**RIPHAH INTERNATIONAL UNIVERSITY, ISLAMABAD**

****

**Lab 13**

**Bachelors of Computer science – 6th semester**

**Subject:** Artificial Intelligence Lab

**Submitted to:** Ma’am Ayesha

**Submitted by:** Areeba Sadaqat

**Sap Id:** 47633

**Date:** 29th April, 2025

**Question 01:**

**Decision Tree Algorithm:**

Train a decision tree to predict whether a student will **Pass (1)** or **Fail (0)** based on their study habits. The dataset is given below

| **Hours\_Studied** | **Sleep\_Hours** | **Tuition\_Attended** | **Pass** |
| --- | --- | --- | --- |
| 2 | 5 | 0 | 0 |
| 4 | 6 | 1 | 1 |
| 1 | 4 | 0 | 0 |
| 5 | 7 | 1 | 1 |
| 3 | 6 | 0 | 0 |
| 6 | 8 | 1 | 1 |
| 4 | 5 | 1 | 1 |
| 2 | 6 | 0 | 0 |

* Use this dataset in your code as features X and labels Y.
* Train a Decision Tree Classifier.
* Predict the result for a new student who:
* Studied 3 hours
* Slept 7 hours
* Did attend tuition (Tuition\_Attended = 1)
* Show the application and decision tree diagram.
* Visualize the decision tree.

import pandas as pd

from sklearn import tree

import matplotlib.pyplot as plt

# Step 1: Load dataset from CSV file

data = pd.read\_csv(r"E:/Sem 6th/AI lab/Lab 13/Tree.csv") # Use raw string or forward slashes

# Step 2: Separate features and labels

X = data.drop("Pass", axis=1) # Features: Hours\_Studied, Sleep\_Hours, Tuition\_Attended

Y = data["Pass"] # Labels: Pass (0/1)

# Step 3: Create and train the Decision Tree model

clf = tree.DecisionTreeClassifier()

clf = clf.fit(X, Y)

# Step 4: Make a prediction

# Predict for: Studied 3 hours, Slept 7 hours, Attended tuition

sample = [[3, 7, 1]] # Input for prediction

prediction = clf.predict(sample)

print("Will the student pass? (1 = Yes, 0 = No):", prediction[0])

# Step 5: Visualize the Decision Tree

plt.figure(figsize=(12, 8))

tree.plot\_tree(clf,

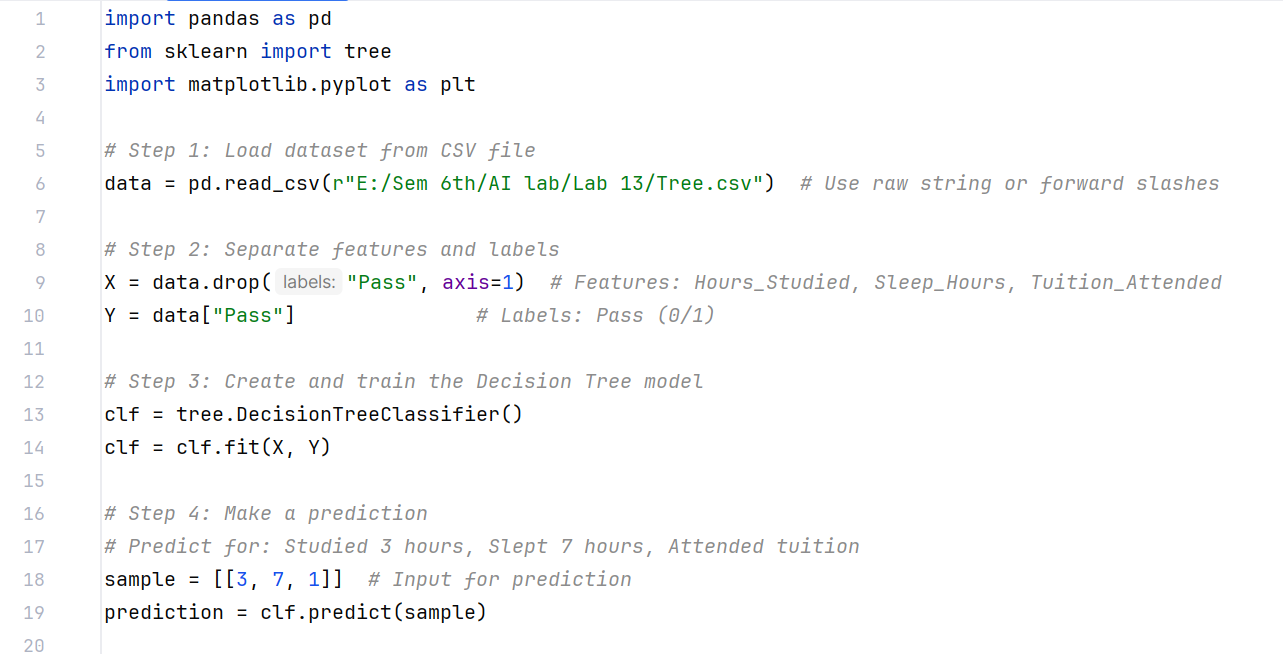
feature\_names=["Hours\_Studied", "Sleep\_Hours", "Tuition\_Attended"],

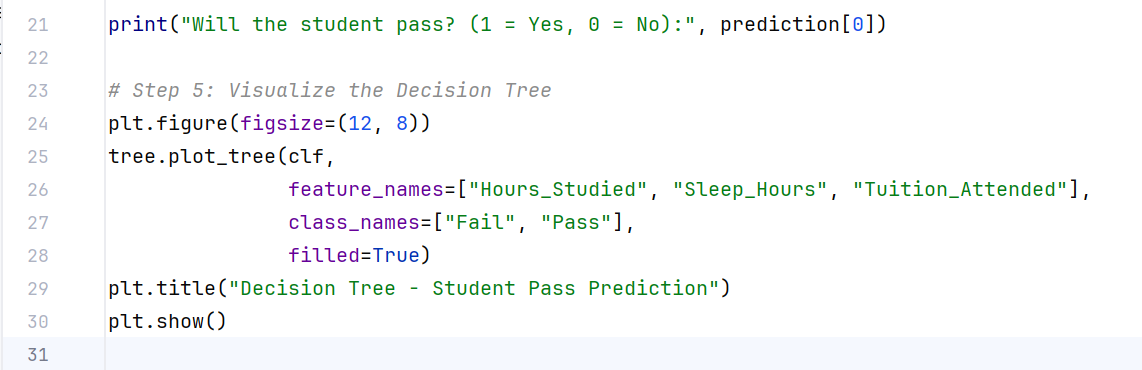
class\_names=["Fail", "Pass"],

filled=True)

plt.title("Decision Tree - Student Pass Prediction")

plt.show()





**Output:**



