In [1]:	<pre>import pandas as pd import numpy as np import seaborn as sns from sklearn.model_selection import train_test_split from sklearn.linear_model import LogisticRegression from sklearn.metrics import accuracy_score from matplotlib import pyplot as plt</pre>
In [2]: Out[2]:	df_train_data=pd.read_csv("C:\\Users\\Pranav\\Desktop\\DATA SCIENCE DATA\\CVC file\\Titanic Survival Prediction using Machine Learning in Python\\train (1).csv") Passengerld Survived Pclass
In [3]:	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 #number of rows and columns df_train_data.shape
Out[3]: In [4]:	<pre>#information about data df_train_data.info() <class 'pandas.core.frame.dataframe'=""> RangeIndex: 891 entries, 0 to 890</class></pre>
	# Column Non-Null Count Dtype
In [5]: Out[5]:	11 Embarked 889 non-null object dtypes: float64(2), int64(5), object(5) memory usage: 83.7+ KB #to find missing value in data df_train_data.isnull().sum() PassengerId 0
out[5]:	Survived 0 Pclass 0 Name 0 Sex 0 Age 177 SibSp 0 Parch 0 Ticket 0
In [6]: Out[6]:	Fare 0 Cabin 687 Embarked 2 dtype: int64 df_train_data['Survived'].value_counts() 0 549
	1 342 Name: Survived, dtype: int64 #drop the cabin calumns tiatanic_data=df_train_data.drop(columns='Cabin',axis=0) #replacing the 'age' value columns with mean value tiatanic_data['Age'].fillna(tiatanic_data['Age'].mean(),inplace=True)
In [9]:	<pre>#replacing the 'Embarked' value columns with mode value print(tiatanic_data['Embarked'].mode()) 0 S Name: Embarked, dtype: object</pre>
In [10]: In [11]: In [12]:	<pre>print(tiatanic_data['Embarked'].mode()[0]) S #replacing the missing value in 'Embarked' tiatanic_data['Embarked'].fillna(tiatanic_data['Embarked'].mode()[0],inplace=True) #to find missing value in data</pre>
Out[12]:	tiatanic_data.isnull().sum() PassengerId 0 Survived 0 Pclass 0 Name 0 Sex 0 Age 0 SibSp 0
In [13]:	Parch 0 Ticket 0 Fare 0 Embarked 0 dtype: int64 #Data analytic #gettin the statistical measures about the data
Out[13]:	PassengerId Survived Pclass Age SibSp Parch Fare count 891.000000
In [14]:	<pre>max 891.000000 1.000000 3.000000 80.000000 6.000000 512.329200 #count plot sns.countplot('Survived', data=tiatanic_data) plt.show() C:\Users\Pranav\Searches\hjhkh\lib\site-packages\seaborn_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positi</pre>
	onal argument will be 'data', and passing other arguments without an explicit keyword will result in an error or misinterpretation. warnings.warn(500 - 400 - 2
In [15]:	#count plot sns.countplot('Sex', data=tiatanic_data) plt.show()
	C:\Users\Pranav\Searches\hjhkh\lib\site-packages\seaborn_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positi onal argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation. warnings.warn(500 400 200 600 700 700 700 700 700 7
In [16]:	#count plot #the number of people Survived gender wise sns.countplot('Sex', hue='Survived', data=tiatanic_data)
	plt.show() C:\Users\Pranav\Searches\hjhkh\lib\site-packages\seaborn_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positi onal argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation. warnings.warn(Survived 0 1 300
In [17]:	#count plot of Pclass
111 [17].	sns.countplot('Pclass', data=tiatanic_data) plt.show() C:\Users\Pranav\Searches\hjhkh\lib\site-packages\seaborn_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positi onal argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation. warnings.warn(500
	400 - 300 -
In [18]:	<pre>#count plot #the number of people Survived Pclass wise sns.countplot('Pclass', hue='Survived', data=tiatanic_data) plt.show() C:\Users\Pranav\Searches\hjhkh\lib\site-packages\seaborn_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positi onal argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation. warnings.warn(</pre>
	350 - Survived 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
In [19]: Out[19]:	tiatanic_data['Sex'].value_counts() male 577 female 314 Name: Sex, dtype: int64
In [20]: Out[20]: In [26]:	<pre>tiatanic_data['Embarked'].value_counts() S 646 C 168 Q 77 Name: Embarked, dtype: int64 tiatanic_data.replace({'Sex':{'male':0,'female':1},'Embarked':{'S':0,'C':1,'Q':2}},inplace=True)</pre>
<pre>In [27]: Out[27]: In [30]:</pre>	tiatanic_data.head() Passengerid Survived Pclass Name Sex Age SibSp Parch Ticket Fare Embarked 0 1 0 3 Braund, Mr. Owen Harris 0 22.0 1 0 A/5 21171 7.2500 0 1 2 1 1 Cumings, Mrs. John Bradley (Florence Briggs Th 1 38.0 1 0 PC 17599 71.2833 1 2 3 1 3 Heikkinen, Miss. Laina 1 26.0 0 0 STON/O2. 3101282 7.9250 0 3 4 1 1 Futrelle, Mrs. Jacques Heath (Lily May Peel) 1 35.0 1 0 113803 53.1000 0 4 5 0 3 Allen, Mr. William Henry 0 35.0 0 0 373450 8.0500 0
In [31]:	<pre>X=tiatanic_data.drop(columns=['PassengerId','Ticket','Name','Survived'],axis=0) y=tiatanic_data['Survived'] print(X) print(y) Pclass Sex Age SibSp Parch Fare Embarked 0</pre>
	1
In [32]:	3
	<pre>X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.1,random_state=2,stratify=y) print("shape of X_train= ",X_train.shape) print("shape of X_test= ",X_test.shape) print("shape of y_train= ",y_train.shape) print("shape of y_test= ",y_test.shape) shape of X_train= (801, 7) shape of X_test= (90, 7) shape of y_train= (801,) shape of y_test= (90,)</pre>
In [35]:	<pre>model=LogisticRegression() model.fit(X_train,y_train) C:\Users\Pranav\Searches\hjhkh\lib\site-packages\sklearn\linear_model_logistic.py:814: ConvergenceWarning: 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 Please also refer to the decumentation for alternative solver entions:</pre>
Out[35]: In [38]:	Please also refer to the documentation for alternative solver options: https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression n_iter_i = _check_optimize_result(LogisticRegression() #model evalution #Accuracy score #accurcy of training data X_train_prediction=model.predict(X_train)
Out[38]:	model.predict(X_train) array([1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
	1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
	1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
	1, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
In [41]:	1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
Out[41].	<pre>array([0, 1, 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,</pre>
In [46]:	training_data_accuracy_score(x_train_prediction,y_train) print('Accuracy on training data:',training_data_accuracy) Accuracy on training data: 0.797752808988764 #accuracy of testing data testing_data_accuracy_score(X_test_prediction,y_test) print('Accuracy on test data:',testing_data_accuracy) Accuracy on test data: 0.8555555555555555555555555555555555555
In []:	