We start by loading the ‘Advanced.csv’ file into a pandas DataFrame to examine its structure. Next, we filter the DataFrame for the 2024 season and handle missing values by dropping rows with NaNs in critical columns ('player', 'tm', 'per', 'mp'). We then normalize each column using MinMaxScaler to standardize the data. After preprocessing, we define our features and split the data into training and test sets using `train_test_split`.

For building the recommender system, we use KMeans clustering to group players into clusters based on their values from our normalized features. We fit the KMeans model on the training data and predict cluster labels for the entire dataset, adding these labels to the original DataFrame for interpretation. To identify top players, we sort them within each cluster based on ‘mp’, and 'per’ highlighting the top performers.

Additionally, we create a function that identifies the three most efficient players for a given team based on the highest combined average of our sorted features. This function filters the DataFrame by team name and locates the player with the highest performance metrics. An example usage of this function demonstrates how to find the three most efficient players for a specified team, such as the Lakers (LAL). This process from data loading and preprocessing to model building and player identification provides a comprehensive approach to determining player efficiency based on advanced metrics.

The final step was to evaluate the precision and accuracy of our clustering model. To do this, we use silhouette score, inertia, and elbow method. The elbow method revealed that the optimal number of clusters for this project was 3. After setting that value according to the elbow method, we observed a silhouette score of ~0.56 for the test data indicating well-defined and distinct clusters. The score is relatively close to baseline, which still reveals room for improvement but is still strong enough to find a relationship between PER and minutes played for quality recommendations.