```
'''                                  ******Entertainer Data Analytics******

Sheet 1: Entertainer - Basic Data

The dataset comprises 70 rows and 3 columns, with each row representing an entertainer. The
columns include:
Name: The name of the actor or actress
Gender: The gender of the entertainer
Birth Year: The year of birth of the entertainer

Dataset Features:
Given the limited number of features in this dataset, we primarily focus on visualizing the
birth year data to derive insights. To illustrate these observations, we will plot a histogram
of the birth year data.
insights.
'''



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

fp = "C:/Users/aashl/OneDrive/Desktop/project 1/Entertainer - Basic Info.xlsx"

data = pd.read_excel(fp,0)

print('The first few rows of the given dataset can be viewed by:', data.head())
#print(data.isnull().sum())
data = data.dropna()              #used to remove the null values from a table
print(data.shape)                 #indicates the number of rows and columns
print(data.index)                 #the number of indices, for example: (0-69)(70 values)
print(data.columns)               #the total number of columns
print(data.info())                #briefs us about the datatype and and the total amount of
memory used.
print(data.dropna())

'''
To illustrate these observations, we will plot a histogram of the birth year data.
'''

plt.figure(figsize=(11,6))
sns.histplot(data=data, x='Birth Year', color='orange', edgecolor='linen', alpha=0.5, bins=5)
plt.title('Distribution of basic info')
plt.xlabel('Birth Year')
plt.ylabel('Gender (traditional)')
plt.show()

'''
Observations:
Upon analyzing the birth year distribution, the following trends are observed:

    1.The majority of actors and actresses were born in the year 1940, followed by notable
numbers in 1900 and 1960.
'''
```

```
'''
                                    Entertainer Data - Breakthrough

The dataset consists of 64 rows and 4 columns, with each row providing information about the
entertainers, including their name, gender, and birth year.

Dataset Features
- **Name**: The name of the actors or actresses.
- **Breakthrough/Hit/Award Nomination**: The title of the movie, show, or notable achievement
that marked their career breakthrough in Hollywood.
- **Year of First Major Award**: The year they received their first Oscar, Grammy, or Emmy,
indicating their rise to prominence.
'''

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

fp1 = "C:/Users/aashl/OneDrive/Desktop/project 1/Entertainer - Breakthrough Info.xlsx"
data1 = pd.read_excel(fp1,0)
print(data1.head())
#print(data.isnull().sum())
data1 = data1.dropna()
print(data1.shape)
print(data1.index)
print(data1.columns)
print(data1.info())

'''We will now examine the distributions of various features in the dataset and calculate
appropriate measures
such as the mean, median, and mode to display our findings.
'''
#Scatterplot between Year of First Oscar/Grammy/Emmy And Year of Breakthrough
plt.figure(figsize=(11,6))
sns.scatterplot(data=data1, x='Year of First Oscar/Grammy/Emmy', y='Year of Breakthrough/#1
Hit/Award Nomination', color='red', edgecolor='linen', alpha=0.5)
plt.title('Distribution of Movie Ratings')
plt.xlabel('Year of first Oscar/Grammy/Emmy')
plt.ylabel('Year of Breakthrough/#1 Hit/Award Nomination')
plt.show()

'''
The scatterplot reveals that the year of the actors' first major award is typically within one
to two years
of their breakthrough year. This suggests a strong correlation between the release of their
breakthrough series
or movie and receiving recognition for their performance.
This finding indicates that notable performances are quickly acknowledged and rewarded in the
entertainment industry.
'''

plt.figure(figsize=(11,6))
sns.histplot(data=data1, x='Year of Breakthrough/#1 Hit/Award Nomination', color='green',
edgecolor='linen', alpha=0.5, bins=5)
plt.title('Distribution of Movie Ratings')
plt.xlabel('Year of Breakthrough/#1 Hit/Award Nomination')
plt.vlines(data1['Year of Breakthrough/#1 Hit/Award Nomination'].mean(),
ymin=0,ymax=40,colors='blue',label='Mean')
plt.vlines(data1['Year of Breakthrough/#1 Hit/Award Nomination'].median(),
ymin=0,ymax=40,colors='red',label='Mean')
plt.show()

'''Observations:
The mean and median are very close to each other, indicating that they are reliable
representatives of the data.
Given their proximity, either the mean or the median can be used as the measure of central
```

```
tendency. '''

#plot distplot using Year of Breakthrough/#1 Hit/Award Nomination
#plt.figure(figsize=(11,6))
#sns.distplot(data1['Year of Breakthrough/#1 Hit/Award Nomination'], color='#e290f1')
#plt.title('distplot of Year of Breakthrough/#1 Hit/Award Nomination')
#plt.xlabel('Year of Breakthrough/#1 Hit/Award Nomination')
#plt.show()

'''Distribution Plot Observations:

The normal distribution is described by the mean and standard deviation.
It is often referred to as a âbell curveâ due to its shape.
In a normal distribution:
The mean equals the median.
There is only one mode.
It is symmetric, decreasing equally on the left, right, and center.
'''

print('\nDetails of Year of Breakthrough/#1 Hit/Award Nomination are as follows: \n')
print('skewness: ', data1['Year of Breakthrough/#1 Hit/Award Nomination'].skew())
print('mean: ', data1['Year of Breakthrough/#1 Hit/Award Nomination'].mean())
print('median: ',data1[ 'Year of Breakthrough/#1 Hit/Award Nomination'].median())

print('\nDetails of Year of First Oscar/Grammy/Emmy are as follows: \n')

print('mean: ', data1[ 'Year of First Oscar/Grammy/Emmy'].mean())
print('median: ',data1['Year of First Oscar/Grammy/Emmy'].median())

rep_values = pd.Series(index=['Year of Breakthrough/#1 Hit/Award Nomination', 'Year of First
Oscar/Grammy/Emmy'],
                       data=[data1['Year of Breakthrough/#1 Hit/Award Nomination'].mean(),
data1['Year of First Oscar/Grammy/Emmy'].mean()])
print("\nAverage value of the years in which breakthrough and the year of first
Oscar/Grammy/Emmy are:\n")
print(rep_values)
```

```
'''                                  ***Entertainer- Last Work***
'''

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

fp1="C:/Users/aashl/OneDrive/Desktop/project 1/Entertainer - Last work Info.xlsx"
data1= pd.read_excel(fp1,0)

print(data1.info())
data_cleaned=data1.dropna()
print(data1.head())
print('The non-null values of the given table can be represented as\n:', data_cleaned)

plt.figure(figsize=(11,6))
sns.histplot(data1['Year of Last Major Work (arguable)'] , color='#b411e8')
plt.title("Histplot")
plt.xlabel('Year of Last Major Work (arguable)')
plt.show()

'''Summary of Year of Last Major Work Analysis
From the histogram and data analysis, it is clear that the year 2016 has the highest number of
occurrences,
with a total count of 30.

This observation suggests that 2016 was a significant year for many entertainers in terms of
their last
notable contributions to their careers.'''

print(data1['Year of Last Major Work (arguable)'].value_counts())
print(data1['Year of Last Major Work (arguable)'].value_counts().index[0])

rep_values = pd.Series(index=['Year of Last Major Work (arguable)', 'Year of
Death'],data=[data1['Year of Last Major Work (arguable)'].mean(), data1['Year of
Death'].mean()])
print(rep_values)

'''After careful and intensive analysis of all the given entertainer datasets, we come to the
following Final conclusions:
    1.From the given data, we can use simple visualisations to get a sense of how data is
distributed.

    2.We can use various measures of central tendency (Mean,Median,Mode) to represent a group
of observations.

    3.The type of central tendency measure to use depends on the type and distribution of the
data.
'''
```