

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix

# Load the dataset
file_path = '/content/titanic dataset.csv' # Change to your dataset path
titanic_data = pd.read_csv(file_path)

# Data Cleaning

# Fill missing values in 'Age' and 'Fare' with median
titanic_data['Age'].fillna(titanic_data['Age'].median(), inplace=True)
titanic_data['Fare'].fillna(titanic_data['Fare'].median(), inplace=True)

# Drop 'Cabin' due to high number of missing values
titanic_data.drop(columns=['Cabin'], inplace=True)

# Convert 'Sex' to numerical values
titanic_data['Sex'] = titanic_data['Sex'].map({'male': 0, 'female': 1})

# One-hot encode 'Embarked'
titanic_data = pd.get_dummies(titanic_data, columns=['Embarked'], drop_first=True)

# Drop unnecessary columns
titanic_data.drop(columns=['Name', 'Ticket', 'PassengerId'], inplace=True)

# Prepare Features and Target
X = titanic_data.drop(columns=['Survived'])
y = titanic_data['Survived']

# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Standardize the data
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Model Building: Logistic Regression
model = LogisticRegression()
model.fit(X_train, y_train)

# Make predictions
y_pred = model.predict(X_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)

print(f"Accuracy: {accuracy:.2f}")
print("Confusion Matrix:")
print(conf_matrix)
```

→ Accuracy: 1.00
Confusion Matrix:
[[50 0]
[0 34]]

