Cowlar Tasks

**Task 1:**

**Summary:**

The provided task performs a regression analysis on a dataset related to car information using the Decision Tree Regressor model. The dataset is read from a CSV file, preprocessed to handle missing values and certain entries, and then used to train the regression model. The model is evaluated on a test set, and the results are printed along with a visualization of the decision tree.

**Approach:**

**Data Reading and Preprocessing:**

The function reads the dataset from the 'car\_data.csv' file using pandas. Missing values and entries with 'Call for price' in the 'Price' column are removed from the dataset.

**Label Encoding:**

Categorical columns ('Make', 'Model', 'Version', 'Assembly', 'Registered City') are label-encoded using LabelEncoder from sklearn.

**Model** **Training**:

The dataset is split into training and testing sets using train\_test\_split. A Decision Tree Regressor is instantiated and trained on the training set (X\_train and y\_train).

**Model Evaluation:**

The model is evaluated on the test set (X\_test and y\_test). The actual and predicted values are printed along with the model score.

**Decision Tree Visualization:**

The decision tree is exported to a 'tree.dot' file for visualization using Graphviz.  
  
**Task 2:  
  
Summary:**

The provided code uses K-Means clustering to group movies based on their average ratings. It visualizes the elbow method to determine the optimal number of clusters, assigns movies to clusters, and then recommends movies similar to a user-input movie based on the cluster. The dataset is read from 'movies\_dataset.csv,' and the movie titles are temporarily removed during processing.  
  
**Approach:**

**Data Reading and Preprocessing:**

The function reads the dataset from 'movies\_dataset.csv' using pandas. The 'title' column is temporarily dropped, and a new column 'average\_rating' is created by taking the mean of the remaining rows. K-Means Clustering:

The k\_means function applies K-Means clustering with varying numbers of clusters (from 1 to 10) and plots the elbow method graph to help determine the optimal number of clusters. Based on the elbow method analysis, the code then fits the K-Means model with the chosen number of clusters (in this case, 3). The movies are assigned to clusters, and a scatter plot is generated to visualize the clustering.

**Movie Recommendation:**

The recommended\_movie function takes the clustered data and prompts the user to input a movie title. It then identifies the cluster to which the input movie belongs and recommends other movies from the same cluster.