TITANIC DATASET ANALYSIS

About the dataset

The Titanic dataset is a popular dataset in the field of machine learning and data analysis. It contains information about the passengers aboard the RMS Titanic, which sank on its maiden voyage in 1912 after hitting an iceberg. The dataset is often used for practicing and demonstrating various data analysis and machine learning techniques.

The dataset typically includes information such as:

1. **PassengerID:** A unique identifier for each passenger.
2. **Survived:** A binary variable indicating whether a passenger survived (1) or did not survive (0).
3. **Pclass (Ticket class):** The class of the ticket the passenger purchased (1st, 2nd, or 3rd).
4. **Name:** The name of the passenger.
5. **Sex:** The gender of the passenger.
6. **Age:** The age of the passenger.
7. **SibSp:** The number of siblings/spouses the passenger had aboard.
8. **Parch:** The number of parents/children the passenger had aboard.
9. **Ticket:** The ticket number.
10. **Fare:** The amount of money the passenger paid for the ticket.
11. **Cabin:** The cabin number where the passenger stayed.
12. **Embarked:** The port where the passenger boarded the Titanic (C = Cherbourg, Q = Queenstown, S = Southampton).

The goal when working with the Titanic dataset is often to predict whether a passenger survived based on the other features. This can be approached as a binary classification problem in machine learning.

The dataset is commonly used for educational purposes, allowing learners to apply various data preprocessing, exploration, and modeling techniques. It has also been used for competitions on platforms like Kaggle, where participants build predictive models based on the dataset to improve their skills in data science and machine learning.

Importing Libraries

In [73]:
```
import sys
import numpy as np
import pandas as pd
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
sns.set_style('darkgrid')
matplotlib.rcParams['font.size'] = 14
matplotlib.rcParams['figure.figsize'] = (13, 5)
matplotlib.rcParams['figure.facecolor'] = '#00000000'
pd.set_option('display.max_columns', 100)
from pandas.plotting import parallel_coordinates
sns.set(style='whitegrid', font_scale=1.3, color_codes=True)
import warnings
warnings.filterwarnings('ignore')
```

Importing Dataset

In [74]:
```
Titanic_data = pd.read_excel("titanic_data.xlsx")
Titanic_data.head()
```

Out[74]:

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Fare	Embarked	survived
0	892	3	Kelly, Mr. James	male	34	0	0	7.8292	Q	0
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47	1	0	7.0000	S	1
2	894	2	Myles, Mr. Thomas Francis	male	62	0	0	9.6875	Q	0
3	895	3	Wirz, Mr. Albert	male	27	0	0	8.6625	S	0
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22	1	1	12.2875	S	1

In this analysis, we will utilize a distplot to visually represent the distribution of ages among the Titanic passengers. A histogram is a graphical representation of the distribution of a dataset, where data is divided into bins, and the frequency of occurrences in each bin is represented by the height of bars. This method allows us to observe patterns, trends, and central tendencies within the age data.By examining the age distribution, we may uncover patterns such as the prevalence of certain age groups, potential outliers, or gaps in the data.

Statistical Analysis of the dataset

In [75]:
```
Titanic_data.describe()
```

Out[75]:

	PassengerId	Pclass	Age	SibSp	Parch	Fare	survived
count	417.000000	417.000000	417.000000	417.000000	417.000000	417.000000	417.000000
mean	1100.635492	2.263789	29.597122	0.448441	0.393285	35.627188	0.364508
std	120.923774	0.842077	12.616793	0.897568	0.982419	55.907576	0.481870
min	892.000000	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000
25%	996.000000	1.000000	23.000000	0.000000	0.000000	7.895800	0.000000
50%	1101.000000	3.000000	27.000000	0.000000	0.000000	14.454200	0.000000
75%	1205.000000	3.000000	35.000000	1.000000	0.000000	31.500000	1.000000
max	1309.000000	3.000000	76.000000	8.000000	9.000000	512.329200	1.000000

Let's interpret the summary statistics provided for the numeric columns in your dataset:

1. **PassengerId:**
   - Count: There are 417 entries in this dataset.
   - Mean: The average PassengerId is approximately 1101.
   - Standard Deviation (std): The standard deviation is 120.92, indicating a spread of PassengerId around the mean.
2. **Pclass (Passenger Class):**
   - Count: All 417 passengers have a Pclass value.
   - Mean: The average Pclass is approximately 2.26.
   - Std: The standard deviation is around 0.84, suggesting some variation in the passenger class.
3. **Age:**
   - Count: There are 417 entries for the Age column.
   - Mean: The average age is approximately 29.60.
   - Std: The standard deviation is about 12.62, indicating a spread in the ages.
4. **SibSp (Number of Siblings/Spouses Aboard):**
   - Count: All 417 passengers have a SibSp value.
   - Mean: The average number of siblings/spouses aboard is about 0.45.
   - Std: The standard deviation is approximately 0.90.
5. **Parch (Number of Parents/Children Aboard):**
   - Count: All 417 passengers have a Parch value.
   - Mean: The average number of parents/children aboard is about 0.39.
   - Std: The standard deviation is around 0.98.
6. **Fare:**
   - Count: There are 417 entries for the Fare column.
   - Mean: The average fare is approximately 35.63.
   - Std: The standard deviation is quite high at 55.91, indicating a wide range of fares.
7. **Survived:**
   - Count: All 417 passengers have a survival status.
   - Mean: The average survival rate is approximately 0.36.
   - Std: The standard deviation is 0.48, indicating variation in survival status.

Interpretation:

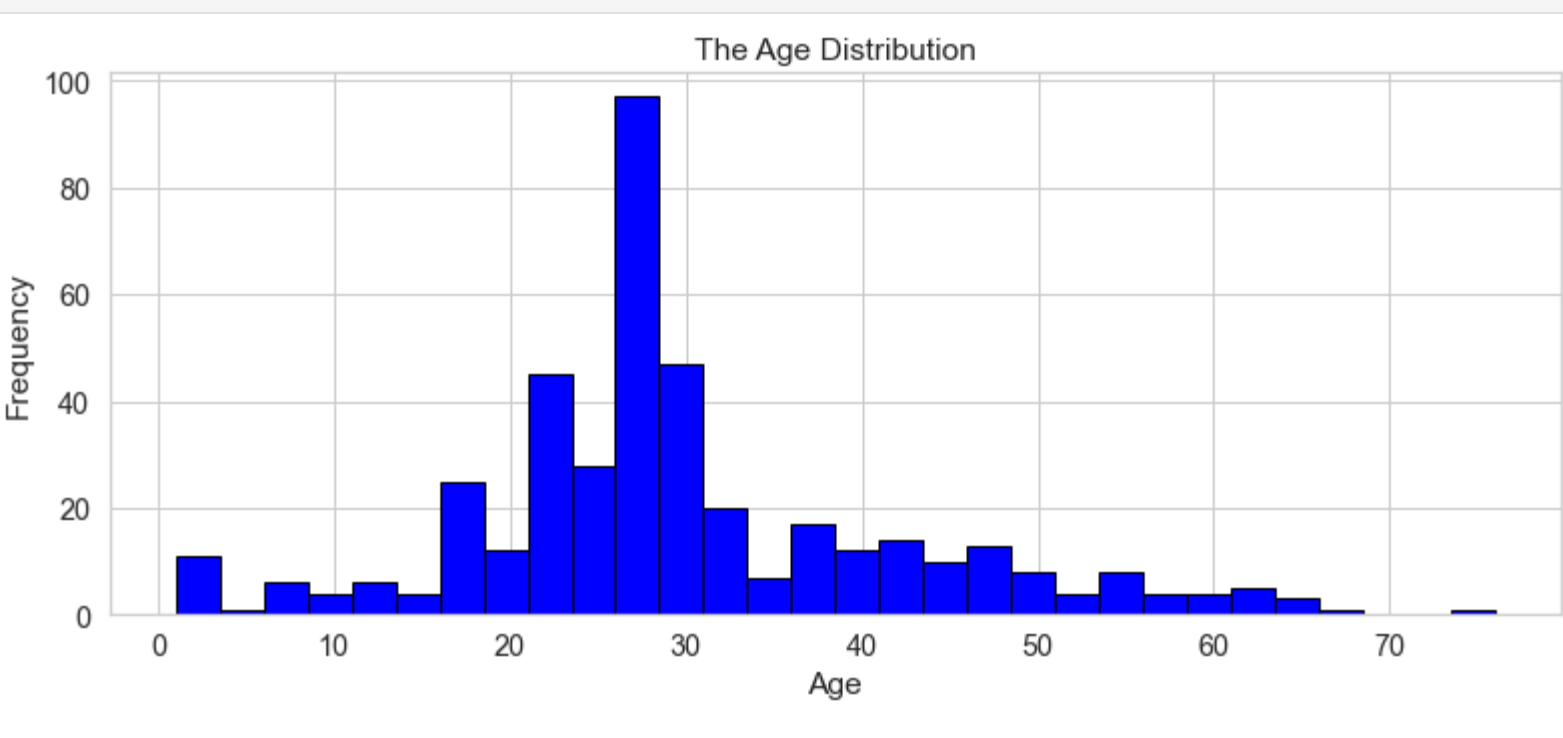
- The average age of passengers is around 29.60, with a standard deviation of 12.62, suggesting a spread in ages.
- Most passengers have a Pclass around 2.26, indicating a mix of second and third-class passengers.
- The average number of siblings/spouses and parents/children aboard is relatively low.
- The average fare is about 35.63, with a wide range of fares as indicated by the high standard deviation.
- The average survival rate is approximately 0.36, suggesting that, on average, about 36% of passengers survived.

These statistics provide a summary of the central tendency, spread, and distribution of the numeric variables in your dataset.

Both datasets contain following variables:

Pclass - Ticket class - a proxy for socio-economic status (SES) 1 - Upper 2 - Middle 3 - Lower Sex SibSp - # of siblings/spouses aboard the Titanic Parch - # of parents/children aboard the Titanic Ticket - Ticket number Fare - Passenger fare Cabin - Cabin Number Embarked - Port of embarktion: C - Cherbourg Q - Queenstown S - Southampton

In [76]:
```
plt.hist(Titanic_data['Age'], bins=30, color='blue', edgecolor='black')
plt.title('The Age Distribution')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.show()
```



The reason for the age analysis could be that: **Younger individuals, especially those in their 20s and 30s, might have been more likely to travel, whether for work, leisure, or immigration.**The Titanic voyage could have attracted a higher number of young and adventurous individuals. **Social and cultural trends of the time might have influenced who was more likely to embark on such a journey. Younger individuals might have been more inclined to travel for exploration or other reasons.

For analysis purpose let's change some numerical value to categorical value

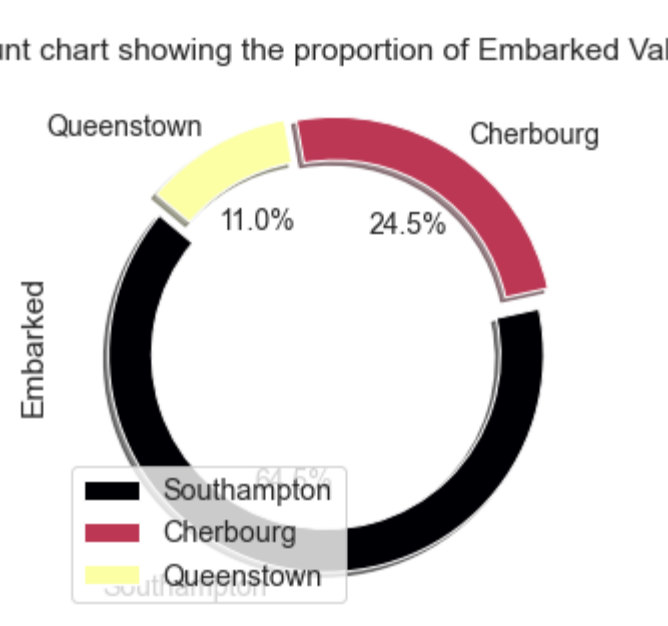
In [77]:
```
class_mapping = {1: 'Upper', 2: 'Middle', 3: 'Lower'}
Titanic_data['Pclass'] = Titanic_data['Pclass'].map(class_mapping)
```

In [78]:
```
Embarked_mapping = {'C' : 'Cherbourg', 'Q' : 'Queenstown', 'S' : 'Southampton'}
Titanic_data['Embarked'] = Titanic_data['Embarked'].map(Embarked_mapping)
```

In [79]:
```
Survived_mapping = {1: 'Yes', 0: 'No'}
Titanic_data['survived'] = Titanic_data['survived'].map(Survived_mapping)
```

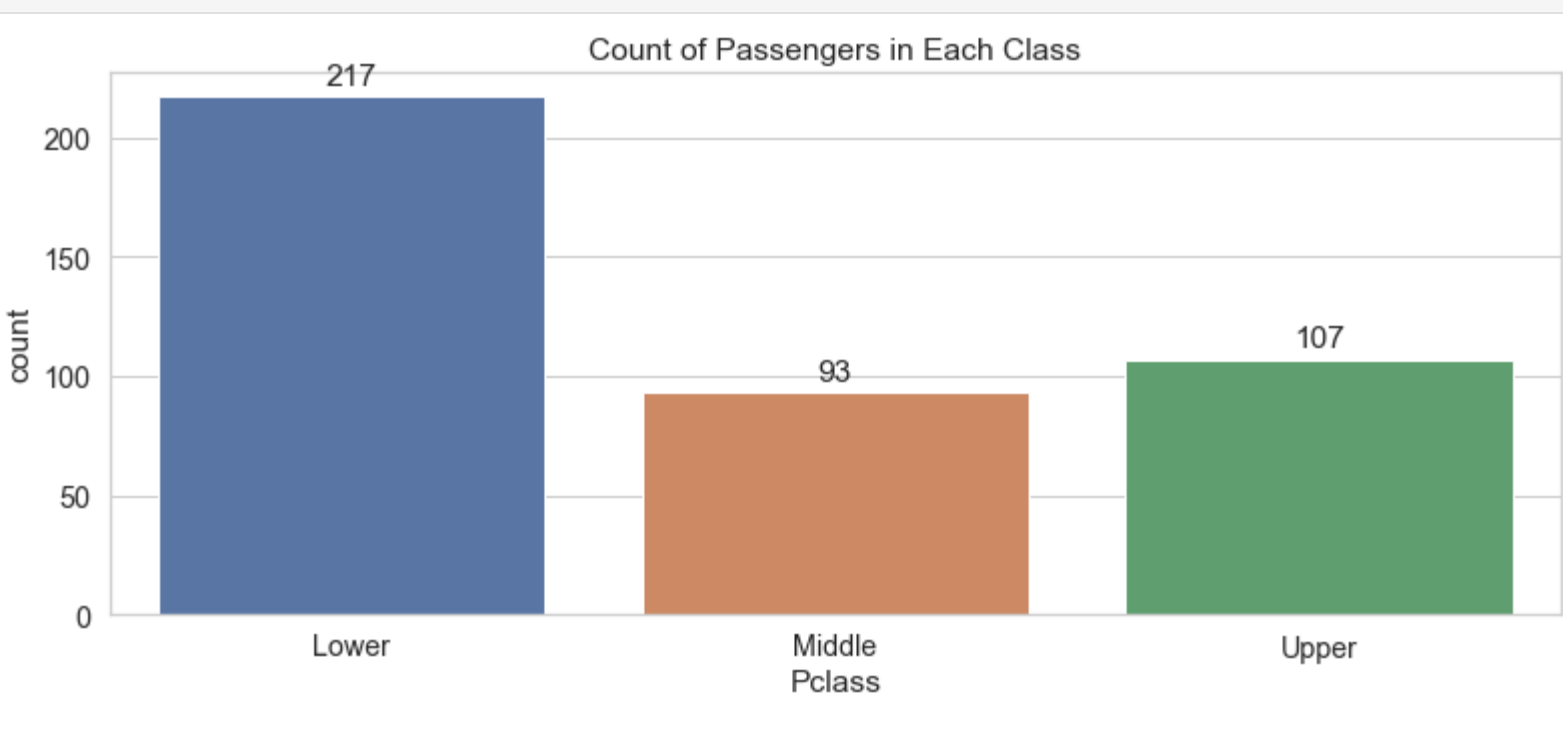
In [98]:
```
Titanic_data['Embarked'].value_counts().plot(kind='pie', explode=np.ones(3)/19, autopct='%3.1f%%', wedgeprops=dict(width=0.2), shadow=True, startangle=140,
                                    cmap='inferno', fontsize=14, legend=True)
plt.title("donunt chart showing the proportion of Embarked Values")
```

Out[98]:
```
Text(0.5, 1.0, 'donunt chart showing the proportion of Embarked Values')
```



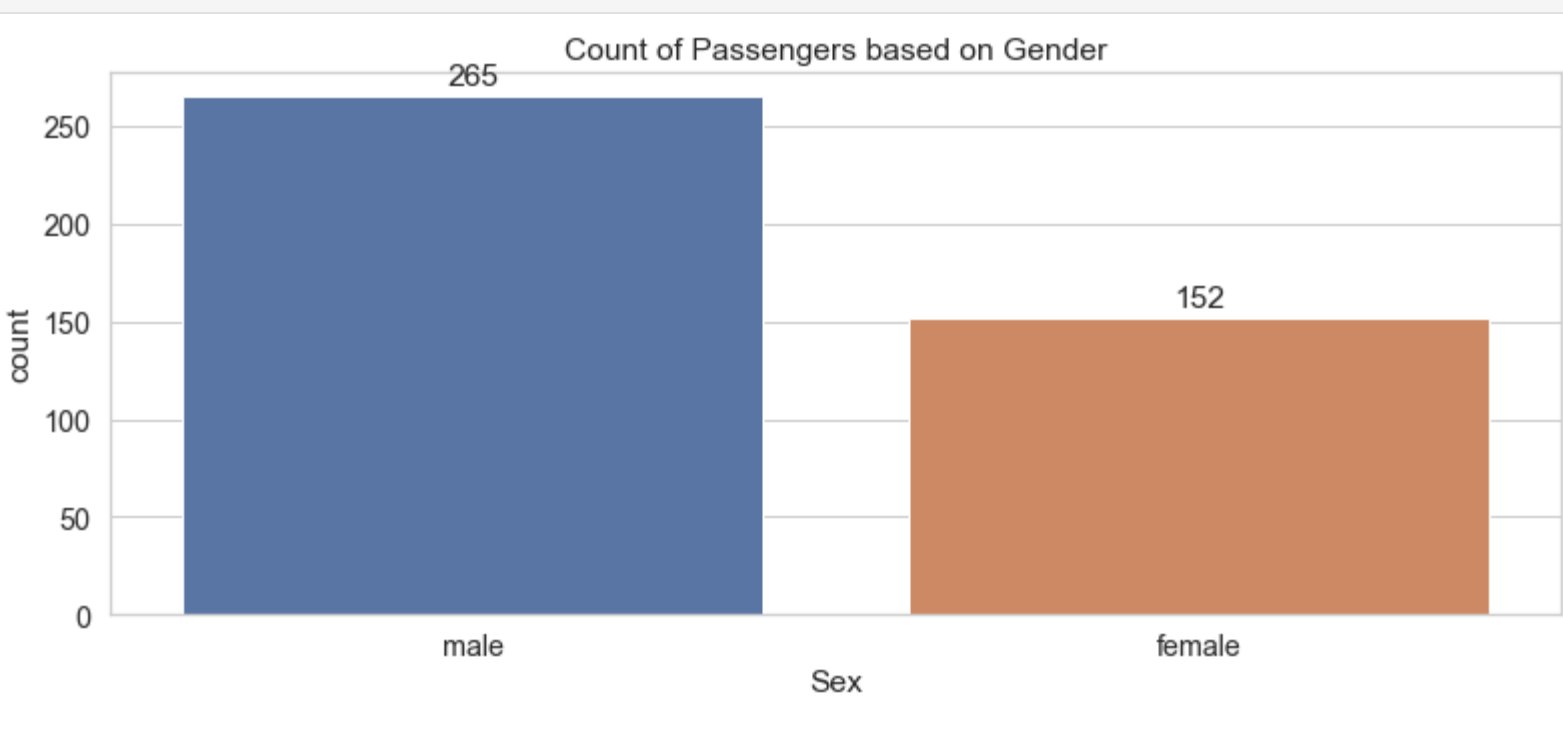
In [81]:
```
ax=sns.countplot(data=Titanic_data, x='Pclass')
for p in ax.patches:
    ax.annotate(f'{p.get_height()}', (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='center', xytext=(0, 10), textcoords='offset points')


plt.title('Count of Passengers in Each Class')
plt.show()
```



In [82]:
```
ax=sns.countplot(data=Titanic_data, x='Sex')
for p in ax.patches:
    ax.annotate(f'{p.get_height()}', (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='center', xytext=(0, 10), textcoords='offset points')


plt.title('Count of Passengers based on Gender')
plt.show()
```



The names are arranged in useful format, with titles included, so extraction of titles should give some insight.

In [83]:
```
Titanic_data['Name'].head()
```

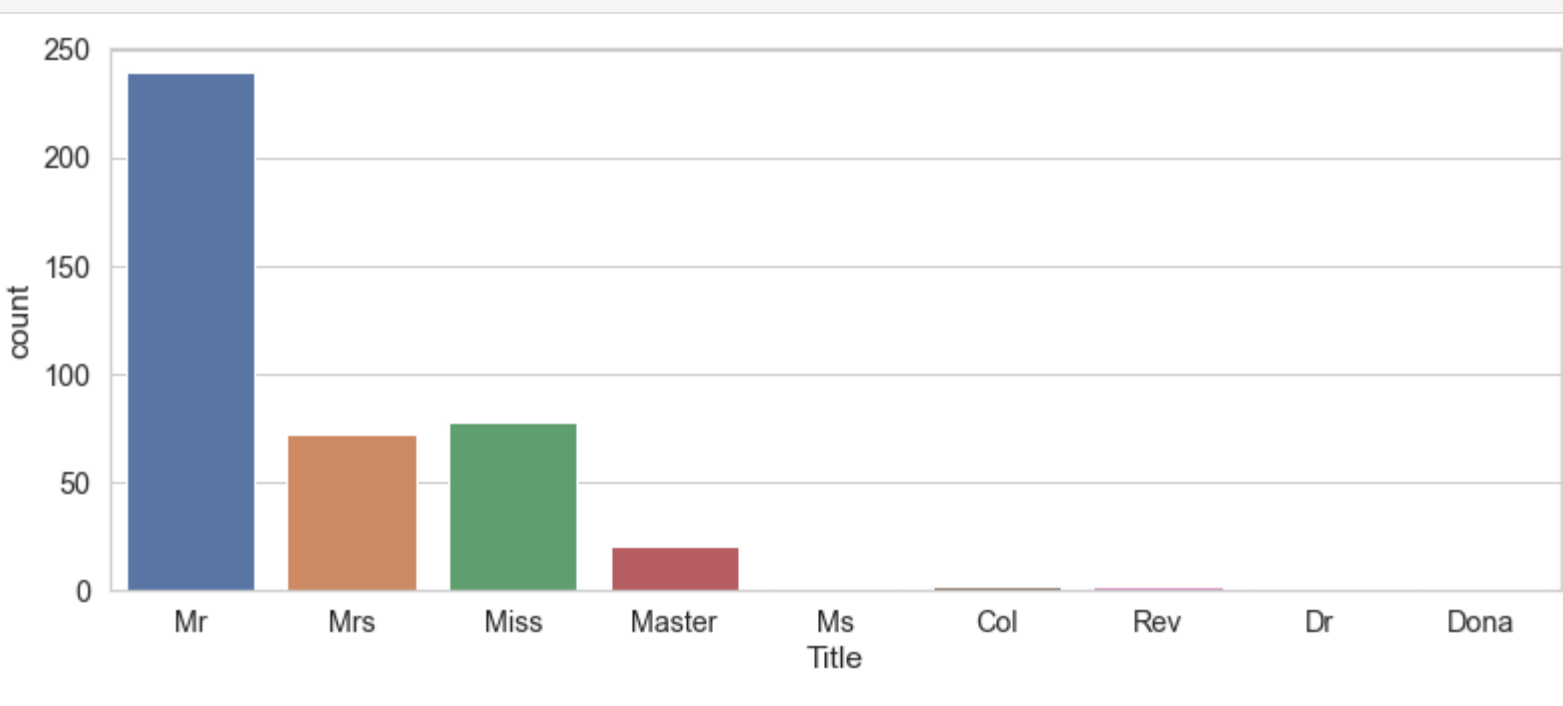
Out[83]:
```
0                              Kelly, Mr. James
1              Wilkes, Mrs. James (Ellen Needs)
2                     Myles, Mr. Thomas Francis
3                              Wirz, Mr. Albert
4    Hirvonen, Mrs. Alexander (Helga E Lindqvist)
Name: Name, dtype: object
```

In [84]:
```
# Adding a Title column based on Name column for both datasets.
Titanic_data['Title'] = Titanic_data['Name'].apply(lambda x: x.split(', ')[1].split('.')[0])
```

In [85]:
```
Titanic_data['Title'].value_counts()
```

Out[85]:
```
Mr        239
Miss       78
Mrs        72
Master     21
Col         2
Rev         2
Ms          1
Dr          1
Dona        1
Name: Title, dtype: int64
```

In [86]:
```
ax=sns.countplot(data=Titanic_data, x='Title')
```



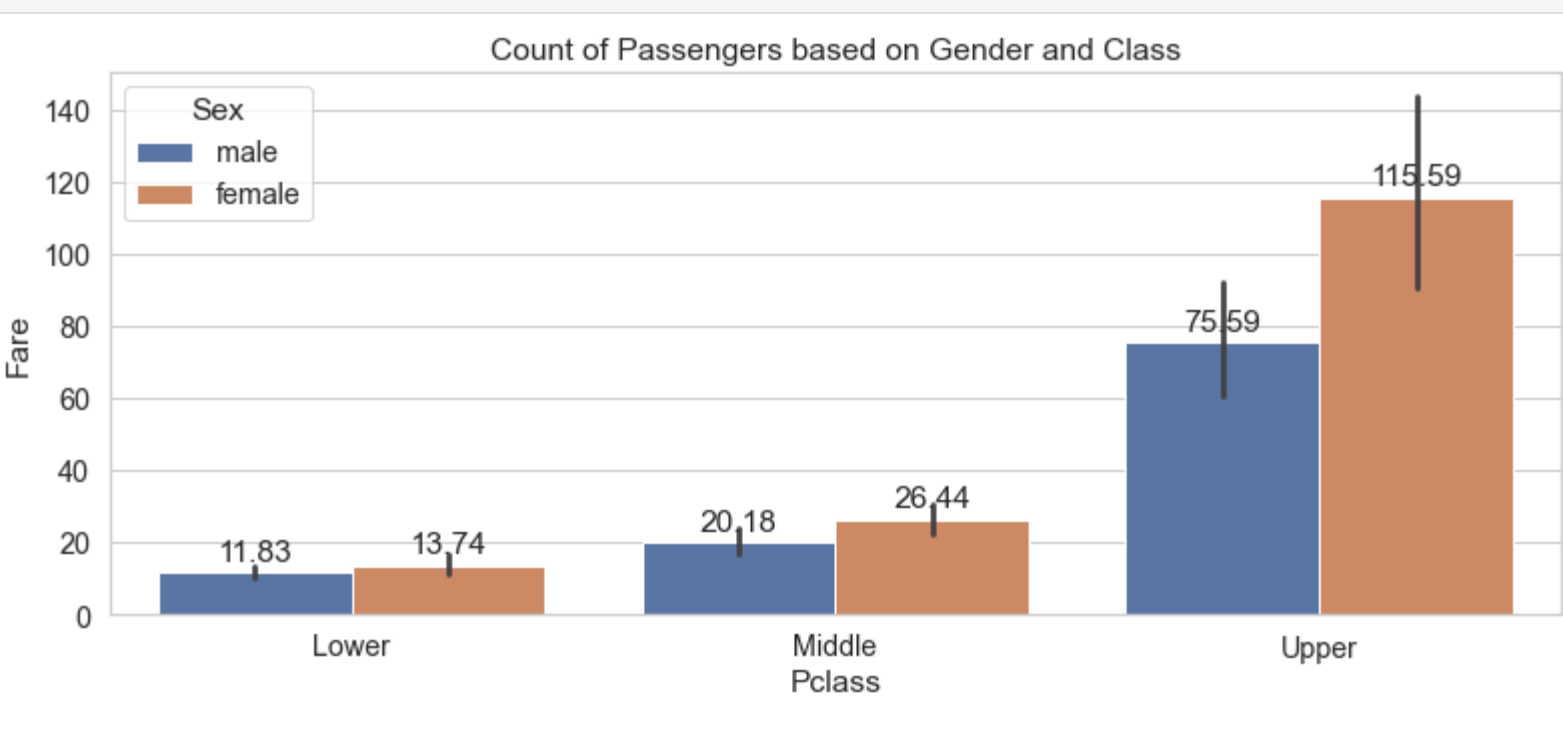
The Average Age based on Pclass

In [87]:
```
Titanic_data.groupby('Pclass').mean('Age')['Age']
```

Out[87]:
```
Pclass
Lower     24.944700
Middle    28.688172
Upper     39.822430
Name: Age, dtype: float64
```

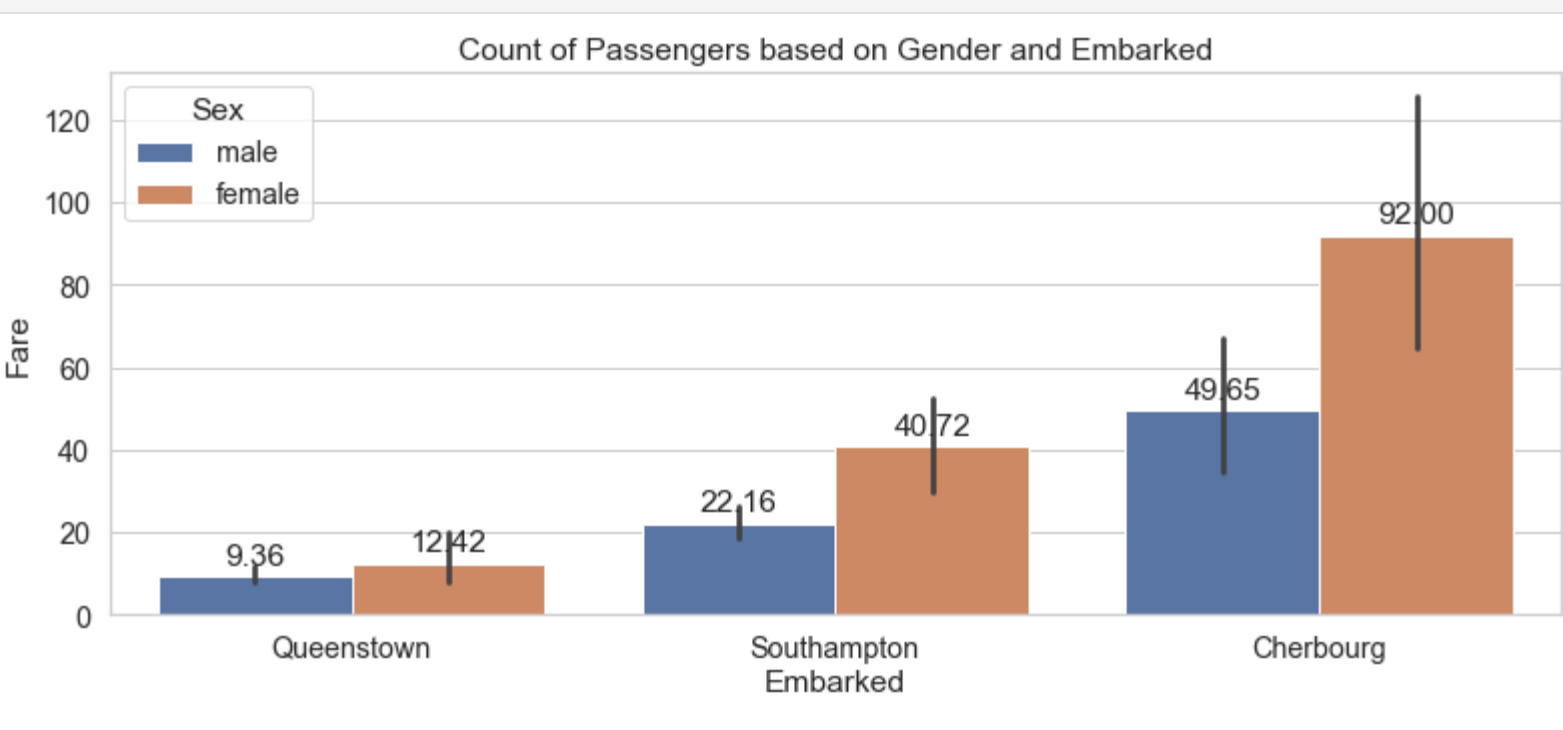
In [88]:
```
ax=sns.barplot(y='Fare', x='Pclass', hue='Sex', data=Titanic_data)
for p in ax.patches:
    ax.annotate(f'{p.get_height():.2f}', (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='center', xytext=(0, 10), textcoords='offset points')


plt.title('Count of Passengers based on Gender and Class')
plt.show()
```



In [89]:
```
ax=sns.barplot(y='Fare', x='Embarked', hue='Sex', data=Titanic_data)
for p in ax.patches:
    ax.annotate(f'{p.get_height():.2f}', (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='center', xytext=(0, 10), textcoords='offset points')


plt.title('Count of Passengers based on Gender and Embarked')
plt.show()
```

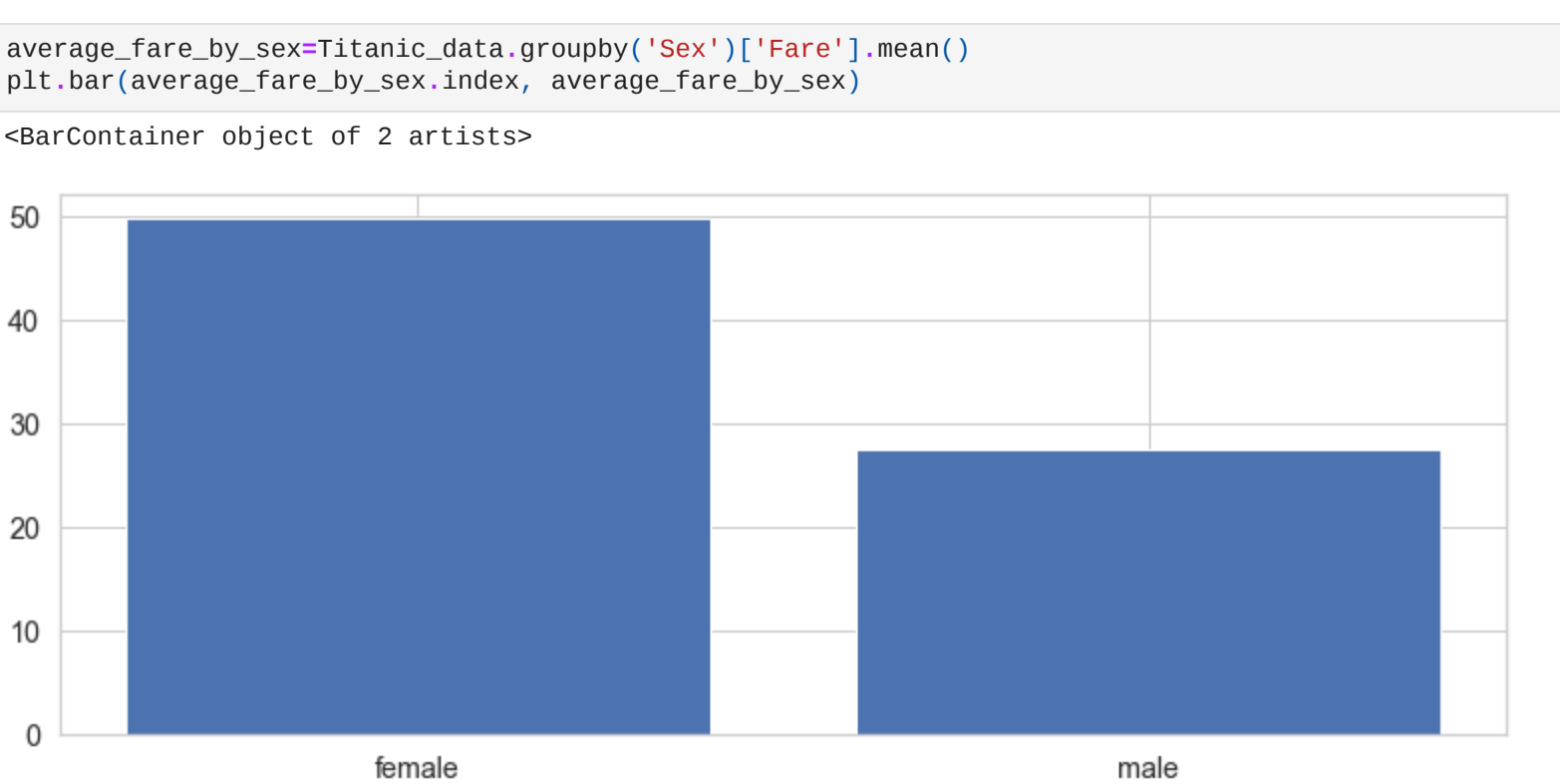


In [90]:
```
Titanic_data.groupby('Sex')['Fare'].mean()
```

Out[90]:
```
Sex
female    49.747699
male      27.527877
Name: Fare, dtype: float64
```

In [91]:
```
average_fare_by_sex=Titanic_data.groupby('Sex')['Fare'].mean()
plt.bar(average_fare_by_sex.index, average_fare_by_sex)
```

Out[91]:
```
<BarContainer object of 2 artists>
```

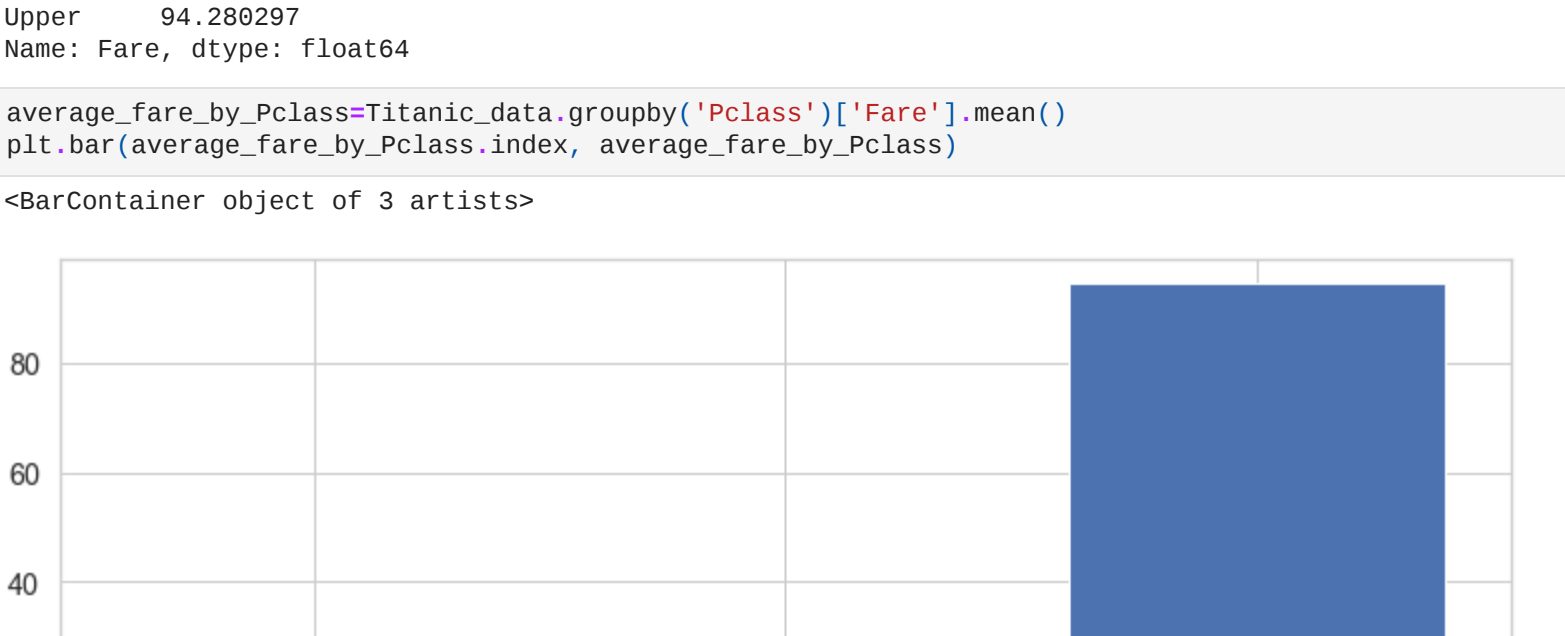


In [92]:
```
Titanic_data.groupby('Pclass')['Fare'].mean()
```

Out[92]:
```
Pclass
Lower     12.459678
Middle    22.202104
Upper     94.280297
Name: Fare, dtype: float64
```

In [93]:
```
average_fare_by_Pclass=Titanic_data.groupby('Pclass')['Fare'].mean()
plt.bar(average_fare_by_Pclass.index, average_fare_by_Pclass)
```

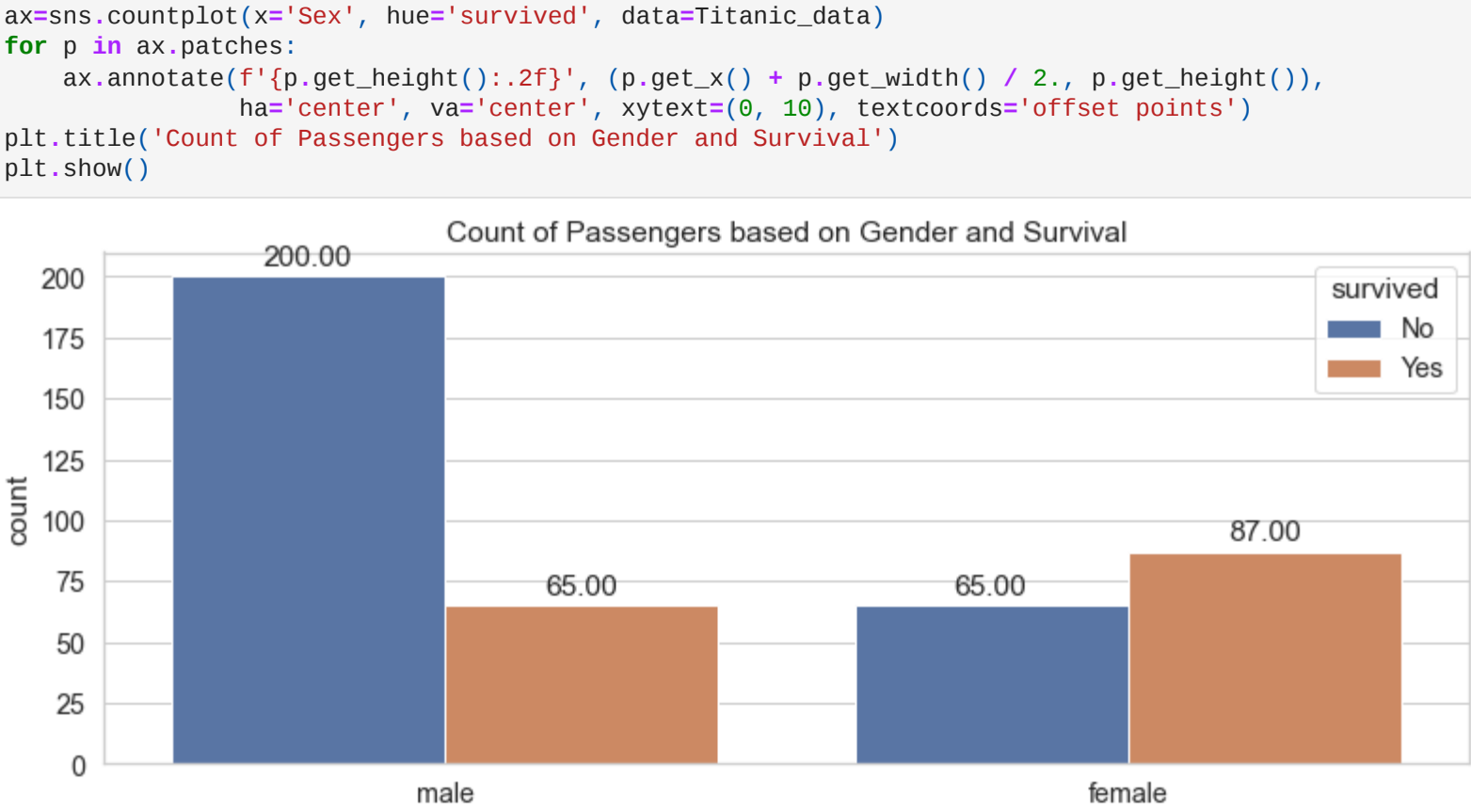
Out[93]:
```
<BarContainer object of 3 artists>
```



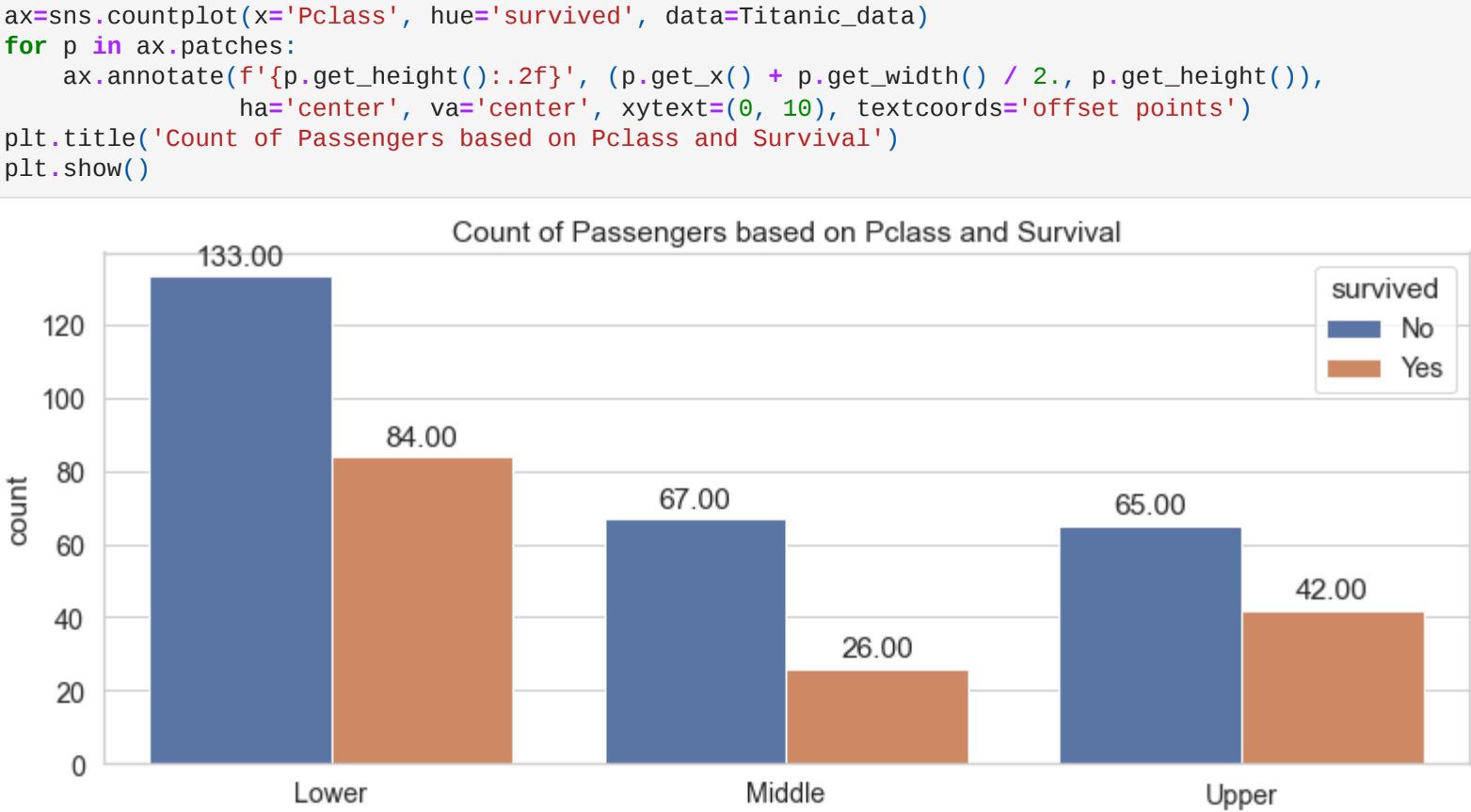
In [94]:
```
sex_counts.values
```

Out[94]:
```
array([265, 152], dtype=int64)
```

In [95]:
```
ax=sns.countplot(x='Sex', hue='survived', data=Titanic_data)
for p in ax.patches:
    ax.annotate(f'{p.get_height():.2f}', (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='center', xytext=(0, 10), textcoords='offset points')
plt.title('Count of Passengers based on Gender and Survival')
plt.show()
```



In [96]:
```
ax=sns.countplot(x='Pclass', hue='survived', data=Titanic_data)
for p in ax.patches:
    ax.annotate(f'{p.get_height():.2f}', (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='center', xytext=(0, 10), textcoords='offset points')
plt.title('Count of Passengers based on Pclass and Survival')
plt.show()
```



In [97]:
```
ax=sns.countplot(x='Embarked', hue='survived', data=Titanic_data)
for p in ax.patches:
    ax.annotate(f'{p.get_height():.2f}', (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='center', xytext=(0, 10), textcoords='offset points')
plt.title('Count of Passengers based on Embarked and Survival')
plt.show()
```