Documentation for class MonitoringandRetraining()

The MonitoringandRetraining() class is responsible for monitoring the behavior of a KMeans model that segments customer data and retraining the model in case it detects a significant drift in the data.

Functions:

- **import_packages()**: Imports the necessary packages for the class to function, including pandas, os, numpy, datetime, and mlflow.
- **__init__(self)**: Initializes the MonitoringandRetraining object with the following attributes:
  - train_start_date: A datetime object representing the start date of the training data.
  - train_end_date: A datetime object representing the end date of the training data.
  - data: A pandas DataFrame containing customer data.
  - reference_data: A pandas DataFrame containing customer data from before the train_end_date.
  - model: A KMeans model trained on the reference_data.
  - silhouette_score: A float representing the silhouette score of the current data.
  - execution_date: A datetime object representing the date the model is being executed.
  - current_data: A pandas DataFrame containing customer data from after the train_end_date.
- **load_data(self)**: Loads customer data from CSV files and processes it into a pandas DataFrame. Sets the reference_data attribute to customer data from before the train_end_date.

- **set_current_data(self, execution_date)**: Sets the execution_date attribute to the input date and sets the current_data attribute to customer data from after the train_end_date up to the execution_date.
- **get_drift_metrics(self)**: Calculates the silhouette score of the current data using the KMeans model trained on the reference_data.
- **log_metric(self)**: Logs the silhouette score, execution date, and train end date to MLflow.
- **set_new_reference_data_and_retrain(self)**: If the silhouette score is below 0.33, sets the train_end_date attribute to the execution_date attribute and reprocesses the data to set a new reference_data attribute. Trains a new KMeans model on the new reference_data and logs it to MLflow. If the silhouette score is above 0.33, prints a message indicating that the model does not need to be retrained.