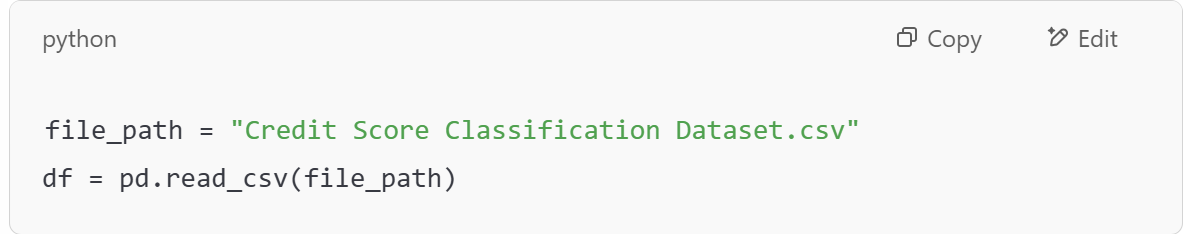
1. Importing Libraries

A screenshot of a computer code

AI-generated content may be incorrect.

* **pandas**: Used for data manipulation and analysis (e.g., loading, cleaning, and transforming data).
* **train\_test\_split**: Function from sklearn.model\_selection to split the data into training and testing sets.
* **DecisionTreeClassifier**: Classifier from sklearn.tree used to build the decision tree model.
* **accuracy\_score & classification\_report**: Metrics from sklearn.metrics to evaluate the model’s performance.
* **matplotlib.pyplot**: Used for plotting and visualizing the decision tree.
* **plot\_tree**: Used to visualize the decision tree after training.

1. Loading the Dataset

* 

**pd.read\_csv()**: Reads the data from a CSV file and stores it in a pandas DataFrame df.

1. Mapping Credit Score Categories to Numeric Values

A screen shot of a computer code

AI-generated content may be incorrect.The **credit score categories (Low, Medium, High)** are mapped to numeric values **(0, 1, 2)** using a dictionary (credit\_score\_mapping).

The **map()** function replaces the original text labels in the "Credit Score" column with numeric values.

1. **Dropping Missing Values**

**A close-up of a computer screen

AI-generated content may be incorrect.**

* **dropna()**: Removes rows where the "Credit Score" column contains missing (NaN) values.
* This ensures that the model is trained on complete data, particularly for the target variable.

1. **Converting the "Credit Score" to Integer**

**A computer screen shot of a computer code

AI-generated content may be incorrect.**

* **astype(int)**: Converts the "Credit Score" column to integer type to ensure it’s suitable for classification.

1. **One-Hot Encoding Categorical Features**

A white background with black text

AI-generated content may be incorrect.

* **pd.get\_dummies()**: One-hot encodes categorical variables.
  + **columns=['Gender', 'Education', 'Marital Status', 'Home Ownership']**: Specifies the columns to encode.
  + **drop\_first=True**: Drops the first category to avoid multicollinearity (the first category becomes the reference category).
* The one-hot encoding converts categorical features into binary columns (e.g., **Gender** might become **Gender\_Male** and **Gender\_Female**).

**7. Splitting Data into Features (X) and Target (y)**

A white rectangular object with black text

AI-generated content may be incorrect.

* **X**: Contains all features (predictors) by dropping the target variable ('Credit Score').
* **y**: Contains the target variable (credit score classification), which we want to predict.

**8. Splitting Data into Training and Testing Sets**

**A white screen with black text

AI-generated content may be incorrect.**

* **train\_test\_split()**: Splits the dataset into **training (80%)** and **testing (20%)** sets.
* **test\_size=0.2**: Specifies that 20% of the data is used for testing.
* **random\_state=42**: Ensures reproducibility of the data split (randomization is the same each time).

**9. Initializing and Training the Decision Tree Classifier**

**A white background with black text

AI-generated content may be incorrect.**

* **DecisionTreeClassifier(random\_state=42)**: Initializes a Decision Tree model with a fixed random state for reproducibility.
* **model.fit(X\_train, y\_train)**: Trains the decision tree on the **training data** (X\_train) and corresponding labels (y\_train).

**10. Making Predictions**

**A close-up of a white card

AI-generated content may be incorrect.**

* **model.predict(X\_test)**: Uses the trained model to predict the credit score classes for the **test data** (X\_test).

**11. Evaluating Model Performance**

A screenshot of a computer code

AI-generated content may be incorrect.

* **accuracy\_score(y\_test, y\_pred)**: Computes the accuracy of the model by comparing the predicted labels (y\_pred) with the true labels (y\_test).
* **classification\_report(y\_test, y\_pred)**: Generates a report that includes precision, recall, F1-score, and support for each class.

**12. Visualizing the Decision Tree**

A screen shot of a computer code

AI-generated content may be incorrect.

* **plt.figure(figsize=(12, 8))**: Sets the size of the plot.
* **plot\_tree()**: Visualizes the trained decision tree.
  + **filled=True**: Colors the nodes based on the c

**OUTPUT**:

Accuracy: 1.0

Classification Report:

precision recall f1-score support

0 1.00 1.00 1.00 1

2 1.00 1.00 1.00 25

accuracy 1.00 26

macro avg 1.00 1.00 1.00 26

weighted avg 1.00 1.00 1.00 26

A diagram of a credit score

AI-generated content may be incorrect.