4.5.3

Activity: Drilling Down to Get Specifics

Maria is pleased with the summary statistics that you generated. Now, she needs information that's more specific. So, she's asked you to more closely examine several subsets of data.

In this activity, you'll use the Pandas `loc` and `iloc` functions to focus on areas of interest in a dataset.

Files

Continue using the Jupyter Notebook file that you created in the previous lesson to write your code.

Download the following files to help you get started:

Drilling Down to Get Specifics files (https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/v2/module_4/4-5-Student_Data_Starter_Code.zip)

sing the Jupyter Notebook file in the `Unsolved` folder, navigate to the "Step 4" section to write your code. Alternatively, you can use the Jupyter Notebook file that you created in the previous lesson to write your code.

Instructions

1. Use `iloc` to select the first row of the DataFrame , as the following code shows:

```
student_df.iloc[0]
```

2. Use `iloc` to display the first 10 rows of the DataFrame, sorted in ascending order by school name, as the following code shows:

```
student_df.iloc[:10].sort_values("school_name")
```

3. Use `loc` to select all the rows for students at public schools from the DataFrame, and store it as `public_schools_df`, as the following code shows:

```
public_schools_df = student_df.loc[student_df["school_type"] == "Public"]
```

4. Display the summary statistics for public schools by calling `describe` on the new DataFrame, as the following code shows:

```
public_schools_df.describe()
```

5. Display the summary statistics for all the students at charter schools, as the following code shows:

```
charter_schools_df = student_df.loc[student_df["school_type"] == "Charter"]
charter_schools_df.describe()
```

6. Store the row that contains the minimum math score, as the following code shows:

```
min_math_score = student_df["math_score"].min()
min_math_row = student_df.loc[student_df["math_score"] == min_math_score]
min_math_row
```

7. Find the average math score of all the students who attend the same school as the student who has the lowest math score, as the following code shows:

```
student_df.loc[student_df["school_name"] == "Green High School", "math_score"].mean()
```

8. Use `loc` with a conditional statement to select all the rows that include math scores less than 70, as the following code shows:

```
student_df.loc[student_df["math_score"] < 70]
```

Solution

How did you do?

You can refer to the solution file in the `Solved` folder, which is in the zipped folder that you downloaded for this activity.

What's Next?

You've done a terrific job with this data analysis so far! You're steadily growing your data analysis skills, and you've already come a long way. Now that you've practiced using `loc` and `iloc` to get specifics for your analysis, you'll next explore what it takes to compare multiple subsets of data to one another.