```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.impute import SimpleImputer
data = pd.read_csv(r"C:/Machine Learning/Boston housing dataset.csv")
print(data.head())
print(data.shape)
print(data.info())
X = data.drop(columns=['MEDV'])
y = data['MEDV']
print("Missing values before processing:")
print(X.isnull().sum())
imputer = SimpleImputer(strategy="mean")
X = pd.DataFrame(imputer.fit_transform(X), columns=X.columns)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)
r2 = r2_score(y_test, y_pred)
print("Mean Squared Error:", mse)
print("Root Mean Squared Error (RMSE):", rmse)
print("R² Score:", r2)
rm_model = LinearRegression()
rm_model.fit(data[['RM']], data['MEDV'])
data['Predicted'] = rm_model.predict(data[['RM']])
sns.set(style='whitegrid')
sns.scatterplot(x='RM', y='MEDV', data=data, label='Actual data', color='blue', alpha=0.5)
sns.lineplot(x='RM', y='Predicted', data=data, label='Regression Line (RM only)', color='red')
plt.title("Linear Regression - RM vs MEDV (Visualization)")
plt.xlabel("Average Number of Rooms (RM)")
plt.ylabel("Median Home Value (MEDV)")
plt.legend()
plt.show()
```