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

# Access CSV

df0= pd.read_csv('/Users/ivanbernal/Downloads/tmdb-movies.csv')

# Exploración de los datos

df0.head(5)
list(df0) # column names
df0.info()
df0.describe()

#Clean, transform , eliminar columnas, ver valores nulos y 0

df1 =
df0.drop(['imdb_id','budget','original_title','cast','homepage','director','tagline','keywords'
,'overview','runtime','production_companies','release_date','budget_adj','revenue_adj'],a
xis=1)
df1.head(5)
df1.info()
df1.describe()

# Extract data 1985-2015

idx0=df1['release_year']>=1985
df2=df1[idx0]
df2.head(5)
df2.info()
df2.describe()

# Filter values 0

df_0=df2.query('revenue == 0')
df_0.describe()

# Replace values 0

df2['revenue'] = df2['revenue'].replace(0, np.NaN)
df2.info()
col = ['revenue']
```

```
df2.dropna(subset = col, how = 'any', inplace = True)

# Verify duplicates and drop

dups = df2[df2.duplicated()]
print(dups)
df2.drop_duplicates(inplace=True)

# Conjunto de datos final

df2.info()

# Visualization revenue vs years

df_revenue_by_year = df2.groupby(['release_year'])['revenue'].sum()
df_revenue_by_year.plot()
plt.title("Revenue Over Time")
plt.xlabel("Years")
plt.ylabel("Revenue in k-millions")
plt.show()

# Caso por géneros

# 1. Division de la columna géneros

df3 = df2
df3['genres'] = df3['genres'].str.split('|', expand = True)
df3['genres'].head(5)
list(df3)

# 2. Group by main genre

df_genres = df3
df_genres_revenue = df_genres.groupby(['genres', 'release_year'])['revenue'].sum()
df_genres_revenue = df_genres_revenue.to_frame().reset_index()


# 3. Visualization

g = sns.FacetGrid(df_genres_revenue, col='genres', hue='genres', col_wrap=4, )
g = g.map(plt.plot, 'release_year', 'revenue')
g = g.map(plt.fill_between, 'release_year', 'revenue',alpha=0.2).set_titles("{col_name}
Genres")
g = g.set_titles("{col_name}")
```

```
plt.subplots_adjust(top=0.92)
g = g.fig.suptitle('Evolution of the value of genres')
plt.show()

# Popularity

df4 = df3
df_pop = df4.groupby(['genres'])['popularity'].mean()
df_pop = df_pop.to_frame().reset_index()
most_pop = df_pop.sort_values(by = ['popularity'], ascending = False)
most10_pop = most_pop.head(10)
less_pop = df_pop.sort_values(by = ['popularity'])
less10_pop = less_pop.head(10)

sns.barplot(x = "genres", y = "popularity", data=most10_pop, palette = "hls", capsize =
0.05, saturation = 8, errcolor = "gray", errwidth = 2, ci = "sd")
plt.show()

sns.barplot(x = "genres", y = "popularity", data=less10_pop, palette = "hls", capsize = 0.05,
saturation = 8, errcolor = "gray", errwidth = 2, ci = "sd")
plt.show()

# Visialización adicional

df5 = df3
df_vote = df4.groupby(['genres'])['vote_average'].mean()
df_vote = df_vote.to_frame().reset_index()
most_vote = df_vote.sort_values(by = ['vote_average'], ascending = False)
most10_vote = most_vote.head(10)
less_vote = df_vote.sort_values(by = ['vote_average'])
less10_vote = less_vote.head(10)

sns.barplot(x = "genres", y = "vote_average", data=most10_vote, palette = "hls", capsize =
0.05, saturation = 8, errcolor = "gray", errwidth = 2, ci = "sd")
plt.show()

sns.barplot(x = "genres", y = "vote_average", data=less10_vote, palette = "hls", capsize =
0.05, saturation = 8, errcolor = "gray", errwidth = 2, ci = "sd")
plt.show()
```