

PyCity Schools Analysis

- As a whole, schools with higher budgets, did not yield better test results. By contrast, schools with higher spending per student actually (\$645 - 675) underperformed compared to schools with smaller budgets (\$585 per student).
- As a whole, smaller and medium sized schools dramatically out-performed large sized schools on passing math performances (89-91% passing vs 67%).
- As a whole, charter schools out-performed the public district schools across all metrics. However, more analysis will be required to glean if the effect is due to school practices or the fact that charter schools tend to serve smaller student populations per school.

Note: Instructions have been included for each segment. You do not have to follow them exactly, but they are included to help you think through the steps.

In [22]:

```
# Dependencies and Setup
import pandas as pd
import numpy as np

# File to Load (Remember to Change These)
school_data_to_load = "data/schools_complete.csv"
student_data_to_load = "data/students_complete.csv"

# Read School and Student Data File and store into Pandas Data Frames
school_data = pd.read_csv('./data/schools_complete.csv')
student_data = pd.read_csv('./data/students_complete.csv')

# Combine the data into a single dataset
school_data_complete = pd.merge(student_data, school_data, how="left", on=["school_name", "student_name"])
school_data_complete.head()

# school_data_complete.count()
```

Out[22]:

	Student ID	student_name	gender	grade	school_name	reading_score	math_score	School ID	typ
0	0	Paul Bradley	M	9th	Huang High School	66	79	0	District
1	1	Victor Smith	M	12th	Huang High School	94	61	0	District
2	2	Kevin Rodriguez	M	12th	Huang High School	90	60	0	District
3	3	Dr. Richard Scott	M	12th	Huang High School	67	58	0	District
4	4	Bonnie Ray	F	9th	Huang High School	97	84	0	District



District Summary

- Calculate the total number of schools
- Calculate the total number of students
- Calculate the total budget
- Calculate the average math score
- Calculate the average reading score
- Calculate the overall passing rate (overall average score), i.e. (avg. math score + avg. reading score)/2
- Calculate the percentage of students with a passing math score (70 or greater)
- Calculate the percentage of students with a passing reading score (70 or greater)
- Create a dataframe to hold the above results
- Optional: give the displayed data cleaner formatting

In []:

```
In [27]: totalSchools = school_data_complete["school_name"].nunique()  
totalSchools
```

Out[27]: 15

```
In [28]: totalStudents = school_data_complete["Student ID"].nunique()  
totalStudents
```

Out[28]: 39170

```
In [29]: totalBudget = school_data["budget"].sum()  
totalBudget
```

Out[29]: 24649428

```
In [30]: averageMathScore = school_data_complete["math_score"].mean()  
averageMathScore
```

Out[30]: 78.98537145774827

```
In [31]: averageReadingScore = school_data_complete["reading_score"].mean()  
averageReadingScore
```

Out[31]: 81.87784018381414

```
In [32]: percentageOverallPassingRate = (averageMathScore + averageReadingScore) / 2  
percentageOverallPassingRate
```

```
Out[32]: 80.43160582078121
```

```
In [33]: passingMath = len(school_data_complete[school_data_complete["math_score"] >= 70])  
percentagePassingMath = passingMath / totalStudents * 100  
percentagePassingMath
```

```
Out[33]: 74.9808526933878
```

School Summary

- Create an overview table that summarizes key metrics about each school, including:
 - School Name
 - School Type
 - Total Students
 - Total School Budget
 - Per Student Budget
 - Average Math Score
 - Average Reading Score
 - % Passing Math
 - % Passing Reading
 - Overall Passing Rate (Average of the above two)
- Create a dataframe to hold the above results

Top Performing Schools (By Passing Rate)

- Sort and display the top five schools in overall passing rate

```
In [38]: grouped_school_df = school_data_complete.groupby(["school_name"])
schoolType = grouped_school_df["type"].first()
totalStudents = grouped_school_df["Student ID"].count()
grouped_school_df
schoolType
totalStudents
```

```
Out[38]: school_name
Bailey High School      4976
Cabrera High School    1858
Figueroa High School   2949
Ford High School       2739
Griffin High School    1468
Hernandez High School  4635
Holden High School     427
Huang High School      2917
Johnson High School    4761
Pena High School       962
Rodriguez High School  3999
Shelton High School    1761
Thomas High School     1635
Wilson High School     2283
Wright High School     1800
Name: Student ID, dtype: int64
```

```
In [35]: totalSchoolBudget = grouped_school_df["budget"].first()
totalSchoolBudget
```

```
Out[35]: school_name
Bailey High School      3124928
Cabrera High School    1081356
Figueroa High School   1884411
Ford High School       1763916
Griffin High School    917500
Hernandez High School  3022020
Holden High School     248087
Huang High School      1910635
Johnson High School    3094650
Pena High School       585858
Rodriguez High School  2547363
Shelton High School    1056600
Thomas High School     1043130
Wilson High School     1319574
Wright High School     1049400
Name: budget, dtype: int64
```

```
In [39]: perStudentBudget = totalSchoolBudget / totalStudents  
perStudentBudget
```

```
Out[39]: school_name  
Bailey High School      628.0  
Cabrera High School    582.0  
Figueroa High School   639.0  
Ford High School       644.0  
Griffin High School    625.0  
Hernandez High School  652.0  
Holden High School     581.0  
Huang High School      655.0  
Johnson High School    650.0  
Pena High School       609.0  
Rodriguez High School  637.0  
Shelton High School    600.0  
Thomas High School     638.0  
Wilson High School     578.0  
Wright High School     583.0  
dtype: float64
```

```
In [40]: averageMathScore = grouped_school_df["math_score"].mean()  
averageReadingScore = grouped_school_df["reading_score"].mean()  
averageMathScore  
averageReadingScore
```

```
Out[40]: school_name  
Bailey High School      81.033963  
Cabrera High School    83.975780  
Figueroa High School   81.158020  
Ford High School       80.746258  
Griffin High School    83.816757  
Hernandez High School  80.934412  
Holden High School     83.814988  
Huang High School      81.182722  
Johnson High School    80.966394  
Pena High School       84.044699  
Rodriguez High School  80.744686  
Shelton High School    83.725724  
Thomas High School     83.848930  
Wilson High School     83.989488  
Wright High School     83.955000  
Name: reading_score, dtype: float64
```

Find the passing rate for math and reading (above 70 points)

```
In [42]: totalMathScore = grouped_school_df["math_score"].count()

passingMath = school_data_complete[school_data_complete["math_score"] >= 70].groupby("school_name").count()

percentagePassingMath = passingMath / totalStudents * 100

percentagePassingMath
```

```
Out[42]: school_name
Bailey High School      66.680064
Cabrera High School    94.133477
Figueroa High School   65.988471
Ford High School       68.309602
Griffin High School    93.392371
Hernandez High School  66.752967
Holden High School     92.505855
Huang High School      65.683922
Johnson High School    66.057551
Pena High School       94.594595
Rodriguez High School  66.366592
Shelton High School    93.867121
Thomas High School     93.272171
Wilson High School     93.867718
Wright High School     93.333333
dtype: float64
```

```
In [43]: totalReadingScore = grouped_school_df["reading_score"].count()

passingRead = school_data_complete[school_data_complete["reading_score"] >= 70].groupby("school_name").count()

percentagePassingReading = passingRead / totalStudents * 100

percentagePassingReading
```

```
Out[43]: school_name
Bailey High School      81.933280
Cabrera High School    97.039828
Figueroa High School   80.739234
Ford High School       79.299014
Griffin High School    97.138965
Hernandez High School  80.862999
Holden High School     96.252927
Huang High School      81.316421
Johnson High School    81.222432
Pena High School       95.945946
Rodriguez High School  80.220055
Shelton High School    95.854628
Thomas High School     97.308869
Wilson High School     96.539641
Wright High School     96.611111
dtype: float64
```

```
In [44]: percentageOverallPassingRate = (percentagePassingMath + percentagePassingReading)  
percentageOverallPassingRate
```

```
Out[44]: school_name  
Bailey High School      74.306672  
Cabrera High School    95.586652  
Figueroa High School   73.363852  
Ford High School       73.804308  
Griffin High School   95.265668  
Hernandez High School  73.807983  
Holden High School    94.379391  
Huang High School     73.500171  
Johnson High School   73.639992  
Pena High School      95.270270  
Rodriguez High School 73.293323  
Shelton High School   94.860875  
Thomas High School    95.290520  
Wilson High School    95.203679  
Wright High School    94.972222  
dtype: float64
```

```
In [45]: school_summary_df = pd.DataFrame({"School Type": schoolType,
                                         "Total Students": totalStudents,
                                         "Total School Budget": totalSchoolBudget,
                                         "Per Student Budget": perStudentBudget,
                                         "Average Math Score": averageMathScore,
                                         "Average Reading Score": averageReadingScore,
                                         "% Passing Math": percentagePassingMath,
                                         "% Passing Reading": percentagePassingReading,
                                         "% Overall Passing Rate": percentageOverallPassingRate})
school_summary_df
```

Out[45]:

<u>school_name</u>	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% F
Bailey High School	District	4976	3124928	628.0	77.048432	81.033963	66.680064	81.933280	74.
Cabrera High School	Charter	1858	1081356	582.0	83.061895	83.975780	94.133477	97.039828	95.
Figueroa High School	District	2949	1884411	639.0	76.711767	81.158020	65.988471	80.739234	73.
Ford High School	District	2739	1763916	644.0	77.102592	80.746258	68.309602	79.299014	73.
Griffin High School	Charter	1468	917500	625.0	83.351499	83.816757	93.392371	97.138965	95.
Hernandez High School	District	4635	3022020	652.0	77.289752	80.934412	66.752967	80.862999	73.
Holden High School	Charter	427	248087	581.0	83.803279	83.814988	92.505855	96.252927	94.
Huang High School	District	2917	1910635	655.0	76.629414	81.182722	65.683922	81.316421	73.
Johnson High School	District	4761	3094650	650.0	77.072464	80.966394	66.057551	81.222432	73.
Peña High School	Charter	962	585858	609.0	83.839917	84.044699	94.594595	95.945946	95.
Rodriguez High School	District	3999	2547363	637.0	76.842711	80.744686	66.366592	80.220055	73.
Shelton High School	Charter	1761	1056600	600.0	83.359455	83.725724	93.867121	95.854628	94.
Thomas High School	Charter	1635	1043130	638.0	83.418349	83.848930	93.272171	97.308869	95.
Wilson High School	Charter	2283	1319574	578.0	83.274201	83.989488	93.867718	96.539641	95.
Wright High School	Charter	1800	1049400	583.0	83.682222	83.955000	93.333333	96.611111	94.

In [46]: school_summary_df = school_summary_df.sort_values(["% Overall Passing Rate"], ascending=False)

Out[46]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% F
school_name									
Cabrera High School	Charter	1858	1081356	582.0	83.061895	83.975780	94.133477	97.039828	95.
Thomas High School	Charter	1635	1043130	638.0	83.418349	83.848930	93.272171	97.308869	95.
Pena High School	Charter	962	585858	609.0	83.839917	84.044699	94.594595	95.945946	95.
Griffin High School	Charter	1468	917500	625.0	83.351499	83.816757	93.392371	97.138965	95.
Wilson High School	Charter	2283	1319574	578.0	83.274201	83.989488	93.867718	96.539641	95.
Wright High School	Charter	1800	1049400	583.0	83.682222	83.955000	93.333333	96.611111	94.
Shelton High School	Charter	1761	1056600	600.0	83.359455	83.725724	93.867121	95.854628	94.
Holden High School	Charter	427	248087	581.0	83.803279	83.814988	92.505855	96.252927	94.
Bailey High School	District	4976	3124928	628.0	77.048432	81.033963	66.680064	81.933280	74.
Hernandez High School	District	4635	3022020	652.0	77.289752	80.934412	66.752967	80.862999	73.
Ford High School	District	2739	1763916	644.0	77.102592	80.746258	68.309602	79.299014	73.
Johnson High School	District	4761	3094650	650.0	77.072464	80.966394	66.057551	81.222432	73.
Huang High School	District	2917	1910635	655.0	76.629414	81.182722	65.683922	81.316421	73.
Figueroa High School	District	2949	1884411	639.0	76.711767	81.158020	65.988471	80.739234	73.
Rodriguez High School	District	3999	2547363	637.0	76.842711	80.744686	66.366592	80.220055	73.

```
In [18]: school_summary_df[["School Type",
    "Total Students",
    "Total School Budget",
    "Per Student Budget",
    "Average Math Score",
    "Average Reading Score",
    "% Passing Math",
    "% Passing Reading",
    "% Overall Passing Rate"]].head()
```

Out[18]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% F
school_name									
Cabrera High School	Charter	1858	1081356	582.0	83.061895	83.975780	94.133477	97.039828	95.
Thomas High School	Charter	1635	1043130	638.0	83.418349	83.848930	93.272171	97.308869	95.
Pena High School	Charter	962	585858	609.0	83.839917	84.044699	94.594595	95.945946	95.
Griffin High School	Charter	1468	917500	625.0	83.351499	83.816757	93.392371	97.138965	95.
Wilson High School	Charter	2283	1319574	578.0	83.274201	83.989488	93.867718	96.539641	95.

Bottom Performing Schools (By Passing Rate)

- Sort and display the five worst-performing schools

```
In [19]: school_summary_df = school_summary_df.sort_values(["% Overall Passing Rate"], ascending=False)

school_summary_df[["School Type",
                   "Total Students",
                   "Total School Budget",
                   "Per Student Budget",
                   "Average Math Score",
                   "Average Reading Score",
                   "% Passing Math",
                   "% Passing Reading",
                   "% Overall Passing Rate"]].head()
```

Out[19]:

school_name	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	F
Rodriguez High School	District	3999	2547363	637.0	76.842711	80.744686	66.366592	80.220055	73.
Figueroa High School	District	2949	1884411	639.0	76.711767	81.158020	65.988471	80.739234	73.
Huang High School	District	2917	1910635	655.0	76.629414	81.182722	65.683922	81.316421	73.
Johnson High School	District	4761	3094650	650.0	77.072464	80.966394	66.057551	81.222432	73.
Ford High School	District	2739	1763916	644.0	77.102592	80.746258	68.309602	79.299014	73.

Math Scores by Grade

- Create a table that lists the average Reading Score for students of each grade level (9th, 10th, 11th, 12th) at each school.
 - Create a pandas series for each grade. Hint: use a conditional statement.
 - Group each series by school
 - Combine the series into a dataframe
 - Optional: give the displayed data cleaner formatting

```
In [25]: grade_summary_df = pd.DataFrame({"9th": grade9th_ds,  
                                         "10th": grade10th_ds,  
                                         "11th": grade11th_ds,  
                                         "12th": grade12th_ds})  
  
grade_summary_df[["9th", "10th", "11th", "12th"]]
```

Out[25]:

	9th	10th	11th	12th
school_name				
Bailey High School	77.083676	76.996772	77.515588	76.492218
Cabrera High School	83.094697	83.154506	82.765560	83.277487
Figueroa High School	76.403037	76.539974	76.884344	77.151369
Ford High School	77.361345	77.672316	76.918058	76.179963
Griffin High School	82.044010	84.229064	83.842105	83.356164
Hernandez High School	77.438495	77.337408	77.136029	77.186567
Holden High School	83.787402	83.429825	85.000000	82.855422
Huang High School	77.027251	75.908735	76.446602	77.225641
Johnson High School	77.187857	76.691117	77.491653	76.863248
Pena High School	83.625455	83.372000	84.328125	84.121547
Rodriguez High School	76.859966	76.612500	76.395626	77.690748
Shelton High School	83.420755	82.917411	83.383495	83.778976
Thomas High School	83.590022	83.087886	83.498795	83.497041
Wilson High School	83.085578	83.724422	83.195326	83.035794
Wright High School	83.264706	84.010288	83.836782	83.644986

```
In [47]: grade9th_ds = school_data_complete.loc[school_data_complete["grade"] == "9th"].gr  
grade9th_ds
```

```
Out[47]: school_name  
Bailey High School      77.083676  
Cabrera High School    83.094697  
Figueroa High School   76.403037  
Ford High School       77.361345  
Griffin High School   82.044010  
Hernandez High School  77.438495  
Holden High School     83.787402  
Huang High School     77.027251  
Johnson High School    77.187857  
Pena High School       83.625455  
Rodriguez High School  76.859966  
Shelton High School   83.420755  
Thomas High School     83.590022  
Wilson High School    83.085578  
Wright High School    83.264706  
Name: math_score, dtype: float64
```

```
In [48]: grade10th_ds = school_data_complete.loc[school_data_complete["grade"] == "10th"].gr  
grade10th_ds
```

```
Out[48]: school_name  
Bailey High School      76.996772  
Cabrera High School    83.154506  
Figueroa High School   76.539974  
Ford High School       77.672316  
Griffin High School   84.229064  
Hernandez High School  77.337408  
Holden High School     83.429825  
Huang High School     75.908735  
Johnson High School    76.691117  
Pena High School       83.372000  
Rodriguez High School  76.612500  
Shelton High School   82.917411  
Thomas High School    83.087886  
Wilson High School    83.724422  
Wright High School    84.010288  
Name: math_score, dtype: float64
```

```
In [49]: grade11th_ds = school_data_complete.loc[school_data_complete["grade"] == "11th"]  
grade11th_ds
```

```
Out[49]: school_name  
Bailey High School      77.515588  
Cabrera High School    82.765560  
Figueroa High School   76.884344  
Ford High School       76.918058  
Griffin High School   83.842105  
Hernandez High School  77.136029  
Holden High School    85.000000  
Huang High School     76.446602  
Johnson High School   77.491653  
Pena High School      84.328125  
Rodriguez High School 76.395626  
Shelton High School   83.383495  
Thomas High School    83.498795  
Wilson High School   83.195326  
Wright High School   83.836782  
Name: math_score, dtype: float64
```

```
In [50]: grade12th_ds = school_data_complete.loc[school_data_complete["grade"] == "12th"]  
grade12th_ds
```

```
Out[50]: school_name  
Bailey High School      76.492218  
Cabrera High School    83.277487  
Figueroa High School   77.151369  
Ford High School       76.179963  
Griffin High School   83.356164  
Hernandez High School  77.186567  
Holden High School    82.855422  
Huang High School     77.225641  
Johnson High School   76.863248  
Pena High School      84.121547  
Rodriguez High School 77.690748  
Shelton High School   83.778976  
Thomas High School    83.497041  
Wilson High School   83.035794  
Wright High School   83.644986  
Name: math_score, dtype: float64
```

Reading Score by Grade

- Perform the same operations as above for reading scores

```
In [30]: grade_summary_df2 = pd.DataFrame({"9th": grade9th_ds2,  
                                         "10th": grade10th_ds2,  
                                         "11th": grade11th_ds2,  
                                         "12th": grade12th_ds2})  
  
grade_summary_df2[["9th", "10th", "11th", "12th"]]
```

Out[30]:

	9th	10th	11th	12th
school_name				
Bailey High School	81.303155	80.907183	80.945643	80.912451
Cabrera High School	83.676136	84.253219	83.788382	84.287958
Figueroa High School	81.198598	81.408912	80.640339	81.384863
Ford High School	80.632653	81.262712	80.403642	80.662338
Griffin High School	83.369193	83.706897	84.288089	84.013699
Hernandez High School	80.866860	80.660147	81.396140	80.857143
Holden High School	83.677165	83.324561	83.815534	84.698795
Huang High School	81.290284	81.512386	81.417476	80.305983
Johnson High School	81.260714	80.773431	80.616027	81.227564
Pena High School	83.807273	83.612000	84.335938	84.591160
Rodriguez High School	80.993127	80.629808	80.864811	80.376426
Shelton High School	84.122642	83.441964	84.373786	82.781671
Thomas High School	83.728850	84.254157	83.585542	83.831361
Wilson High School	83.939778	84.021452	83.764608	84.317673
Wright High School	83.833333	83.812757	84.156322	84.073171

```
In [51]: grade9th_ds2 = school_data_complete.loc[school_data_complete["grade"] == "9th"].  
grade9th_ds2
```

```
Out[51]: school_name  
Bailey High School      81.303155  
Cabrera High School    83.676136  
Figueroa High School   81.198598  
Ford High School       80.632653  
Griffin High School   83.369193  
Hernandez High School  80.866860  
Holden High School     83.677165  
Huang High School     81.290284  
Johnson High School    81.260714  
Pena High School       83.807273  
Rodriguez High School  80.993127  
Shelton High School   84.122642  
Thomas High School     83.728850  
Wilson High School    83.939778  
Wright High School    83.833333  
Name: reading_score, dtype: float64
```

```
In [52]: grade10th_ds2 = school_data_complete.loc[school_data_complete["grade"] == "10th"].  
grade10th_ds2
```

```
Out[52]: school_name  
Bailey High School      80.907183  
Cabrera High School    84.253219  
Figueroa High School   81.408912  
Ford High School       81.262712  
Griffin High School   83.706897  
Hernandez High School  80.660147  
Holden High School     83.324561  
Huang High School     81.512386  
Johnson High School    80.773431  
Pena High School       83.612000  
Rodriguez High School  80.629808  
Shelton High School   83.441964  
Thomas High School    84.254157  
Wilson High School    84.021452  
Wright High School    83.812757  
Name: reading_score, dtype: float64
```

```
In [53]: grade11th_ds2 = school_data_complete.loc[school_data_complete["grade"] == "11th"]
grade11th_ds2
```

```
Out[53]: school_name
Bailey High School      80.945643
Cabrera High School    83.788382
Figueroa High School   80.640339
Ford High School       80.403642
Griffin High School    84.288089
Hernandez High School  81.396140
Holden High School     83.815534
Huang High School      81.417476
Johnson High School    80.616027
Pena High School       84.335938
Rodriguez High School  80.864811
Shelton High School    84.373786
Thomas High School     83.585542
Wilson High School     83.764608
Wright High School     84.156322
Name: reading_score, dtype: float64
```

```
In [54]: grade12th_ds2 = school_data_complete.loc[school_data_complete["grade"] == "12th"]
grade12th_ds2
```

```
Out[54]: school_name
Bailey High School      80.912451
Cabrera High School    84.287958
Figueroa High School   81.384863
Ford High School       80.662338
Griffin High School    84.013699
Hernandez High School  80.857143
Holden High School     84.698795
Huang High School      80.305983
Johnson High School    81.227564
Pena High School       84.591160
Rodriguez High School  80.376426
Shelton High School    82.781671
Thomas High School     83.831361
Wilson High School     84.317673
Wright High School     84.073171
Name: reading_score, dtype: float64
```

Scores by School Spending

- Create a table that breaks down school performances based on average Spending Ranges (Per Student). Use 4 reasonable bins to group school spending. Include in the table each of the following:
 - Average Math Score
 - Average Reading Score

- % Passing Math
- % Passing Reading
- Overall Passing Rate (Average of the above two)

In [55]: `# Sample bins. Feel free to create your own bins.`

```
spending_bins = [0, 585, 615, 645, 675]
group_names = ["<$585", "$585-615", "$615-645", "$645-675"]
```

In [56]: `school_data_complete['spending_bins'] = pd.cut(school_data_complete['budget']/sch`

```
school_data_complete['spending_bins']
```

Out[56]: 0 \$645-675

1 \$645-675

2 \$645-675

3 \$645-675

4 \$645-675

...

39165 \$615-645

39166 \$615-645

39167 \$615-645

39168 \$615-645

39169 \$615-645

Name: spending_bins, Length: 39170, dtype: category

Categories (4, object): ['\$<585' < '\$585-615' < '\$615-645' < '\$645-675']

In [57]: `by_spending = school_data_complete.groupby('spending_bins')`

```
by_spending
```

Out[57]: <pandas.core.groupby.generic.DataFrameGroupBy object at 0x000002AE6B9D6CD0>

In [58]: `avg_math = by_spending['math_score'].mean()`

```
avg_math
```

Out[58]: `spending_bins`

<\$585	83.363065
--------	-----------

\$585-615	83.529196
-----------	-----------

\$615-645	78.061635
-----------	-----------

\$645-675	77.049297
-----------	-----------

Name: math_score, dtype: float64

In [59]: `pass_math = school_data_complete[school_data_complete['math_score'] >= 70].groupby('size')`

Out[59]: `spending_bins`

<\$585	93.702889
\$585-615	94.124128
\$615-645	71.400428
\$645-675	66.230813

 Name: Student ID, dtype: float64

In [60]: `pass_read = school_data_complete[school_data_complete['reading_score'] >= 70].groupby('size')`

Out[60]: `spending_bins`

<\$585	96.686558
\$585-615	95.886889
\$615-645	83.614770
\$645-675	81.109397

 Name: Student ID, dtype: float64

In [61]: `overall = school_data_complete[(school_data_complete['reading_score'] >= 70) & (school_data_complete['math_score'] >= 70)].groupby('size')`

Out[61]: <pandas.core.groupby.generic.SeriesGroupBy object at 0x000002AE6B98AA60>

Scores by School Size

- Perform the same operations as above, based on school size.

In [63]: `# Sample bins. Feel free to create your own bins.`
`size_bins = [0, 1000, 2000, 5000]`
`group_names = ["Small (<1000)", "Medium (1000-2000)", "Large (2000-5000)"]`

In [64]: `school_data_complete['size_bins'] = pd.cut(school_data_complete['size'], size_bins)`
`by_size = school_data_complete.groupby('size_bins')`
`by_size`

Out[64]: <pandas.core.groupby.generic.DataFrameGroupBy object at 0x000002AE6B9DBA00>

Look for the total count of test scores that pass 70% or higher

```
In [65]: math_pass_size = school_data_complete[school_data_complete['math_score'] >= 70].size  
math_pass_size
```

```
◀ ━━━━━━ ▶
```

```
Out[65]: size_bins  
Small (<1000)      0.939525  
Medium (1000-2000)  0.936165  
Large (2000-5000)   0.686524  
Name: Student ID, dtype: float64
```

```
In [66]: read_pass_size = school_data_complete[school_data_complete['reading_score'] >= 70].size  
read_pass_size
```

```
◀ ━━━━━━ ▶
```

```
Out[66]: size_bins  
Small (<1000)      0.960403  
Medium (1000-2000)  0.967731  
Large (2000-5000)   0.821252  
Name: Student ID, dtype: float64
```

```
In [67]: overall = school_data_complete[(school_data_complete['reading_score'] >= 70) & (school_data_complete['math_score'] >= 70)].size  
overall
```

```
◀ ━━━━━━ ▶
```

```
Out[67]: size_bins  
Small (<1000)      125200  
Medium (1000-2000)  772300  
Large (2000-5000)   1655300  
Name: Student ID, dtype: int64
```

Scores by School Type

- Perform the same operations as above, based on school type.

```
In [68]: school_type = school_data_complete.groupby("type")  
school_type
```

```
◀ ━━━━━━ ▶
```

```
Out[68]: <pandas.core.groupby.generic.DataFrameGroupBy object at 0x000002AE6BB3B250>
```

Find counts of the passing 70 or higher score for the both test

```
In [69]: pass_math = school_data_complete[school_data_complete['math_score'] >= 70].groupby('type').size()
```

```
Out[69]: type
Charter      0.937018
District     0.665184
Name: Student ID, dtype: float64
```

```
In [70]: pass_read = school_data_complete[school_data_complete['reading_score'] >= 70].groupby('type').size()
```

```
Out[70]: type
Charter      0.966459
District     0.809052
Name: Student ID, dtype: float64
```

```
In [71]: overall = school_data_complete[(school_data_complete['reading_score'] >= 70) & (school_data_complete['math_score'] >= 70)].groupby('type').size()
```

```
Out[71]: type
Charter      1104300
District     1448500
Name: Student ID, dtype: int64
```

```
In [72]: scores_school_type = pd.DataFrame({
    'Total Passing Math': pass_math,
    'Total Passing Reading': pass_read,
    "% Overall Passing": overall})
scores_school_type
```

```
Out[72]:
Total Passing Math  Total Passing Reading  % Overall Passing
type
-----
```

type	Total Passing Math	Total Passing Reading	% Overall Passing
Charter	0.937018	0.966459	1104300
District	0.665184	0.809052	1448500

```
In [73]: scores_school_type.index.name = "Type of School"
```

In [74]: scores_school_type

Out[74]:

Type of School	Total Passing Math	Total Passing Reading	% Overall Passing
----------------	--------------------	-----------------------	-------------------

Type of School	Total Passing Math	Total Passing Reading	% Overall Passing
Charter	0.937018	0.966459	1104300
District	0.665184	0.809052	1448500

In []: