In [1]:
```
import pandas as pd
```

In [2]:
```
data=pd.read_csv("/home/placement/Downloads/Titanic Dataset.csv")
```

In [3]:
```
data
```

Out[3]:

	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
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows × 12 columns

In [4]:
```
data.tail()
```

Out[4]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.00	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.00	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.45	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.00	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.75	NaN	Q

In [5]:
```
data.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 [11]:
```
data.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 [12]:
```
list(data)
```

Out[12]:
```
['PassengerId',
 'Survived',
 'Pclass',
 'Name',
 'Sex',
 'Age',
 'SibSp',
 'Parch',
 'Ticket',
 'Fare',
 'Cabin',
 'Embarked']
```

In [13]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
df=pd.read_csv("/home/placement/Downloads/Titanic Dataset.csv")
```

```
In [20]: df.head
```

```
Out[20]: <bound method NDFrame.head of      PassengerId  Survived  Pclass  \
0              1         0       3
1              2         1       1
2              3         1       3
3              4         1       1
4              5         0       3
..           ...       ...     ...
886          887         0       2
887          888         1       1
888          889         0       3
889          890         1       1
890          891         0       3

                                                  Name     Sex   Age  SibSp  \
0                              Braund, Mr. Owen Harris    male  22.0      1
1    Cumings, Mrs. John Bradley (Florence Briggs Th...  female  38.0      1
2                               Heikkinen, Miss. Laina  female  26.0      0
3         Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                             Allen, Mr. William Henry    male  35.0      0
..                                                 ...     ...   ...    ...
886                              Montvila, Rev. Juozas    male  27.0      0
887                       Graham, Miss. Margaret Edith  female  19.0      0
888           Johnston, Miss. Catherine Helen "Carrie"  female   NaN      1
889                              Behr, Mr. Karl Howell    male  26.0      0
890                                Dooley, Mr. Patrick    male  32.0      0

     Parch            Ticket     Fare Cabin Embarked
0        0         A/5 21171   7.2500   NaN        S
1        0          PC 17599  71.2833   C85        C
2        0  STON/O2. 3101282   7.9250   NaN        S
3        0            113803  53.1000  C123        S
4        0            373450   8.0500   NaN        S
..     ...               ...      ...   ...      ...
886      0            211536  13.0000   NaN        S
887      0            112053  30.0000   B42        S
888      2        W./C. 6607  23.4500   NaN        S
889      0            111369  30.0000  C148        C
890      0            370376   7.7500   NaN        Q
```

```
[891 rows x 12 columns]>
```

In [21]:
```
data
```

Out[21]:

	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
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows × 12 columns

In [22]:
```
data["Pclass"].unique()
```

Out[22]:
```
array([3, 1, 2])
```

In [25]:
```
data["Survived"].unique()
```

Out[25]:
```
array([0, 1])
```

```
In [26]: data["Name"].unique()
```

```
       'Chibnall, Mrs. (Edith Martha Bowerman)',
       'Skoog, Mrs. William (Anna Bernhardina Karlsson)',
       'Baumann, Mr. John D', 'Ling, Mr. Lee',
       'Van der hoef, Mr. Wyckoff', 'Rice, Master. Arthur',
       'Johnson, Miss. Eleanor Ileen', 'Sivola, Mr. Antti Wilhelm',
       'Smith, Mr. James Clinch', 'Klasen, Mr. Klas Albin',
       'Lefebre, Master. Henry Forbes', 'Isham, Miss. Ann Elizabeth',
       'Hale, Mr. Reginald', 'Leonard, Mr. Lionel',
       'Sage, Miss. Constance Gladys', 'Pernot, Mr. Rene',
       'Asplund, Master. Clarence Gustaf Hugo',
       'Becker, Master. Richard F', 'Kink-Heilmann, Miss. Luise Gretchen',
       'Rood, Mr. Hugh Roscoe',
       'O\'Brien, Mrs. Thomas (Johanna "Hannah" Godfrey)',
       'Romaine, Mr. Charles Hallace ("Mr C Rolmane")',
       'Bourke, Mr. John', 'Turcin, Mr. Stjepan', 'Pinsky, Mrs. (Rosa)',
       'Carbines, Mr. William',
       'Andersen-Jensen, Miss. Carla Christine Nielsine',
       'Navratil, Master. Michel M',
       'Brown, Mrs. James Joseph (Margaret Tobin)',
       'Lurette, Miss. Elise', 'Mernagh, Mr. Robert',
```

```
In [27]: data["Sex"].unique()

Out[27]: array(['male', 'female'], dtype=object)
```

```
In [28]: data["Age"].unique()

Out[28]: array([22.  , 38.  , 26.  , 35.  ,   nan, 54.  ,  2.  , 27.  , 14.  ,
                 4.  , 58.  , 20.  , 39.  , 55.  , 31.  , 34.  , 15.  , 28.  ,
                 8.  , 19.  , 40.  , 66.  , 42.  , 21.  , 18.  ,  3.  ,  7.  ,
                49.  , 29.  , 65.  , 28.5 ,  5.  , 11.  , 45.  , 17.  , 32.  ,
                16.  , 25.  ,  0.83, 30.  , 33.  , 23.  , 24.  , 46.  , 59.  ,
                71.  , 37.  , 47.  , 14.5 , 70.5 , 32.5 , 12.  ,  9.  , 36.5 ,
                51.  , 55.5 , 40.5 , 44.  ,  1.  , 61.  , 56.  , 50.  , 36.  ,
                45.5 , 20.5 , 62.  , 41.  , 52.  , 63.  , 23.5 ,  0.92, 43.  ,
                60.  , 10.  , 64.  , 13.  , 48.  ,  0.75, 53.  , 57.  , 80.  ,
                70.  , 24.5 ,  6.  ,  0.67, 30.5 ,  0.42, 34.5 , 74.  ])
```

```
In [30]: data["SibSp"].unique()

Out[30]: array([1, 0, 3, 4, 2, 5, 8])
```

```
In [31]: data["Parch"].unique()

Out[31]: array([0, 1, 2, 5, 3, 4, 6])
```

```
In [32]: data["Ticket"].unique()
```

```
        '248740', '244361', '229236', '248733', '31418', '386525',
        'C.A. 37671', '315088', '7267', '113510', '2695', '2647', '345783',
        '237671', '330931', '330980', 'SC/PARIS 2167', '2691',
        'SOTON/O.Q. 3101310', 'C 7076', '110813', '2626', '14313',
        'PC 17477', '11765', '3101267', '323951', 'C 7077', '113503',
        '2648', '347069', 'PC 17757', '2653', 'STON/O 2. 3101293',
        '349227', '27849', '367655', 'SC 1748', '113760', '350034',
        '3101277', '350052', '350407', '28403', '244278', '240929',
        'STON/O 2. 3101289', '341826', '4137', '315096', '28664', '347064',
        '29106', '312992', '349222', '394140', 'STON/O 2. 3101269',
        '343095', '28220', '250652', '28228', '345773', '349254',
        'A/5. 13032', '315082', '347080', 'A/4. 34244', '2003', '250655',
        '364851', 'SOTON/O.Q. 392078', '110564', '376564', 'SC/AH 3085',
        'STON/O 2. 3101274', '13507', 'C.A. 18723', '345769', '347076',
        '230434', '65306', '33638', '113794', '2666', '113786', '65303',
        '113051', '17453', 'A/5 2817', '349240', '13509', '17464',
        'F.C.C. 13531', '371060', '19952', '364506', '111320', '234360',
        'A/S 2816', 'SOTON/O.Q. 3101306', '113792', '36209', '323592',
        '315089', 'SC/AH Basle 541', '7553', '31027', '3460', '350060',
        '3101298', '239854', 'A/5 3594', '4134', '11771', 'A.5. 18509',
        '65304', 'SOTON/OQ 3101317', '113787', 'PC 17609', 'A/4 45380',
In [33]: data["Fare"].unique()
```

```
Out[33]: array([  7.25  ,  71.2833,   7.925 ,  53.1   ,   8.05  ,   8.4583,
                 51.8625,  21.075 ,  11.1333,  30.0708,  16.7   ,  26.55  ,
                 31.275 ,   7.8542,  16.    ,  29.125 ,  13.    ,  18.    ,
                  7.225 ,  26.    ,   8.0292,  35.5   ,  31.3875, 263.    ,
                  7.8792,   7.8958,  27.7208, 146.5208,   7.75  ,  10.5   ,
                 82.1708,  52.    ,   7.2292,  11.2417,   9.475 ,  21.    ,
                 41.5792,  15.5   ,  21.6792,  17.8   ,  39.6875,   7.8   ,
                 76.7292,  61.9792,  27.75  ,  46.9   ,  80.    ,  83.475 ,
                 27.9   ,  15.2458,   8.1583,   8.6625,  73.5   ,  14.4542,
                 56.4958,   7.65  ,  29.    ,  12.475 ,   9.    ,   9.5   ,
                  7.7875,  47.1   ,  15.85  ,  34.375 ,  61.175 ,  20.575 ,
                 34.6542,  63.3583,  23.    ,  77.2875,   8.6542,   7.775 ,
                 24.15  ,   9.825 ,  14.4583, 247.5208,   7.1417,  22.3583,
                  6.975 ,   7.05  ,  14.5   ,  15.0458,  26.2833,   9.2167,
                 79.2   ,   6.75  ,  11.5   ,  36.75  ,   7.7958,  12.525 ,
                 66.6   ,   7.3125,  61.3792,   7.7333,  69.55  ,  16.1   ,
                 15.75  ,  20.525 ,  55.    ,  25.925 ,  33.5   ,  30.6958,
                 25.4667,  28.7125,   0.    ,  15.05  ,  39.    ,  22.025 ,
                 50.    ,   8.4042,   6.4958,  10.4625,  18.7875,  31.    ,
                113.275 ,  27.    ,  76.2917,  90.    ,   9.35  ,  13.5   ,
                  7.55  ,  26.25  ,  12.275 ,   7.125 ,  52.5542,  20.2125,
                 86.5   , 512.3292,  79.65  , 153.4625, 135.6333,  19.5   ,
                 29.7   ,  77.9583,  20.25  ,  78.85  ,  91.0792,  12.875 ,
                  8.85  , 151.55  ,  30.5   ,  23.25  ,  12.35  , 110.8833,
                108.9   ,  24.    ,  56.9292,  83.1583, 262.375 ,  14.    ,
                164.8667, 134.5   ,   6.2375,  57.9792,  28.5   , 133.65  ,
                 15.9   ,   9.225 ,  35.    ,  75.25  ,  69.3   ,  55.4417,
                211.5   ,   4.0125, 227.525 ,  15.7417,   7.7292,  12.    ,
                120.    ,  12.65  ,  18.75  ,   6.8583,  32.5   ,   7.875 ,
                 14.4   ,  55.9   ,   8.1125,  81.8583,  19.2583,  19.9667,
                 89.1042,  38.5   ,   7.725 ,  13.7917,   9.8375,   7.0458,
                  7.5208,  12.2875,   9.5875,  49.5042,  78.2667,  15.1   ,
                  7.6292,  22.525 ,  26.2875,  59.4   ,   7.4958,  34.0208,
                 93.5   , 221.7792, 106.425 ,  49.5   ,  71.    ,  13.8625,
                  7.8292,  39.6   ,  17.4   ,  51.4792,  26.3875,  30.    ,
                 40.125 ,   8.7125,  15.    ,  33.    ,  42.4   ,  15.55  ,
                 65.    ,  32.3208,   7.0542,   8.4333,  25.5875,   9.8417,
                  8.1375,  10.1708, 211.3375,  57.    ,  13.4167,   7.7417,
                  9.4833,   7.7375,   8.3625,  23.45  ,  25.9292,   8.6833,
                  8.5167,   7.8875,  37.0042,   6.45   ,   6.95   ,   8.3    ,
                  6.4375,  39.4    ,  14.1083,  13.8583,  50.4958,   5.     ,
                  9.8458,  10.5167])
```

```
In [34]: data["Cabin"].unique()
```

```
Out[34]: array([nan, 'C85', 'C123', 'E46', 'G6', 'C103', 'D56', 'A6',
                'C23 C25 C27', 'B78', 'D33', 'B30', 'C52', 'B28', 'C83', 'F33',
                'F G73', 'E31', 'A5', 'D10 D12', 'D26', 'C110', 'B58 B60', 'E101',
                'F E69', 'D47', 'B86', 'F2', 'C2', 'E33', 'B19', 'A7', 'C49', 'F4',
                'A32', 'B4', 'B80', 'A31', 'D36', 'D15', 'C93', 'C78', 'D35',
                'C87', 'B77', 'E67', 'B94', 'C125', 'C99', 'C118', 'D7', 'A19',
                'B49', 'D', 'C22 C26', 'C106', 'C65', 'E36', 'C54',
                'B57 B59 B63 B66', 'C7', 'E34', 'C32', 'B18', 'C124', 'C91', 'E40',
                'T', 'C128', 'D37', 'B35', 'E50', 'C82', 'B96 B98', 'E10', 'E44',
                'A34', 'C104', 'C111', 'C92', 'E38', 'D21', 'E12', 'E63', 'A14',
                'B37', 'C30', 'D20', 'B79', 'E25', 'D46', 'B73', 'C95', 'B38',
                'B39', 'B22', 'C86', 'C70', 'A16', 'C101', 'C68', 'A10', 'E68',
                'B41', 'A20', 'D19', 'D50', 'D9', 'A23', 'B50', 'A26', 'D48',
                'E58', 'C126', 'B71', 'B51 B53 B55', 'D49', 'B5', 'B20', 'F G63',
                'C62 C64', 'E24', 'C90', 'C45', 'E8', 'B101', 'D45', 'C46', 'D30',
                'E121', 'D11', 'E77', 'F38', 'B3', 'D6', 'B82 B84', 'D17', 'A36',
                'B102', 'B69', 'E49', 'C47', 'D28', 'E17', 'A24', 'C50', 'B42',
                'C148'], dtype=object)
```

```
In [39]: data["Embarked"].unique()
```

```
Out[39]: array(['S', 'C', 'Q', nan], dtype=object)
```

```
In [43]: data1=data.drop(['PassengerId','Cabin','Name','Ticket','SibSp','Parch'],axis=1)
```

In [44]:
```
data1.head(10)
```

Out[44]:

	Survived	Pclass	Sex	Age	Fare	Embarked
0	0	3	male	22.0	7.2500	S
1	1	1	female	38.0	71.2833	C
2	1	3	female	26.0	7.9250	S
3	1	1	female	35.0	53.1000	S
4	0	3	male	35.0	8.0500	S
5	0	3	male	NaN	8.4583	Q
6	0	1	male	54.0	51.8625	S
7	0	3	male	2.0	21.0750	S
8	1	3	female	27.0	11.1333	S
9	1	2	female	14.0	30.0708	C

In [45]:
```
data1['Sex']=data1['Sex'].map({'male':1,'female':0})
```

In [48]:
```
data2=data1.fillna(data.median())#
```

```
/tmp/ipykernel_5146/1516598085.py:1: FutureWarning: The default value of numeric_only in DataFrame.median i
s deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is
deprecated. Select only valid columns or specify the value of numeric_only to silence this warning.
  data2=data1.fillna(data.median())#
```

In [49]:

```
data2.isna().sum()
```

Out[49]:

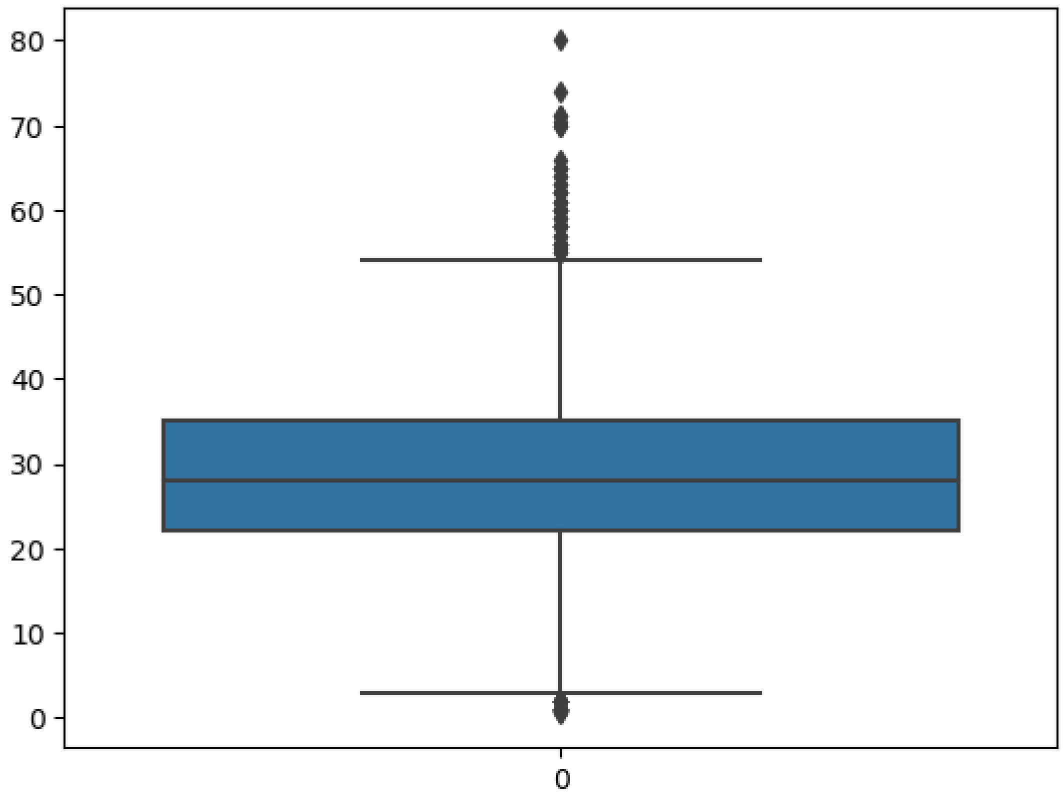
```
Survived    0
Pclass      0
Sex         0
Age         0
Fare        0
Embarked    2
dtype: int64
```

In [50]:

```
import seaborn as sns
import matplotlib.pyplot as plt
sns.boxplot(data2.Age)
```

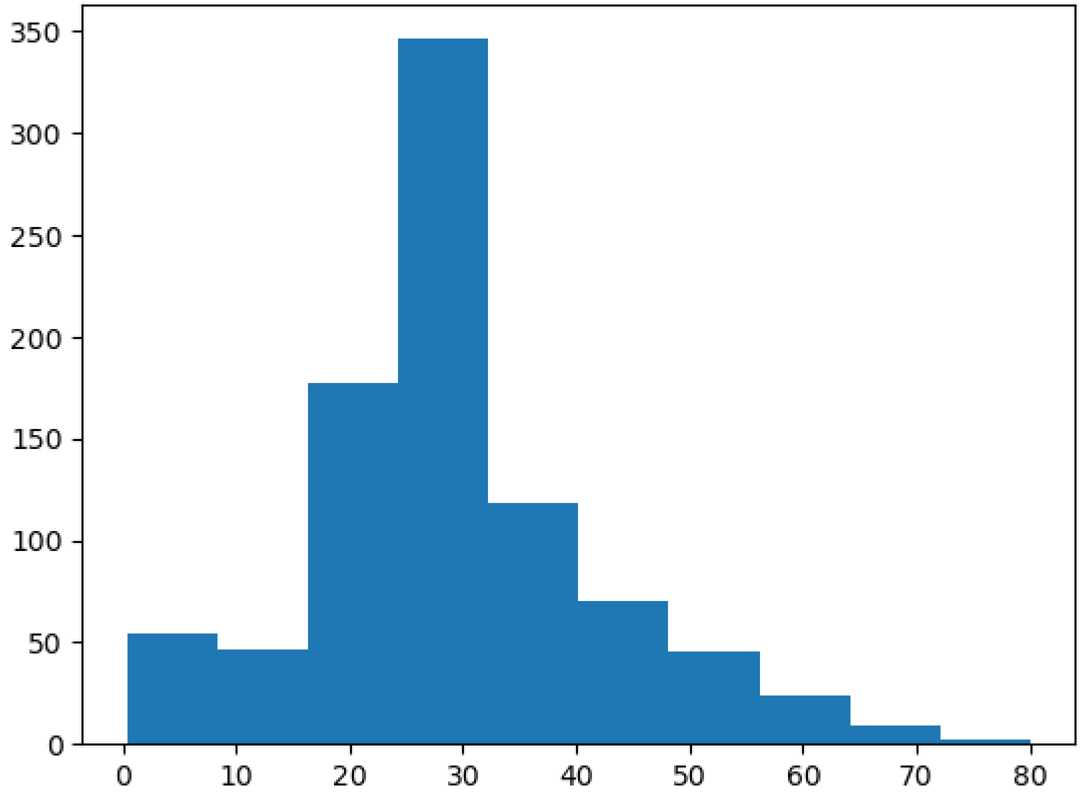
Out[50]:

```
<Axes: >
```



```
In [51]: plt.hist(data2['Age'])
```

```
Out[51]: (array([ 54.,  46., 177., 346., 118.,  70.,  45.,  24.,   9.,   2.]),
          array([ 0.42 ,  8.378, 16.336, 24.294, 32.252, 40.21 , 48.168, 56.126,
                 64.084, 72.042, 80.   ]),
          <BarContainer object of 10 artists>)
```

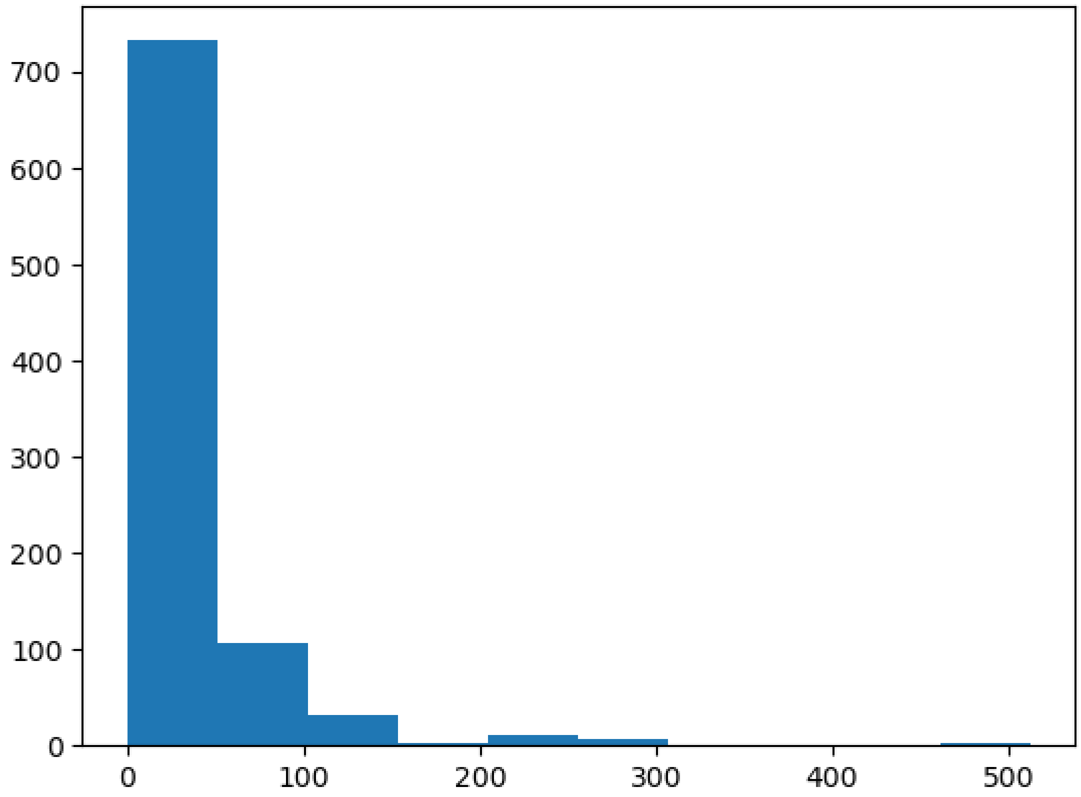


In [52]:

```
plt.hist(data2['Fare'])
```

Out[52]:

```
(array([732., 106.,  31.,   2.,  11.,   6.,   0.,   0.,   0.,   3.]),
 array([  0.     ,  51.23292, 102.46584, 153.69876, 204.93168, 256.1646 ,
        307.39752, 358.63044, 409.86336, 461.09628, 512.3292 ]),
 <BarContainer object of 10 artists>)
```



In [53]:
```
data2.describe()
```

Out[53]:

	Survived	Pclass	Sex	Age	Fare
count	891.000000	891.000000	891.000000	891.000000	891.000000
mean	0.383838	2.308642	0.647587	29.361582	32.204208
std	0.486592	0.836071	0.477990	13.019697	49.693429
min	0.000000	1.000000	0.000000	0.420000	0.000000
25%	0.000000	2.000000	0.000000	22.000000	7.910400
50%	0.000000	3.000000	1.000000	28.000000	14.454200
75%	1.000000	3.000000	1.000000	35.000000	31.000000
max	1.000000	3.000000	1.000000	80.000000	512.329200

In [54]:
```
data2['Age'].unique()
```

Out[54]:
```
array([22.  , 38.  , 26.  , 35.  , 28.  , 54.  ,  2.  , 27.  , 14.  ,
        4.  , 58.  , 20.  , 39.  , 55.  , 31.  , 34.  , 15.  ,  8.  ,
       19.  , 40.  , 66.  , 42.  , 21.  , 18.  ,  3.  ,  7.  , 49.  ,
       29.  , 65.  , 28.5 ,  5.  , 11.  , 45.  , 17.  , 32.  , 16.  ,
       25.  ,  0.83, 30.  , 33.  , 23.  , 24.  , 46.  , 59.  , 71.  ,
       37.  , 47.  , 14.5 , 70.5 , 32.5 , 12.  ,  9.  , 36.5 , 51.  ,
       55.5 , 40.5 , 44.  ,  1.  , 61.  , 56.  , 50.  , 36.  , 45.5 ,
       20.5 , 62.  , 41.  , 52.  , 63.  , 23.5 ,  0.92, 43.  , 60.  ,
       10.  , 64.  , 13.  , 48.  ,  0.75, 53.  , 57.  , 80.  , 70.  ,
       24.5 ,  6.  ,  0.67, 30.5 ,  0.42, 34.5 , 74.  ])
```

```
In [55]: data2.groupby(['Age']).count()
```

```
Out[55]:
         Survived  Pclass  Sex  Fare  Embarked
   Age
  0.42          1       1    1     1         1
  0.67          1       1    1     1         1
  0.75          2       2    2     2         2
  0.83          2       2    2     2         2
  0.92          1       1    1     1         1
   ...        ...     ...  ...   ...       ...
 70.00          2       2    2     2         2
 70.50          1       1    1     1         1
 71.00          2       2    2     2         2
 74.00          1       1    1     1         1
 80.00          1       1    1     1         1

88 rows × 5 columns
```

In [56]:
```
data2.head(10)
```

Out[56]:

	Survived	Pclass	Sex	Age	Fare	Embarked
0	0	3	1	22.0	7.2500	S
1	1	1	0	38.0	71.2833	C
2	1	3	0	26.0	7.9250	S
3	1	1	0	35.0	53.1000	S
4	0	3	1	35.0	8.0500	S
5	0	3	1	28.0	8.4583	Q
6	0	1	1	54.0	51.8625	S
7	0	3	1	2.0	21.0750	S
8	1	3	0	27.0	11.1333	S
9	1	2	0	14.0	30.0708	C

In [57]:

```
data2['Pclass']=data2['Pclass'].map({1:'F',2:'S',3:'Third'})
data2.head(10)
```

Out[57]:

	Survived	Pclass	Sex	Age	Fare	Embarked
0	0	Third	1	22.0	7.2500	S
1	1	F	0	38.0	71.2833	C
2	1	Third	0	26.0	7.9250	S
3	1	F	0	35.0	53.1000	S
4	0	Third	1	35.0	8.0500	S
5	0	Third	1	28.0	8.4583	Q
6	0	F	1	54.0	51.8625	S
7	0	Third	1	2.0	21.0750	S
8	1	Third	0	27.0	11.1333	S
9	1	S	0	14.0	30.0708	C

In [58]:

```
data2=pd.get_dummies(data2)
data2.head()
```

Out[58]:

	Survived	Sex	Age	Fare	Pclass_F	Pclass_S	Pclass_Third	Embarked_C	Embarked_Q	Embarked_S
0	0	1	22.0	7.2500	0	0	1	0	0	1
1	1	0	38.0	71.2833	1	0	0	1	0	0
2	1	0	26.0	7.9250	0	0	1	0	0	1
3	1	0	35.0	53.1000	1	0	0	0	0	1
4	0	1	35.0	8.0500	0	0	1	0	0	1

In [59]:

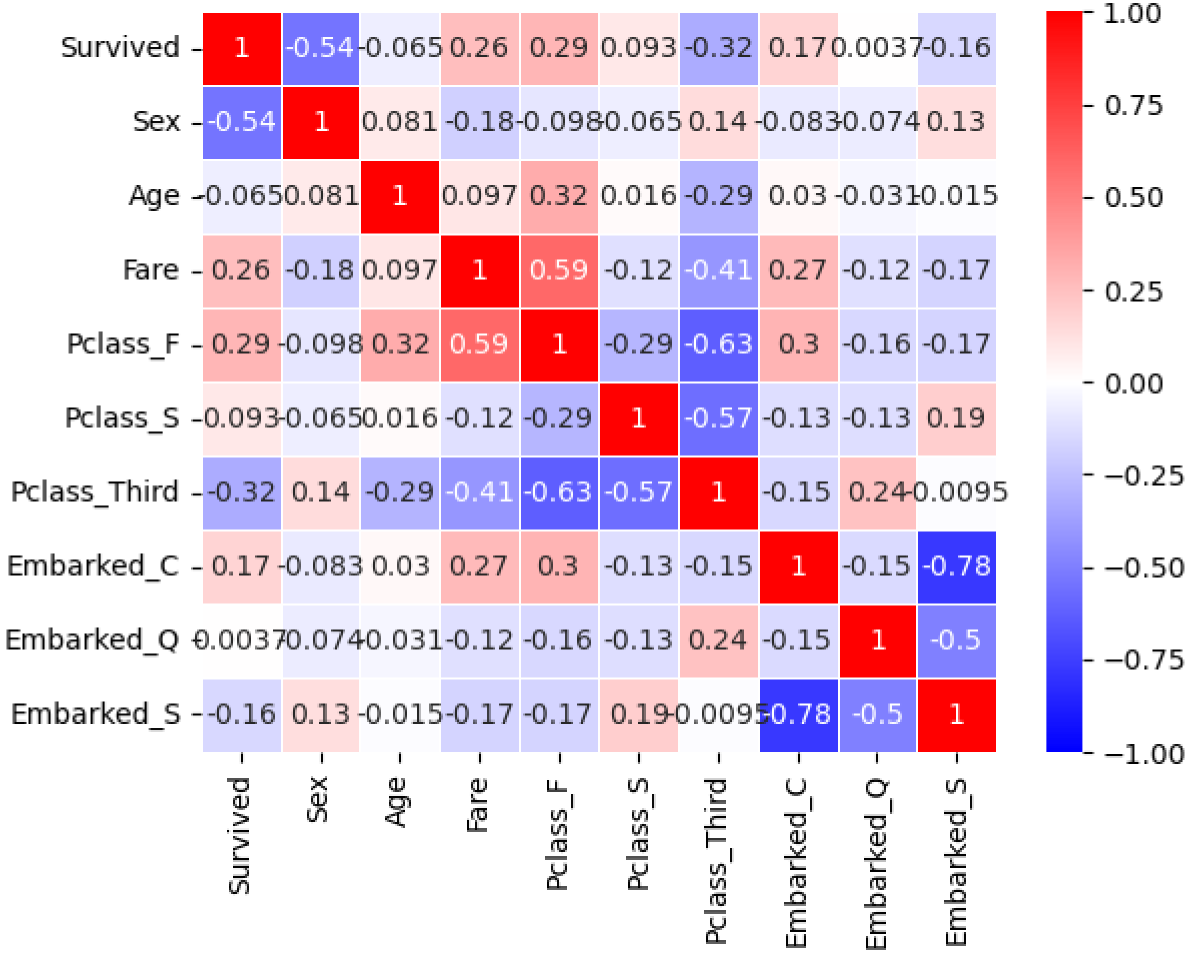
```
cor_mat=data2.corr()
cor_mat
```

Out[59]:

	Survived	Sex	Age	Fare	Pclass_F	Pclass_S	Pclass_Third	Embarked_C	Embarked_Q	Embarked_S
Survived	1.000000	-0.543351	-0.064910	0.257307	0.285904	0.093349	-0.322308	0.168240	0.003650	-0.155660
Sex	-0.543351	1.000000	0.081163	-0.182333	-0.098013	-0.064746	0.137143	-0.082853	-0.074115	0.125722
Age	-0.064910	0.081163	1.000000	0.096688	0.323896	0.015831	-0.291955	0.030248	-0.031415	-0.014665
Fare	0.257307	-0.182333	0.096688	1.000000	0.591711	-0.118557	-0.413333	0.269335	-0.117216	-0.166603
Pclass_F	0.285904	-0.098013	0.323896	0.591711	1.000000	-0.288585	-0.626738	0.296423	-0.155342	-0.170379
Pclass_S	0.093349	-0.064746	0.015831	-0.118557	-0.288585	1.000000	-0.565210	-0.125416	-0.127301	0.192061
Pclass_Third	-0.322308	0.137143	-0.291955	-0.413333	-0.626738	-0.565210	1.000000	-0.153329	0.237449	-0.009511
Embarked_C	0.168240	-0.082853	0.030248	0.269335	0.296423	-0.125416	-0.153329	1.000000	-0.148258	-0.778359
Embarked_Q	0.003650	-0.074115	-0.031415	-0.117216	-0.155342	-0.127301	0.237449	-0.148258	1.000000	-0.496624
Embarked_S	-0.155660	0.125722	-0.014665	-0.166603	-0.170379	0.192061	-0.009511	-0.778359	-0.496624	1.000000

In [60]:
```
import seaborn as sns
sns.heatmap(cor_mat,vmax=1,vmin=-1,annot=True,linewidths=.5,cmap='bwr')
```

Out[60]:
```
<Axes: >
```



In [61]:
```
data2.groupby('Survived').count()
```

Out[61]:

	Sex	Age	Fare	Pclass_F	Pclass_S	Pclass_Third	Embarked_C	Embarked_Q	Embarked_S
Survived									
0	549	549	549	549	549	549	549	549	549
1	342	342	342	342	342	342	342	342	342

In [63]:
```
y=data2['Survived']
X=data2.drop('Survived',axis=1)
```

In [64]:
```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.33,random_state=42)
```

In [65]:

```
from sklearn.linear_model import LogisticRegression
classifier=LogisticRegression()
classifier.fit(X_train,y_train)
```

```
/home/placement/anaconda3/lib/python3.10/site-packages/sklearn/linear_model/_logistic.py:458: ConvergenceWa
rning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html (https://scikit-learn.org/stable/modules/pre
processing.html)
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression (https://scikit-learn.or
g/stable/modules/linear_model.html#logistic-regression)
  n_iter_i = _check_optimize_result(
```

Out[65]:

```
▾ LogisticRegression

LogisticRegression()
```

In [66]:

```
y_pred=classifier.predict(X_test)
```

In [67]:

```
y_pred
```

Out[67]:

```
array([0, 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0,
       1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0,
       1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 1,
       0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1,
       0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0,
       1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 0,
       0, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 1,
       0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0,
       0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 1, 0,
       1, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0,
       0, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 1,
       0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0,
       0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0,
       1, 0, 0, 0, 0, 0, 1, 1, 0])
```

In [ ]: