1 Phase 1 Project 1- Microsoft Movie Industry Analysis

1.1 Author: Jhonathan D. Herrera-Shaikh

1.2 Overview

1.2.1 Business Understanding & Business Problem

Microsoft, an American multinational technology company that produces computer software, consumer electronics, personal computers, and related services headquartered at the Microsoft Redmond campus located in Redmond, Washington, United States, wishes to enter the movie industry as a strategic company objective given the high returns of companies in this industry.

The problem for Microsoft is that they do not know how to analyze vast amount of data residing in a collection of databases available to gain insight into the market. They wish to collect, prepare, and extract valuable business insights from these databases that will help them make great tactical initial decisions. Decisions for example, as to what type of movies should they initially support and why.And, if once supported, when would it be a good time to promote and publish these films. Given the vast amount of data available in databases about movies, they really need help from a data scientist firm that can help them overcome their challenge, and realize the beggining steps of their strategic objective.

Fortunately for Microsoft, the best data scientists consulting companies in the world have started to form. One of them, and probably the most reliable,is Pandas Soldiers Data Science Inc. run by one man, CEO Jhonathan David Herrera-Shaikh, a leader in this field. Microsoft turns to him for help them acquire information so that they can intelligently invest their dollars with the highest chance of being successful

Here in this presentation, therefore I will gain access to available databasases and the moviesinformation in it. I have to find information that would be relevant for Microsoft executives to enter the market. I'll first find descriptive information about movies in the movie industry. Second, I will decide which data factors are the most relevant and can have the most value for decision making. Third, I'll make an analysis on the data and provide insight and analytical findings while finally I will recommend the 3 best course of actions recommendations for the business exectives in the initial stages of their strategic objective.

1.3 Data Story

In this presentation I explored all 11 databases provided, however utilized 5 databases for my recooomendation including:

1. studio_gross : Industry competitors data about studios, the competitors of Microsoft. Identifying the names of the studio and their gross revenues both domestically and internationally.
2. bon.movie_gross.csv: Movie data. Identifying how much revenues has movie titles brought and in what year.
3. tn.movie_budgets.csv: Movie data. Identifying the budget for each production and how much money did each movie made as well as each movie's realese date.
4. opus.data: Free and available database available to the public for free with information about movies including their genres and revenue
5. imdb.title.basic.csv: Movie data. Identifying runtimes of movies, movie's titles and genres.

1.4 Recommendations

My analysis of the movie industry, achieved by munching data and utilizing visualizations on the databases for analysis, showed three main business relevant insights for business decision making and are as follows:

1. Microsoft should focus on understanding their competition, the top 5 market share holders their revenue streams preparing business strategies for both the domestic and foreign markets as the top market share holders recognize revenue from both streams.
2. Microsoft is first focus on 3 main genres initially, Drama, Action, Adventure and Comedy (possibly Thrillers as fourth one). This because there is more information and most sales for genre type, and we should start production knowing where to focus. Now that we know this, since Microsoft knows now begin, the next step is discovering more about them. I also recommend to approach getting to know these genres further, extracting databases specifically for these genre types to extract detailed value as to what factors influence these genres.¶
3. Microsoft is first focus Adventure and Action productions, and consider Musicals. This because the worldwide gross in these genre types provide the highest opportunity of worldwide gross earnings. Now that we know this, since Microsoft wants to be profitable and drive up revenues, we now now where that is likely to occur. I also recommend to approach getting to know these genres further, extracting databases specifically for these genre types to extract detailed value as to what factors influence these genres.

1.5 Methodology

The methodology to arrive at a recommendation on each database analyzed consisted of three steps. I did my set up for EDA (Exploratory Data Analysis) successfully, first loading, understanding,and cleaning the databases. To accomplish this, I imported the necessary libraries from Python, reading and exploring the data in the the databases through Pandas, and identifying modification and cleaning opportunities as necessary, finally visualizing and making a recommendation. In summary:

Step 1: Data Load & Transform to Pandas

Step 2: Data Understanding & Cleaning

Step 3: Data Visualization, Analysis, Recommendation

1.6 Future Work:

In addition to these initial recommendations. There is still additional insight that can be drawn from Microsoft databases. This includes, for example, impact of run times per movies genre or type in relationship to world gross.

2 Data Load & Transformation

2.0.0.1 Importing Libraries

In [233]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from bs4 import BeautifulSoup
import requests

%matplotlib inline
```

2.0.0.2 Loading Databases & Transforming into Pandas

In [234]:
```
#transforming the CSV files
df_studio          = pd.read_csv ('data/studio_gross.csv',       float_prec
df_bommoviegross   = pd.read_csv ('data/bom.movie_gross.csv',    float_prec
df_rating          = pd.read_csv('data/imdb.title.ratings.csv',  float_prec
df_basics          = pd.read_csv('data/imdb.name.basics.csv')
df_titles          = pd.read_csv('data/imdb.title.akas.csv')
df_titlebasics     = pd.read_csv('data/imdb.title.basics.csv')
df_crew            = pd.read_csv('data/imdb.title.crew.csv')
df_budget          = pd.read_csv('data/tn.movie_budgets.csv')
df_principals      = pd.read_csv('data/imdb.title.principals.csv')
imdb                = pd.read_csv('Data/imdb.title.crew.csv')

#transforming the TSV file
df_tsvmovie        = pd.read_csv('data/rt.movie_info.tsv', delimiter='\t')
```

2.0.0.3 First taking a look at the head and tail, shape and information. With the goal to understand the data and conduct EDL (Exploratory Data Analysis)

In [155]:

```
# Observing and exploring database 1
df_studio.head()
```

Out[155]:

	studio	domestic_gross	foreign_gross	total_gross
0	BV	415000000.0	652000000.0	1.067000e+09
1	BV	334200000.0	691300000.0	1.025500e+09
2	WB	296000000.0	664300000.0	9.603000e+08
3	WB	292600000.0	535700000.0	8.283000e+08
4	P/DW	238700000.0	513900000.0	7.526000e+08

In [156]:

```
df_studio.tail()
```

Out[156]:

	studio	domestic_gross	foreign_gross	total_gross
3382	Magn.	6200.0	0.0	6200.0
3383	FM	4800.0	0.0	4800.0
3384	Sony	2500.0	0.0	2500.0
3385	Synergetic	2400.0	0.0	2400.0
3386	Grav.	1700.0	0.0	1700.0

In [157]:

```
df_bommoviegross.head()
```

Out[157]:

	title	studio	domestic_gross	foreign_gross	year
0	Toy Story 3	BV	415000000.0	652000000	2010
1	Alice in Wonderland (2010)	BV	334200000.0	691300000	2010
2	Harry Potter and the Deathly Hallows Part 1	WB	296000000.0	664300000	2010
3	Inception	WB	292600000.0	535700000	2010
4	Shrek Forever After	P/DW	238700000.0	513900000	2010

In [158]:

```
df_bommoviegross.tail()
```

Out[158]:

	title	studio	domestic_gross	foreign_gross	year
3382	The Quake	Magn.	6200.0	NaN	2018
3383	Edward II (2018 re-release)	FM	4800.0	NaN	2018
3384	El Pacto	Sony	2500.0	NaN	2018
3385	The Swan	Synergetic	2400.0	NaN	2018
3386	An Actor Prepares	Grav.	1700.0	NaN	2018

In [159]:
```
df_rating.head()
```

Out[159]:

	tconst	averagerating	numvotes
0	tt10356526	8.3	31
1	tt10384606	8.9	559
2	tt1042974	6.4	20
3	tt1043726	4.2	50352
4	tt1060240	6.5	21

In [160]:
```
df_rating.tail()
```

Out[160]:

	tconst	averagerating	numvotes
73851	tt9805820	8.1	25
73852	tt9844256	7.5	24
73853	tt9851050	4.7	14
73854	tt9886934	7.0	5
73855	tt9894098	6.3	128

In [162]:
```
df_basics.head()
```

Out[162]:

	nconst	primary_name	birth_year	death_year	primary_professio
0	nm0061671	Mary Ellen Bauder	NaN	NaN	miscellaneous,production_manager,produce
1	nm0061865	Joseph Bauer	NaN	NaN	composer,music_department,sound_departmen
2	nm0062070	Bruce Baum	NaN	NaN	miscellaneous,actor,write
3	nm0062195	Axel Baumann	NaN	NaN	camera_department,cinematographer,art_departmen
4	nm0062798	Pete Baxter	NaN	NaN	production_designer,art_department,set_decorato

In [163]:

```
df_basics.tail()
```

Out[163]:

	nconst	primary_name	birth_year	death_year	primary_profession	known_for_titles
606643	nm9990381	Susan Grobes	NaN	NaN	actress	NaN
606644	nm9990690	Joo Yeon So	NaN	NaN	actress	tt9090932,tt8737130
606645	nm9991320	Madeline Smith	NaN	NaN	actress	tt8734436,tt9615610
606646	nm9991786	Michelle Modigliani	NaN	NaN	producer	NaN
606647	nm9993380	Pegasus Envoyé	NaN	NaN	director,actor,writer	tt8743182

In [164]:

```
df_titles.head()
```

Out[164]:

	title_id	ordering	title	region	language	types	attributes	is_original_title
0	tt0369610	10	Джурасик свят	BG	bg	NaN	NaN	0.0
1	tt0369610	11	Jurashikku warudo	JP	NaN	imdbDisplay	NaN	0.0
2	tt0369610	12	Jurassic World: O Mundo dos Dinossauros	BR	NaN	imdbDisplay	NaN	0.0
3	tt0369610	13	O Mundo dos Dinossauros	BR	NaN	NaN	short title	0.0
4	tt0369610	14	Jurassic World	FR	NaN	imdbDisplay	NaN	0.0

In [165]:

```
df_titles.tail()
```

Out[165]:

	title_id	ordering	title	region	language	types	attributes	is_original_title
331698	tt9827784	2	Sayonara kuchibiru	NaN	NaN	original	NaN	1.0
331699	tt9827784	3	Farewell Song	XWW	en	imdbDisplay	NaN	0.0
331700	tt9880178	1	La atención	NaN	NaN	original	NaN	1.0
331701	tt9880178	2	La atención	ES	NaN	NaN	NaN	0.0
331702	tt9880178	3	The Attention	XWW	en	imdbDisplay	NaN	0.0

In [166]:

```
df_titles.shape
```

Out[166]:

```
(331703, 8)
```

In [167]:
```
df_titlebasics.head()
```

Out[167]:

	tconst	primary_title	original_title	start_year	runtime_minutes	genres
0	tt0063540	Sunghursh	Sunghursh	2013	175.0	Action,Crime,Drama
1	tt0066787	One Day Before the Rainy Season	Ashad Ka Ek Din	2019	114.0	Biography,Drama
2	tt0069049	The Other Side of the Wind	The Other Side of the Wind	2018	122.0	Drama
3	tt0069204	Sabse Bada Sukh	Sabse Bada Sukh	2018	NaN	Comedy,Drama
4	tt0100275	The Wandering Soap Opera	La Telenovela Errante	2017	80.0	Comedy,Drama,Fantasy

In [168]:
```
df_titlebasics.columns
```

Out[168]:
```
Index(['tconst', 'primary_title', 'original_title', 'start_year',
       'runtime_minutes', 'genres'],
      dtype='object')
```

In [169]:
```
df_crew.head()
```

Out[169]:

	tconst	directors	writers
0	tt0285252	nm0899854	nm0899854
1	tt0438973	NaN	nm0175726,nm1802864
2	tt0462036	nm1940585	nm1940585
3	tt0835418	nm0151540	nm0310087,nm0841532
4	tt0878654	nm0089502,nm2291498,nm2292011	nm0284943

In [170]:
```
df_crew.tail()
```

Out[170]:

	tconst	directors	writers
146139	tt8999974	nm10122357	nm10122357
146140	tt9001390	nm6711477	nm6711477
146141	tt9001494	nm10123242,nm10123248	NaN
146142	tt9004986	nm4993825	nm4993825
146143	tt9010172	NaN	nm8352242

In [171]:

```
df_budget.head()
```

Out[171]:

	id	release_date	movie	production_budget	domestic_gross	worldwide_gross
0	1	Dec 18, 2009	Avatar	$425,000,000	$760,507,625	$2,776,345,279
1	2	May 20, 2011	Pirates of the Caribbean: On Stranger Tides	$410,600,000	$241,063,875	$1,045,663,875
2	3	Jun 7, 2019	Dark Phoenix	$350,000,000	$42,762,350	$149,762,350
3	4	May 1, 2015	Avengers: Age of Ultron	$330,600,000	$459,005,868	$1,403,013,963
4	5	Dec 15, 2017	Star Wars Ep. VIII: The Last Jedi	$317,000,000	$620,181,382	$1,316,721,747

In [172]:

```
df_budget.tail()
```

Out[172]:

	id	release_date	movie	production_budget	domestic_gross	worldwide_gross
5777	78	Dec 31, 2018	Red 11	$7,000	$0	$0
5778	79	Apr 2, 1999	Following	$6,000	$48,482	$240,495
5779	80	Jul 13, 2005	Return to the Land of Wonders	$5,000	$1,338	$1,338
5780	81	Sep 29, 2015	A Plague So Pleasant	$1,400	$0	$0
5781	82	Aug 5, 2005	My Date With Drew	$1,100	$181,041	$181,041

In [173]:

```
df_principals.head()
```

Out[173]:

	tconst	ordering	nconst	category	job	characters
0	tt0111414	1	nm0246005	actor	NaN	["The Man"]
1	tt0111414	2	nm0398271	director	NaN	NaN
2	tt0111414	3	nm3739909	producer	producer	NaN
3	tt0323808	10	nm0059247	editor	NaN	NaN
4	tt0323808	1	nm3579312	actress	NaN	["Beth Boothby"]

In [174]:
```
df_principals.tail()
```

Out[174]:

	tconst	ordering	nconst	category	job	characters
1028181	tt9692684	1	nm0186469	actor	NaN	["Ebenezer Scrooge"]
1028182	tt9692684	2	nm4929530	self	NaN	["Herself","Regan"]
1028183	tt9692684	3	nm10441594	director	NaN	NaN
1028184	tt9692684	4	nm6009913	writer	writer	NaN
1028185	tt9692684	5	nm10441595	producer	producer	NaN

In [175]:
```
df_tsvmovie.head()
```

Out[175]:

	id	synopsis	rating	genre	director	writer	theater_date	dvd_
0	1	This gritty, fast-paced, and innovative police...	R	Action and Adventure|Classics|Drama	William Friedkin	Ernest Tidyman	Oct 9, 1971	Se
1	3	New York City, not-too-distant-future: Eric Pa...	R	Drama|Science Fiction and Fantasy	David Cronenberg	David Cronenberg|Don DeLillo	Aug 17, 2012	J
2	5	Illeana Douglas delivers a superb performance ...	R	Drama|Musical and Performing Arts	Allison Anders	Allison Anders	Sep 13, 1996	Ap
3	6	Michael Douglas runs afoul of a treacherous su...	R	Drama|Mystery and Suspense	Barry Levinson	Paul Attanasio|Michael Crichton	Dec 9, 1994	Au
4	7	NaN	NR	Drama|Romance	Rodney Bennett	Giles Cooper	NaN	

3 1st Data Analysis Process : Using 1 Data Base for Analysis

I chose to explore and clean this database because is base to my first business recommendation

In [176]:

```
# Observing and exploring database 1
df_studio.head()
```

Out[176]:

	studio	domestic_gross	foreign_gross	total_gross
0	BV	415000000.0	652000000.0	1.067000e+09
1	BV	334200000.0	691300000.0	1.025500e+09
2	WB	296000000.0	664300000.0	9.603000e+08
3	WB	292600000.0	535700000.0	8.283000e+08
4	P/DW	238700000.0	513900000.0	7.526000e+08

In [177]:

```
df_studio.tail()
```

Out[177]:

	studio	domestic_gross	foreign_gross	total_gross
3382	Magn.	6200.0	0.0	6200.0
3383	FM	4800.0	0.0	4800.0
3384	Sony	2500.0	0.0	2500.0
3385	Synergetic	2400.0	0.0	2400.0
3386	Grav.	1700.0	0.0	1700.0

In [178]:

```
#looking deeper into the database
df_studio.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3387 entries, 0 to 3386
Data columns (total 4 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   studio          3382 non-null   object 
 1   domestic_gross  3359 non-null   float64
 2   foreign_gross   3387 non-null   float64
 3   total_gross     3359 non-null   float64
dtypes: float64(3), object(1)
memory usage: 106.0+ KB
```

In [179]:

```
df_studio.shape
```

Out[179]:

```
(3387, 4)
```

In [180]:

```
#understanding what are the top 5 values count
top5 = df_studio.studio.value_counts().head(5)
```

In [181]:
```
top5
```

Out[181]:
```
IFC      166
Uni.     147
WB       140
Magn.    136
Fox      136
Name: studio, dtype: int64
```

3.0.0.1 Database 'studio_gross' - Visualization, Analysis and Recommendation

In [182]:
```
#cleaning by viewing and summing studios by total gross
df_studio.groupby(['studio']).sum()
```

Out[182]:

	domestic_gross	foreign_gross	total_gross
studio			
3D	6100000.0	9900000.0	16000000.0
A23	164200.0	0.0	164200.0
A24	324194200.0	238462200.0	562656400.0
ADC	248200.0	0.0	248200.0
AF	2142900.0	3500000.0	5642900.0
...	...	...	...
XL	458000.0	0.0	458000.0
YFG	1100000.0	0.0	1100000.0
Yash	31631400.0	272825100.0	304392100.0
Zee	1100000.0	571000.0	1671000.0
Zeit.	5663500.0	20300000.0	25963500.0

257 rows × 3 columns

In [183]:
```
df_histo = df_studio.groupby(['studio']).agg('sum')
```

In [184]:
```
df_histo= df_histo.sort_values('total_gross', ascending=False).head(10)
```

In [185]:
```
df_histo.index
```

Out[185]:
```
Index(['BV', 'Fox', 'WB', 'Uni.', 'Sony', 'Par.', 'WB (NL)', 'LGF', 'LG/
S',
       'P/DW'],
      dtype='object', name='studio')
```

In [186]:

```
font = {'family' : 'Helvetica',
        'weight' : 'bold',
        'size'   : 10}

plt.rc('font', **font)
```

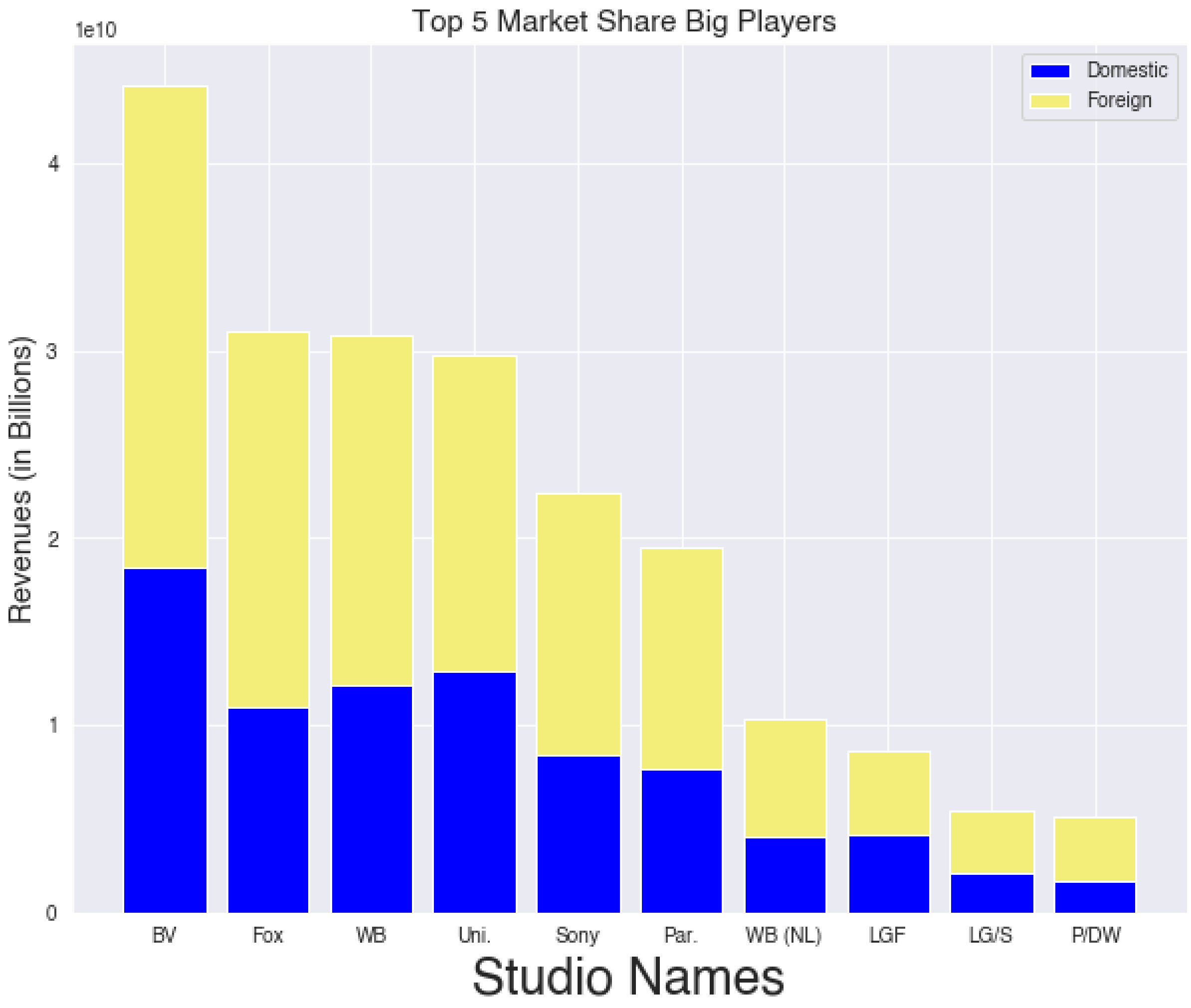
In [187]:

```
plt.figure(figsize=(10,8))

studios = df_histo.index
dom_gross = df_histo.domestic_gross
for_gross = df_histo.foreign_gross
tot_gross = df_histo.total_gross

plt.bar(range(len(studios)), dom_gross, color='blue')
plt.bar(range(len(studios)), for_gross, color='#f2ee79', bottom=dom_gross)
plt.title('Top 5 Market Share Big Players ', fontsize=15)
plt.xlabel('Studio Names', fontsize=25)
plt.ylabel('Revenues (in Billions)', fontsize=15)
plt.xticks(range(len(studios)), studios)

plt.legend(['Domestic', 'Foreign'])
plt.show();
```



3.0.1 Analysis

Microsoft would want to know the market first. Based on this premise, I'd recommend Microsoft first to understand who their biggest competitors are. We have discovered who they are and their revenue by stream (foreign or domestic). Knowing this will set up other analysis that are more tailored towards understanding these top competitors. Microsoft can't underestimate the foreign markets. This is because the top players play in both, domestic can foreign. We understand how big of the market share do they really hold.

3.1 Recommendation 1

My first recommendation to Microsoft is to research that BV, Fox, WB, Uni, and Sony, because they have a hold on the the largest share of the market, and we should mimic their production methodology (how they operate, who are their leaders, what movie(s)/genres makes them the most money). Since Microsoft wants to be in the big leagues, I also recommend to approach the research knowing focusing on understanding the business strategy in both foreign and the domestic market strategy, as these top players clearly earn revenue from both streams.

4 2nd Data Analysis Process: Uses 2 DB's- Join for analysis

4.0.1 database: df_budget. Containing information about movies, production budgets, domestic and worlwide gross.

In [235]:

```
#Looking inside the database budget again
df_budget.head()
```

Out[235]:

	id	release_date	movie	production_budget	domestic_gross	worldwide_gross
0	1	Dec 18, 2009	Avatar	$425,000,000	$760,507,625	$2,776,345,279
1	2	May 20, 2011	Pirates of the Caribbean: On Stranger Tides	$410,600,000	$241,063,875	$1,045,663,875
2	3	Jun 7, 2019	Dark Phoenix	$350,000,000	$42,762,350	$149,762,350
3	4	May 1, 2015	Avengers: Age of Ultron	$330,600,000	$459,005,868	$1,403,013,963
4	5	Dec 15, 2017	Star Wars Ep. VIII: The Last Jedi	$317,000,000	$620,181,382	$1,316,721,747

In [197]:

```
df_budget.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5782 entries, 0 to 5781
Data columns (total 6 columns):
 #   Column             Non-Null Count  Dtype 
---  ------             --------------  ----- 
 0   id                 5782 non-null   int64 
 1   release_date       5782 non-null   object
 2   movie              5782 non-null   object
 3   production_budget  5782 non-null   object
 4   domestic_gross     5782 non-null   object
 5   worldwide_gross    5782 non-null   object
dtypes: int64(1), object(5)
memory usage: 271.2+ KB
```

In [198]:

```
df_budget.dtypes
```

Out[198]:

```
id                    int64
release_date         object
movie                object
production_budget    object
domestic_gross       object
worldwide_gross      object
dtype: object
```

In [199]:

```
#dropping columns
df_budget.drop(['domestic_gross'], axis=1, inplace=True)
```

In [200]:

```
#finding out data types
df_budget.dtypes
```

Out[200]:

```
id                    int64
release_date         object
movie                object
production_budget    object
worldwide_gross      object
dtype: object
```

In [201]:

```
# I want to change the columns to plain number integers I can work later wi
df_budget.head(2)
```

Out[201]:

	id	release_date	movie	production_budget	worldwide_gross
0	1	Dec 18, 2009	Avatar	$425,000,000	$2,776,345,279
1	2	May 20, 2011	Pirates of the Caribbean: On Stranger Tides	$410,600,000	$1,045,663,875

In [202]:

```
# removing dollar signs and commas from dollar amounts
# converting dollar amounts from strings into integers
df_budget['production_budget'] = df_budget['production_budget'].str.replace
df_budget['worldwide_gross']   = df_budget['worldwide_gross'].str.replace('
df_budget.head(2)
```

Out[202]:

	id	release_date	movie	production_budget	worldwide_gross
0	1	Dec 18, 2009	Avatar	425000000	2776345279
1	2	May 20, 2011	Pirates of the Caribbean: On Stranger Tides	410600000	1045663875

4.0.2 database: df1. Opus. Containg movie titles by genre, budget, and production years- This is available on the web. Downloaded to my pc and brought to for the purpose of this analysis.

In [95]:

```
#I downloaded a new data base that has genres only joined it with my budget
df1= pd.read_csv('/Users/jonax/Documents/opus_df.csv')
```

In [96]:

```
#looking at the head
df1.head()
```

Out[96]:

	Unnamed: 0	title	production_year	budget	dom_gross	int_gross	creative_type	prod_me
0	0	Madea's Family Reunion	2006	10000000	63257940	62581	Contemporary Fiction	Live A
1	1	Krrish	2006	10000000	1430721	31000000	Science Fiction	Live A
2	2	End of the Spear	2006	10000000	11748661	175380	Historical Fiction	Live A
3	3	A Prairie Home Companion	2006	10000000	20342852	6373339	Contemporary Fiction	Live A
4	4	Saw III	2006	10000000	80238724	83638091	Contemporary Fiction	Live A

In [97]:
```
#it's a very clearn data base, I'll change the 'title', to 'movie' to condu
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1936 entries, 0 to 1935
Data columns (total 9 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Unnamed: 0       1936 non-null   int64
 1   title            1936 non-null   object
 2   production_year  1936 non-null   int64
 3   budget           1936 non-null   int64
 4   dom_gross        1936 non-null   int64
 5   int_gross        1936 non-null   int64
 6   creative_type    1923 non-null   object
 7   prod_method      1925 non-null   object
 8   genre            1926 non-null   object
dtypes: int64(5), object(4)
memory usage: 136.2+ KB
```

In [98]:
```
# renaming certain columns
df1 = df1.rename(columns={'title':'movie'})
```

In [99]:
```
#verifying it column has been changed
df1.head(2)
```

Out[99]:

	Unnamed: 0	movie	production_year	budget	dom_gross	int_gross	creative_type	prod_metho
0	0	Madea's Family Reunion	2006	10000000	63257940	62581	Contemporary Fiction	Live Acti
1	1	Krrish	2006	10000000	1430721	31000000	Science Fiction	Live Acti

In [100]:
```
#setting the index to movie on this database
df1.set_index('movie', inplace=True)
```

4.1 Conducting an Inner Join : df_super

In [101]:

```
#joining two database (inner) at movie columun
df_super= df_budget.join(df1, how='inner')
df_super
```

Out[101]:

	id	release_date	production_budget	worldwide_gross	Unnamed: 0	production_year	b
movie							
10 Days in a Madhouse	48	Nov 11, 2015	12000000	14616	1492	2015	120
10,000 B.C.	51	Mar 7, 2008	105000000	269065678	483	2008	1050
12 Rounds	37	Mar 27, 2009	20000000	17306648	558	2009	200
12 Strong	64	Jan 19, 2018	35000000	71118378	1836	2017	350
12 Years a Slave	18	Oct 18, 2013	20000000	181025343	1246	2013	200
...	...	...	...	...	...	...	
Zootopia	57	Mar 4, 2016	150000000	1019429616	1627	2015	1500
Zulu	82	Dec 31, 2013	16000000	1844228	1233	2013	160
Zwartboek	48	Apr 6, 2007	22000000	27238354	226	2007	220
mother!	59	Sep 15, 2017	30000000	42531076	1819	2017	300
xXx: Return of Xander Cage	15	Jan 20, 2017	85000000	345033359	1742	2016	850

1932 rows × 12 columns

4.1.0.1 The New Joined Dabase characteristics

In [102]:
```
df_super.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 1932 entries, 10 Days in a Madhouse to xXx: Return of Xander Cage
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype 
---  ------             --------------  ----- 
 0   id                 1932 non-null   int64 
 1   release_date       1932 non-null   object
 2   production_budget  1932 non-null   int64 
 3   worldwide_gross    1932 non-null   int64 
 4   Unnamed: 0         1932 non-null   int64 
 5   production_year    1932 non-null   int64 
 6   budget             1932 non-null   int64 
 7   dom_gross          1932 non-null   int64 
 8   int_gross          1932 non-null   int64 
 9   creative_type      1920 non-null   object
 10  prod_method        1922 non-null   object
 11  genre              1924 non-null   object
dtypes: int64(8), object(4)
memory usage: 196.2+ KB
```

In [103]:
```
#dropping unnecesary columns on my joint database
df_super.drop(['Unnamed: 0', 'release_date', 'creative_type', 'dom_gross',
```

In [104]:
```
#looking at the cleaned database
df_super
```

Out[104]:

	id	production_budget	worldwide_gross	production_year	budget	int_gross	
movie							
10 Days in a Madhouse	48	12000000	14616	2015	12000000	0	
10,000 B.C.	51	105000000	269065678	2008	105000000	174281477	
12 Rounds	37	20000000	17306648	2009	20000000	5071954	
12 Strong	64	35000000	71118378	2017	35000000	25298665	
12 Years a Slave	18	20000000	181025343	2013	20000000	124353350	
...	...	...	...	...	...	...	
Zootopia	57	150000000	1019429616	2015	150000000	678436100	
Zulu	82	16000000	1844228	2013	16000000	1844228	Thri
Zwartboek	48	22000000	27238354	2007	22000000	22839822	Thri
mother!	59	30000000	42531076	2017	30000000	24731072	Thri
xXx: Return of Xander Cage	15	85000000	345033359	2016	85000000	300134946	

1932 rows × 7 columns

In [105]:
```
#I'll make the numbers a bit more manageable for both production budget and
df_super ['production_budget']= df_super['production_budget']/1000000
```

In [106]:
```
df_super ['worldwide_gross']= df_super['worldwide_gross']/1000000
```

In [107]:

```
#checking it out
df_super
```

Out[107]:

	id	production_budget	worldwide_gross	production_year	budget	int_gross	
movie							
10 Days in a Madhouse	48	12.0	0.014616	2015	12000000	0	
10,000 B.C.	51	105.0	269.065678	2008	105000000	174281477	
12 Rounds	37	20.0	17.306648	2009	20000000	5071954	
12 Strong	64	35.0	71.118378	2017	35000000	25298665	
12 Years a Slave	18	20.0	181.025343	2013	20000000	124353350	
...	...	...	...	...	...	...	
Zootopia	57	150.0	1019.429616	2015	150000000	678436100	
Zulu	82	16.0	1.844228	2013	16000000	1844228	Thrille
Zwartboek	48	22.0	27.238354	2007	22000000	22839822	Thrille
mother!	59	30.0	42.531076	2017	30000000	24731072	Thrille
xXx: Return of Xander Cage	15	85.0	345.033359	2016	85000000	300134946	

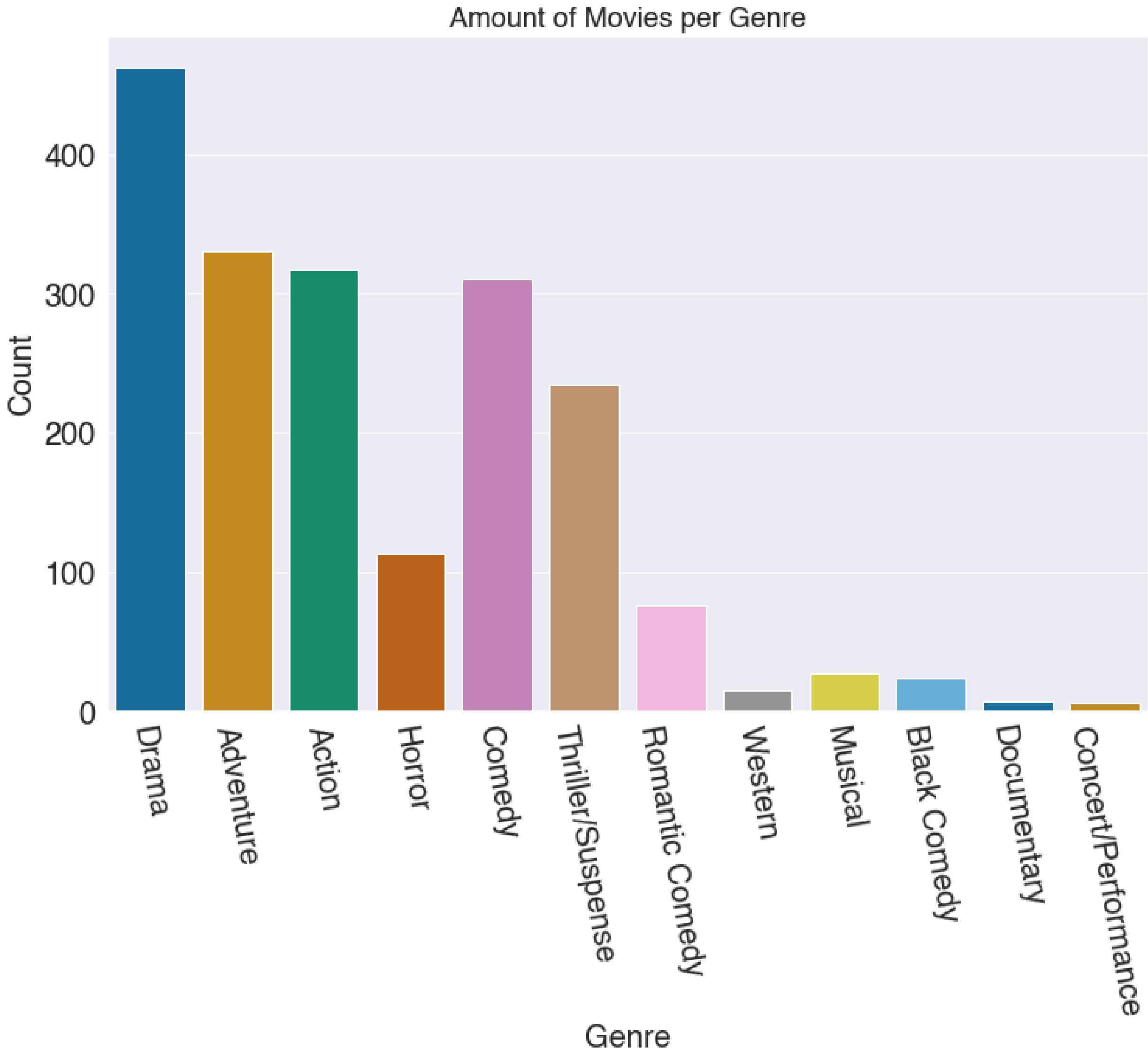
1932 rows × 7 columns

4.1.1 Now that I have a Joined Database (df_super) I'll visualize and analyse to propose recommendations

4.1.1.1 Plotting my chart

In [232]:

```
# plotting the number of movies per genre in dataset to find out insight an
plt.figure(figsize=(12,8))
sns.countplot(x='genre', data=df_super, palette='colorblind')
plt.title('Amount of Movies per Genre', fontsize=18)
plt.ylabel('Count', fontsize=20)
plt.xlabel('Genre', fontsize=20)
plt.xticks(fontsize=20)
plt.yticks(fontsize=20)
plt.xticks(rotation=-80);
```



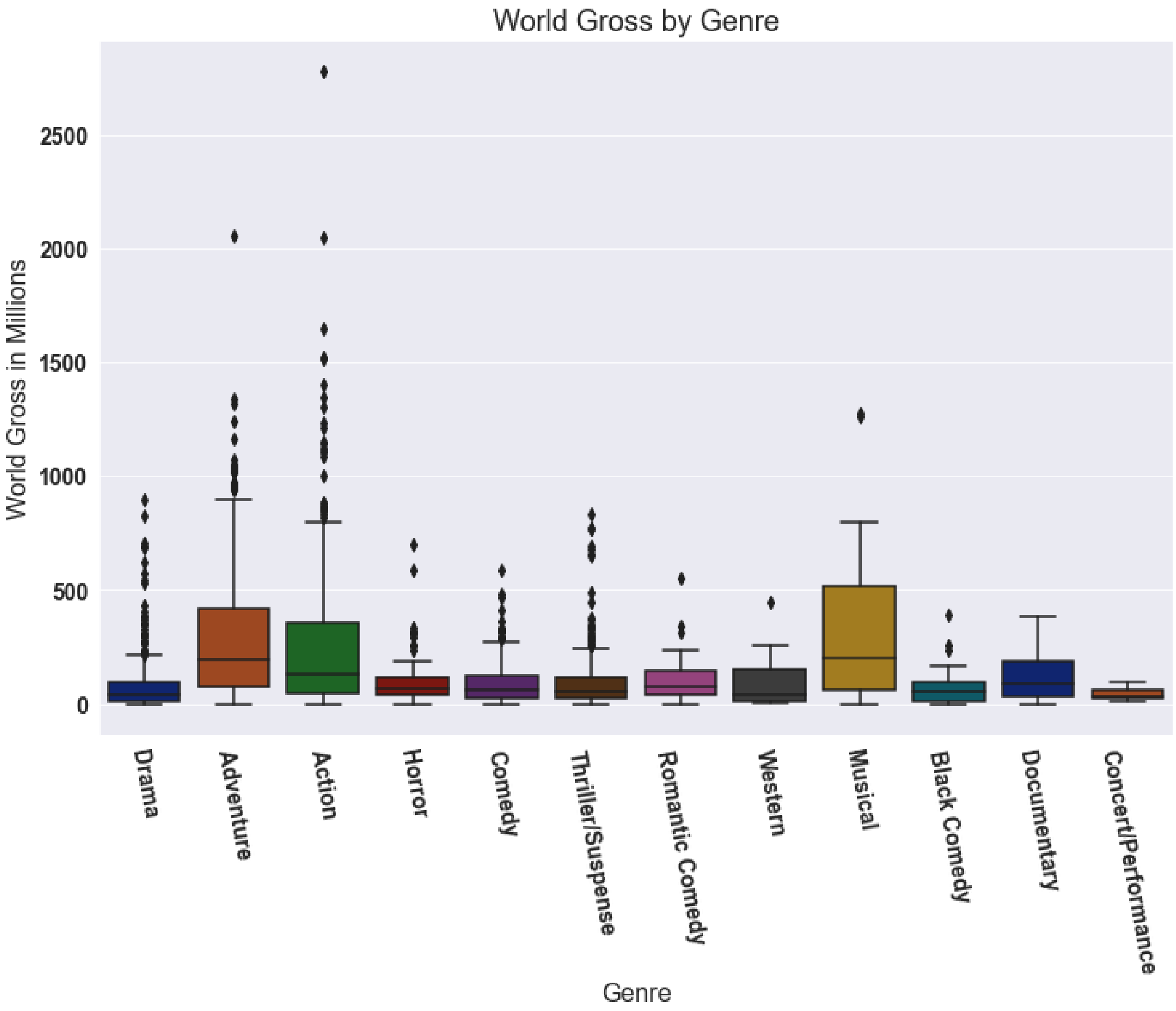
4.2 Analysis

4.2.0.1 Microsoft would want to know what movies sell the most. Based on this premise, we have discovered what genres are the most popular out of 1,932 movie titles and each title being identified by genre type we can realize valuable information. Knowing these main genres will provide guidance as to the next step, to set up tailored analysis on these genres that will tell us how to make these type of movies succesfully. Microsoft can't start making Romantic Comedy, Documentaries as these are clearly the least estimated by the public.

4.3 Recommendiation 2

4.3.0.1 My second recommendation to Microsoft is first focus on 3 main genres initially, Drama, Action, Adventure and Comedy (possibly Thrillers as fourth one). This because there is more information and most sales for genre type, and we should start production knowing where to focus. Now that we know this, since Microsoft knows now begin, the next step is discovering more about them. I also recommend to approach getting to know these genres further, extracting databases specifically for these genre types to extract detailed value as to what factors influence these genres.

In [109]:
```
#For my visualizations, I found it helpful to show my plots both with and w
# generating box plot of world gross statistics per genre
plt.figure(figsize=(12,8))
sns.set_style('darkgrid')
sns.boxplot(x='genre', y='worldwide_gross', data=df_super, palette='dark')
plt.xticks(rotation=-80)
plt.ylabel('World Gross in Millions', fontsize=16)
plt.xlabel('Genre', fontsize = 16)
plt.title('World Gross by Genre', fontsize = 18)
plt.xticks(fontsize=14)
plt.yticks(fontsize=14);
#saved in images as fig2
#plt.subplots_adjust(bottom=0.2)
#plt.savefig('./images/fig2.png')
```



4.4 Analysis

4.4.0.1 Microsoft would also want to know about money. Worldwide gross earnings is a very good factor in knowing this. Through this anlysis, the chart above makes us visualize the reality for worwide gross and tells us the about opportunity that there is in making world gross in million by genre. With this information we can understand what types of movies generate the highest gross worlwide and can make a recommendation as well as noting

interest findings. For example, in our previous graph we saw the genre type musical as low in count of movies, we see that it has a high potential to make worlwide gross.

4.4.1 Recommendation 3

4.4.1.1 My third recommendation to Microsoft is first focus Adventure and Action productions, and consider Musicals. This because the worldwide gross in these genre types provide the highest opportunity of worldwide gross earnings. Now that we know this, since Microsoft wants to be profitable and drive up revenues, we now now where that is likely to occur. I also recommend to approach getting to know these genres further, extracting databases specifically for these genre types to extract detailed value as to what factors influence these genres.

5 Summary & Conclusions

5.1 My analysis of the movie industry, achieved by munching data and utilizing visualizations on the databases for analysis, showed three main business relevant insights for business decision making and are as follows:

1. Microsoft should focus on understanding their competition, the top 5 market share holders their revenue streams preparing business strategies for both the domestic and foreign markets as the top market share holders recognize revenue from both streams.
2. Microsoft is first focus on 3 main genres initially, Drama, Action, Adventure and Comedy (possibly Thrillers as fourth one). This because there is more information and most sales for genre type, and we should start production knowing where to focus. Now that we know this, since Microsoft knows now begin, the next step is discovering more about them. I also recommend to approach getting to know these genres further, extracting databases specifically for these genre types to extract detailed value as to what factors influence these genres.¶
3. Microsoft is first focus Adventure and Action productions, and consider Musicals. This because the worldwide gross in these genre types provide the highest opportunity of worldwide gross earnings. Now that we know this, since Microsoft wants to be profitable and drive up revenues, we now now where that is likely to occur. I also recommend to approach getting to know these genres further, extracting databases specifically for these genre types to extract detailed value as to what factors influence these genres.

5.1.0.1 This was a great practice project. It made me practice what I've learned in Phase 1. It forced me to think as a real world data scientist. Moreover, it made me see my strenghts but also my weaknesses in areas I need to improve moving forward. I was able to make recommendations with my new knowledge to Microsoft and I'm very happy about that.