```
In [1]:  #AKSHATA JADHAV
         #TE-A-24
         #Batch TA1-TA2
```

```
In [2]:  import pandas as pd
         import numpy as np
```

```
In [3]:  df=pd.read_csv("train.csv")
```

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

Out[4]:

	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 [5]:  df.columns
```

Out[5]: Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
                'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
               dtype='object')

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

Out[6]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

```
In [7]: 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 [8]: df.shape
```

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

```
In [9]: df.isna().sum()
```

```
Out[9]: 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 [10]: def fun1(value):
             if (value == "male"):
                 return 1
             elif (value == "female"):
                 return 0
             else:
                 pass
```

```
In [11]: df["Sex"].value_counts()
```

```
Out[11]: Sex
         male      577
         female    314
         Name: count, dtype: int64
```

```
In [12]: df["Sex"] = df["Sex"].apply(fun1)
```

```
In [13]: df.head(10)
```

Out[13]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	1	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	0	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	0	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	0	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	1	35.0	0	0	373450	8.0500	NaN	S
5	6	0	3	Moran, Mr. James	1	NaN	0	0	330877	8.4583	NaN	Q
6	7	0	1	McCarthy, Mr. Timothy J	1	54.0	0	0	17463	51.8625	E46	S
7	8	0	3	Palsson, Master. Gosta Leonard	1	2.0	3	1	349909	21.0750	NaN	S
8	9	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	0	27.0	0	2	347742	11.1333	NaN	S
9	10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	0	14.0	1	0	237736	30.0708	NaN	C

In [14]:
```python
def fun2(value):
    if (np.isnan(value) == False):
        return int(value)
```

In [16]:
```python
df["Age"] = df["Age"].apply(fun2)
```

In [17]:
```python
df.head()
```

Out[17]:

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

In [18]:
```python
df_age_greater_than_25 = df[df["Age"] > 25.0]
```

In [19]:
```python
df_age_greater_than_25.head()
```

Out[19]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	0	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	0	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	0	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	1	35.0	0	0	373450	8.0500	NaN	S
6	7	0	1	McCarthy, Mr. Timothy J	1	54.0	0	0	17463	51.8625	E46	S

In [20]:
```python
df.columns = ["passenger_id", "survived", "p_class", "name", "sex", "age", "sib_sp", "parch",
```

In [21]:
```python
df.head()
```

Out[21]:

	passenger_id	survived	p_class	name	sex	age	sib_sp	parch	ticket	fare	cabin	embarked
0	1	0	3	Braund, Mr. Owen Harris	1	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	0	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	0	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	0	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	1	35.0	0	0	373450	8.0500	NaN	S

In [22]:
```python
df["age"].value_counts()
```

Out[22]:
```
age
24.0    31
22.0    27
28.0    27
30.0    27
18.0    26
        ..
66.0     1
53.0     1
80.0     1
12.0     1
74.0     1
Name: count, Length: 71, dtype: int64
```

In [23]:
```python
def fun3(value):
    if (value < 20):
        return "teenager"
    elif (value >= 20 and value < 40):
        return "young"
    elif (value >= 40 and value < 60):
        return "middle aged"
    elif (value >= 60):
        return "senior citizen"
    else:
        pass
```

In [24]:
```python
df["age"] = df["age"].apply(fun3)
```

In [25]:
```python
df.head(10)
```

	passenger_id	survived	p_class	name	sex	age	sib_sp	parch	ticket	fare	cabin	embarke
0	1	0	3	Braund, Mr. Owen Harris	1	young	1	0	A/5 21171	7.2500	NaN	
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	0	young	1	0	PC 17599	71.2833	C85	
2	3	1	3	Heikkinen, Miss. Laina	0	young	0	0	STON/O2. 3101282	7.9250	NaN	
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	0	young	1	0	113803	53.1000	C123	
4	5	0	3	Allen, Mr. William Henry	1	young	0	0	373450	8.0500	NaN	
5	6	0	3	Moran, Mr. James	1	None	0	0	330877	8.4583	NaN	
6	7	0	1	McCarthy, Mr. Timothy J	1	middle aged	0	0	17463	51.8625	E46	
7	8	0	3	Palsson, Master. Gosta Leonard	1	teenager	3	1	349909	21.0750	NaN	
8	9	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	0	young	0	2	347742	11.1333	NaN	
9	10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	0	teenager	1	0	237736	30.0708	NaN	

In [ ]: