**Instructions**

Using Pandas and Jupyter Notebook, create a report that includes the following data. Your report must include a written description of at least two observable trends based on the data.

**Hint:** Check out the sample solution called PyCitySchools\_starter.ipynb located in the .zip file to review the desired format for this assignment.

**District Summary**

Perform the necessary calculations and then create a high-level snapshot of the district's key metrics in a DataFrame.

Include the following:

* Total number of unique schools
* Total students
* Total budget
* Average math score
* Average reading score
* % passing math (the percentage of students who passed math)
* % passing reading (the percentage of students who passed reading)
* % overall passing (the percentage of students who passed math AND reading)

**School Summary**

Perform the necessary calculations and then create a DataFrame that summarizes key metrics about each school.

Include the following:

* School name
* School type
* Total students
* Total school budget
* Per student budget
* Average math score
* Average reading score
* % passing math (the percentage of students who passed math)
* % passing reading (the percentage of students who passed reading)
* % overall passing (the percentage of students who passed math AND reading)

**Highest-Performing Schools (by % Overall Passing)**

Sort the schools by % Overall Passing in descending order and display the top 5 rows.

Save the results in a DataFrame called "top\_schools".

**Lowest-Performing Schools (by % Overall Passing)**

Sort the schools by % Overall Passing in ascending order and display the top 5 rows.

Save the results in a DataFrame called "bottom\_schools".

**Math Scores by Grade**

Perform the necessary calculations to create a DataFrame that lists the average math score for students of each grade level (9th, 10th, 11th, 12th) at each school.

**Reading Scores by Grade**

Create a DataFrame that lists the average reading score for students of each grade level (9th, 10th, 11th, 12th) at each school.

**Scores by School Spending**

Create a table that breaks down school performance based on average spending ranges (per student).

Use the code provided below to create four bins with reasonable cutoff values to group school spending.

spending\_bins = [0, 585, 630, 645, 680]

labels = ["<$585", "$585-630", "$630-645", "$645-680"]

Use pd.cut to categorize spending based on the bins.

Use the following code to then calculate mean scores per spending range.

spending\_math\_scores = school\_spending\_df.groupby(["Spending Ranges (Per Student)"])["Average Math Score"].mean()

spending\_reading\_scores = school\_spending\_df.groupby(["Spending Ranges (Per Student)"])["Average Reading Score"].mean()

spending\_passing\_math = school\_spending\_df.groupby(["Spending Ranges (Per Student)"])["% Passing Math"].mean()

spending\_passing\_reading = school\_spending\_df.groupby(["Spending Ranges (Per Student)"])["% Passing Reading"].mean()

overall\_passing\_spending = school\_spending\_df.groupby(["Spending Ranges (Per Student)"])["% Overall Passing"].mean()

Use the scores above to create a DataFrame called spending\_summary.

Include the following metrics in the table:

* Average math score
* Average reading score
* % passing math (the percentage of students who passed math)
* % passing reading (the percentage of students who passed reading)
* % overall passing (the percentage of students who passed math AND reading)

**Scores by School Size**

Use the following code to bin the per\_school\_summary.

size\_bins = [0, 1000, 2000, 5000]

labels = ["Small (<1000)", "Medium (1000-2000)", "Large (2000-5000)"]

Use pd.cut on the "Total Students" column of the per\_school\_summary DataFrame.

Create a DataFrame called size\_summary that breaks down school performance based on school size (small, medium, or large).

**Scores by School Type**

Use the per\_school\_summary DataFrame from the previous step to create a new DataFrame called type\_summary.

This new DataFrame should show school performance based on the "School Type".