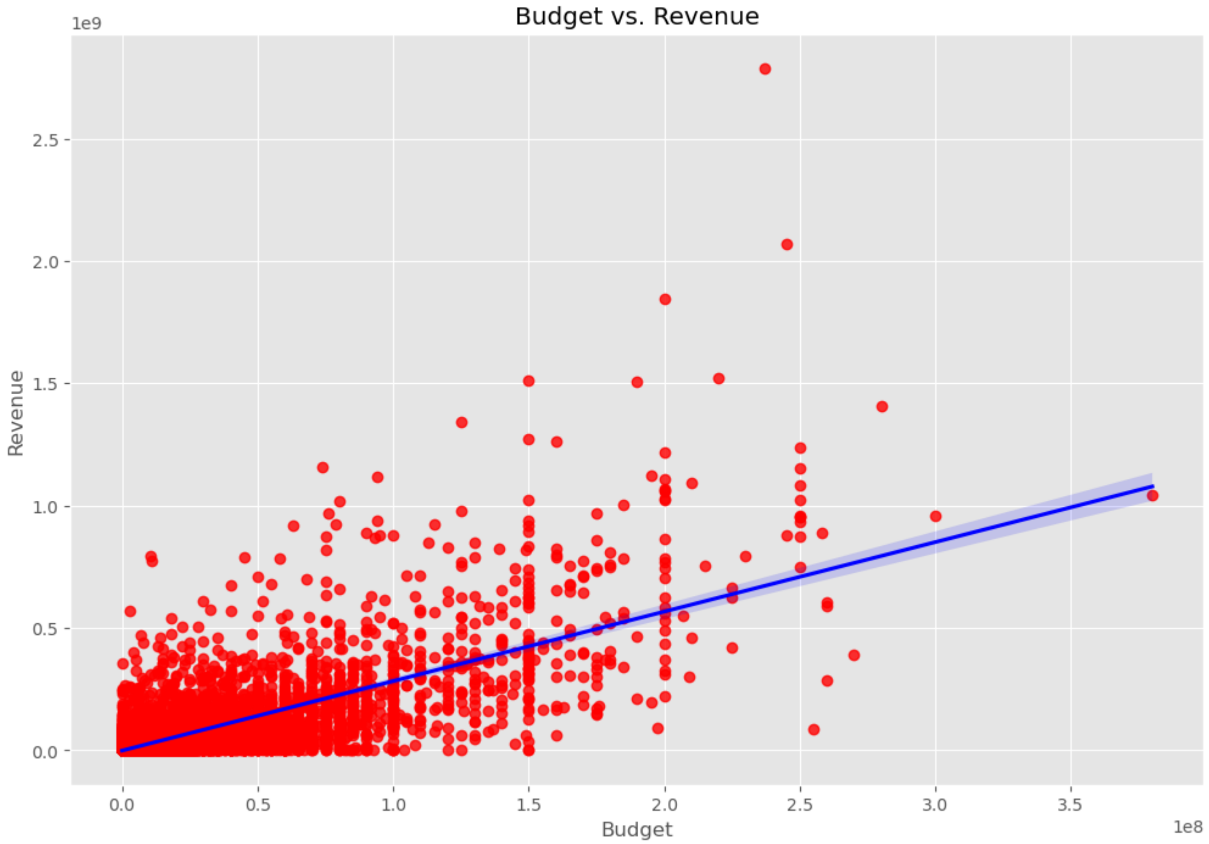
In [1]:

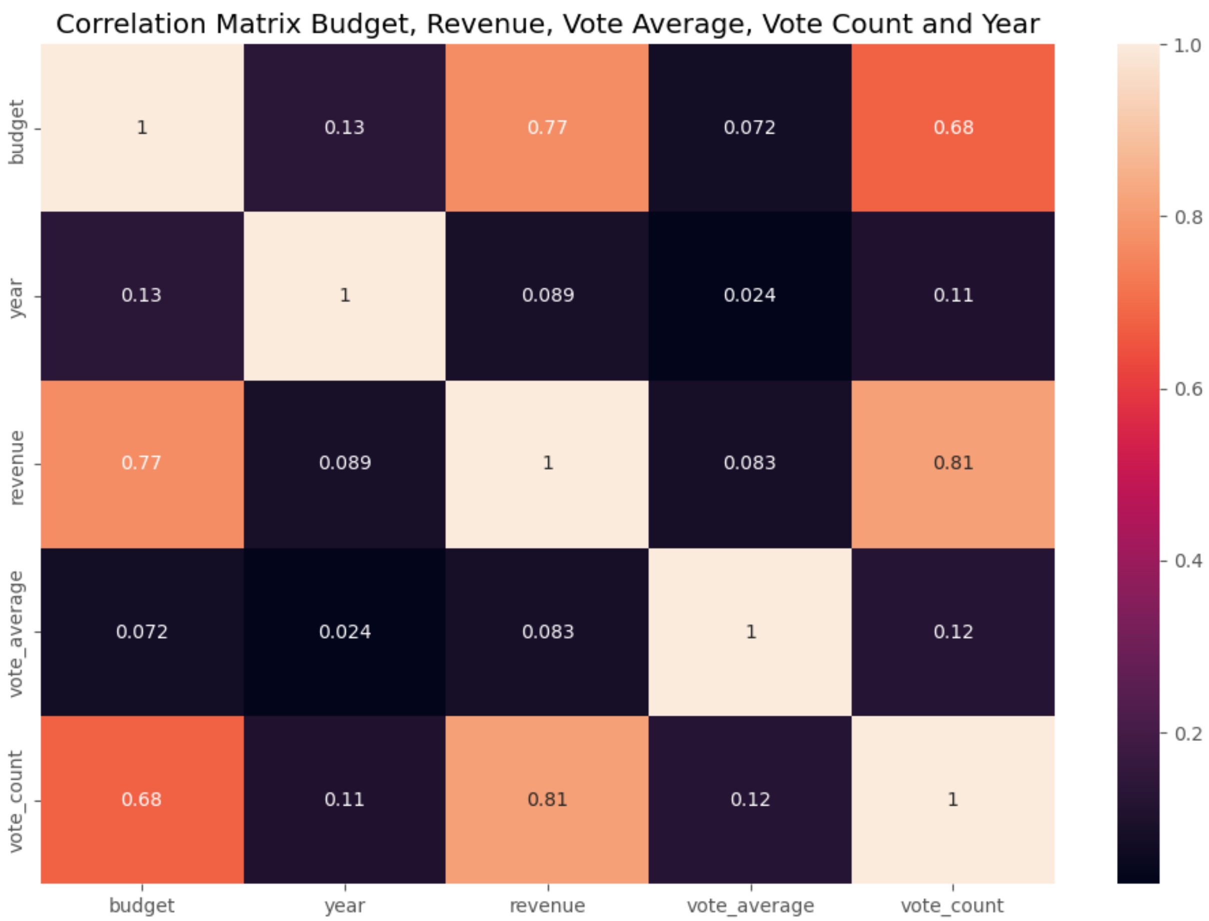
```
import pandas as pd
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
plt.style.use ('ggplot')
from matplotlib.pyplot import figure

%matplotlib inline
matplotlib.rcParams['figure.figsize'] = (12,8)



df = pd.read_csv(r"C:\Users\Dana\Desktop\Datasets\SQL Projects Examples\Database\te
for col in df.columns:
    pct_missing=np.mean(df[col].isnull())
    print ('{}-{}%'.format(col, pct_missing))
df.head()
df.dtypes

# we will check correlation between budget and revenue

from datetime import datetime
df['release_date'] = pd.to_datetime(df['release_date'],dayfirst=True) #.dt.strftime
df.drop_duplicates()
df=df.sort_values(by=['revenue'],  ascending=True)
df

# Scatterplot Budget vs Revenue
#plt.scatter(x=df['budget'], y=df['revenue'])
sns.regplot(x ='budget', y = 'revenue', data=df , scatter_kws={"color":"red"},  lin
df['revenue']=df['revenue'].astype('Int64')
df['budget']=df['budget'].astype('Int64')
plt.title ('Budget vs. Revenue')
plt.xlabel ('Budget')
plt.ylabel ('Revenue')
plt.show()
```

```
adult-0.0%
imdb_id-0.0003973422202600383%
budget-2.2074567890333546e-05%
id-2.2074567890333546e-05%
original_language-0.00026489481468400255%
release_date-0.003598154566124368%
revenue-0.002075009381691353 5%
status-0.003863049380808371%
title-0.001986711110130019%
vote_average-0.0%
vote_count-0.0017438908633363502%
imdb_id2-0.0003973422202600383%
imdb-0.0%
```



In [2]:

```
import pandas as pd
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
plt.style.use ('ggplot')
from matplotlib.pyplot import figure

%matplotlib inline
matplotlib.rcParams['figure.figsize'] = (12,8)



df = pd.read_csv(r"C:\Users\Dana\Desktop\Datasets\SQL Projects Examples\Database\te
df['year'] = df['release_date'].astype(str).str[6:]
df['year']  = pd.to_numeric(df['year'])
df1 = df[['budget',  'year'  ,'revenue' ,'vote_average','vote_count']]
df1.head()
correlation_matrix = df1.corr(method='pearson')
sns.heatmap(correlation_matrix, annot = True)
plt.title ('Correlation Matrix Budget, Revenue, Vote Average, Vote Count and Year')
plt.show()
```



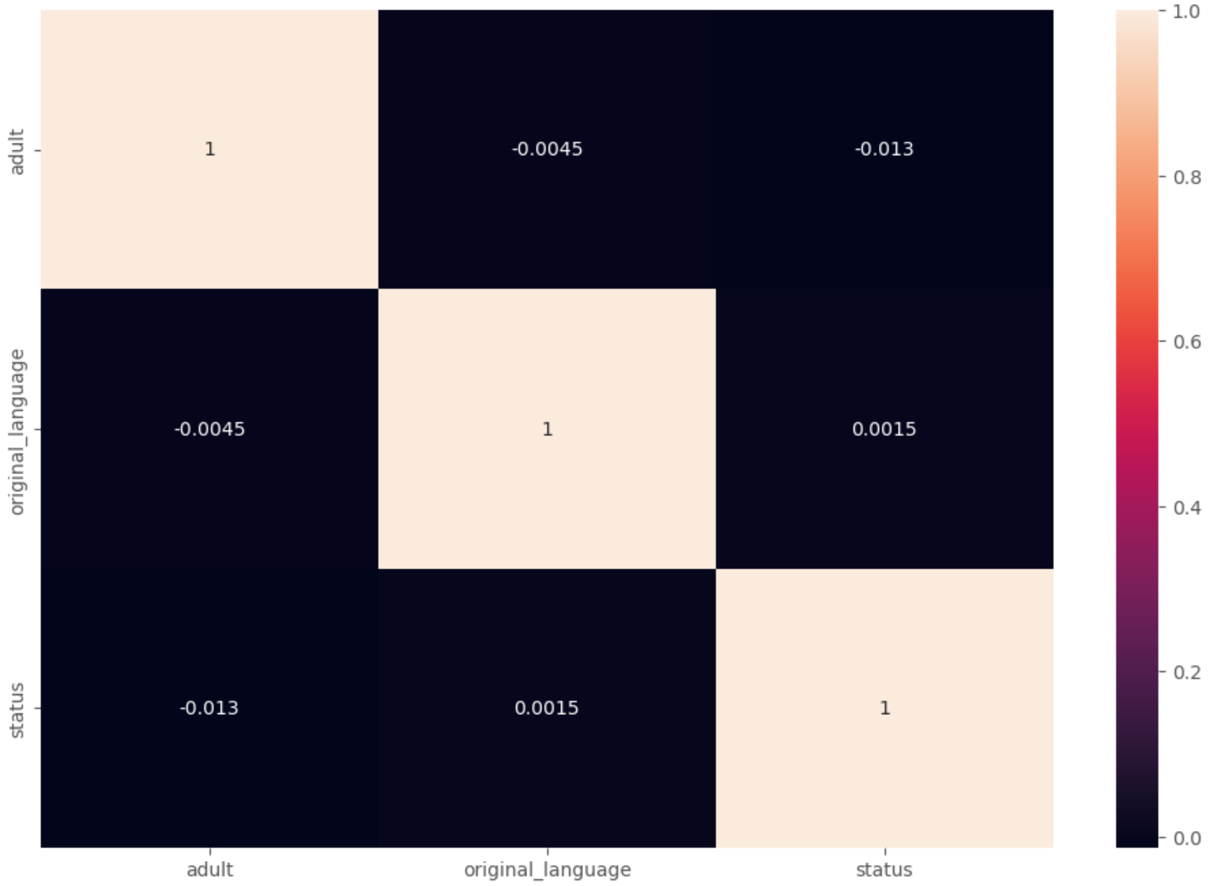
	budget	year	revenue	vote_average	vote_count
budget	1	0.13	0.77	0.072	0.68
year	0.13	1	0.089	0.024	0.11
revenue	0.77	0.089	1	0.083	0.81
vote_average	0.072	0.024	0.083	1	0.12
vote_count	0.68	0.11	0.81	0.12	1

In [3]:

```
import pandas as pd
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
plt.style.use ('ggplot')
from matplotlib.pyplot import figure

%matplotlib inline
matplotlib.rcParams['figure.figsize'] = (12,8)




df2 = pd.read_csv(r"C:\Users\Dana\Desktop\Datasets\SQL Projects Examples\Database\t
df3=df2
#df2 = df2[['adult', 'budget','revenue' , 'original_language' , 'status' ,'vote_ave
#for col_name in df2.columns:
    #if(df2.[col_name].dtypes == 'object'):
      # df2.[col_name]= df2.[col_name].astype('category')
#df2.[col_name]= df2.[col_name].cat.names
df3 = df3[['adult',  'original_language' , 'status' ]]
df3

df3['adult']=df3['adult'].astype('category').cat.codes
df3['original_language']=df3['original_language'].astype('category').cat.codes
df3['status']=df3['status'].astype('category').cat.codes
df3.corr()
```

```
correlation_matrix = df3.corr(method='pearson')
sns.heatmap(correlation_matrix, annot = True)
plt.show()
```

```
C:\Users\Dana\AppData\Local\Temp\ipykernel_12144\4172043848.py:24: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/u
ser_guide/indexing.html#returning-a-view-versus-a-copy
  df3['adult']=df3['adult'].astype('category').cat.codes
C:\Users\Dana\AppData\Local\Temp\ipykernel_12144\4172043848.py:25: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/u
ser_guide/indexing.html#returning-a-view-versus-a-copy
  df3['original_language']=df3['original_language'].astype('category').cat.codes
C:\Users\Dana\AppData\Local\Temp\ipykernel_12144\4172043848.py:26: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/u
ser_guide/indexing.html#returning-a-view-versus-a-copy
  df3['status']=df3['status'].astype('category').cat.codes
```

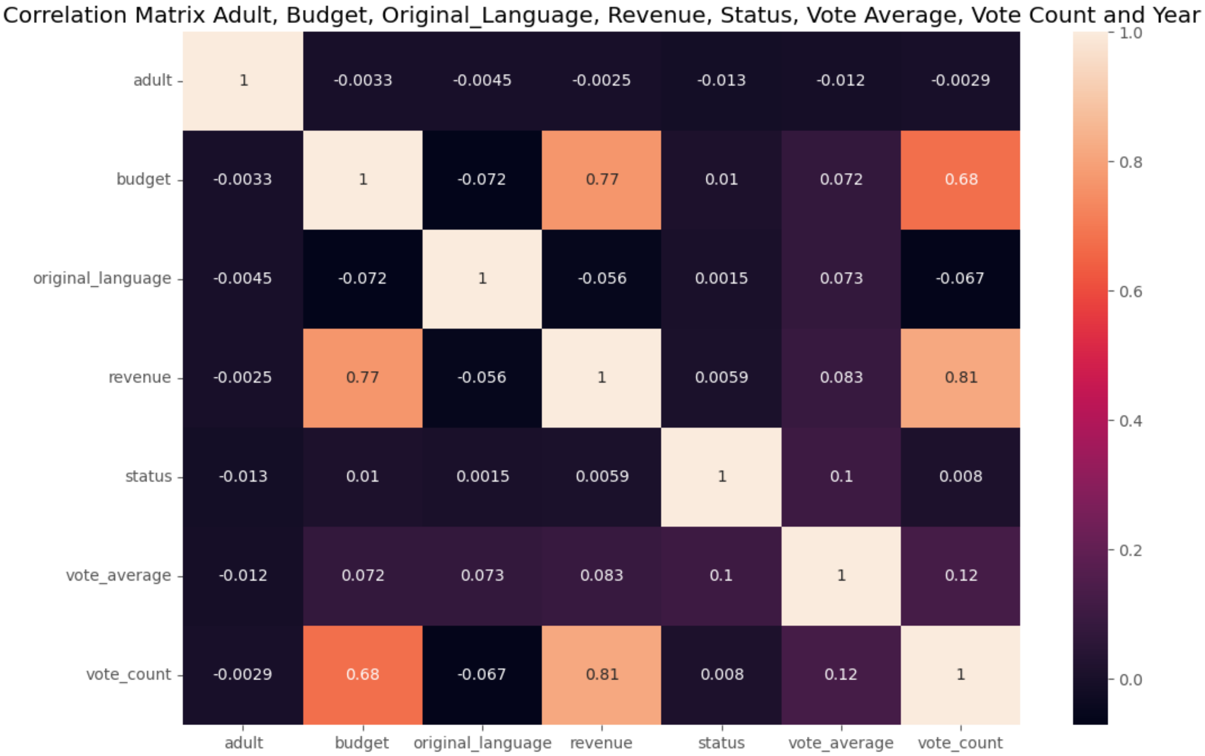


In [ ]:

```
import pandas as pd
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
plt.style.use ('ggplot')
from matplotlib.pyplot import figure

%matplotlib inline
matplotlib.rcParams['figure.figsize'] = (12,8)



df4 = pd.read_csv(r"C:\Users\Dana\Desktop\Datasets\SQL Projects Examples\Database\t
df5=df4
df5['year'] = df5['release_date'].astype(str).str[6:]
df5['year']  = pd.to_numeric(df5['year'])
df5.head()
```

In [6]:

```
import pandas as pd
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
plt.style.use ('ggplot')
from matplotlib.pyplot import figure

%matplotlib inline
matplotlib.rcParams['figure.figsize'] = (12,8)

#transformation of boolean and categorical columns to numerical

df6 = pd.read_csv(r"C:\Users\Dana\Desktop\Datasets\SQL Projects Examples\Database\t
df6.dtypes
if (df6['status'].dtype =='object'):
    df6['status'] = df6['status'].astype('category')
    df6['status'] = df6['status'].cat.codes
if (df6['adult'].dtype =='bool'):
    df6['adult'] = df6['adult'].astype('category')
    df6['adult'] = df6['adult'].cat.codes
if (df6['original_language'].dtype =='object'):
    df6['original_language'] = df6['original_language'].astype('category')
    df6['original_language'] = df6['original_language'].cat.codes
df6
df6 = df6[['adult', 'budget', 'original_language' , 'revenue' , 'status' ,'vote_ave
correlation_matrix = df6.corr(method='pearson')
sns.heatmap(correlation_matrix, annot = True)
plt.title ('Correlation Matrix Adult, Budget, Original_Language, Revenue, Status, V
plt.show()

# THere is correlation between Budget, Revenue and Vote_Count. There is no correlat
```



	adult	budget	original_language	revenue	status	vote_average	vote_count
adult	1	-0.0033	-0.0045	-0.0025	-0.013	-0.012	-0.0029
budget	-0.0033	1	-0.072	0.77	0.01	0.072	0.68
original_language	-0.0045	-0.072	1	-0.056	0.0015	0.073	-0.067
revenue	-0.0025	0.77	-0.056	1	0.0059	0.083	0.81
status	-0.013	0.01	0.0015	0.0059	1	0.1	0.008
vote_average	-0.012	0.072	0.073	0.083	0.1	1	0.12
vote_count	-0.0029	0.68	-0.067	0.81	0.008	0.12	1

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

```
In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:
```