

```
In [3]: from google.colab import drive
drive.mount('/content/drive')
```

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

```
In [4]: import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
```

```
In [5]: entertainer_data = pd.read_excel("/content/drive/MyDrive/Entertainer Data (1)/Ente
entertainer_data
```

```
Out[5]:
```

	Entertainer	Gender (traditional)	Birth Year
0	Adele	F	1988
1	Angelina Jolie	F	1975
2	Aretha Franklin	F	1942
3	Bette Davis	F	1908
4	Betty White	F	1922
...
65	Tom Hanks	M	1956
66	Tony Bennett	M	1926
67	Wayne Newton	M	1942
68	Will Smith	M	1968
69	Willie Nelson	M	1933

70 rows × 3 columns

```
In [8]: entertainer_data.shape
```

```
Out[8]: (70, 3)
```

```
In [6]: entertainer_data2 = pd.read_excel("/content/drive/MyDrive/Entertainer Data (1)/Ente
entertainer_data2
```

Out[6]:

	Entertainer	Year of Breakthrough/#1 Hit/Award Nomination	Breakthrough Name	Year of First Oscar/Grammy/Emmy
0	Adele	2008	19	2009.0
1	Angelina Jolie	1999	Girl, Interrupted	1999.0
2	Aretha Franklin	1967	I Never Loved a Man (The Way I Love You)	1968.0
3	Bette Davis	1934	Of Human Bondage	1935.0
4	Betty White	1952	Life with Elizabeth	1976.0
...
65	Tom Hanks	1984	Splash	1993.0
66	Tony Bennett	1951	Because of You	1963.0
67	Wayne Newton	1972	Daddy, Don't You Walk So Fast	NaN
68	Will Smith	1990	The Fresh Prince of Bel-Air	1988.0
69	Willie Nelson	1975	Red Headed Stranger	1976.0

70 rows × 4 columns

```
In [7]: entertainer_data3 = pd.read_excel("/content/drive/MyDrive/Entertainer Data (1)/Ente
entertainer_data3
```

Out[7]:

	Entertainer	Year of Last Major Work (arguable)	Year of Death
0	Adele	2016	NaN
1	Angelina Jolie	2016	NaN
2	Aretha Franklin	2014	NaN
3	Bette Davis	1989	1989.0
4	Betty White	2016	NaN
...
65	Tom Hanks	2016	NaN
66	Tony Bennett	2016	NaN
67	Wayne Newton	2016	NaN
68	Will Smith	2016	NaN
69	Willie Nelson	2016	NaN

70 rows × 3 columns

```
In [9]: entertainer_data2.shape
```

```
Out[9]: (70, 4)
```

```
In [10]: entertainer_data3.shape
```

```
Out[10]: (70, 3)
```

```
In [11]: #joining three dataset
merged_data = pd.concat([entertainer_data, entertainer_data2, entertainer_data3], axis=1)
merged_data.reset_index(drop=True, inplace=True)
merged_data
```

```
Out[11]:
```

	Entertainer	Gender (traditional)	Birth Year	Entertainer	Year of Breakthrough/#1 Hit/Award Nomination	Breakthrough Name	Year of Oscar/Grammy/E
0	Adele	F	1988	Adele	2008		2
1	Angelina Jolie	F	1975	Angelina Jolie	1999	Girl, Interrupted	1
2	Aretha Franklin	F	1942	Aretha Franklin	1967	I Never Loved a Man (The Way I Love You)	1
3	Bette Davis	F	1908	Bette Davis	1934	Of Human Bondage	1
4	Betty White	F	1922	Betty White	1952	Life with Elizabeth	1
...
65	Tom Hanks	M	1956	Tom Hanks	1984	Splash	1
66	Tony Bennett	M	1926	Tony Bennett	1951	Because of You	1
67	Wayne Newton	M	1942	Wayne Newton	1972	Daddy, Don't You Walk So Fast	
68	Will Smith	M	1968	Will Smith	1990	The Fresh Prince of Bel- Air	1
69	Willie Nelson	M	1933	Willie Nelson	1975	Red Headed Stranger	1

70 rows × 10 columns

```
In [15]: print(merged_data['Entertainer'])
```

	Entertainer	Entertainer	Entertainer
0	Adele	Adele	Adele
1	Angelina Jolie	Angelina Jolie	Angelina Jolie
2	Aretha Franklin	Aretha Franklin	Aretha Franklin
3	Bette Davis	Bette Davis	Bette Davis
4	Betty White	Betty White	Betty White
..
65	Tom Hanks	Tom Hanks	Tom Hanks
66	Tony Bennett	Tony Bennett	Tony Bennett
67	Wayne Newton	Wayne Newton	Wayne Newton
68	Will Smith	Will Smith	Will Smith
69	Willie Nelson	Willie Nelson	Willie Nelson

[70 rows x 3 columns]

```
In [59]: #remove duplicate column 'entertainer'
new_data = merged_data.loc[:,~merged_data.columns.duplicated()]
new_data
```

Out[59]:

	Entertainer	Gender (traditional)	Birth Year	Year of Breakthrough/#1 Hit/Award Nomination	Breakthrough Name	Year of First Oscar/Grammy/Emmy	Year of Movie V (argua
0	Adele	F	1988	2008	19	2009.0	
1	Angelina Jolie	F	1975	1999	Girl, Interrupted	1999.0	
2	Aretha Franklin	F	1942	1967	I Never Loved a Man (The Way I Love You)	1968.0	
3	Bette Davis	F	1908	1934	Of Human Bondage	1935.0	
4	Betty White	F	1922	1952	Life with Elizabeth	1976.0	
...	
65	Tom Hanks	M	1956	1984	Splash	1993.0	
66	Tony Bennett	M	1926	1951	Because of You	1963.0	
67	Wayne Newton	M	1942	1972	Daddy, Don't You Walk So Fast	NaN	
68	Will Smith	M	1968	1990	The Fresh Prince of Bel- Air	1988.0	
69	Willie Nelson	M	1933	1975	Red Headed Stranger	1976.0	

70 rows x 8 columns

```
In [47]: #Checking Columns
col = list(new_data)
print(col)
```

```
['Entertainer', 'Gender (traditional)', 'Birth Year', 'Year of Breakthrough/#1 Hit/Award Nomination', 'Breakthrough Name', 'Year of First Oscar/Grammy/Emmy', 'Year of Last Major Work (arguable)', 'Year of Death']
```

```
In [18]: #Checking Null Values
new_data.isnull().sum()
```

```
Out[18]:
```

	0
Entertainer	0
Gender (traditional)	0
Birth Year	0
Year of Breakthrough/#1 Hit/Award Nomination	0
Breakthrough Name	0
Year of First Oscar/Grammy/Emmy	6
Year of Last Major Work (arguable)	0
Year of Death	40

dtype: int64

```
In [19]: new_data.shape
```

```
Out[19]: (70, 8)
```

```
In [20]: import seaborn as sns
import matplotlib.pyplot as plt
```

```
In [64]: #Renaming Columns

new_data = new_data.rename(columns={'Gender (traditional)': 'Gender', 'Year of First
new_data
```

Out[64]:

	Entertainer	Gender	Birth Year	Year of Breakthrough/#1 Hit/Award Nomination	Breakthrough Name	Year of First Award	Year of Last Major Work (arguable)	Year of Death
0	Adele	F	1988	2008	19	2009.0	2016	NaN
1	Angelina Jolie	F	1975	1999	Girl, Interrupted	1999.0	2016	NaN
2	Aretha Franklin	F	1942	1967	I Never Loved a Man (The Way I Love You)	1968.0	2014	NaN
3	Bette Davis	F	1908	1934	Of Human Bondage	1935.0	1989	1989.0
4	Betty White	F	1922	1952	Life with Elizabeth	1976.0	2016	NaN
...
65	Tom Hanks	M	1956	1984	Splash	1993.0	2016	NaN
66	Tony Bennett	M	1926	1951	Because of You	1963.0	2016	NaN
67	Wayne Newton	M	1942	1972	Daddy, Don't You Walk So Fast	NaN	2016	NaN
68	Will Smith	M	1968	1990	The Fresh Prince of Bel-Air	1988.0	2016	NaN
69	Willie Nelson	M	1933	1975	Red Headed Stranger	1976.0	2016	NaN

70 rows × 8 columns

In [25]: *#count of male and female*

```
gender_counts = new_data['Gender'].value_counts()

print(f"Number of males: {gender_counts['M']}")
print(f"Number of females: {gender_counts['F']}")
```

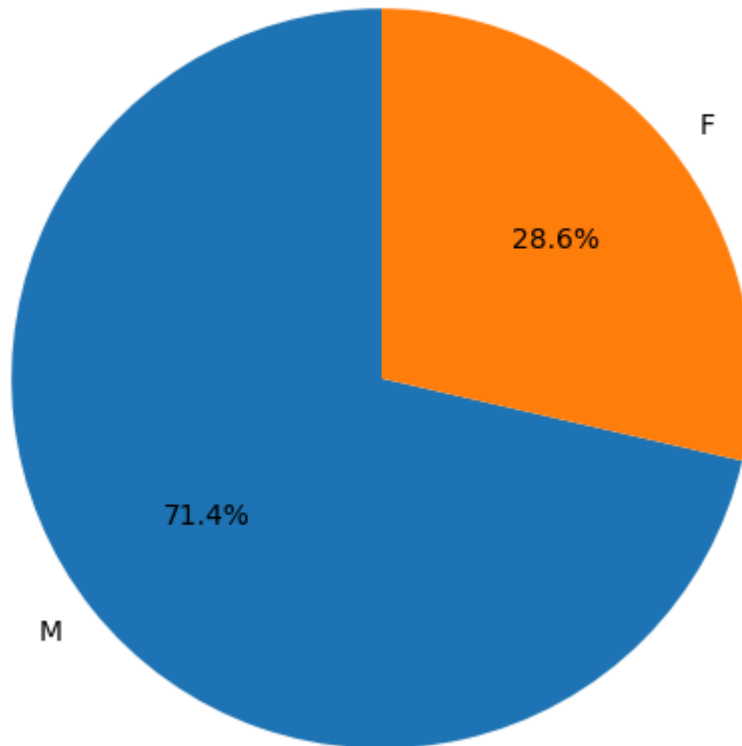
Number of males: 50
 Number of females: 20

In [26]: *#Gender Distribuion representation using pie chart*

```
gender_dist= new_data['Gender'].value_counts()

# Creating the pie chart
plt.figure(figsize=(6,6))
plt.pie(gender_dist, labels=gender_dist.index, autopct='%1.1f%%', startangle=90)
plt.title('Gender Distribution')
plt.show()
```

Gender Distribution



```
In [65]: #Average age of entertainers to receive oscar/grammy or emmy

new_data['Age at first award'] = new_data['Year of First Award'] - new_data['Birth
average_age_by_gender = new_data.groupby('Gender')[['Age at first award']].mean()

x= average_age_by_gender.index
y= average_age_by_gender.values.flatten().astype(int)

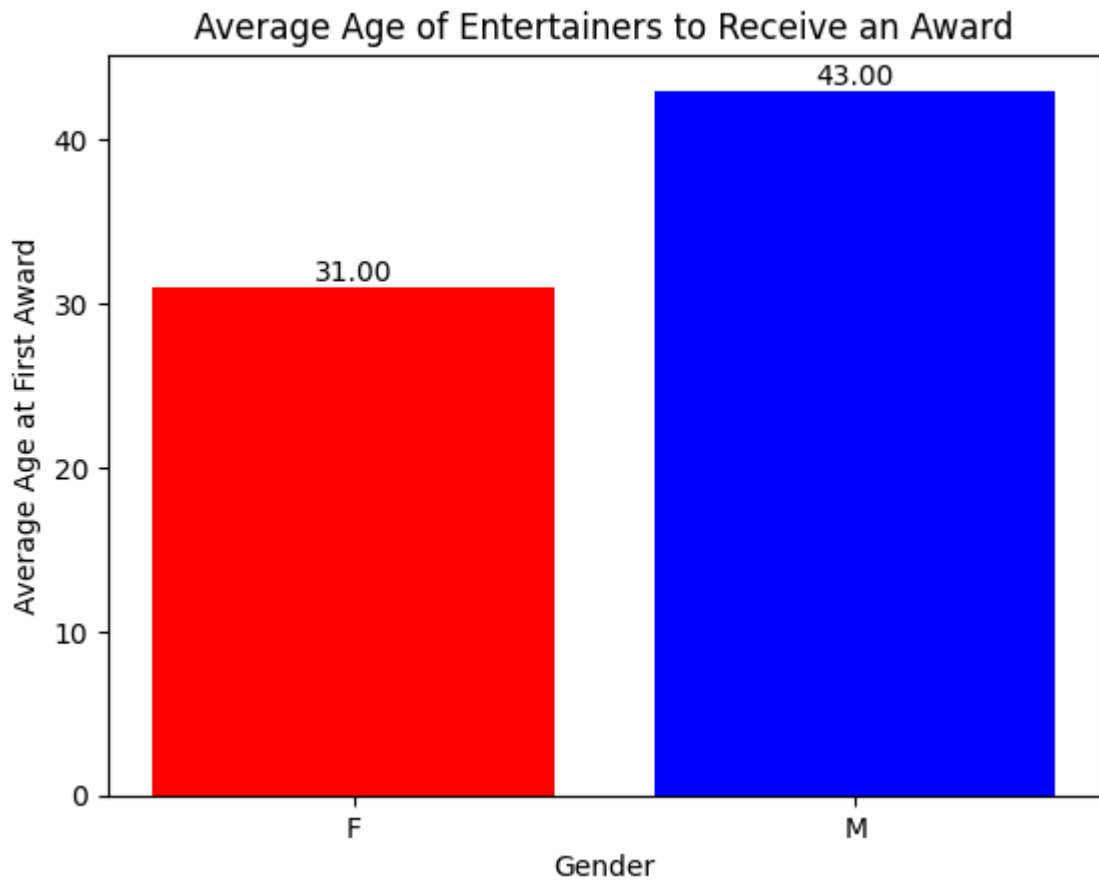
def add_value_label(x,y):
    for i in range (len(x)):
        plt.text(i, y[i], f"{y[i]:.2f}", ha='center', va='bottom')

plt.bar(x, y, color = ['red' , 'blue'])

add_value_label(x, y)

plt.title('Average Age of Entertainers to Receive an Award')
plt.xlabel('Gender')
plt.ylabel('Average Age at First Award')

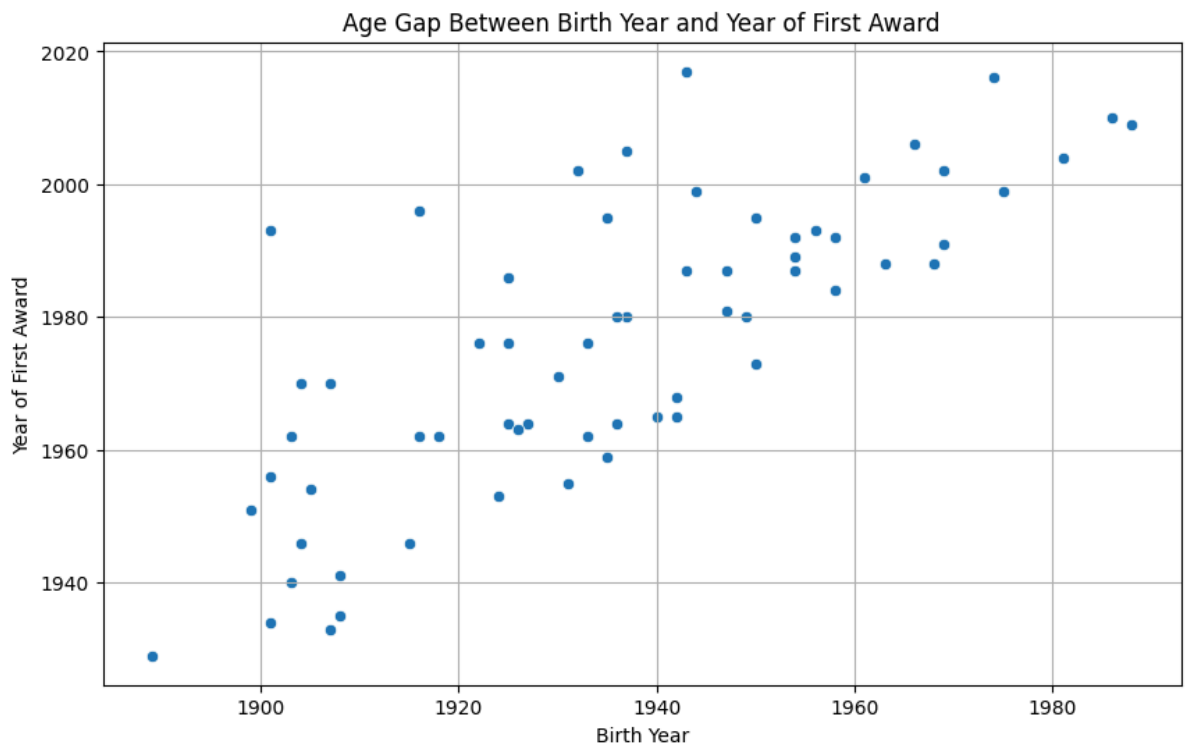
plt.show()
```



In [28]: *#Age gap between Birth Year and Year of First Award using scatter plot*

```
new_data = new_data[['Year of First Award', 'Birth Year']]

plt.figure(figsize=(10, 6))
sns.scatterplot(x='Birth Year', y='Year of First Award', data=new_data)
plt.xlabel('Birth Year')
plt.ylabel('Year of First Award')
plt.title('Age Gap Between Birth Year and Year of First Award')
plt.grid(True)
plt.show()
```

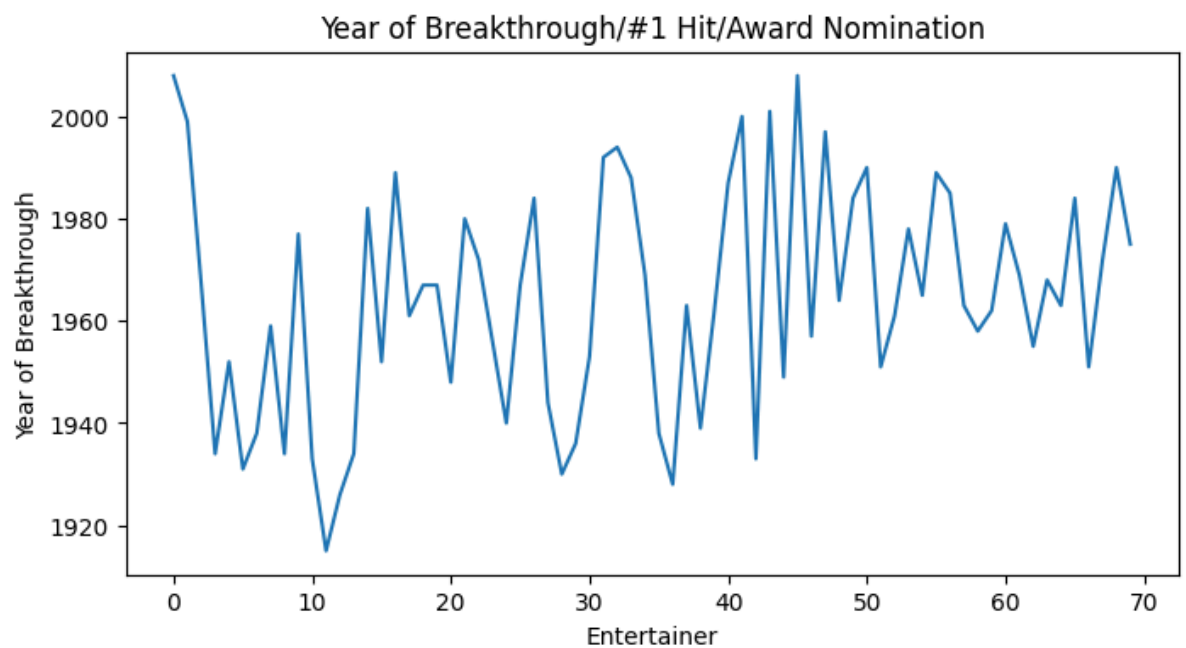



In [66]: *#Finding age at first award*

```
new_data['Age at First Award'] = new_data['Year of First Award'] - new_data['Birth Year']
avg_age = new_data.groupby('Gender')[['Year of First Award']].mean()
labels = new_data.index.tolist()
```

In [34]: **from** matplotlib **import** pyplot **as** plt

```
entertainer_data2['Year of Breakthrough/#1 Hit/Award Nomination'].plot(kind='line',
plt.gca().spines[['top', 'right']].set_visible(True)
plt.xlabel('Entertainer')
plt.ylabel('Year of Breakthrough')
plt.show()
```



In [36]: new_data.head()

Out[36]:

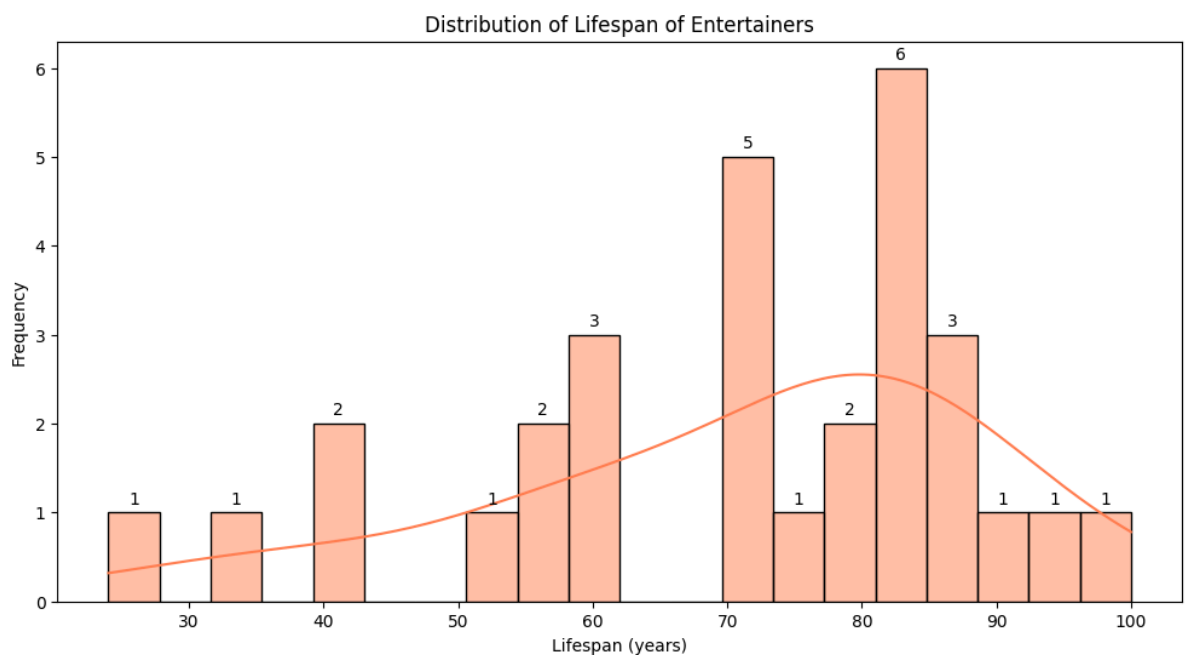
	Year of First Award	Birth Year
0	2009.0	1988
1	1999.0	1975
2	1968.0	1942
3	1935.0	1908
4	1976.0	1922

In [55]: *#LIFESPAN OF ENTERTAINERS*

```
ent_life = new_data['Year of Death'] - new_data['Birth Year']
ent_life

plt.figure(figsize=(12, 6))
ax = sns.histplot(ent_life.dropna(), bins=20, kde=True, color='coral')
plt.title('Distribution of Lifespan of Entertainers')
plt.xlabel('Lifespan (years)')
plt.ylabel('Frequency')

for p in ax.patches:
    height = p.get_height()
    if height > 0:
        ax.text(p.get_x() + p.get_width() / 2., height + 0.1, '{:.0f}'.format(height))
plt.show()
```



In []: `new_data.info()`

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 70 entries, 0 to 69
```

```
Data columns (total 9 columns):
```

#	Column	Non-Null Count	Dtype
0	Entertainer	70 non-null	object
1	Gender	70 non-null	object
2	Birth Year	70 non-null	int64
3	Year of Breakthrough/#1 Hit/Award Nomination	70 non-null	int64
4	Breakthrough Name	70 non-null	object
5	Year of First Oscar/Grammy/Emmy	64 non-null	float64
6	Year of Last Major Work (arguable)	70 non-null	int64
7	Year of Death	30 non-null	float64
8	Age at first award	64 non-null	float64

```
dtypes: float64(3), int64(3), object(3)
```

```
memory usage: 5.0+ KB
```

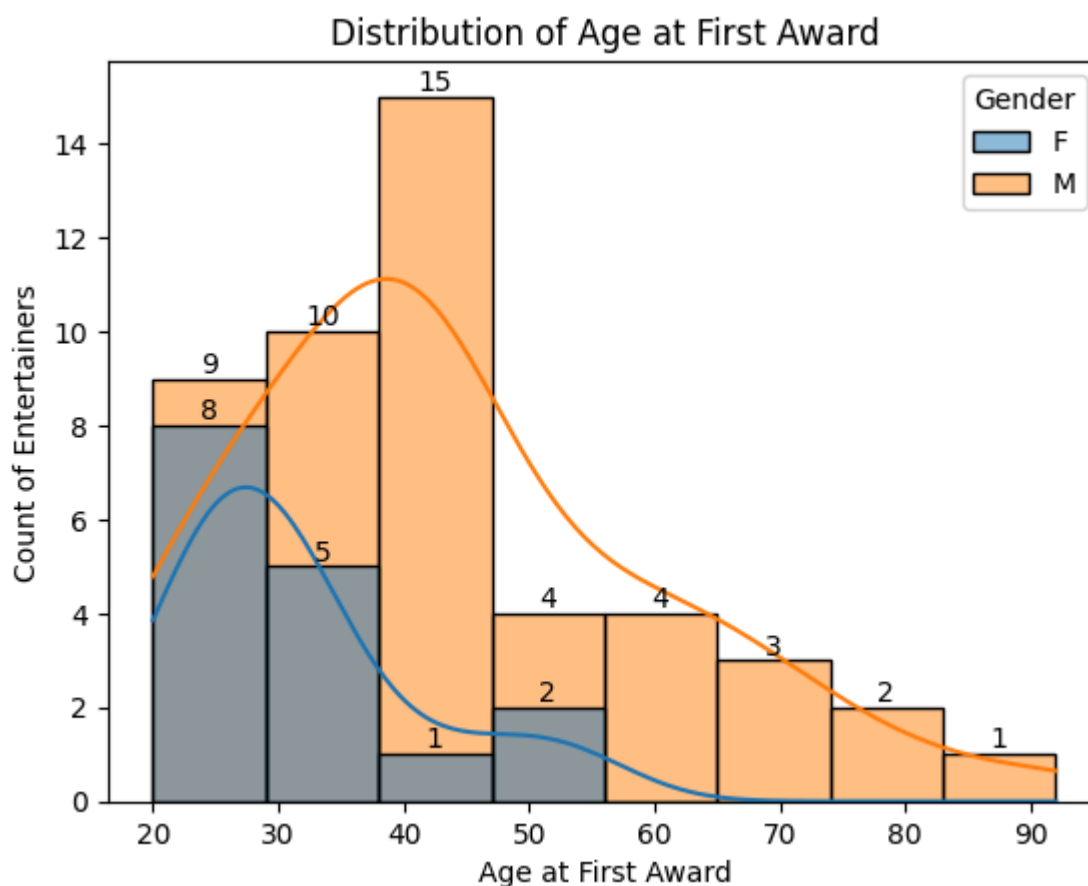
```
In [67]: # Creating a distplot for 'Age at First Award'
```

```
ax = sns.histplot(data=new_data, x='Age at First Award', hue='Gender', kde=True, e

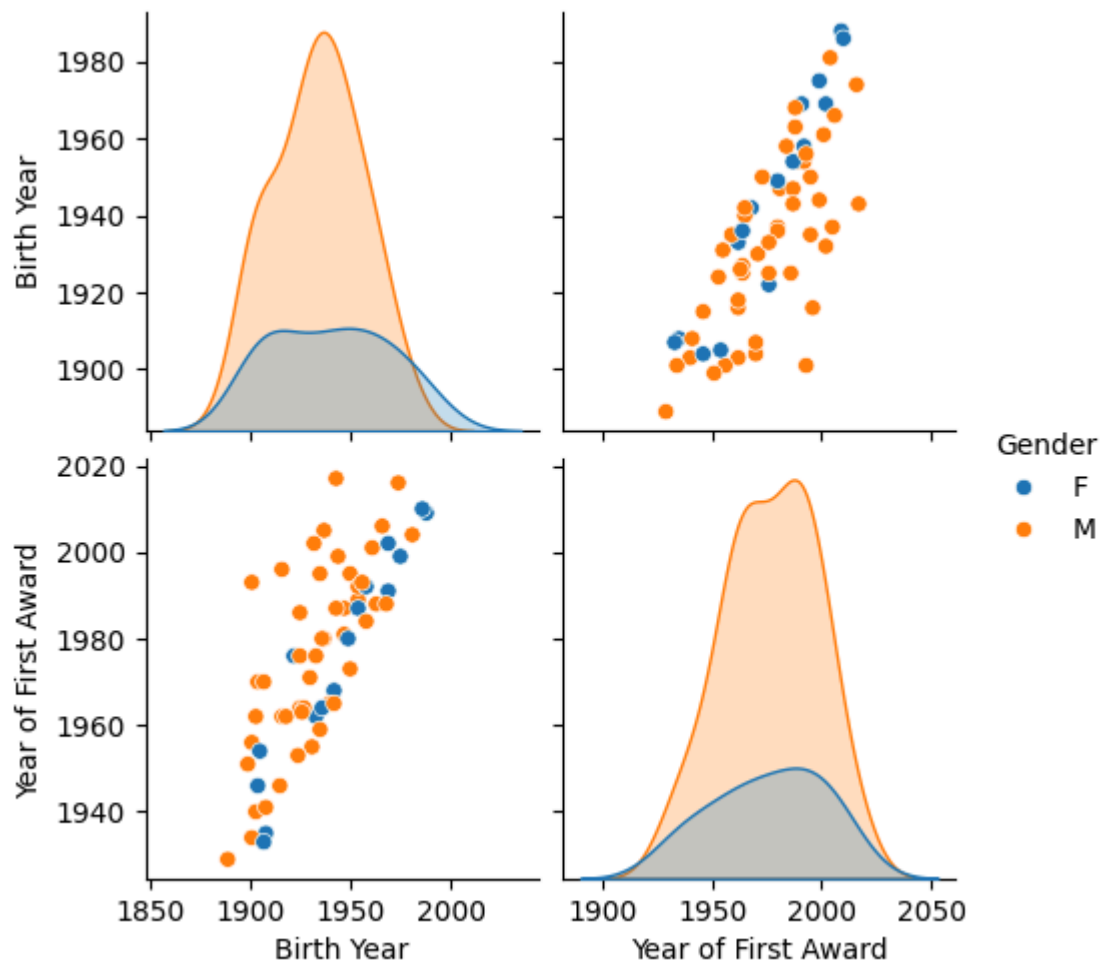
plt.title('Distribution of Age at First Award')
plt.xlabel('Age at First Award')
plt.ylabel('Count of Entertainers')

for p in ax.patches:
    height = p.get_height()
    if height > 0:
        ax.annotate(f'{height:.0f}',
                    (p.get_x() + p.get_width() / 2., height),
                    ha='center',
                    va='bottom',
                    fontsize=10,
                    color='black')

plt.show()
```



```
In [51]: sns.pairplot(new_data, hue='Gender', vars=['Birth Year', 'Year of First Award'], di
plt.show()
```



This analysis has provided valuable insights into the careers and demographics of a group of entertainers. We observed the distribution of genders, calculated the average age at which entertainers receive their first major award, and explored the relationship between birth year, year of breakthrough, and year of first award.

Notably, we found that male entertainers tend to receive their first major award slightly earlier than their female counterparts. The distribution of lifespans provided a glimpse into the longevity of careers in the entertainment industry.

Further research could delve deeper into the factors contributing to the observed trends, such as genre, type of entertainment, and socio-cultural influences. This could help us better understand the dynamics of success and longevity in the entertainment world.

```
In [73]: !jupyter nbconvert --to html - /ENTERTAINER_DATA.ipynb
```

```
[NbConvertApp] WARNING | pattern '-' matched no files
[NbConvertApp] Converting notebook /ENTERTAINER_DATA.ipynb to html
[NbConvertApp] Writing 1017113 bytes to /ENTERTAINER_DATA.html
```