In [43]:
```
import pandas as pd
```

In [44]:
```
data=pd.read_csv("/home/placement/Downloads/Titanic Dataset.csv")
```

In [45]:
```
data.describe()
```

Out[45]:

| | 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 [46]:
```
data.head(5)
```

Out[46]:

| | 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 [47]:
```
data.isna().sum()
```

Out[47]:
```
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 [48]:
```
data.head(10)
```

Out[48]:

| | 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 |
| **5** | 6 | 0 | 3 | Moran, Mr. James | male | NaN | 0 | 0 | 330877 | 8.4583 | NaN | Q |
| **6** | 7 | 0 | 1 | McCarthy, Mr. Timothy J | male | 54.0 | 0 | 0 | 17463 | 51.8625 | E46 | S |
| **7** | 8 | 0 | 3 | Palsson, Master. Gosta Leonard | male | 2.0 | 3 | 1 | 349909 | 21.0750 | NaN | S |
| **8** | 9 | 1 | 3 | Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg) | female | 27.0 | 0 | 2 | 347742 | 11.1333 | NaN | S |
| **9** | 10 | 1 | 2 | Nasser, Mrs. Nicholas (Adele Achem) | female | 14.0 | 1 | 0 | 237736 | 30.0708 | NaN | C |

In [49]:
```
data['Survived'].unique()
```

Out[49]:
```
array([0, 1])
```

In [50]:
```
data['SibSp'].unique()
```

Out[50]:
```
array([1, 0, 3, 4, 2, 5, 8])
```

In [51]:
```
data['Age'].unique()
```

Out[51]:
```
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 [52]:
```
data1=data.drop(['PassengerId','Ticket','Cabin','Name','SibSp','Parch'],axis=1)
```

In [53]:
```
data1
```

Out[53]:

| | 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 |
| **...** | ... | ... | ... | ... | ... | ... |
| **886** | 0 | 2 | male | 27.0 | 13.0000 | S |
| **887** | 1 | 1 | female | 19.0 | 30.0000 | S |
| **888** | 0 | 3 | female | NaN | 23.4500 | S |
| **889** | 1 | 1 | male | 26.0 | 30.0000 | C |
| **890** | 0 | 3 | male | 32.0 | 7.7500 | Q |

891 rows × 6 columns

In [54]:
```
data1.fillna(35,inplace=True)
```

In [55]:
```
data1
```

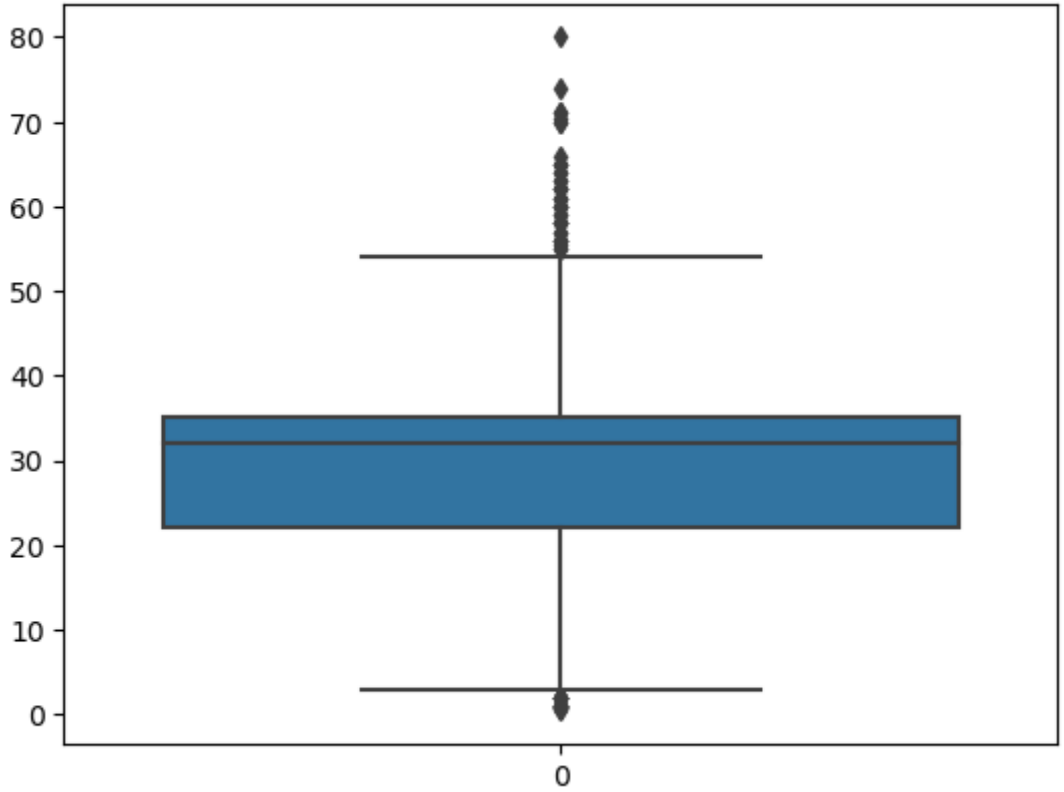
Out[55]:

| | 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 |
| **...** | ... | ... | ... | ... | ... | ... |
| **886** | 0 | 2 | male | 27.0 | 13.0000 | S |
| **887** | 1 | 1 | female | 19.0 | 30.0000 | S |
| **888** | 0 | 3 | female | 35.0 | 23.4500 | S |
| **889** | 1 | 1 | male | 26.0 | 30.0000 | C |
| **890** | 0 | 3 | male | 32.0 | 7.7500 | Q |

891 rows × 6 columns

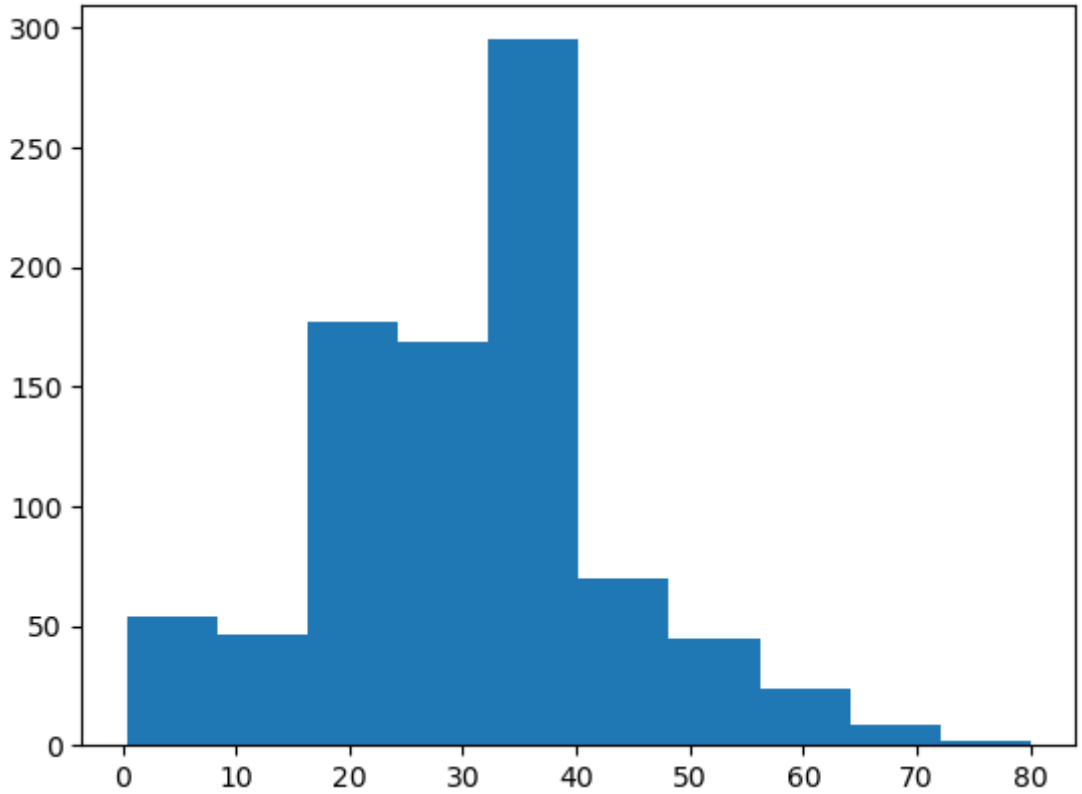
In [56]:
```
import seaborn as sns
import matplotlib.pyplot as plt
sns.boxplot (data1.Age)
```

Out[56]: <Axes: >



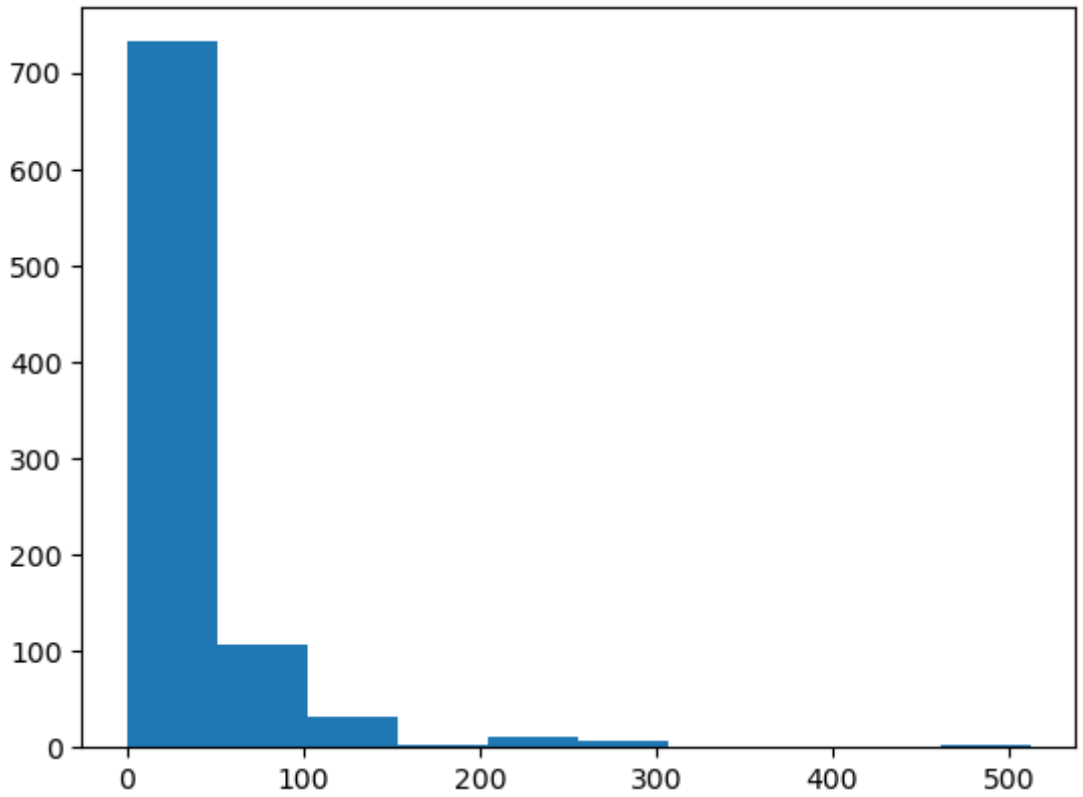
In [57]:
```
plt.hist(data1['Age'])
```

Out[57]:
```
(array([ 54.,  46., 177., 169., 295.,  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 [58]:
```
plt.hist(data1['Fare'])
```

Out[58]:
```
(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 [59]:
```
data1.describe()
```

Out[59]:

| | Survived | Pclass | Age | Fare |
|---|---|---|---|---|
| **count** | 891.000000 | 891.000000 | 891.000000 | 891.000000 |
| **mean** | 0.383838 | 2.308642 | 30.752155 | 32.204208 |
| **std** | 0.486592 | 0.836071 | 13.173100 | 49.693429 |
| **min** | 0.000000 | 1.000000 | 0.420000 | 0.000000 |
| **25%** | 0.000000 | 2.000000 | 22.000000 | 7.910400 |
| **50%** | 0.000000 | 3.000000 | 32.000000 | 14.454200 |
| **75%** | 1.000000 | 3.000000 | 35.000000 | 31.000000 |
| **max** | 1.000000 | 3.000000 | 80.000000 | 512.329200 |

In [60]:
```
data1['Age'].unique()
```

Out[60]:
```
array([22.  , 38.  , 26.  , 35.  , 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 [61]:
```
data1.groupby(['Age']).count()
```

Out[61]:

| Age | Survived | Pclass | Sex | Fare | Embarked |
|---|---|---|---|---|---|
| **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 [62]:
```
data1['Pclass']=data1['Pclass'].map({1:'F',2:'S',3:'Third'})
```

In [63]:
```
data1.isna().sum()
```

Out[63]:
```
Survived    0
Pclass      0
Sex         0
Age         0
Fare        0
Embarked    0
dtype: int64
```

In [64]:
```
data1=pd.get_dummies(data1)
```

In [65]:
```
data1.shape
```

Out[65]: (891, 12)

In [66]:
```
data1.head(500)
```

Out[66]:

| | Survived | Age | Fare | Pclass_F | Pclass_S | Pclass_Third | Sex_female | Sex_male | Embarked_35 | Embarked_C | Embarked_Q | Embarked_S |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 22.0 | 7.2500 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| **1** | 1 | 38.0 | 71.2833 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| **2** | 1 | 26.0 | 7.9250 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| **3** | 1 | 35.0 | 53.1000 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| **4** | 0 | 35.0 | 8.0500 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **495** | 0 | 35.0 | 14.4583 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| **496** | 1 | 54.0 | 78.2667 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| **497** | 0 | 35.0 | 15.1000 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| **498** | 0 | 25.0 | 151.5500 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| **499** | 0 | 24.0 | 7.7958 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |

500 rows × 12 columns

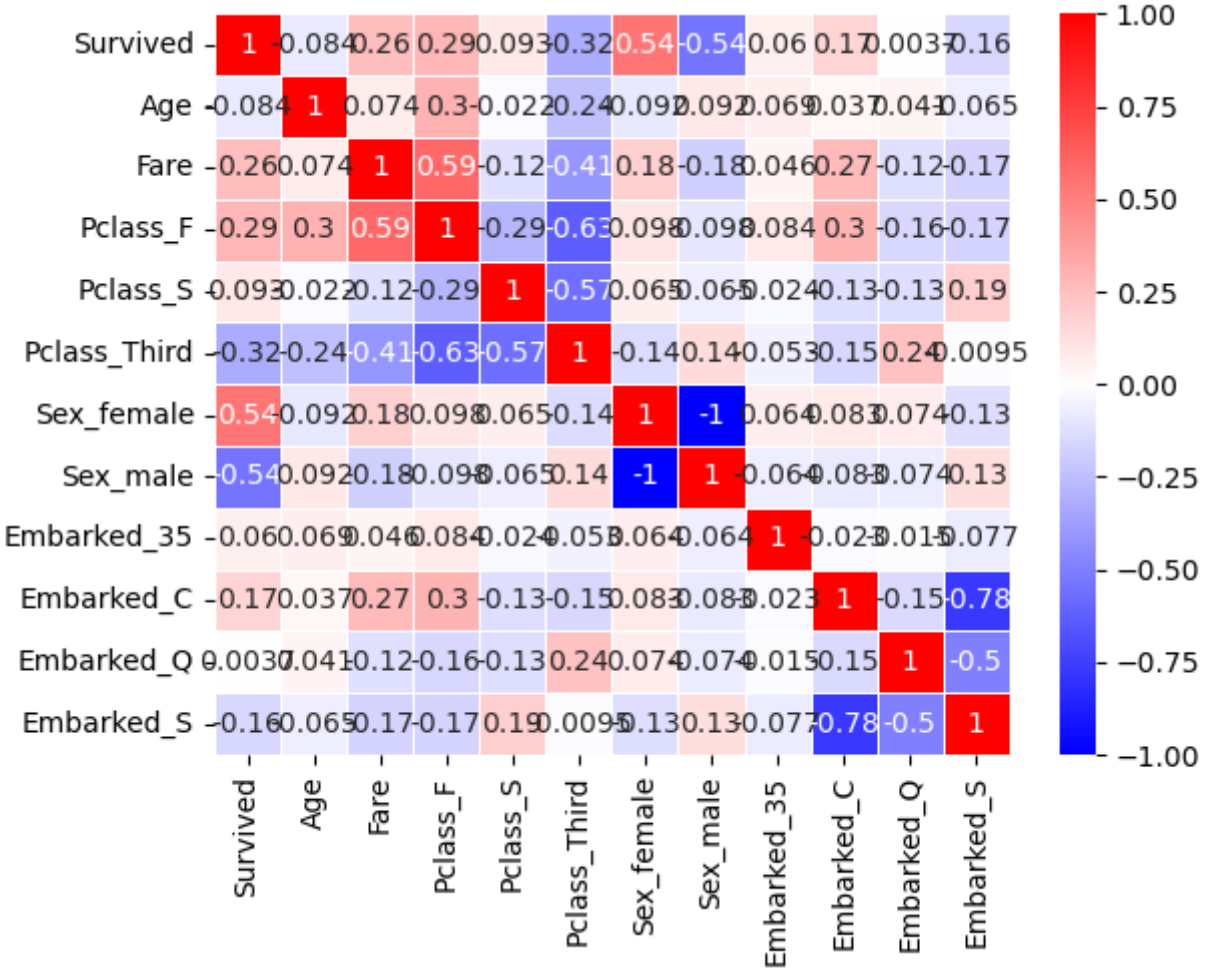
In [67]:
```
cor_mat=data1.corr()
cor_mat
```

Out[67]:

| | Survived | Age | Fare | Pclass_F | Pclass_S | Pclass_Third | Sex_female | Sex_male | Embarked_35 | Embarked_C | Embarked_ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Survived** | 1.000000 | -0.083713 | 0.257307 | 0.285904 | 0.093349 | -0.322308 | 0.543351 | -0.543351 | 0.060095 | 0.168240 | 0.00365 |
| **Age** | -0.083713 | 1.000000 | 0.074199 | 0.302149 | -0.022021 | -0.242412 | -0.091930 | 0.091930 | 0.069343 | 0.036953 | 0.04052 |
| **Fare** | 0.257307 | 0.074199 | 1.000000 | 0.591711 | -0.118557 | -0.413333 | 0.182333 | -0.182333 | 0.045646 | 0.269335 | -0.11721 |
| **Pclass_F** | 0.285904 | 0.302149 | 0.591711 | 1.000000 | -0.288585 | -0.626738 | 0.098013 | -0.098013 | 0.083847 | 0.296423 | -0.15534 |
| **Pclass_S** | 0.093349 | -0.022021 | -0.118557 | -0.288585 | 1.000000 | -0.565210 | 0.064746 | -0.064746 | -0.024197 | -0.125416 | -0.12730 |
| **Pclass_Third** | -0.322308 | -0.242412 | -0.413333 | -0.626738 | -0.565210 | 1.000000 | -0.137143 | 0.137143 | -0.052550 | -0.153329 | 0.23744 |
| **Sex_female** | 0.543351 | -0.091930 | 0.182333 | 0.098013 | 0.064746 | -0.137143 | 1.000000 | -1.000000 | 0.064296 | 0.082853 | 0.07411 |
| **Sex_male** | -0.543351 | 0.091930 | -0.182333 | -0.098013 | -0.064746 | 0.137143 | -1.000000 | 1.000000 | -0.064296 | -0.082853 | -0.07411 |
| **Embarked_35** | 0.060095 | 0.069343 | 0.045646 | 0.083847 | -0.024197 | -0.052550 | 0.064296 | -0.064296 | 1.000000 | -0.022864 | -0.01458 |
| **Embarked_C** | 0.168240 | 0.036953 | 0.269335 | 0.296423 | -0.125416 | -0.153329 | 0.082853 | -0.082853 | -0.022864 | 1.000000 | -0.14825 |
| **Embarked_Q** | 0.003650 | 0.040528 | -0.117216 | -0.155342 | -0.127301 | 0.237449 | 0.074115 | -0.074115 | -0.014588 | -0.148258 | 1.00000 |
| **Embarked_S** | -0.155660 | -0.065062 | -0.166603 | -0.170379 | 0.192061 | -0.009511 | -0.125722 | 0.125722 | -0.076588 | -0.778359 | -0.49662 |

In [68]:
```
sns.heatmap(cor_mat,vmax=1,vmin=-1,annot=True,linewidths=.5,cmap='bwr')
```

Out[68]: <Axes: >



In [69]:
```
data.groupby('Survived').count()
```

Out[69]:

| Survived | PassengerId | Pclass | Name | Sex | Age | SibSp | Parch | Ticket | Fare | Cabin | Embarked |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 549 | 549 | 549 | 549 | 424 | 549 | 549 | 549 | 549 | 68 | 549 |
| **1** | 342 | 342 | 342 | 342 | 290 | 342 | 342 | 342 | 342 | 136 | 340 |

In [70]:
```
y=data1['Survived']
x=data1.drop('Survived',axis=1)
```

In [71]:
```
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 [73]:
```
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[73]:
```
LogisticRegression()
```
**In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.**
**On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.**

In [74]:
```
y_pred=classifier.predict(x_test)
```

In [75]:
```
y_pred
```

Out[75]:
```
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, 0, 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, 1, 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 [76]:
```
from sklearn.metrics import confusion_matrix
confusion_matrix(y_test,y_pred)
```

Out[76]:
```
array([[155,  20],
       [ 36,  84]])
```

In [78]:
```
from sklearn.metrics import accuracy_score
accuracy_score(y_test,y_pred)
```

Out[78]:
```
0.8101694915254237
```

In [79]:
```
y
```

Out[79]:
```
0      0
1      1
2      1
3      1
4      0
      ..
886    0
887    1
888    0
889    1
890    0
Name: Survived, Length: 891, dtype: int64
```

In [ ]:
```

```