

```
# Import necessary libraries
import pandas as pd
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier, plot_tree
from sklearn.metrics import accuracy_score
import matplotlib.pyplot as plt

# Part A: Data Preprocessing

# Load dataset from URL
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/car/car.data"
column_names = ['buying', 'maint', 'doors', 'persons', 'lug_boot', 'safety', 'class']
df = pd.read_csv(url, names=column_names)

# Encode categorical features
label_encoders = {}
for col in df.columns:
    le = LabelEncoder()
    df[col] = le.fit_transform(df[col])
    label_encoders[col] = le # Store encoders in case needed later

# Split dataset (80% train, 20% test)
X = df.drop('class', axis=1)
y = df['class']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Part B: Decision Tree Classifier using Gini Index

# Initialize and train the classifier
clf_gini = DecisionTreeClassifier(criterion='gini', random_state=42)
clf_gini.fit(X_train, y_train)

# Make predictions and evaluate
y_pred_gini = clf_gini.predict(X_test)
accuracy_gini = accuracy_score(y_test, y_pred_gini)
print("Accuracy using Gini Index:", accuracy_gini)

# Visualize the tree
plt.figure(figsize=(20, 10))
plot_tree(clf_gini, feature_names=X.columns, class_names=label_encoders['class'].classes_, filled=True)
plt.title("Decision Tree using Gini Index")
plt.show()
```

↻ Accuracy using Gini Index: 0.9739884393063584

Decision Tree using Gini Index

