# Predicting IMDb Scores

To load and preprocess a **IMDb**(Internet Movie Database) dataset for analysis, you can follow these general steps using Python and Pandas. Make sure you have a IMDb dataset in a suitable format available.

**1.Import Libraries :**

Start by importing the necessary Python libraries, including Pandas, to load and preprocess the dataset.

**2. Load the IMDb Dataset :**

Load the **IMDb** dataset into a Pandas DataFrame. You can use **pd.read_csv()** for CSV files, but the method may vary depending on the file format.

In [1]:
```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt
```

In [2]:
```
na_values = ["\\N", "nan"]
df=pd.read_csv("/kaggle/input/imdb-dataset/name.basics.tsv/data.tsv",sep='\t',low_memo
ry=False, na_values=na_values)
```

In [3]:
```
df=df.dropna()
```

In [4]:
```
df.head()
```

Out[4]:

| | nconst | primaryName | birthYear | deathYear | primaryProfession | knownForTitles |
|---|---|---|---|---|---|---|
| 0 | nm0000001 | Fred Astaire | 1899.0 | 1987.0 | soundtrack,actor,miscellaneous | tt0053137,tt0072308,tt0050419,tt0031 |
| 1 | nm0000002 | Lauren Bacall | 1924.0 | 2014.0 | actress,soundtrack | tt0075213,tt0037382,tt0038355,tt0117 |
| 3 | nm0000004 | John Belushi | 1949.0 | 1982.0 | actor,soundtrack,writer | tt0078723,tt0080455,tt0077975,tt0072 |
| 4 | nm0000005 | Ingmar Bergman | 1918.0 | 2007.0 | writer,director,actor | tt0069467,tt0050976,tt0083922,tt0050 |
| 5 | nm0000006 | Ingrid Bergman | 1915.0 | 1982.0 | actress,soundtrack,producer | tt0038109,tt0036855,tt0034583,tt0038 |

In [5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 179159 entries, 0 to 12920210
Data columns (total 6 columns):
 #   Column             Non-Null Count   Dtype  
---  ------             --------------   -----  
 0   nconst             179159 non-null  object 
 1   primaryName        179159 non-null  object 
 2   birthYear          179159 non-null  float64
 3   deathYear          179159 non-null  float64
 4   primaryProfession  179159 non-null  object 
 5   knownForTitles     179159 non-null  object 
dtypes: float64(2), object(4)
memory usage: 9.6+ MB
```

In [6]:
```
df.head()
```

Out[6]:

| | nconst | primaryName | birthYear | deathYear | primaryProfession | knownForTitles |
|---|---|---|---|---|---|---|
| 0 | nm0000001 | Fred Astaire | 1899.0 | 1987.0 | soundtrack,actor,miscellaneous | tt0053137,tt0072308,tt0050419,tt0031 |
| 1 | nm0000002 | Lauren Bacall | 1924.0 | 2014.0 | actress,soundtrack | tt0075213,tt0037382,tt0038355,tt0117 |
| 3 | nm0000004 | John Belushi | 1949.0 | 1982.0 | actor,soundtrack,writer | tt0078723,tt0080455,tt0077975,tt0072 |
| 4 | nm0000005 | Ingmar Bergman | 1918.0 | 2007.0 | writer,director,actor | tt0069467,tt0050976,tt0083922,tt0050 |
| 5 | nm0000006 | Ingrid Bergman | 1915.0 | 1982.0 | actress,soundtrack,producer | tt0038109,tt0036855,tt0034583,tt0038 |

In [7]:
```
df["birthYear"]=df["birthYear"].astype("int32")
```

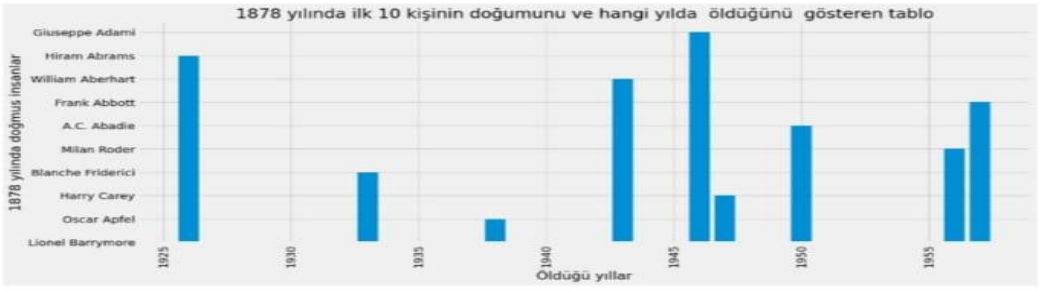
In [8]:
```
birth_Year=list(df["birthYear"].unique())
birth_Year.sort()
```

In [9]:
```
birth_Year=birth_Year[340:]
```

In [10]:
```
primary=list(df["primaryProfession"].unique())
```

In [11]:
```
# her bir yılda doğan insanların ne zaman öldüklerini gösteren ilk 10 kişinin 1877 yılın
dan itibaren tablosu
for i in birth_Year:
    c=df[df["birthYear"]==i]

    plt.style.use("fivethirtyeight")
    plt.figure(figsize=(16,6))
    plt.bar(c["deathYear"][:10],c["primaryName"][:10])
    plt.xlabel("Öldüğü yıllar")
    plt.title(f"{i} yılında ilk 10 kişinin doğumunu ve hangi yılda  öldüğünü  gösteren
tablo")
    plt.xticks(rotation=90)
    plt.ylabel(f"{i} yılında doğmus insanlar")
    plt.show()
```



1880 yılında ilk 10 kişinin doğumunu ve hangi yılda öldüğünü gösteren tablo
Viktor Aden
Fritz Achterberg
Chris van Abkoude
Bror Abelli
Legaren à Hiller
Robert Stolz
Hans Moser
Montagu Love
Gilbert M. "Broncho Billy" Anderson
W.C. Fields
1880 yılında doğmus insanlar
1940
1945
1950
1955
1960
1965
1970
1975
Öldüğü yıllar


1881 yılında ilk 10 kişinin doğumunu ve hangi yılda öldüğünü gösteren tablo
Dan Agar
John G. Adolfi
Lowden Adams
Walter Ackerman
Jesus Hermenegildo Abitia
Achmed Abdullah
Taso Abashidze
Asta Nielsen
Pedro de Cordoba
Cecil B. DeMille
1881 yılında doğmus insanlar
1935
1940
1945
1950
1955
1960
1965
1970
Öldüğü yıllar


1882 yılında ilk 10 kişinin doğumunu ve hangi yılda öldüğünü gösteren tablo
Igor Stravinsky
Richard Hageman
Roy F. Overbaugh
André Barlatier
James Joyce
John Hughes
Noah Beery
John W. Brown
John Barrymore
Bela Lugosi
1882 yılında doğmus insanlar
1940
1945
1950
1955
1960
1965
1970
Öldüğü yıllar


1883 yılında ilk 10 kişinin doğumunu ve hangi yılda öldüğünü gösteren tablo
Muriel Aked
Juan Aguilar
Stella Adams
Frank R. Adams
Maurice Abrahams
David Abel
James E. Abbe
Franz Doelle
Joseph Crehan
Douglas Fairbanks
1883 yılında doğmus insanlar
1930
1940
1950
1960
1970
Öldüğü yıllar


1884 yılında ilk 10 kişinin doğumunu ve hangi yılda öldüğünü gösteren tablo
Robyn Adair
Ilona Cs. Aczél
Max Abramson
Hector Abbas
Hans Theyer
Nikolay P. Cherkasov
Harry Langdon
Hugh Herbert
Harrison Ford
Billie Burke
1884 yılında doğmus insanlar
1940
1945
1950
1955
1960
1965
1970
Öldüğü yıllar

In [12]:
```
na_values = ["\\N", "nan"]
df1=pd.read_csv("/kaggle/input/imdb-dataset/title.basics.tsv/data.tsv",sep='\t',low_me
mory=False, na_values=na_values)
```

In [13]:
```
df1.head()
```

Out[13]:

| | tconst | titleType | primaryTitle | originalTitle | isAdult | startYear | endYear | runtimeMinutes | genres |
|---|---|---|---|---|---|---|---|---|---|
| 0 | tt0000001 | short | Carmencita | Carmencita | 0.0 | 1894.0 | NaN | 1 | Documentary,Sh |
| 1 | tt0000002 | short | Le clown et ses chiens | Le clown et ses chiens | 0.0 | 1892.0 | NaN | 5 | Animation,Short |
| 2 | tt0000003 | short | Pauvre Pierrot | Pauvre Pierrot | 0.0 | 1892.0 | NaN | 4 | Animation,Comec |
| 3 | tt0000004 | short | Un bon bock | Un bon bock | 0.0 | 1892.0 | NaN | 12 | Animation,Short |
| 4 | tt0000005 | short | Blacksmith Scene | Blacksmith Scene | 0.0 | 1893.0 | NaN | 1 | Comedy,Short |

In [14]:
```
title_Type=list(df1["titleType"].unique())
```

In [15]:
```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10233937 entries, 0 to 10233936
Data columns (total 9 columns):
 #   Column          Dtype
---  ------          -----
 0   tconst          object
 1   titleType       object
 2   primaryTitle    object
 3   originalTitle   object
 4   isAdult         float64
 5   startYear       float64
 6   endYear         float64
 7   runtimeMinutes  object
 8   genres          object
dtypes: float64(3), object(6)
memory usage: 702.7+ MB
```

In [16]:
```
df1=df1.dropna()
```

In [17]:
```
df1.info()
```

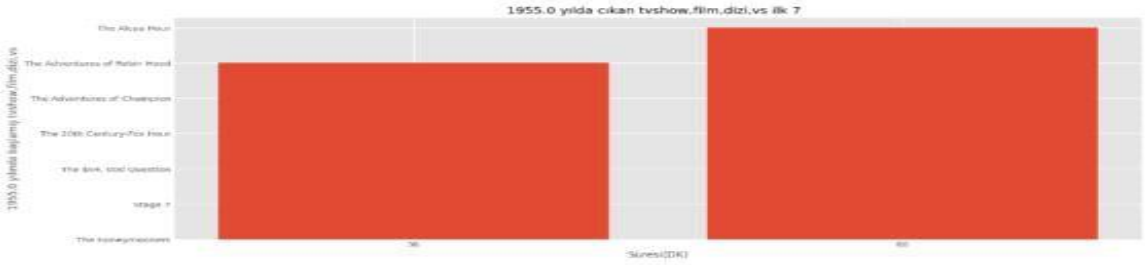
```
<class 'pandas.core.frame.DataFrame'>
Index: 52603 entries, 35174 to 10233631
Data columns (total 9 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   tconst          52603 non-null  object
 1   titleType       52603 non-null  object
 2   primaryTitle    52603 non-null  object
 3   originalTitle   52603 non-null  object
 4   isAdult         52603 non-null  float64
 5   startYear       52603 non-null  float64
 6   endYear         52603 non-null  float64
 7   runtimeMinutes  52603 non-null  object
 8   genres          52603 non-null  object
dtypes: float64(3), object(6)
memory usage: 4.0+ MB
```

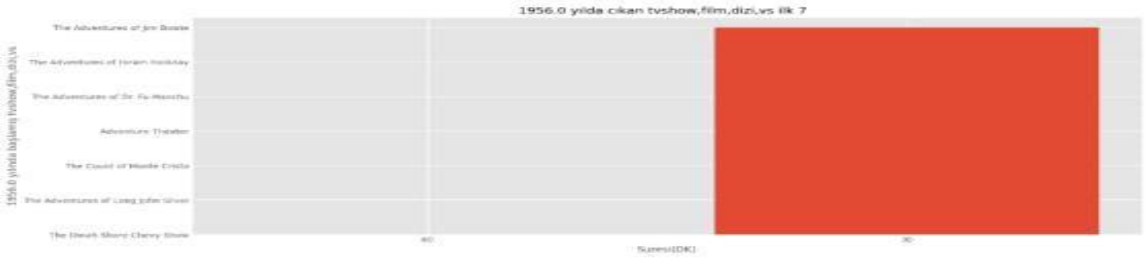
In [18]:
```
start_Year=list(df1["startYear"].unique())
```

[19]:
```
start_Year.sort()
```

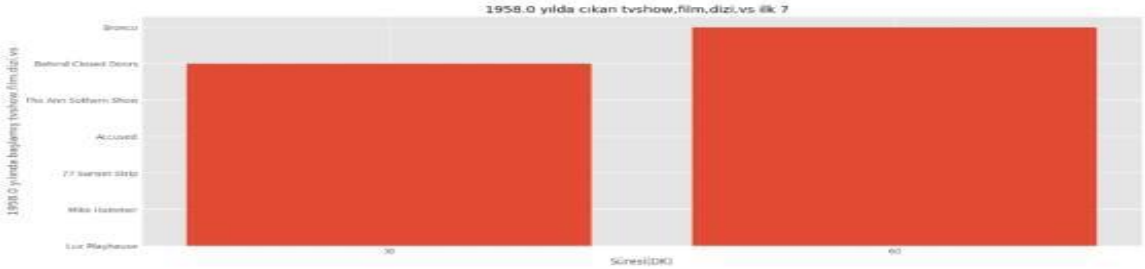
[20]:
```
start_Year=start_Year[20:]
```

[21]:
```
for i in start_Year:
    a=df1[df1["startYear"]==i]
    plt.style.use("ggplot")
    plt.figure(figsize=(16,6))
    plt.bar(a["runtimeMinutes"][:7],a["primaryTitle"][:7])
    plt.xlabel("Süresi(DK)")
    plt.title(f"{i} yılda cıkan tvshow,film,dizi,vs ilk 7  ")
    plt.ylabel(f"{i} yılında başlamış tvshow,film,dizi,vs")
    plt.show()
```







## Conclusion :

These are the general steps to load and preprocess a **IMDb** dataset using Python and Pandas. Remember that the specific preprocessing steps and operations may vary depending on the structure of your dataset and your analysis objectives.