22/7/25 Preprocessing using pandas and simple imputer.

AIM:

To load titanic dataset from csv, handle missing values using simple imputer, analyze key passenger features, filter passenger based on candidates, and prepare data for model training and testing.

Procedure /Algorithm:

Step 1: load titanic.csv into a dataframe

Step 2: Explore dataset shape, info and summary statistics.

Step 3: use simple Imputer to fill missing Age.

Step 4: Fill missing cabin with "unknown" and embaced with mode.

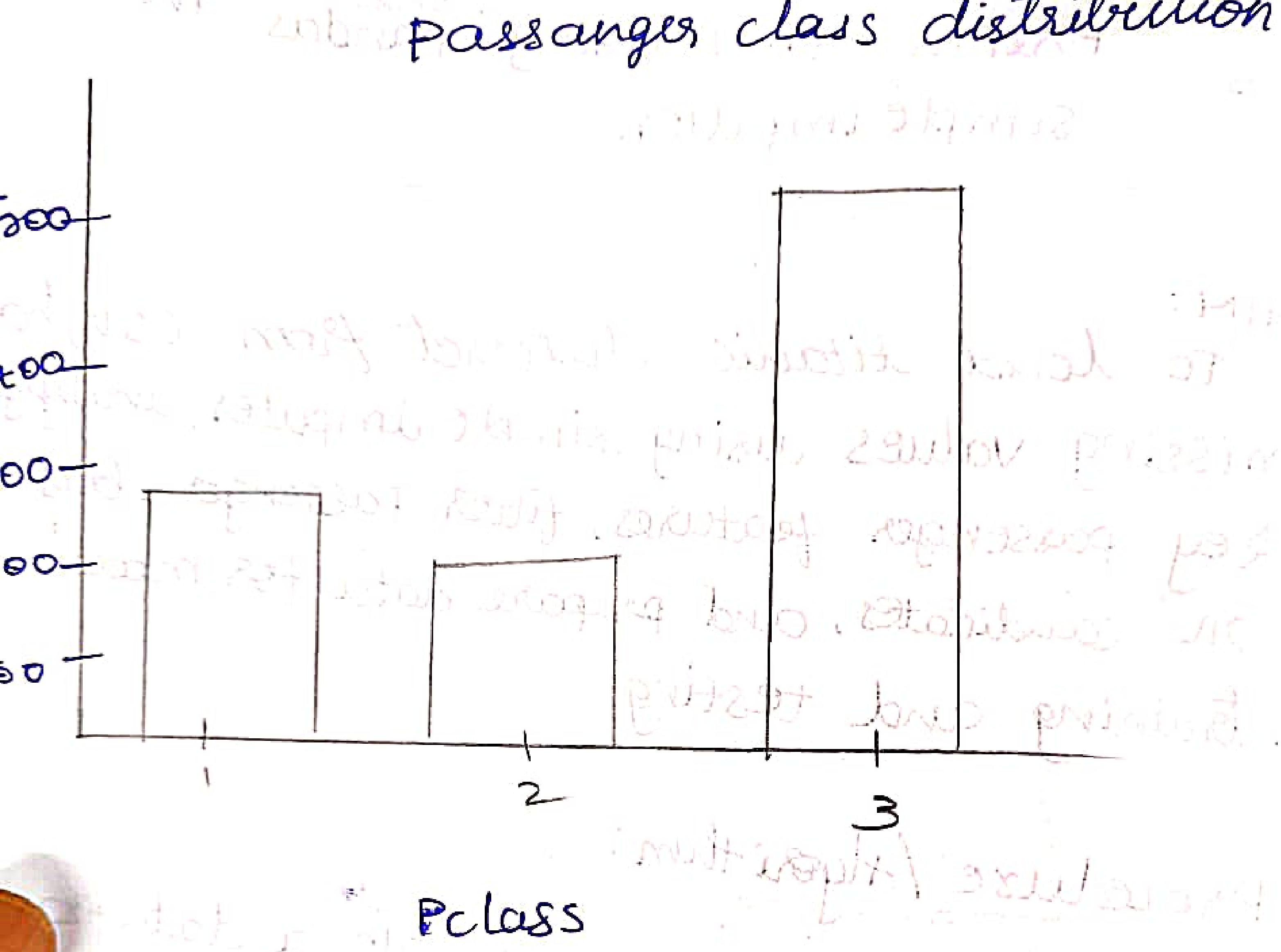
Step 5: visualize passenger class.

Step 6: Filter passangers by genders, Survival, class, age fav embancation, family abroad, and survival status.

Step 7: Identity top oldest survivors and zeros - fare passangers

Step 8: Split training and testing sets.

output:



PROGRAM:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.impute import Simple Imputer
from sklearn.model_selection import train_
                                    test_split

df = sns.load_dataset ('titanic')
df['age'] = Simple Imputer (strategy = 'mean')
            fit_transform (df[['age']])
df['deck'] = df['deck'].cate.add_categories
                                   ('unknown')
df['deck] = df['deck'].fillna('unknown')
df['embanced'] = df['embanced'].fillna
      (df['embanced'].mode()[0])
sns.countplot (x='pclass', data=df) plt.title
      ('passanger class distribution')
  plt.show()
Print ("Females who survived:", df[(df.
   sex = 'female') & (df.survived ==1)].inde
                             .tolist()
Print(" 3rd class passangers under 18:",
   df[(df.pclass ==3) & (df.age<18)].
                    index.tolist()
```

passangers who paid zero tax : 15 passangers

Training set size :712

Testing set size =179

```
Print (" (1st class passangers older than "
    df[(df. pclass ==1) & (df.age>40)].
                              index .to list())

Print (" 1st class passangers older than 40
    who survived: ", df [(df. Pclass ==1) &(df age>
    40) & (df-survived ==1) ].index. to list())
```

RESULT:

The Program successfully indentifies passangers with zero fax and efficiently splits the datasets into 80% training and 20% testing sets, ensuring reproducibility and readiness for machine learning tasks.