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

In [3]:
```
titanic = pd.read_csv('C:\\24 - Machine_Learning\\download files\\titanic.csv')
titanic
```

Out[3]:

	PassengerId	Pclass	Name	Gender	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	Q
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	S
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	Q
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	S
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...
413	1305	3	Spector, Mr. Woolf	male	NaN	0	0	A.5. 3236	8.0500	NaN	S
414	1306	1	Oliva y Ocana, Dona. Fermina	female	39.0	0	0	PC 17758	108.9000	C105	C
415	1307	3	Saether, Mr. Simon Sivertsen	male	38.5	0	0	SOTON/O.Q. 3101262	7.2500	NaN	S
416	1308	3	Ware, Mr. Frederick	male	NaN	0	0	359309	8.0500	NaN	S
417	1309	3	Peter, Master. Michael J	male	NaN	1	1	2668	22.3583	NaN	C

418 rows × 11 columns

In [4]:
```
titanic.head()
# by default head will show first 5 records
```

Out[4]:

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

In [5]:
```
titanic.head(10)
# to show first 10 records
```

Out[5]:

	PassengerId	Pclass	Name	Gender	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	Q
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	S
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	Q
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	S
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	S
5	897	3	Svensson, Mr. Johan Cervin	male	14.0	0	0	7538	9.2250	NaN	S
6	898	3	Connolly, Miss. Kate	female	30.0	0	0	330972	7.6292	NaN	Q
7	899	2	Caldwell, Mr. Albert Francis	male	26.0	1	1	248738	29.0000	NaN	S
8	900	3	Abrahim, Mrs. Joseph (Sophie Halaut Easu)	female	18.0	0	0	2657	7.2292	NaN	C
9	901	3	Davies, Mr. John Samuel	male	21.0	2	0	A/4 48871	24.1500	NaN	S

In [6]:
```
titanic.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   Gender       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 [7]:
```
titanic.isnull()
# returns true if null value
```

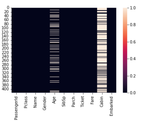
Out[7]:

	PassengerId	Pclass	Name	Gender	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	False	False	False	False	False	False	False	False	False	True	False
1	False	False	False	False	False	False	False	False	False	True	False
2	False	False	False	False	False	False	False	False	False	True	False
3	False	False	False	False	False	False	False	False	False	True	False
4	False	False	False	False	False	False	False	False	False	True	False
...	...	...	...	...	...	...	...	...	...	...	...
413	False	False	False	False	True	False	False	False	False	True	False
414	False	False	False	False	False	False	False	False	False	False	False
415	False	False	False	False	False	False	False	False	False	True	False
416	False	False	False	False	True	False	False	False	False	True	False
417	False	False	False	False	True	False	False	False	False	True	False

418 rows × 11 columns

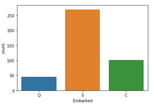
In [8]:
```
sns.heatmap(titanic.isnull())
```

Out[8]:
```
<AxesSubplot:>
```



In [9]:
```
# MISSING DATA
sns.countplot(x='Embarked', data=titanic)
```

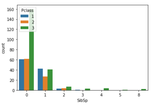
Out[9]:
```
<AxesSubplot:xlabel='Embarked', ylabel='count'>
```



In [ ]:
```
# From Passenger list most of the passengers are Embarked "S" class
```

In [10]:
```
sns.countplot(x='SibSp', hue='Pclass', data=titanic)
```

Out[10]:
```
<AxesSubplot:xlabel='SibSp', ylabel='count'>
```



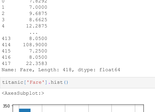
In [ ]:
```
# From Passengers list most of the passengers are SibSp(Siblings&Spouse) i.e.dependence "0" class
# more passenger with 0 dependence
```

In [12]:
```
titanic['Fare']
```

Out[12]:
```
0       7.8292
1       7.0000
2       9.6875
3       8.6625
4      12.2875
        ...   
413     8.0500
414   108.9000
415     7.2500
416     8.0500
417    22.3583
Name: Fare, Length: 418, dtype: float64
```

In [13]:
```
titanic['Fare'].hist()
```

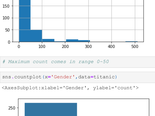
Out[13]:
```
<AxesSubplot:>
```



In [ ]:
```
# Maximum count comes in range 0-50
```

In [16]:
```
sns.countplot(x='Gender',data=titanic)
```

Out[16]:
```
<AxesSubplot:xlabel='Gender', ylabel='count'>
```



In [ ]:
```
# From the graph we can identify the count of male & female
```

In [18]:
```
titanic['Age'].hist()
```

Out[18]:
```
<AxesSubplot:>
```

In [ ]:
```
# From the histogram we can get the maximum no of age group
```

In [ ]:
```
# DATA CLEANING
```

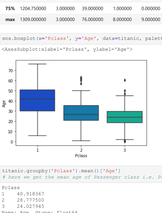
In [19]:
```
titanic.describe()
```

Out[19]:

	PassengerId	Pclass	Age	SibSp	Parch	Fare
count	418.000000	418.000000	332.000000	418.000000	418.000000	417.000000
mean	1100.500000	2.265550	30.272590	0.447368	0.392344	35.627188
std	120.810458	0.841838	14.181209	0.896760	0.981429	55.907576
min	892.000000	1.000000	0.170000	0.000000	0.000000	0.000000
25%	996.250000	1.000000	21.000000	0.000000	0.000000	7.895800
50%	1100.500000	3.000000	27.000000	0.000000	0.000000	14.454200
75%	1204.750000	3.000000	39.000000	1.000000	0.000000	31.500000
max	1309.000000	3.000000	76.000000	8.000000	9.000000	512.329200

In [20]:
```
sns.boxplot(x='Pclass', y='Age', data=titanic, palette='winter')
```

Out[20]:
```
<AxesSubplot:xlabel='Pclass', ylabel='Age'>
```



In [21]:
```
titanic.groupby('Pclass').mean()['Age']
# here we get the mean age of Passenger class i.e. Pclass
```

Out[21]:
```
Pclass
1    40.918367
2    28.777500
3    24.027945
Name: Age, dtype: float64
```

In [28]:
```
def impute_Age(cols):
    Age = cols[0]
    Pclass = cols[1]

    if pd.isnull(Age):

        if Pclass == 1:
            return 41

        elif Pclass == 2:
            return 29

        else:
            return 24

    else:
        return Age
```

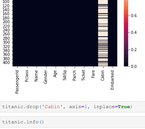
In [29]:
```
titanic['Age'] = titanic[['Age','Pclass']].apply(impute_Age, axis=1)
```

In [30]:
```
titanic['Age']
```

Out[30]:
```
0      34.5
1      47.0
2      62.0
3      27.0
4      22.0
       ... 
413    24.0
414    39.0
415    38.5
416    24.0
417    24.0
Name: Age, Length: 418, dtype: float64
```

In [31]:
```
sns.heatmap(titanic.isnull())
```

Out[31]:
```
<AxesSubplot:>
```



In [32]:
```
titanic.drop('Cabin', axis=1, inplace=True)
```

In [33]:
```
titanic.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 418 entries, 0 to 417
Data columns (total 10 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   Gender       418 non-null    object 
 4   Age          418 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   Embarked     418 non-null    object 
dtypes: float64(2), int64(4), object(4)
memory usage: 32.8+ KB
```