```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

df=pd.read_csv(r"C:\Users\ASUS\Documents\pythonStack\DS_PR\
titanic.csv")
df.head()
```

```
   survived  pclass     sex   age  sibsp  parch     fare embarked
class  \
0          0       3    male  22.0      1      0   7.2500        S
Third
1          1       1  female  38.0      1      0  71.2833        C
First
2          1       3  female  26.0      0      0   7.9250        S
Third
3          1       1  female  35.0      1      0  53.1000        S
First
4          0       3    male  35.0      0      0   8.0500        S
Third

     who  adult_male  deck  embark_town alive  alone
0    man        True   NaN  Southampton    no  False
1  woman       False     C    Cherbourg   yes  False
2  woman       False   NaN  Southampton   yes   True
3  woman       False     C  Southampton   yes  False
4    man        True   NaN  Southampton    no   True
```

```python
(df['fare']<50 ).value_counts()
```

```
fare
True     730
False    161
Name: count, dtype: int64
```

```python
# Step 3: Plot the histogram for the 'fare' column to see its
distribution
# kde=True adds a Kernel Density Estimate line for smoothness
# kde=True adds a smooth curve (Kernel Density Estimate) over the
histogram
# 1. Helps visualize the underlying probability distribution of the
data
# 2. Smooths out noise in histogram by estimating density across
values
# 3. Makes it easier to detect patterns like skewness or multiple
peaks
# 4. Useful for understanding how values are concentrated (e.g., most
fares are low)
# 5. Complements the histogram by showing a continuous distribution
line
```

```
sns.histplot(df['fare'], kde=True, bins=10)
plt.title('Distribution of Ticket Fare for Titanic Passengers')  # Set
the title
plt.xlabel('Fare')  # Label for x-axis
plt.ylabel('Frequency')  # Label for y-axis

Text(0, 0.5, 'Frequency')
```


Distribution of Ticket Fare for Titanic Passengers