Task 1: Perform Data Cleaning

In [2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
from matplotlib import style
import seaborn as sns
```

In [3]:
```
#import csv file
df = pd.read_csv('Teck task 1.csv')
```

Data cleaning

In [4]:
```
#check the data type
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  891 non-null    int64  
 1   Survived     891 non-null    int64  
 2   Pclass       891 non-null    int64  
 3   Name         891 non-null    object 
 4   Sex          891 non-null    object 
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64  
 7   Parch        891 non-null    int64  
 8   Ticket       891 non-null    object 
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object 
 11  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

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

Out[5]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

In [6]:
```
#check the null values
pd.isnull(df).sum()
```

Out[6]:
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

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

Out[8]:
```
(891, 12)
```

In [15]:
```
df.dropna(inplace=True)
```

In [16]:
```
df.shape
```

Out[16]:
```
(183, 12)
```

In [11]:
```
#Drop duplicate values
df.drop_duplicates(subset = 'PassengerId', keep = "first").head(2)
```

Out[11]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C

In [17]:
```
#float to int
df['Age'] = df['Age'].astype('int')
```

In [18]:
```
df['Fare'] = df['Fare'].astype('int')
```

In [20]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 183 entries, 1 to 889
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype 
---  ------       --------------  ----- 
 0   PassengerId  183 non-null    int64 
 1   Survived     183 non-null    int64 
 2   Pclass       183 non-null    int64 
 3   Name         183 non-null    object
 4   Sex          183 non-null    object
 5   Age          183 non-null    int32 
 6   SibSp        183 non-null    int64 
 7   Parch        183 non-null    int64 
 8   Ticket       183 non-null    object
 9   Fare         183 non-null    int32 
 10  Cabin        183 non-null    object
 11  Embarked     183 non-null    object
dtypes: int32(2), int64(5), object(5)
memory usage: 17.2+ KB
```

In [34]:
```
#column's name corrections
df.rename(columns = {'PassengerId':'Passenger ID'},inplace = True)
```

In [33]:
```
df.rename(columns = {'Pclass':'Passenger Class'},inplace = True)
```

In [23]:
```
df.rename(columns = {'SibSp':'Sibling/Spouse'},inplace = True)
```

In [24]:
```
df.rename(columns = {'Parch':'Parents/Children'},inplace = True)
```

In [35]:
```
df.describe()
```

Out[35]:

	Passenger Id	Survived	Passenger Class	Age	Sibling/Spouse	Parents/Children	Fare
count	183.000000	183.000000	183.000000	183.000000	183.000000	183.000000	183.000000
mean	455.366120	0.672131	1.191257	35.661202	0.464481	0.475410	78.273224
std	247.052476	0.470725	0.515187	15.654054	0.644159	0.754617	76.362868
min	2.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000
25%	263.500000	0.000000	1.000000	24.000000	0.000000	0.000000	29.000000
50%	457.000000	1.000000	1.000000	36.000000	0.000000	0.000000	57.000000
75%	676.000000	1.000000	1.000000	47.500000	1.000000	1.000000	90.000000
max	890.000000	1.000000	3.000000	80.000000	3.000000	4.000000	512.000000

Steps for Data cleaning

1. Import the CSV file.

2. Check the datatype in the given csv file by using, df.info(). However, check the details of the rows and columns' values and their ## datatypes. Here some of the datatype's values are in integers and two are in float.

3. Changed the value from float to int by using, df['Fare'] = df['Fare'].astype('int'). Check the corrected values from df.head().

4. Check the null values by using pd.isnull(df).sum(). Drop the null values by using, df.dropna(inplace=True).

5. Drop the duplicate values, df.drop_duplicates(subset = 'PassengerId', keep = "first").head(2). Here the passenger id is used to drop the duplicate because the passenger ID is one of the unique things.

6. Some of the given column names are in short form so changed them in clear format for the visualization. Rename the given name by using, df.rename(columns={'SibSp':'Sibling/Spouse'},inplace=True)

7. df.shape shown the rows and columns values.

8. df.describe calculate the mean, count, std, min, max, and %age of each row.

In [ ]: