In [ ]:
```
from matplotlib import pyplot as plt
from IPython import display
import pandas as pd
import numpy as np
import csv
import sqlite3
```

Proof of Concept Goal - Show Studio Can Be Successful At Turning A Profit

Accomplishing this goal involves looking at types of movies that lend themselves to high profitability. In addition, partnering with another succesful studio that is experienced in making these types of movies will help to lay the framework for the new studio's success.

The Bottom Line - Demonstrating profitability at the start will encourage studios, producers, filmcrew and actors to participate in future projects.

1. Focus On Profitability by Selecting Highest Profitable Sub-genre

2. Keep Production Budget Low to Minimize Risk

3. Partner with Successful Studio Experienced in Making Selected Sub-genre

Sub-genre is the best way to breakdown types of movies as it can define the entire production process. For example, below are some of the top grossing movies for the Action/Adventure/Sci-Fi genre:

In [ ]:
```
df_action = pd.read_csv('data/ideal_only_aas.csv',index_col=0)
df_action.head()
```

Out[ ]:

	Movie Title	Budget (USD)	Total Gross (USD)	Profitability (GM%)	Release Year	Average IMDB Rating	Director
436	Jurassic World	$215,000,000	$2,301,125,489	90.66	2015	7.0	Colin Trevorrow
870	The Avengers	$225,000,000	$2,141,215,444	89.49	2012	8.0	Joss Whedon
53	Black Panther	$200,000,000	$2,048,317,790	90.24	2018	7.3	Ryan Coogler
464	Avengers: Age of Ultron	$330,600,000	$1,862,019,831	82.25	2015	7.3	Joss Whedon
54	Jurassic World: Fallen Kingdom	$170,000,000	$1,723,492,559	90.14	2018	6.2	J.A. Bayona

Immediately what comes to mind are elaborate film sets, stunt teams and a star-studded cast. These requirements keep a production budget relatively high compared to other genres. In contrast, Compare this to top grosing movies from the Adventure/Animation/Comedy genre:

In [ ]:
```
df_anim = pd.read_csv('data/ideal_only_aac.csv',index_col=0)
df_anim.head()
```

Out[ ]:

	Movie Title	Budget (USD)	Total Gross (USD)	Profitability (GM%)	Release Year	Average IMDB Rating	Director
751	Monsters University	$200,000,000	$1,012,076,658	80.24	2013	7.2	Dan Scanlon
1178	Shrek Forever After	$165,000,000	$994,981,460	83.42	2010	6.3	Mike Mitchell
1018	The Smurfs	$110,000,000	$706,363,481	84.43	2011	5.4	Raja Gosnell
51	Hotel Transylvania 3: Summer Vacation	$65,000,000	$694,580,054	90.64	2018	6.3	Genndy Tartakovsky
908	Wreck-It Ralph	$165,000,000	$685,924,198	75.94	2012	7.7	Rich Moore

Most animated films require no physical set and no stunt team. A lot of an animated film's budget will go into the animation and modeling of the movie versus the cast budget. On average the budget for an animated movie will be lower than that of an Action/Adventure/Sci-Fi movie.

In [ ]:
```
def clean_money(budget):
    '''
    Take money value stored as string and convert to integer
    '''
    split_list = budget.split(',')
    split_list[0] = split_list[0][1:]
    answer = ''
    for part in split_list:
        answer += part

    return int(answer)

print(f'Mean of Top Action/Adventure/Sci-fi Production Budgets: ${df_action["Bud
print(f'Mean of Top Adventure/Animation/Comedy Production Budgets: ${df_anim["Bu
```

```
Mean of Top Action/Adventure/Sci-fi Production Budgets: $191.88 million
Mean of Top Adventure/Animation/Comedy Production Budgets: $111.4 million
```

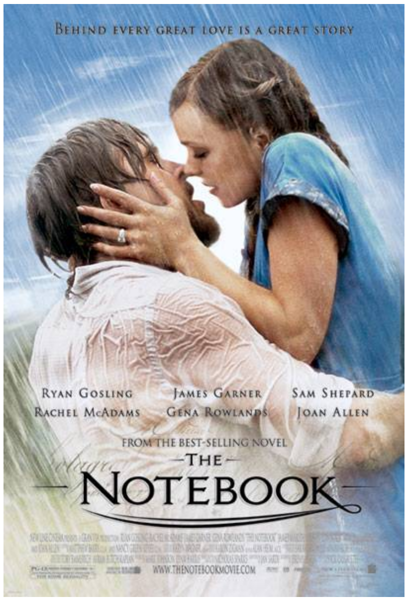
These defining differences between genres are not just between vastly different types like animation and action. They also exist within the genre subsets. For example, the adventure/animation/comedy genre has the highest average budget when compared to other genre amalgams that include animation.

It's important to note that these genre labels with multiple are not grouping together of the individual genres. Instead, each genre is being used as a descriptor to create a brand new, more specific genre.This leads to the importance of genre specificity.

Sub-Genre Specificity:

Gone are the days of single genre movies. It is impossible to group movies by a single core genre name (i.e Action, Animation, Comedy, Drama, Romance). There is a difference between

The Notebook and *50 First Dates*. One is a Romantic/Drama and the other is a Romantic/Comedy. These movies are created differently, marketed differently and viewed by different audiences.



≠

Viewing Production Budget as A Source of Risk:

"*It takes money to make money.*" - Sol Luckman

The above saying carries a lot of weight. As seen in the results below, a higher production budget equates to a higher gross income. However, the business decision as defined at the start is not looking for highest gross income but highest profitability. In this case, production budget now serves as a source of risk. The higher the production budget the more money is being risked on the film.

The goal is proof of concept - can the studio make a profitable movie.

Data Exploration

The primary data sources came from IMDB Datasets and Box Office Mojo by IMDB.

The data columns important to this business decision are:

- `Production Budget`
- `Total Gross Income`
- `Genre`
- `Studio Name`
- `Movie Title`
- `Director Name`
- `Oscar Award`

Data Cleaning

For this decision there were some restrictions placed on the datasets to better represent the current film-making environment.

Only movies with:

- Production Budget and Total Gross Income Data
- A Recent Release Date (2000 or Later)
- A Common Genre (Having 20 or More Movies Released Since 2000)

```
- Common Genres - Action/Adventure/Sci-Fi, Adventure/Animation/Comedy,
Comedy/Romance
- Uncommon Genres - Crime/Drama/Musical, Comedy/Documentary/Horror
```

The data was also restricted to an IQR of 80% with respect to profitability which is defined in the Feature Exploration section. The cleaning process can be found in the appendix folder.

Feature Engineering

Profitability is the key feature when looking at a business' financial health. Operating at a loss will ensure a swift end. For this study Profitability will be defined as Percent Gross Margin.

Profitability:

Define Profitability as Percent Gross Margin ($GM\%$).

$GM\% = \frac{I-B}{B}$

Where:

$I =$ Total Gross Income

$B =$ Budget

Profitability Explained:

In words, Profitability is what percent of every dollar made is above the cost. The max Profitability a movie can have is 100% (all profit and no costs). There is no limit on the lower end of Profitability. For Example, if a movie that cost 1,000,000 to make only makes 750,000 that would be a Profitability of -33% or a profit loss. The higher the production budget and the lower the gross income, the lower the Profitability.

Creating Profitability Feature:

The profitability feature was created using the total gross income and production budget numbers from the Box Office Mojo dataset and added as its own column in the genre dataset.

Data Analysis

From the IMDB and Budget datasets, two final datasets were created:

- The Genre Dataset - compares genre, profitability, budget and director - each row is a movie
- The Studio Dataset - compares studio and total gross income - each row is a movie

Profitability and Production Budget By Genre

Importing the Genre Dataset Grouped By Genre

In [ ]:

```
# Importing the grouped by genre dataset
df_idea1_g_grouped = pd.read_csv('data/idea1_main_genre_grouped.csv')

#Reset genre to be the index after importing csv
df_idea1_g_grouped.index = df_idea1_g_grouped['main_genre']
df_idea1_g_grouped.drop('main_genre',axis=1,inplace=True)
df_clean = df_idea1_g_grouped.copy(deep=True)

def dirty_money(dollar_int):
    '''
    Changes dollars as integer to dollars as string with commas and $ sign
    '''
    dollar_int = int(dollar_int)
    dollar_str = str(dollar_int)[::-1]
    final_dollar = ''
    counter = 0
    for number in dollar_str:
        counter += 1
        final_dollar += number
        if counter % 3 == 0:
            final_dollar += ','
    if final_dollar[::-1][0] == ',':
        final_dollar = final_dollar[:-1]
        return '$' + final_dollar[::-1]
    else:
        return '$' + final_dollar[::-1]

df_clean['mean_budget'] = df_idea1_g_grouped['mean_budget'].map(dirty_money)
df_clean.rename({'gross_margin':'Mean Profitability (GM%)','mean_budget':'Mean B
df_clean.filter(['Mean Profitability (GM%)', 'Mean Budget (USD)']).head()
```

Out[ ]:

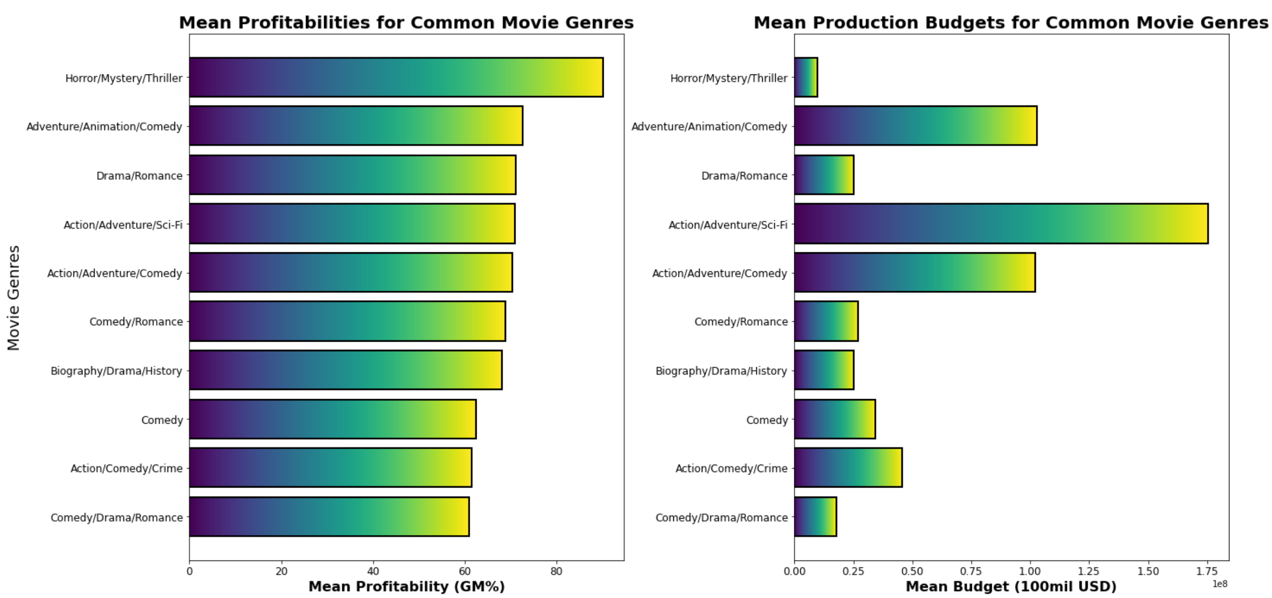
main_genre	Mean Profitability (GM%)	Mean Budget (USD)
Horror/Mystery/Thriller	90.182759	$9,780,172
Adventure/Animation/Comedy	72.574925	$102,768,656
Drama/Romance	71.060690	$25,263,448
Action/Adventure/Sci-Fi	71.002308	$175,530,769
Action/Adventure/Comedy	70.281429	$102,071,428

Above are the top 5 entries in the Genre Dataset grouped by genre. It is sorted by the mean profitability of each genre. Below shows the mean profitabilities of each genre as a bar graph.

In [ ]:

```
def gradientbars(bars):
    grad = np.atleast_2d(np.linspace(0,1,256))
    ax = bars[0].axes
```

```
    lim = ax.get_xlim()+ax.get_ylim()
    for bar in bars:
        bar.set_zorder(1)
        bar.set_facecolor("none")
        x,y = bar.get_xy()
        w, h = bar.get_width(), bar.get_height()
        ax.imshow(grad, extent=[x,x+w,y,y+h], aspect="auto", zorder=0)
    ax.axis(lim)

fig, axes = plt.subplots(nrows=1,ncols=2)

bar = axes[0].barh(y=df_ideal_g_grouped.index[::-1],width=df_ideal_g_grouped['gr
bar1 = axes[1].barh(y=df_ideal_g_grouped.index[::-1],width=df_ideal_g_grouped['m

gradientbars(bar)
gradientbars(bar1)
fig.set_figheight(10)
fig.set_figwidth(20)
fig.tight_layout(pad=4)
axes[0].tick_params(axis='both',which='major',labelsize=12)
axes[1].tick_params(axis='both',which='major',labelsize=12)
axes[0].set_title('Mean Profitabilities for Common Movie Genres',fontsize=20,fon
axes[1].set_title('Mean Production Budgets for Common Movie Genres',fontsize=20,
axes[0].set_xlabel('Mean Profitability (GM%)',fontsize=16,fontweight='bold')
axes[1].set_xlabel('Mean Budget (100mil USD)',fontsize=16,fontweight='bold')
axes[0].set_ylabel('Movie Genres',fontsize=18)
axes[1].set_ylabel('')

plt.show()
```

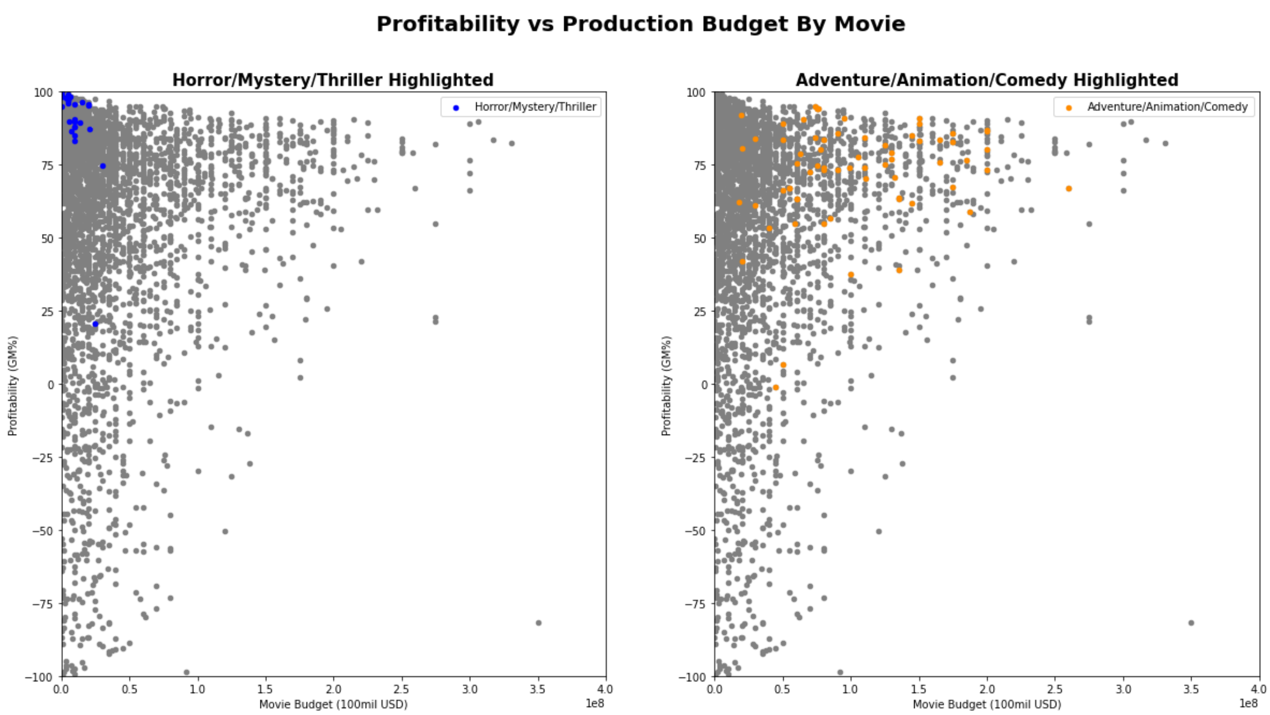


Not only is **Horror/Mystery/Thriller** at the top for profit, it also has by far the lowest budget costs compared to the other genres in the top 7 most profitable. **Adventure/Animation/Comedy** comes in second for profitability but is second highest for production budget.

The Horror/Mystery/Thriller Genre has the Lowest Production Budget and Highest Profitability

The next two graphs shows Profitability v Production Budget by movie. The movies within the top two genres for profitability (as discovered above) are highlighted.

In [ ]:
```
df_idea1_g = pd.read_csv('data/idea1_main_genre.csv')

# Plotting
fig, axes = plt.subplots(nrows=1,ncols=2)
fig.set_figheight(10)
fig.set_figwidth(20)
# fig.set_facecolor("")
plt1_list = [df_idea1_g_grouped.index[0]]
plt2_list = [df_idea1_g_grouped.index[1]]

df_idea1_g[df_idea1_g['release_year'] >= 2000].plot(ax=axes[0],kind='scatter' ,
df_idea1_g[(df_idea1_g['release_year'] >= 2000) & (df_idea1_g['main_genre'].isin

df_idea1_g[df_idea1_g['release_year'] >= 2000].plot(ax=axes[1],kind='scatter' ,
df_idea1_g[(df_idea1_g['release_year'] >= 2000) & (df_idea1_g['main_genre'].isin

axes[0].set_title('Horror/Mystery/Thriller Highlighted',fontsize=15,fontweight='
axes[1].set_title('Adventure/Animation/Comedy Highlighted',fontsize=15,fontweigh
fig.suptitle('Profitability vs Production Budget By Movie',fontsize=20, fontweig
for i in [0,1]:
        axes[i].set_ylabel('Profitability (GM%)')
        axes[i].set_xlabel('Movie Budget (100mil USD)')
        axes[i].set_ylim(-100,100)
        axes[i].set_xlim(0,4e8)
        # axes[i].set_facecolor("")
plt.show()
```



The left graph shows horror/mystery/thriller movies clustered to the top left. This indicates high profitability and some of the lowest movie budgets across all genres. The next most profitable genre is shown in the right graph. This shows a much larger spread on production budget.

Horror/Mystery/Thriller Genre Director Performance - For Chosing Director

The table below shows the top 4 grossing Horror/Mystery/Thriller movies with director information.

```
In [ ]: # Read in csv
df_director = pd.read_csv('data/ideal_only_hmt.csv',index_col=0)
df_director['Budget (USD)'] = df_director['Budget (USD)'].map(lambda x: x[:-8] +
df_director['Total Gross (USD)'] = df_director['Total Gross (USD)'].map(lambda x

df_director.filter(['Movie Title', 'Budget (USD)', 'Total Gross (USD)']).iloc[:4
```

Out[ ]:

	Movie Title	Budget (USD)	Total Gross (USD)
705	The Conjuring	$20 MM	$455 MM
152	Annabelle: Creation	$15 MM	$407 MM
560	Annabelle	$6 MM	$341 MM
37	Insidious: The Last Key	$10 MM	$235 MM

James Wan has directed, produced or written half of the most successful horror/mystery/thriller movies of the last two decades. Four of those movies were the top four grossing movies since 2000. He was also the main director for the top grossing horror/mystery/thriller of all time - The Conjuring.

Top Performing Studio Data For Choosing Studio To Partner with

To choose a studio we want to look at the top performing studios that have been involved with horror movies. The below table shows the top 10 studios sorted by total gross income.

```
In [ ]: df_top_grossing_studios = pd.read_csv('data/top_grossing_studios.csv')
df_top_grossing_studios.index = df_top_grossing_studios[df_top_grossing_studios.
df_top_grossing_studios.drop(df_top_grossing_studios.columns[0], axis = 1, inpla
df_top_grossing_studios.index.names = ['']
df_top_grossing_studios.iloc[1:]
```

Out[ ]:

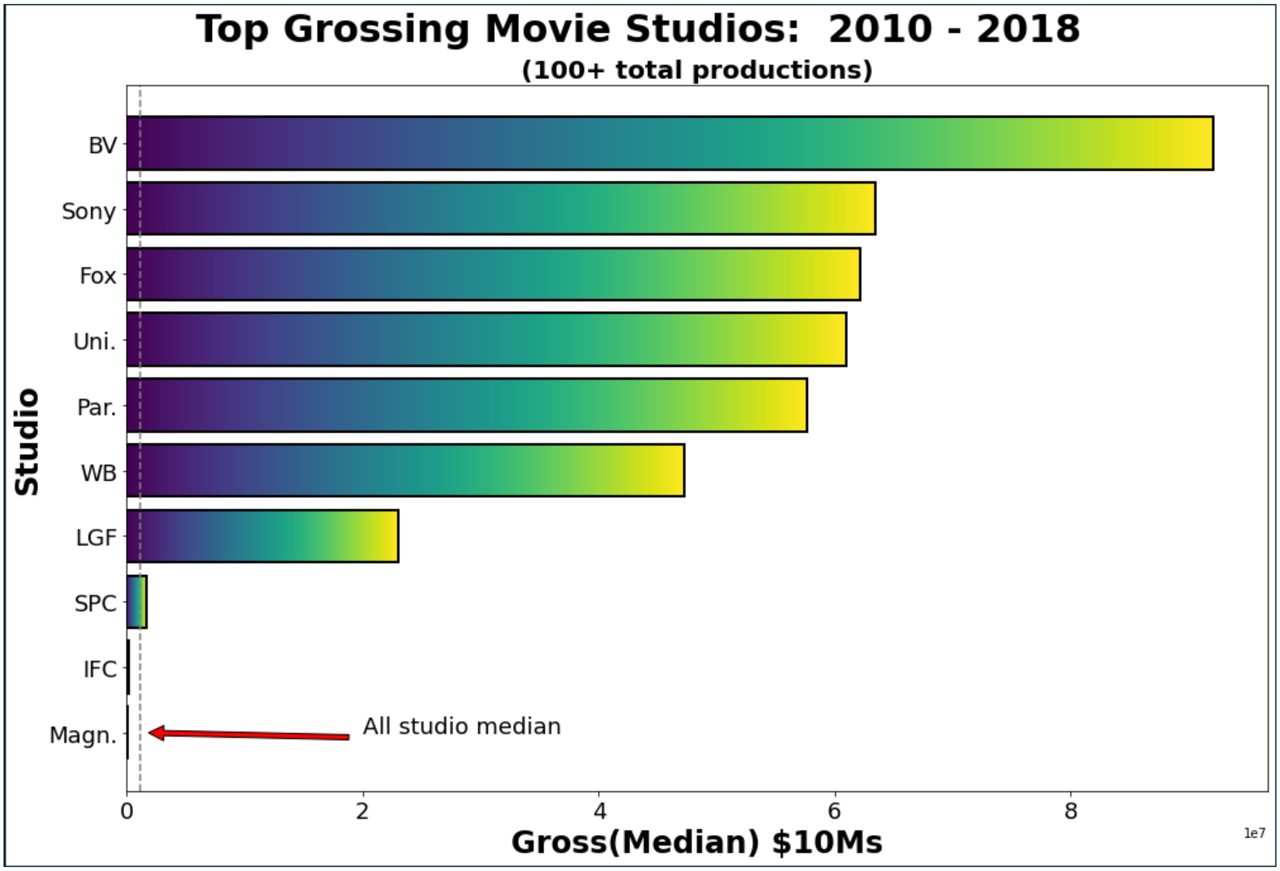
	BV	Sony	Fox	Uni.	Par.	WB
max_gross	936700000	404500000	363100000	652300000	312400000	448100000
min_gross	48500	2500	2800000	22000	6700	139000
total_gross	18419029199	8459683098	10949499997	12902393000	7685870699	12168046000
total_films	106	109	136	147	101	140
mean_gross	173764426	77611771	80511029	87771380	76097729	86914614
median_gross	92100000	63500000	62150000	61000000	57700000	47250000

BV is the top grossing studio and supports Disney. The plot below shows a cleare drop off in terms of total gross income after Warner Brothers.

```
In [ ]: above_100_studios = list(df_top_grossing_studios.keys())
medians = []
for median in df_top_grossing_studios:
    medians.append(int((df_top_grossing_studios[median]['median_gross'])))
median_total_gr = 1100000
```

In [ ]:
```
x2 = above_100_studios
x2.reverse()
y2 = medians
y2.reverse()

fig, top_studios = plt.subplots(figsize = (15,10),linewidth=3, edgecolor='#04253
fig.subplots_adjust(top=0.85)

fig.set_facecolor("white")
fig.suptitle('Top Grossing Movie Studios:  2010 - 2018',fontsize=30,fontweight='
fig.subplots_adjust(right=.96,top=.9)

def gradientbars(bars):
    grad = np.atleast_2d(np.linspace(0,1,256))
    ax = bars[0].axes
    lim = ax.get_xlim()+ax.get_ylim()
    for bar in bars:
        bar.set_zorder(1)
        bar.set_facecolor("none")
        x,y = bar.get_xy()
        w, h = bar.get_width(), bar.get_height()
        ax.imshow(grad, extent=[x,x+w,y,y+h], aspect="auto", zorder=0)
    ax.axis(lim)

top_studios.axvline(median_total_gr, ls='--', color='grey')
top_studios.annotate('All studio median', fontsize=18, xy=(900000,'Magn.'), xyte
plt.xticks(fontsize=18)
top_studios.ticklabel_format(useOffset=True, style='sci')
plt.yticks(fontsize=18)

bar = top_studios.barh(x2,y2, color = '#00750c', linewidth=2, edgecolor="black")

gradientbars(bar)

top_studios.set_title("(100+ total productions)", fontsize=20, fontweight='bold'
top_studios.set_xlabel("Gross(Median) $10Ms", fontsize=24, fontweight='bold')
top_studios.set_ylabel("Studio", fontsize=24,fontweight='bold')
```

Out[ ]: ''



Both Paramount and Warner Bros work with horror/mystery/thrillers. Paramount has released movies such as Paranormal Acitvity series while Warner Bros is responsible for The Conjuring universe. Either studio would be a good choice to partner with.

All studio average median is represented by the vertical dashed line

BV = Buena Vista (Disney)

Uni. = Universal Studios

Par. = Paramount Pictures

WB = Warner Bros. Studios

LGF = Lionsgate Entertainment Corp.

SPC = Sony Pictures Classics

IFC = Independent Film Channel (owned by AMC)

Magn. = Magnolia Pictures

The Oscars

Looking at the IMDB title basics and the oscars awards dataframe

In [ ]:
```
Titlebasics = pd.read_csv('data/title.basics.csv')
oscars = pd.read_csv('data/the_oscar_award.csv')
```

In [ ]:
```
Titlebasics
```

Out[ ]:

	tconst	primary_title	original_title	start_year	runtime_minutes	genres
0	tt0063540	Sunghursh	Sunghursh	2013	175.0	Action,Crime,Drama

	tconst	primary_title	original_title	start_year	runtime_minutes	genres
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
...	...	...	...	...	...	...
146139	tt9916538	Kuambil Lagi Hatiku	Kuambil Lagi Hatiku	2019	123.0	Drama
146140	tt9916622	Rodolpho Teóphilo - O Legado de um Pioneiro	Rodolpho Teóphilo - O Legado de um Pioneiro	2015	NaN	Documentary
146141	tt9916706	Dankyavar Danka	Dankyavar Danka	2013	NaN	Comedy
146142	tt9916730	6 Gunn	6 Gunn	2017	116.0	NaN
146143	tt9916754	Chico Albuquerque - Revelações	Chico Albuquerque - Revelações	2013	NaN	Documentary

146144 rows × 6 columns

In [ ]:

```
oscars
```

Out[ ]:

	year_film	year_ceremony	ceremony	category	name	film	winner
0	1927	1928	1	ACTOR	Richard Barthelmess	The Noose	False
1	1927	1928	1	ACTOR	Emil Jannings	The Last Command	True
2	1927	1928	1	ACTRESS	Louise Dresser	A Ship Comes In	False
3	1927	1928	1	ACTRESS	Janet Gaynor	7th Heaven	True
4	1927	1928	1	ACTRESS	Gloria Swanson	Sadie Thompson	False
...	...	...	...	...	...	...	...
10390	2019	2020	92	WRITING (Original Screenplay)	Screenplay by Bong Joon Ho, Han Jin Won; Story...	Parasite	True

	year_film	year_ceremony	ceremony	category	name	film	winner
10391	2019	2020	92	JEAN HERSHOLT HUMANITARIAN AWARD	Geena Davis	NaN	True
10392	2019	2020	92	HONORARY AWARD	David Lynch	NaN	True
10393	2019	2020	92	HONORARY AWARD	Wes Studi	NaN	True
10394	2019	2020	92	HONORARY AWARD	Lina Wertmüller	NaN	True

10395 rows × 7 columns

Combining the title basics and oscar awards dataframe

In order to relate the oscar winnings/nominations to a genre, we'll have to merge the dataframes. Upon analyzing the two, we can relate the data by movie title using the "primary_title" and "start_year" columns in the Titlebasics dataframe to the "film" and "year_film" columns in the oscars dataframe

In [ ]:

```
Oscar_w_genre = oscars.merge(Titlebasics, how="left", left_on=['film','year_film
Oscar_w_genre
```

Out[ ]:

	year_film	year_ceremony	ceremony	category	name	film	winner	t
0	1927	1928	1	ACTOR	Richard Barthelmess	The Noose	False	
1	1927	1928	1	ACTOR	Emil Jannings	The Last Command	True	
2	1927	1928	1	ACTRESS	Louise Dresser	A Ship Comes In	False	
3	1927	1928	1	ACTRESS	Janet Gaynor	7th Heaven	True	
4	1927	1928	1	ACTRESS	Gloria Swanson	Sadie Thompson	False	
...	...	...	...	...	...	...	...	
10424	2019	2020	92	WRITING (Original Screenplay)	Screenplay by Bong Joon Ho, Han Jin Won; Story...	Parasite	True	tt67
10425	2019	2020	92	JEAN HERSHOLT HUMANITARIAN AWARD	Geena Davis	NaN	True	
10426	2019	2020	92	HONORARY AWARD	David Lynch	NaN	True	

	year_film	year_ceremony	ceremony	category	name	film	winner	t
10427	2019	2020	92	HONORARY AWARD	Wes Studi	NaN	True	
10428	2019	2020	92	HONORARY AWARD	Lina Wertmüller	NaN	True	

10429 rows × 13 columns

Getting only the data we want

We'll be analyzing the oscar awards by genre going from 2010 onward. We'll set up a new data frame that will only contain movies from the past 11 years and remove any NA values from the "genres" column.

In [ ]:

```
Oscar_since2010 = Oscar_w_genre[(Oscar_w_genre['year_film'] >= 2010)].sort_value
```

In [ ]:

```
Oscar_since2010.dropna(subset = ['genres'], inplace=True)
Oscar_since2010
```

Out[ ]:

	year_film	year_ceremony	ceremony	category	name	film	winner	t
9142	2010	2011	83	ACTOR IN A LEADING ROLE	Javier Bardem	Biutiful	False	tt116
9229	2010	2011	83	BEST PICTURE	Anne Rosellini and Alix Madigan-Yorkin, Producers	Winter's Bone	False	tt139
9228	2010	2011	83	BEST PICTURE	Scott Rudin, Ethan Coen and Joel Coen, Producers	True Grit	False	tt140
9227	2010	2011	83	BEST PICTURE	Darla K. Anderson, Producer	Toy Story 3	False	tt04:
9226	2010	2011	83	BEST PICTURE	Scott Rudin, Dana Brunetti, Michael De Luca an...	The Social Network	False	tt12ɛ
...	...	...	...	...	...	...	...	
10334	2019	2020	92	DIRECTING	Bong Joon Ho	Parasite	True	tt1037
10332	2019	2020	92	DIRECTING	Sam Mendes	1917	False	tt857
10331	2019	2020	92	DIRECTING	Todd Phillips	Joker	False	tt728

	year_film	year_ceremony	ceremony	category	name	film	winner	t
10328	2019	2020	92	COSTUME DESIGN	Jacqueline Durran	Little Women	True	tt328
10343	2019	2020	92	DOCUMENTARY (Short Subject)	John Haptas and Kristine Samuelson	Life Overtakes Me	False	tt920

1025 rows × 13 columns

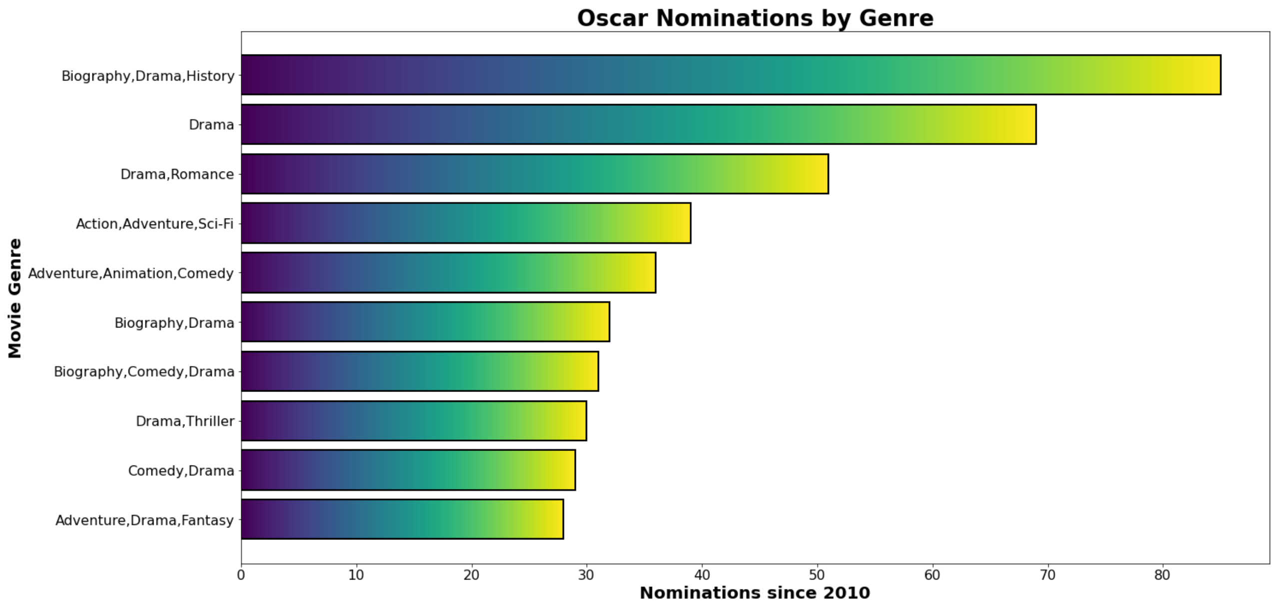
Count up the genres and plot

Now we'll analyze how many total oscars nominations per genre there were since 2010 and plot them.

In [ ]:
```
Oscar_since2010C = Oscar_since2010['genres'].value_counts()[:10]
Oscar_since2010C
```

Out[ ]:
```
Biography,Drama,History        85
Drama                          69
Drama,Romance                  51
Action,Adventure,Sci-Fi        39
Adventure,Animation,Comedy     36
Biography,Drama                32
Biography,Comedy,Drama         31
Drama,Thriller                 30
Comedy,Drama                   29
Adventure,Drama,Fantasy        28
Name: genres, dtype: int64
```

In [ ]:
```
fig, axes = plt.subplots()

bar = axes.barh(y=Oscar_since2010C.index[::-1],width=list(Oscar_since2010C)[::-1

gradientbars(bar)
fig.set_figheight(10)
fig.set_figwidth(20)
fig.tight_layout(pad=4)
axes.tick_params(axis='both',which='major',labelsize=16)
axes.set_title('Oscar Nominations by Genre',fontsize=26,fontweight='bold')
axes.set_xlabel('Nominations since 2010',fontsize=20,fontweight='bold')
axes.set_ylabel('Movie Genre',fontsize=20,fontweight='bold')

plt.show()
```



In [ ]:

Oscar Wins

We can also analyze the number of wins by genre since 2010.

In [ ]:

```
Oscar_Wins = Oscar_since2010[(Oscar_w_genre['winner'] == True)].sort_values(by='
Oscar_Wins
```

```
<ipython-input-20-64824dc55e63>:1: UserWarning: Boolean Series key will be reind
exed to match DataFrame index.
  Oscar_Wins = Oscar_since2010[(Oscar_w_genre['winner'] == True)].sort_values(by
='year_film', ascending=True)
```

Out[ ]:

	year_film	year_ceremony	ceremony	category	name	film	winner	
9224	2010	2011	83	BEST PICTURE	Iain Canning, Emile Sherman and Gareth Unwin, ...	The King's Speech	True	tt15
9176	2010	2011	83	COSTUME DESIGN	Colleen Atwood	Alice in Wonderland	True	tt1(
9177	2010	2011	83	COSTUME DESIGN	Colleen Atwood	Alice in Wonderland	True	tt2(
9185	2010	2011	83	DIRECTING	Tom Hooper	The King's Speech	True	tt15
9190	2010	2011	83	DOCUMENTARY (Feature)	Charles Ferguson and Audrey Marrs	Inside Job	True	tt16
...	...	...	...	...	...	...	...	

	year_film	year_ceremony	ceremony	category	name	film	winner	
10382	2019	2020	92	BEST PICTURE	Kwak Sin Ae and Bong Joon Ho, Producers	Parasite	True	tt6
10381	2019	2020	92	BEST PICTURE	Kwak Sin Ae and Bong Joon Ho, Producers	Parasite	True	tt10
10334	2019	2020	92	DIRECTING	Bong Joon Ho	Parasite	True	tt10
10407	2019	2020	92	SOUND MIXING	Mark Taylor and Stuart Wilson	1917	True	tt8
10328	2019	2020	92	COSTUME DESIGN	Jacqueline Durran	Little Women	True	tt3

220 rows × 13 columns

In [ ]:

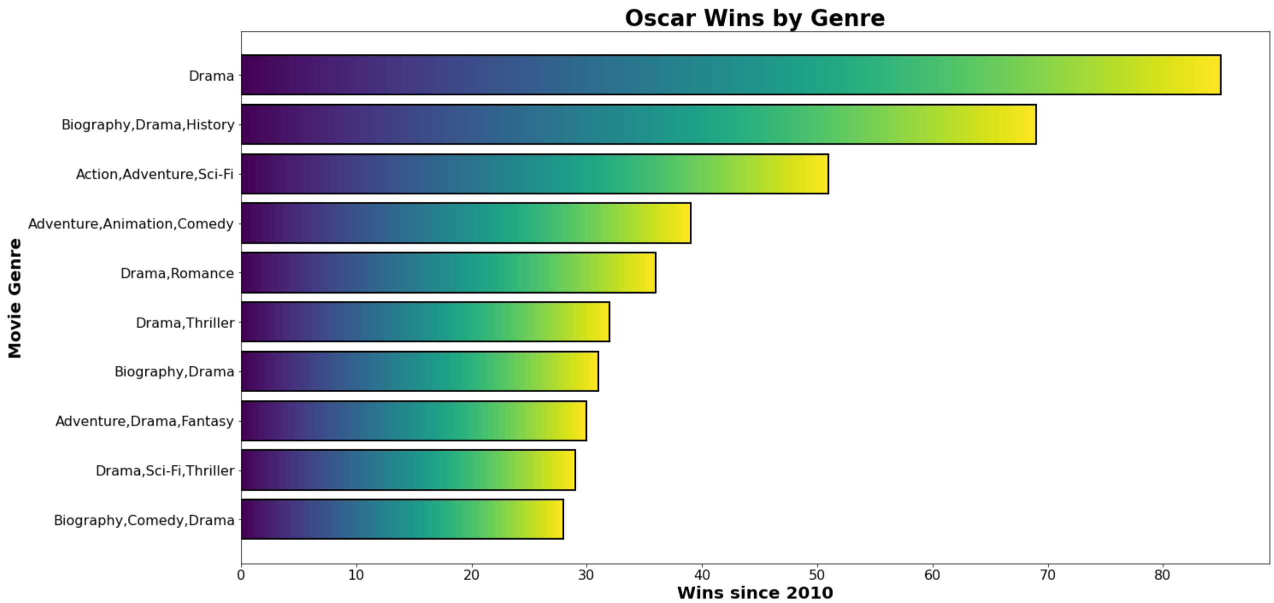
```
Oscar_WinsC = Oscar_Wins['genres'].value_counts()[:10]
Oscar_WinsC
```

Out[ ]:

```
Drama                          17
Biography,Drama,History        17
Action,Adventure,Sci-Fi        13
Adventure,Animation,Comedy     11
Drama,Romance                   9
Drama,Thriller                  9
Biography,Drama                 9
Adventure,Drama,Fantasy         8
Drama,Sci-Fi,Thriller           7
Biography,Comedy,Drama          6
Name: genres, dtype: int64
```

In [ ]:

```
fig, axes = plt.subplots()

bar = axes.barh(y=Oscar_WinsC.index[::-1],width=list(Oscar_since2010C)[::-1],lin

gradientbars(bar)
fig.set_figheight(10)
fig.set_figwidth(20)
fig.tight_layout(pad=4)
axes.tick_params(axis='both',which='major',labelsize=16)
axes.set_title('Oscar Wins by Genre',fontsize=26,fontweight='bold')
axes.set_xlabel('Wins since 2010',fontsize=20,fontweight='bold')
axes.set_ylabel('Movie Genre',fontsize=20,fontweight='bold')

plt.show()
```



Conlusions

- **Horror/Mystery/Thriller genre should be the focus for the first movies from the new Microsoft Studio.** These types of movies show the highest profitability and low required budget for success.
- **James Wan should be looked at as director of first movies** as he has been apart of 50% of the top grossing horror/mystery/thriller movies of the past two decades.
- **Paramount or Warner Bros. would be the best studios to partner with** as they both have experience in releasing successful horror/mystery/thrillers

Next Steps

Recommendations with regards to the Oscar Academy Awards

As a movie studio, you may decide to join the glorious pursuit of the ever covetted Oscar academy award. When you're ready to throw your hat into the ring, we highly recommend switching your main genre focus from Horror to Drama based on the wins and nominations since 2010.

Citations

https://mowe.studio/how-much-does-animation-cost-vs-live-action/

https://www.imdb.com/interfaces/

https://help.imdb.com/article/imdbpro/industry-research/box-office-mojo-by-imdbpro-faq/GCWTV4MQKGWRAUAP?ref_=mojo_ftr_help#