**1. Adding a New Column Using df.assign()**

* **Question: Use the df.assign() method to add a new column called BMI that calculates the Body Mass Index (BMI) of the players using the formula: BMI = Weight / (Height in meters)^2. Assume the Height column is in feet and inches, and you need to convert it to meters first.**
* **Hint: You can chain .assign() to add multiple transformations.**

**2. Adding a Column Based on Conditional Logic Using df.assign()**

* **Question: Add a new column called Salary\_Tier using the df.assign() method, which classifies players as 'High' if their salary is above 5,000,000, and 'Low' otherwise.**
* **Hint: Use a lambda function inside df.assign().**

**3. Selecting Rows and Columns Using df.loc()**

* **Question: Use the df.loc() method to select and display all columns for the players whose Position is "PG" (Point Guard).**
* **Hint: Use df.loc[df['Position'] == 'PG', :].**

**4. Modifying Data Using df.loc()**

* **Question: Use the df.loc() method to change the Team of all players whose College is "Texas" to "New Team".**
* **Hint: Use df.loc[df['College'] == 'Texas', 'Team'] = 'New Team'.**

**5. Selecting Rows and Columns Using df.iloc()**

* **Question: Use the df.iloc() method to select and display the first 5 rows and the first 3 columns of the DataFrame.**
* **Hint: Use df.iloc[:5, :3].**

**6. Modifying Specific Cells Using df.iloc()**

* **Question: Use the df.iloc() method to modify the salary of the player in the third row to 10,000,000.**
* **Hint: Use df.iloc[2, 'Salary\_Column\_Index'] = 10000000 where Salary\_Column\_Index is the index of the salary column.**

**7. Dropping Rows Based on a Condition**

* **Question: Drop all rows from the DataFrame where the player’s Age is less than 24.**
* **Hint: Use df.drop(df[df['Age'] < 24].index, inplace=True).**

**8. Dropping a Column Using drop()**

* **Question: Use the drop() method to remove the College column from the DataFrame.**
* **Hint: Use df.drop('College', axis=1, inplace=True).**

**9. Truncating the DataFrame Using truncate()**

* **Question: Use the truncate() method to keep only the first 10 rows and remove the rest.**
* **Hint: Use df.truncate(after=9).**

**10. Sorting the DataFrame Based on a Single Column**

* **Question: Sort the DataFrame by Salary in descending order.**
* **Hint: Use df.sort\_values(by='Salary', ascending=False).**

**11. Sorting the DataFrame by Multiple Columns**

* **Question: Sort the DataFrame first by Team in alphabetical order and then by Salary in descending order within each team.**
* **Hint: Use df.sort\_values(by=['Team', 'Salary'], ascending=[True, False]).**