stadilaista tangoa etsimässä	2010	90.0	Documentary,M
...	...	...	...	...	...	
2172	7000.0	841926.0	Pet	2016	94.0	Horror,Thr
2173	7000.0	71644.0	Stag Night of the Dead	2010	81.0	Action,Comedy,Ho
2174	7000.0	900.0	Mr. Nice	2010	121.0	Biography,Comedy,Cr
2175	6000.0	240495.0	Marley	2012	144.0	Biography,Documentary,M
2176	5000.0	1338.0	Welcome to the Rileys	2010	110.0	Dra

2177 rows × 6 columns

In [17]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2177 entries, 0 to 2176
Data columns (total 6 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   production_budget  2177 non-null   float64
 1   worldwide_gross    2177 non-null   float64
 2   original_title     2177 non-null   object 
 3   start_year         2177 non-null   int32  
 4   runtime_minutes    2177 non-null   float64
 5   genres             2115 non-null   object 
dtypes: float64(3), int32(1), object(2)
memory usage: 93.7+ KB
```

In [18]:
```
# Dropping null rows under the genres column as they are not replacable by another metho
df = df.dropna(subset=['genres'])
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 2115 entries, 0 to 2176
Data columns (total 6 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   production_budget  2115 non-null   float64
 1   worldwide_gross    2115 non-null   float64
 2   original_title     2115 non-null   object 
 3   start_year         2115 non-null   int32  
 4   runtime_minutes    2115 non-null   float64
 5   genres             2115 non-null   object 
dtypes: float64(3), int32(1), object(2)
memory usage: 107.4+ KB
```

In [19]:
```
# Checking if the data is fully clean and without missing data and ready for visualizati
df.isna().sum()
```

Out[19]:
```
production_budget    0
worldwide_gross      0
original_title       0
start_year           0
runtime_minutes      0
genres               0
dtype: int64
```

In [20]:
```
# Storing the data as a csv file
df.to_csv('cleaned_data.csv')
```

In [ ]:

In [ ]: