



```
In [1]: import pandas as pd

school_data_to_load = "C:\\Users\\me\\Desktop\\BBC\\COPY_but-ind-data-pt-06-2020-u-c\\Unit_04_Pandas\\Homework\\PyCitySchools\\school_data_to_load.csv"
student_data_to_load = "C:\\Users\\me\\Desktop\\BBC\\COPY_but-ind-data-pt-06-2020-u-c\\Unit_04_Pandas\\Homework\\PyCitySchools\\student_data_to_load.csv"

school_data = pd.read_csv(school_data_to_load)
student_data = pd.read_csv(student_data_to_load)

school_data_complete = pd.merge(student_data, school_data, how="left", on=["school_name", "school_name"])
```

```
In [2]: school_data_complete["school_name"].unique()

Out[2]: array(['Huang High School', 'Figueroa High School', 'Shelton High School',
              'Hernandez High School', 'Griffin High School',
              'Wilson High School', 'Cabrera High School', 'Bailey High School',
              'Holden High School', 'Pena High School', 'Wright High School',
              'Rodriguez High School', 'Johnson High School', 'Ford High School',
              'Thomas High School'], dtype=object)
```

```
In [3]: total_schools = school_data_complete["school_name"].unique()
total_school_count = len(total_schools)
```

```
In [4]: total_kids = school_data_complete["student_name"].count()
```

```
In [5]: total_budget = school_data["budget"].sum()
```

```
In [6]: average_math = school_data_complete["math_score"].mean() / 100
average_reading = school_data_complete["reading_score"].mean() / 100
```

```
In [8]: math_bins = 0, 69.9, 100
math_names = ["Failing", "Passing"]
Math_df = pd.cut(school_data_complete["math_score"], math_bins, labels=math_names, include_lowest=True)
total_mpass_scores = Math_df.value_counts("Passing")
```

```
In [9]: reading_bins = 0, 69.9, 100
reading_names = ["Failing", "Passing"]
Reading_df = pd.cut(school_data_complete["reading_score"], reading_bins, labels=reading_names, include_lowest=True)
total_rpass_scores = Reading_df.value_counts("Passing")
```

PyCitySchools_OFFICAL SUBMISS

+

localhost:8888/notebooks/Unit_04_Pandas/Homework/PyCitySchools/PyCitySchools_OFFICAL%20SUBMISSION_WaycottChristina.ipynb

Vyatt Mom WGU WGU Inbox All Mail 47 Centene BBC

jupyter

PyCitySchools_OFFICAL SUBMISSION_WaycottChristina (autosaved)

Logout

File Edit View Insert Cell Kernel Help

Not Trusted Python 3

+

⬆

⬇

▶ Run

⏏

↺

▶▶

Markdown

In [10]:

total_math_pass = total_mpass_scores["Passing"]
total_reading_pass = total_rpass_scores["Passing"]

In [11]:

overall = school_data_complete[(school_data_complete["math_score"] >= 70)
& (school_data_complete["reading_score"] >= 70)].count()["student_name"]
overall_percent = overall / total_kids

In [12]:

district_summary = pd.DataFrame({"Total Schools": [total_school_count],
"Total Students": [total_kids],
"Total Budget": [total_budget],
"Average Math Score": [average_math],
"Average Reading Score": [average_reading],
"% Passing Math": [total_math_pass],
"% Passing Reading": [total_reading_pass],
"% Overall Passing": [overall_percent]})

district_summary

Out[12]:

	Total Schools	Total Students	Total Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing
0	15	39170	24649428	0.789854	0.818778	0.749809	0.858055	0.651723

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 percentage of students with a passing math score (70 or greater)
- Calculate the percentage of students with a passing reading score (70 or greater)
- Calculate the percentage of students who passed math and reading (% Overall Passing)
- Create a dataframe to hold the above results
- Optional: give the displayed data cleaner formatting

In [5]:

###DO NOT USE!!!!!!!!!!!!!!!###

Out[5]:

	Total Schools	Total Students	Total Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing
0	15	39,170	\$24,649,428.00	78.985371	81.87784	74.980853	85.805463	65.172326

- 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 (The percentage of students that passed math and reading.)
- Create a dataframe to hold the above results

```
In [13]: total_students = school_data_complete.groupby(["school_name"])["student_name"].count()
school_budget = school_data_complete.groupby(["school_name"])["budget"].mean()
per school budget = school budget / total students
```

```
In [14]: average_math = school_data_complete.groupby(["school_name"])["math_score"].mean()
         average_read = school_data_complete.groupby(["school_name"])["reading_score"].mean()
```

```
In [15]: math_score = school_data_complete[(school_data_complete["math_score"] >= 70)]
math_percent = math_score.groupby(["school_name"])["math score"].count() / total_students
```

```
In [16]: read_score = school_data_complete[(school_data_complete["reading_score"] >= 70)]
         read_percent = read_score.groupby(["school_name"])["reading_score"].count() / total_students
```

```
In [17]: overall_score = school_data_complete[(school_data_complete["math_score"] >= 70)
        & (school_data_complete["reading_score"] >= 70)]
        overall_percent = overall_score.groupby(["school name"])["student name"].count() / total_students
```

```
In [18]: s_types = school_data.set_index(["school_name"])["type"]
```

[illegible]

work/PyCity x PyCitySchools_OFFICAL SUBMISS x +

localhost:8888/notebooks/Unit_04_Pandas/Homework/PyCitySchools/PyCitySchools_OFFICAL%20SUBMISSION_WaycottChristina.ipynb

Wyatt Mom WGU WGU Inbox All Mail 47 Centene BBC

jupyter PyCitySchools_OFFICAL SUBMISSION_WaycottChristina (autosaved)

Python 3

Logout

File Edit View Insert Cell Kernel Help

Not Trusted | Python 3

In [21]:

school_summary = pd.DataFrame({"School Type": s_types,
"Total Students": total_students,
"Total School Budget": school_budget,
"Per Student Budget": per_school_budget,
"Average Math Score": average_math,
"Average Reading Score": average_read,
"% Passing Math": math_percent,
"% Passing Reading": read_percent,
"% Overall Passing": overall_percent})

school_summary

Out[21]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing	
	Bailey High School	District	4976	3124928	628.0	77.048432	81.033963	0.666801	0.819333	0.546423
	Cabrera High School	Charter	1858	1081356	582.0	83.061895	83.975780	0.941335	0.970398	0.913348
	Figueroa High School	District	2949	1884411	639.0	76.711767	81.158020	0.659885	0.807392	0.532045
	Ford High School	District	2739	1763916	644.0	77.102592	80.746258	0.683096	0.792990	0.542899
	Griffin High School	Charter	1468	917500	625.0	83.351499	83.816757	0.933924	0.971390	0.905995
	Hernandez High School	District	4635	3022020	652.0	77.289752	80.934412	0.667530	0.808630	0.535275
	Holden High School	Charter	427	248087	581.0	83.803279	83.814988	0.925059	0.962529	0.892272
	Huang High School	District	2917	1910635	655.0	76.629414	81.182722	0.656839	0.813164	0.535139
	Johnson High School	District	4761	3094650	650.0	77.072464	80.966394	0.680576	0.812224	0.535392
	Pena High School	Charter	962	585858	609.0	83.839917	84.044099	0.945946	0.959459	0.905405
	Rodriguez High School	District	3999	2547383	637.0	76.842711	80.744686	0.663666	0.802201	0.529882
	Shelton High School	Charter	1761	1056600	600.0	83.359455	83.725724	0.938671	0.958546	0.898921
	Thomas High School	Charter	1635	1043130	638.0	83.418349	83.848930	0.932722	0.973089	0.909480
	Wilson High School	Charter	2283	1319574	578.0	83.274201	83.989488	0.938677	0.965396	0.905826
	Wright High School	Charter	1800	1049400	583.0	83.682222	83.955000	0.933333	0.966111	0.903333

In [9]:

DO NOT USE!!!!!!!!!!!!!!###

Out[9]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing	
	Bailey High School	District	4976	\$3,124,928.00	\$628.00	77.048432	81.033963	66.680064	81.933280	54.642283

Top Performing Schools (By % Overall Passing)

- Sort and display the top five performing schools by % overall passing.

In [22]: `school_summary = school_summary.sort_values("% Overall Passing", ascending=False)
school_summary.head()`

Out[22]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing
Cabrera High School	Charter	1858	1081356	582.0	83.061895	83.975780	0.941335	0.970398	0.913348
Thomas High School	Charter	1635	1043130	638.0	83.418349	83.848930	0.932722	0.973089	0.909480
Griffin High School	Charter	1468	917500	625.0	83.351499	83.816757	0.933924	0.971390	0.905995
Wilson High School	Charter	2283	1319574	578.0	83.274201	83.989488	0.938677	0.965396	0.905826
Pena High School	Charter	982	585858	609.0	83.839917	84.044699	0.945946	0.959459	0.905405

In [10]: `###DO NOT USE!!!!!!!!!!!!!!!###`

Out[10]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing
Cabrera High School	Charter	1858	\$1,081,356.00	\$582.00	83.061895	83.975780	94.133477	97.039828	91.334789
Thomas High School	Charter	1635	\$1,043,130.00	\$638.00	83.418349	83.848930	93.272171	97.308889	90.948012
Griffin High School	Charter	1468	\$917,500.00	\$625.00	83.351499	83.816757	93.392371	97.138965	90.599455
Wilson High School	Charter	2283	\$1,319,574.00	\$578.00	83.274201	83.989488	93.867718	96.539641	90.582587
Pena High School	Charter	982	\$585,858.00	\$609.00	83.839917	84.044699	94.594595	95.945946	90.540541

Bottom Performing Schools (By % Overall Passing)

- Sort and display the five worst-performing schools by % overall passing.

```
In [23]: school_summary = school_summary.sort_values("% Overall Passing")
school_summary.head()
```

Out[23]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing
Rodriguez High School	District	3999	2547363	637.0	76.842711	80.744686	0.663666	0.802201	0.529882
Figueroa High School	District	2949	1884411	639.0	76.711767	81.158020	0.659885	0.807392	0.532045
Huang High School	District	2917	1910635	655.0	76.629414	81.182722	0.656839	0.813164	0.535139
Hernandez High School	District	4635	3022020	652.0	77.289752	80.934412	0.667530	0.808630	0.535275
Johnson High School	District	4761	3094650	650.0	77.072464	80.986394	0.660576	0.812224	0.535392

```
In [11]: ###DO NOT USE!!!!!!!!!!!!!!!###
```

Out[11]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing
Rodriguez High School	District	3999	\$2,547,363.00	\$637.00	76.842711	80.744686	66.366592	80.220055	52.988247
Figueroa High School	District	2949	\$1,884,411.00	\$639.00	76.711767	81.158020	65.988471	80.739234	53.204476
Huang High School	District	2917	\$1,910,635.00	\$655.00	76.629414	81.182722	65.683922	81.316421	53.513884
Hernandez High School	District	4635	\$3,022,020.00	\$652.00	77.289752	80.934412	66.752967	80.862999	53.527508
Johnson High School	District	4761	\$3,094,650.00	\$650.00	77.072464	80.986394	66.057551	81.222432	53.539172

- ```
In [24]: ▶ ninth = school_data_complete[(school_data_complete["grade"] == "9th")]
 tenth = school_data_complete[(school_data_complete["grade"] == "10th")]
 eleventh = school_data_complete[(school_data_complete["grade"] == "11th")]
 twelfth = school_data_complete[(school_data_complete["grade"] == "12th")]

In [25]: ▶ ninth_math_score = ninth.groupby(["school_name"])["math_score"].mean()
 tenth_math_score = tenth.groupby(["school_name"])["math_score"].mean()
 eleventh_math_score = eleventh.groupby(["school_name"])["math_score"].mean()
 twelfth_math_score = twelfth.groupby(["school_name"])["math_score"].mean()

In [26]: ▶ math_summary = pd.DataFrame({"9th": ninth_math_score,
 "10th": tenth_math_score,
 "11th": eleventh_math_score,
 "12th": twelfth_math_score})
 math_summary
```

Out[26]:

|                       | 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      | 77.381345 | 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 | 77.188567 |
| 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 | 83.836762 | 83.644968 |

## Reading Score by Grade

- Perform the same operations as above for reading scores

```
In [28]: ▶ r ninth = school_data_complete[(school_data_complete["grade"] == "9th")]
 ▶ rtenth = school_data_complete[(school_data_complete["grade"] == "10th")]
 ▶ releventh = school_data_complete[(school_data_complete["grade"] == "11th")]
 ▶ rtwelfth = school_data_complete[(school_data_complete["grade"] == "12th")]
```

```
In [29]: ▶ ninth_reading_score = rninth.groupby(["school_name"])["reading_score"].mean()
 ▶ tenth_reading_score = rtenth.groupby(["school_name"])["reading_score"].mean()
 ▶ eleventh_reading_score = releventh.groupby(["school_name"])["reading_score"].mean()
 ▶ twelfth_reading_score = rtwelfth.groupby(["school_name"])["reading_score"].mean()
```

```
In [30]: ▶ reading_summary = pd.DataFrame({"9th": ninth_reading_score,
 ▶ "10th": tenth_reading_score,
 ▶ "11th": eleventh_reading_score,
 ▶ "12th": twelfth_reading_score})
 ▶ reading_summary
```

```
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 | 81.384883 |
| 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 | 80.680147 | 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   | 81.280714 | 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 | 83.441984 | 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 |



- 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 [44]: spending_data = pd.DataFrame({"School Type": s_types,
 "Total Students": total_students,
 "Total School Budget": school_budget,
 "Per Student Budget": per_school_budget,
 "Average Math Score": average_math,
 "Average Reading Score": average_read,
 "% Passing Math": math_percent,
 "% Passing Reading": read_percent,
 "% Overall Passing": overall_percent}))
```

```
In [36]: student_bins = [0, 584, 629, 644, 675]
student_bins_names = ["<$584", "$585-629", "$630-644", "$645-675"]
```

```
In [59]: spending_data["Spending Ranges (Per Student)"] = pd.cut(spending_data["Per Student Budget"], student_bins, labels = student
```

```
In [60]: spending_math_scores = spending_data.groupby(["Spending Ranges (Per Student)"]).mean()["Average Math Score"]
spending_read_scores = spending_data.groupby(["Spending Ranges (Per Student)"]).mean()["Average Reading Score"]
spending_math_per = spending_data.groupby(["Spending Ranges (Per Student)"]).mean()["% Passing Math"]
spending_read_per = spending_data.groupby(["Spending Ranges (Per Student)"]).mean()["% Passing Reading"]
spending_overall_per = spending_data.groupby(["Spending Ranges (Per Student)"]).mean()["% Overall Passing"]
```

```
In [61]: spending_summary = pd.DataFrame({"Average Math Score": spending_math_scores,
 "Average Reading Score": spending_read_scores,
 "% Passing Math": spending_math_per,
 "% Passing Reading": spending_read_per,
 "% Overall Passing": spending_overall_per})

spending_summary
```

Out[61]:

| Average Math Score | Average Reading Score | % Passing Math | % Passing Reading | % Overall Passing |
|--------------------|-----------------------|----------------|-------------------|-------------------|
|--------------------|-----------------------|----------------|-------------------|-------------------|

| Spending Ranges (Per Student) |           |           |          |          |          |  |
|-------------------------------|-----------|-----------|----------|----------|----------|--|
| <\$584                        | 83.455399 | 83.933814 | 0.934601 | 0.986109 | 0.903695 |  |
| \$585-629                     | 81.899826 | 83.155286 | 0.871335 | 0.927182 | 0.814186 |  |
| \$630-644                     | 78.518855 | 81.624473 | 0.734842 | 0.843918 | 0.628577 |  |
| \$645-675                     | 76.997210 | 81.027843 | 0.661648 | 0.811340 | 0.535299 |  |

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

```
In [49]: size_students = school_data_complete.groupby(["size"])[["student_name"].count()
size_av_math = school_data_complete.groupby(["size"])[["math_score"].mean()
size_av_read = school_data_complete.groupby(["size"])[["reading_score"].mean()
size_math_score = school_data_complete[school_data_complete["math_score"] >= 70]
size_math_percent = size_math_score.groupby(["size"])[["math_score"].count() / size_students
size_read_score = school_data_complete[school_data_complete["reading_score"] >= 70]
size_read_percent = size_read_score.groupby(["size"])[["reading_score"].count() / size_students
size_overall = school_data_complete[school_data_complete["math_score"] >= 70
& (school_data_complete["reading_score"] >= 70)]
size_overall_percent = size_overall.groupby(["size"])[["student_name"].count() / size_students
school_size = school_data_complete.groupby(["size"])[["school_name"].count()]
```

```
In [56]: size_data = pd.DataFrame({"Average Math Score": size_av_math,
"Average Reading Score": size_av_read,
"% Passing Math": size_math_percent,
"% Passing Reading": size_read_percent,
"% Overall Passing": size_overall_percent,
"School Size": school_size})
```

```
In [57]: size_bins = [0, 1000, 2000, 5000]
size_bins_names = ["Small (<1000)", "Medium (1000-2000)", "Large (2000-5000)"]
size_data["School Size"] = pd.cut(size_data["School Size"], size_bins, labels = size_bins_names)
```

```
In [54]: size_math_scores = size_data.groupby(["School Size"]).mean()["Average Math Score"]
size_read_scores = size_data.groupby(["School Size"]).mean()["Average Reading Score"]
size_math_per = size_data.groupby(["School Size"]).mean()["% Passing Math"]
size_read_per = size_data.groupby(["School Size"]).mean()["% Passing Reading"]
size_overall_per = size_data.groupby(["School Size"]).mean()["% Overall Passing"]
```

```
In [58]: size_summary = pd.DataFrame({"Average Math Score": size_math_scores,
"Average Reading Score": size_read_scores,
"% Passing Math": size_math_per,
"% Passing Reading": size_read_per,
"% Overall Passing": size_overall_per})
size_summary
```

```
Out[58]:
```

|                    | Average Math Score | Average Reading Score | % Passing Math | % Passing Reading | % Overall Passing |
|--------------------|--------------------|-----------------------|----------------|-------------------|-------------------|
| School Size        |                    |                       |                |                   |                   |
| Small (<1000)      | 83.821598          | 83.929843             | 0.935502       | 0.960994          | 0.898839          |
| Medium (1000-2000) | 83.374684          | 83.864438             | 0.935997       | 0.967907          | 0.906215          |
| Large (2000-5000)  | 77.746417          | 81.344493             | 0.699834       | 0.827666          | 0.582880          |

## Scores by School Type

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

```
In [64]: type_students = school_data_complete.groupby(["type"])["student_name"].count()
type_math_average = school_data_complete.groupby(["type"])["math_score"].mean()
type_reading_average = school_data_complete.groupby(["type"])["reading_score"].mean()
type_math_scores = school_data_complete[school_data_complete["math_score"] >= 70]
type_math_percent = type_math_scores.groupby(["type"])["math_score"].count() / type_students
type_reading_scores = school_data_complete[school_data_complete["reading_score"] >= 70]
type_reading_percent = type_reading_scores.groupby(["type"])["math_score"].count() / type_students
type_overall = school_data_complete[(school_data_complete["math_score"] >= 70)
& (school_data_complete["reading_score"] >= 70)]
type_overall_passing = type_overall.groupby(["type"])["student_name"].count() / type_students
type_s_types = school_data.set_index(["type"])["size"]
```

```
In [65]: type_summary = pd.DataFrame({"Average Math Score": type_math_average,
"Average Reading Score": type_reading_average,
"Math Percentage": type_math_percent,
"Reading Percentage": type_reading_percent,
"Overall Passing Percentage": type_overall_passing})
type_summary
```

```
Out[65]:
```

|          | Average Math Score | Average Reading Score | Math Percentage | Reading Percentage | Overall Passing Percentage |
|----------|--------------------|-----------------------|-----------------|--------------------|----------------------------|
| type     |                    |                       |                 |                    |                            |
| Charter  | 83.406183          | 83.902821             | 0.937018        | 0.966459           | 0.905609                   |
| District | 76.987026          | 80.962485             | 0.665184        | 0.809052           | 0.536959                   |

```
In [24]: ###DO NOT USE!!!!!!!!!!!!!!!###
```

```
Out[24]:
```

|             | Average Math Score | Average Reading Score | % Passing Math | % Passing Reading | % Overall Passing |
|-------------|--------------------|-----------------------|----------------|-------------------|-------------------|
| School Type |                    |                       |                |                   |                   |
| Charter     | 83.473852          | 83.896421             | 93.620830      | 96.586489         | 90.432244         |
| District    | 76.956733          | 80.966836             | 66.548453      | 80.799062         | 53.672208         |

```
In []:
```