Task 1 -- Perform Data Cleaning

Clean a dataset by removing missing values and outliers

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
df=pd.read_csv("test.csv")
df.head()
```

Out[2]:

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Eml
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	

In [129…
```
df.shape
```

Out[129]:
```
(418, 11)
```

In [131…
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 418 entries, 0 to 417
Data columns (total 11 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  418 non-null    int64  
 1   Pclass       418 non-null    int64  
 2   Name         418 non-null    object 
 3   Sex          418 non-null    object 
 4   Age          332 non-null    float64
 5   SibSp        418 non-null    int64  
 6   Parch        418 non-null    int64  
 7   Ticket       418 non-null    object 
 8   Fare         417 non-null    float64
 9   Cabin        91 non-null     object 
 10  Embarked     418 non-null    object 
dtypes: float64(2), int64(4), object(5)
memory usage: 36.0+ KB
```

In [182…
```
df.describe(include = 'all')
```

Out[182]:

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	
count	418.000000	418.000000	418	418	332.000000	418.000000	418.000000	4.160
unique	NaN	NaN	418	2	NaN	NaN	NaN	
top	NaN	NaN	Kelly, Mr. James	male	NaN	NaN	NaN	
freq	NaN	NaN	1	266	NaN	NaN	NaN	
mean	1100.500000	2.265550	NaN	NaN	30.272590	0.447368	0.392344	4.538
std	120.810458	0.841838	NaN	NaN	14.181209	0.896760	0.981429	2.293
min	892.000000	1.000000	NaN	NaN	0.170000	0.000000	0.000000	2.500
25%	996.250000	1.000000	NaN	NaN	21.000000	0.000000	0.000000	1.369
50%	1100.500000	3.000000	NaN	NaN	27.000000	0.000000	0.000000	5.333
75%	1204.750000	3.000000	NaN	NaN	39.000000	1.000000	0.000000	3.460
max	1309.000000	3.000000	NaN	NaN	76.000000	8.000000	9.000000	2.310

Missing values treatment

In [183…
```
df.isnull().sum()
```

Out[183]:
```
PassengerId      0
Pclass           0
Name             0
Sex              0
Age             86
SibSp            0
Parch            0
Ticket           2
Fare             1
Cabin          327
Embarked         0
dtype: int64
```

In [ ]:
```
df["Age"]
```

In [132…
```
# check datatype
df.dtypes
```

Out[132]:
```
PassengerId      int64
Pclass           int64
Name            object
Sex             object
Age            float64
SibSp            int64
Parch            int64
Ticket          object
Fare           float64
Cabin           object
Embarked        object
dtype: object
```

In [116…
```
pd.set_option('display.max_rows',None)
```

In [139…
```
# To change Datatype

df[['Ticket']] = df[['Ticket']].apply(pd.to_numeric)
print(df.dtypes)
```

```
PassengerId      int64
Pclass           int64
Name            object
Sex             object
Age            float64
SibSp            int64
Parch            int64
Ticket         float64
Fare           float64
Cabin           object
Embarked        object
dtype: object
```

In [69]:
```
df.Age.fillna(30,inplace=True)
```

In [101…
```
# Remove extra specific character
df["Ticket"] = [x.strip("A-Z") for x in df["Ticket"]]

# Remove all extra character
df['Ticket'] = df['Ticket'].str.replace('\D', '', regex=True)
```

In [105…
```
df.head(10)
```

Out[105]:

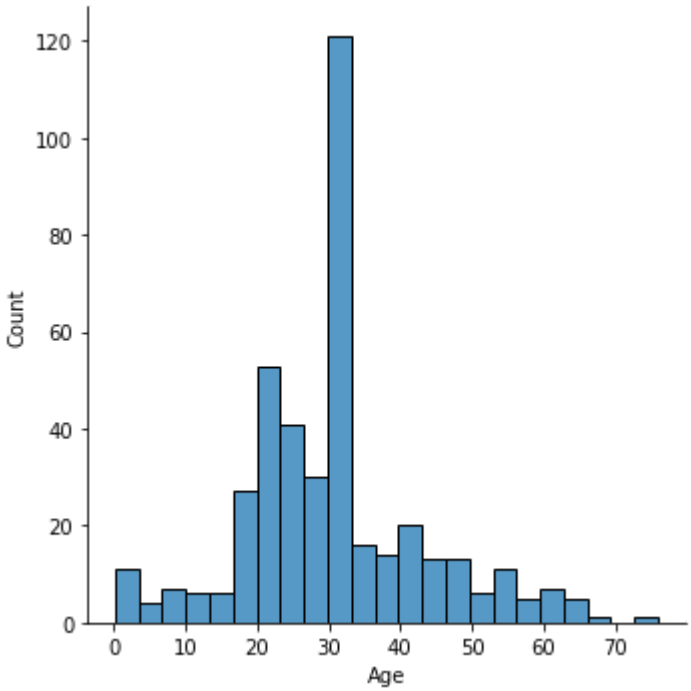
	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	E
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	
5	897	3	Svensson, Mr. Johan Cervin	male	14.0	0	0	7538	9.2250	NaN	
6	898	3	Connolly, Miss. Kate	female	30.0	0	0	330972	7.6292	NaN	
7	899	2	Caldwell, Mr. Albert Francis	male	26.0	1	1	248738	29.0000	NaN	
8	900	3	Abrahim, Mrs. Joseph (Sophie Halaut Easu)	female	18.0	0	0	2657	7.2292	NaN	
9	901	3	Davies, Mr. John Samuel	male	21.0	2	0	448871	24.1500	NaN	

In [ ]:

Detect outlier

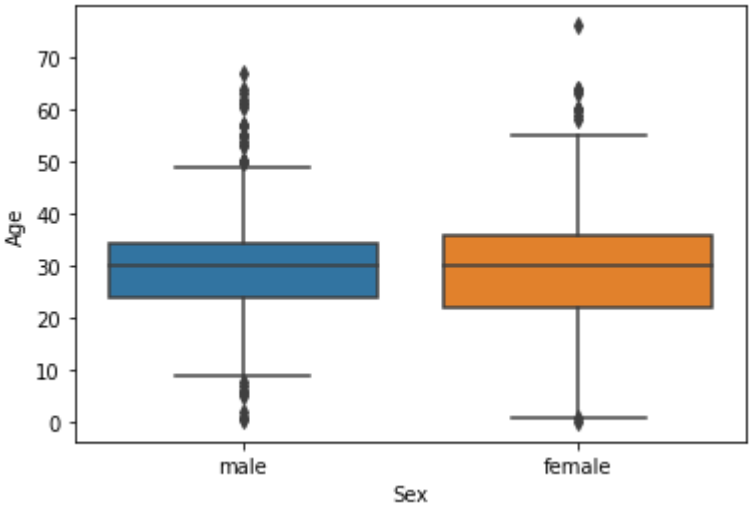
In [71]:
```
# check outlier
sns.displot(df["Age"])
```

Out[71]:
```
<seaborn.axisgrid.FacetGrid at 0x7fa8d31dfd90>
```



In [72]:
```
# Here we can see the outlier, now will try to removie or replce acording th
sns.boxplot(x='Sex',y='Age',data=df)
```

Out[72]:
```
<AxesSubplot:xlabel='Sex', ylabel='Age'>
```



Treat outlier

In [172…
```
high_boud = df["Age"].quantile(.95)
high_boud
```

Out[172]:
```
57.0
```

In [ ]:
```
low_boud = df["Age"].quantile(0.5)
low_boud
```

In [167…
```
df[df["Age"]<high_boud]
df.head(10)
```

Out[167]:

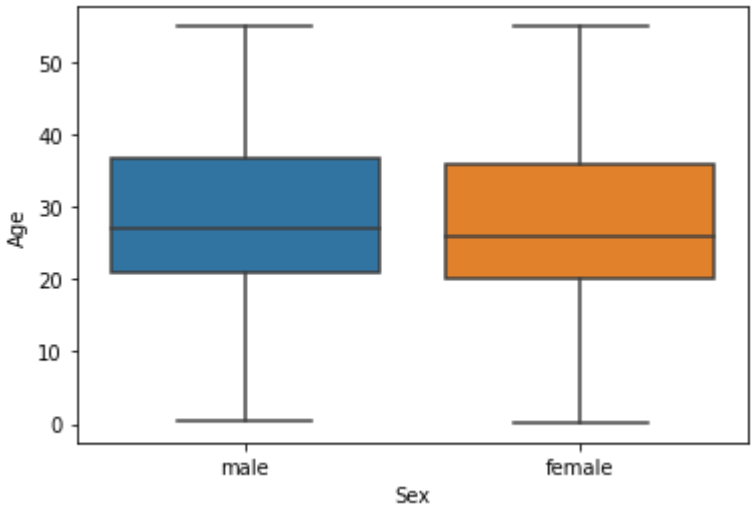
	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	E
0	892	3	Kelly, Mr. James	male	34.5	0	0	33091.0	7.8292	NaN	
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	36327.0	7.0000	NaN	
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	24027.0	9.6875	NaN	
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	31515.0	8.6625	NaN	
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	310129.0	12.2875	NaN	
5	897	3	Svensson, Mr. Johan Cervin	male	14.0	0	0	753.0	9.2250	NaN	
6	898	3	Connolly, Miss. Kate	female	30.0	0	0	33097.0	7.6292	NaN	
7	899	2	Caldwell, Mr. Albert Francis	male	26.0	1	1	24873.0	29.0000	NaN	
8	900	3	Abrahim, Mrs. Joseph (Sophie Halaut Easu)	female	18.0	0	0	265.0	7.2292	NaN	
9	901	3	Davies, Mr. John Samuel	male	21.0	2	0	44887.0	24.1500	NaN	

In [173…

```
show=df[df["Age"]<high_boud]
sns.boxplot(x="Sex",y="Age",data=show)
```

Out[173]:

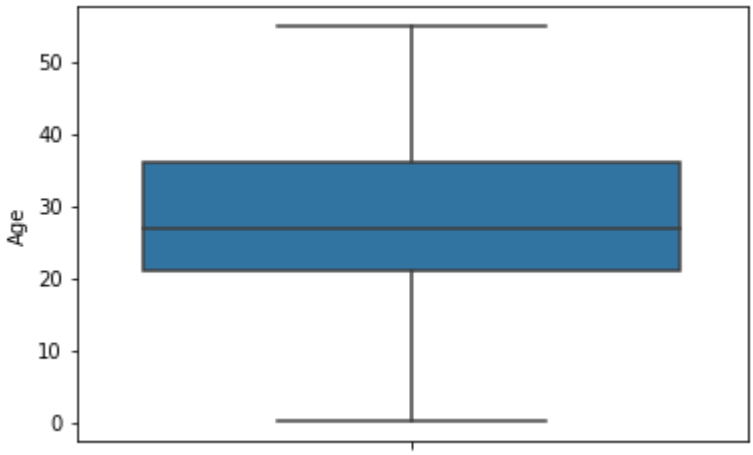
```
<AxesSubplot:xlabel='Sex', ylabel='Age'>
```



In [ ]:

In [165…
```
df1=df[df['Age']<max]
sns.boxplot(y='Age',data=df1)
```

Out[165]: <AxesSubplot:ylabel='Age'>



In [38]:
```
mult_conditions = df[(df['Sex'] == 'female') & (df['Age'] > 50)]
mult_conditions.head()
```

Out[38]:

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
48	940	1	Bucknell, Mrs. William Robert (Emma Eliza Ward)	female	60.0	0	0	11813	76.2917	D15
69	961	1	Fortune, Mrs. Mark (Mary McDougald)	female	60.0	1	4	19950	263.0000	C23 C25 C27
77	969	1	Cornell, Mrs. Robert Clifford (Malvina Helen L...	female	55.0	2	0	11770	25.7000	C101
96	988	1	Cavendish, Mrs. Tyrell William (Julia Florence...	female	76.0	1	0	19877	78.8500	C46
114	1006	1	Straus, Mrs. Isidor (Rosalie Ida Blun)	female	63.0	1	0	PC 17483	221.7792	C55 C57

In [54]:
```
df2=df[["Sex","Age"]]
df2.head()
```

Out[54]:

	Sex	Age
0	male	34.5
1	female	47.0
2	male	62.0
3	male	27.0
4	female	22.0

In [55]:
```
df["Age"].describe()
```

Out[55]:
```
count    332.000000
mean      30.272590
std       14.181209
min        0.170000
25%       21.000000
50%       27.000000
75%       39.000000
max       76.000000
Name: Age, dtype: float64
```

In [62]:
```
df[df["Age"]<max]
```

Out[62]:

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	C
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	
5	897	3	Svensson, Mr. Johan Cervin	male	14.0	0	0	7538	9.2250	
...	...	...	...	...	...	...	...	...	...	
409	1301	3	Peacock, Miss. Treasteall	female	3.0	1	1	SOTON/O.Q. 3101315	13.7750	
411	1303	1	Minahan, Mrs. William Edward (Lillian E Thorpe)	female	37.0	1	0	19928	90.0000	
412	1304	3	Henriksson, Miss. Jenny Lovisa	female	28.0	0	0	347086	7.7750	
414	1306	1	Oliva y Ocana, Dona. Fermina	female	39.0	0	0	PC 17758	108.9000	C
415	1307	3	Saether, Mr. Simon Sivertsen	male	38.5	0	0	SOTON/O.Q. 3101262	7.2500	

313 rows × 11 columns

In [152…
```
min=df["Age"].quantile(0.5)
min
```

Out[152]:
```
27.0
```

In [151…
```
max=df["Age"].quantile(0.95)
max
```

Out[151]:
```
57.0
```

In [ ]: