1. **What is the purpose of `pd.read_csv()` in this code?**

   - The `pd.read_csv()` function reads a CSV file into a pandas DataFrame. It is used to load the dataset from the specified file path for analysis and processing.

2. **What does `data.head()` do?**

   - `data.head()` returns the first 5 rows of the DataFrame, allowing us to quickly inspect the initial data and its structure.

3. **Why do we use `data.isnull().sum()`?**

   - This function checks for missing values in the dataset and provides the count of missing values in each column, helping us identify any data quality issues.

4. **What is the purpose of the `sns.histplot()` function?**

   - `sns.histplot()` is used to plot the distribution of a numerical variable, in this case, `medv`, which represents the target variable. It provides insights into its distribution.

5. **What does the `kde=True` argument in `sns.histplot()` do?**

   - The `kde=True` argument overlays a kernel density estimate (KDE) curve on the histogram, providing a smooth estimate of the probability density function.

6. **What does `data.corr()` compute?**

   - `data.corr()` computes the correlation matrix, showing the relationships between different numerical columns. It indicates how strongly features are related to one another.

7. **Why do we round the correlation values to two decimal places?**

- Rounding the correlation values to two decimal places makes the output easier to read and interpret, without losing important information.

8. What is the significance of `sns.heatmap()` in this context?

- `sns.heatmap()` visualizes the correlation matrix as a heatmap, with color intensity representing the strength of the correlation between features.

9. Why do we use scatter plots between features and the target variable?

- Scatter plots help to visually explore the relationship between individual features (`rm` and `lstat`) and the target variable (`medv`), making it easier to identify patterns or trends.

10. What is the role of `train_test_split()`?

- `train_test_split()` splits the dataset into training and testing sets, allowing the model to be trained on one portion and evaluated on a separate unseen portion of the data.

11. Why is the test size set to 0.2 in `train_test_split()`?

- The test size of 0.2 indicates that 20% of the data will be used for testing, while 80% will be used for training the model. This is a common train-test split ratio.

12. What is the purpose of `model.fit()`?

- `model.fit()` trains the linear regression model using the provided training data (`X_train` and `Y_train`), allowing the model to learn the relationships between features and the target.

13. What does `model.predict()` do?

- `model.predict()` uses the trained model to make predictions on the test set (`X_test`), generating predicted values of the target variable.

14. What is the purpose of calculating the Mean Squared Error (MSE)?

- MSE quantifies the difference between the predicted values and the actual values. A lower MSE indicates better model performance.

15. What does a high MSE value indicate about the model's performance?

- A high MSE suggests that the model's predictions deviate significantly from the actual values, indicating poor performance.

16. Why do we use `plt.tight_layout()`?

- `plt.tight_layout()` adjusts the layout of the plots to avoid overlapping elements and ensure that the visualizations are displayed clearly.

17. Why is the figure size set using `sns.set()`?

- Setting the figure size with `sns.set()` ensures that all plots are large enough to be easily read and interpreted, providing a consistent visual experience.

18. What does the `cmap='coolwarm'` argument in `sns.heatmap()` do?

- The `cmap='coolwarm'` argument specifies the color palette used for the heatmap, where "cool" represents negative correlations and "warm" represents positive correlations.

19. Why do we select only certain features like `rm` and `lstat` for modeling?

- These features (`rm` and `lstat`) are selected because they show strong correlations with the target variable (`medv`), potentially leading to a better model.

20. What is the target variable in this dataset?

- The target variable (`medv`) represents the median value of homes in thousands of dollars and is the value we aim to predict using the features.

21. Why do we use `sns.set(rc={'figure.figsize': (11.7, 8.27)})`?

- This sets the default figure size for Seaborn plots to ensure that all visualizations are consistently large and appropriately sized for easy interpretation.

22. How does the heatmap help in model selection?

- The heatmap shows which features are most correlated with the target variable, helping us select the most relevant features for the model.

23. Why are scatter plots useful for visualizing relationships between variables?

- Scatter plots provide a visual representation of how two variables are related, helping to detect patterns, trends, and potential outliers.

24. What is the purpose of the `random_state=42` argument in `train_test_split()`?

- The `random_state=42` ensures that the data split is reproducible, so the same training and test sets are used each time the code is run.

25. What does the `plt.show()` function do?

- `plt.show()` renders the plot to the screen, making the visualizations visible for inspection. It is required to display any plot created using Matplotlib.