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 [1]:
```
# 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(school_data_to_load)
student_data = pd.read_csv(student_data_to_load)

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

# school_data_complete.count()
```

Out[1]:

	Student ID	student_name	gender	grade	school_name	reading_score	math_score	School ID	type	size	bu
0	0	Paul Bradley	M	9th	Huang High School	66	79	0	District	2917	191
1	1	Victor Smith	M	12th	Huang High School	94	61	0	District	2917	191
2	2	Kevin Rodriguez	M	12th	Huang High School	90	60	0	District	2917	191
3	3	Dr. Richard Scott	M	12th	Huang High School	67	58	0	District	2917	191
4	4	Bonnie Ray	F	9th	Huang High School	97	84	0	District	2917	191

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 [ ]:
```
# Create a District Summary
```

In [2]:
```
# Total number of schools
school_count = school_data_complete['School ID'].nunique()
school_count
```

Out[2]:
```
15
```

In [3]:
```
# Total number of students
student_count = school_data_complete['Student ID'].count()
student_count
```

Out[3]:
```
39170
```

In [4]:
```
# Total budget
total_budget = school_data_complete['budget'].sum()
total_budget
```

Out[4]:
```
82932329558
```

In [5]:
```
# Average math score
avg_math = school_data_complete['math_score'].mean()
avg_math
```

Out[5]:
```
78.98537145774827
```

In [6]:
```
# Average reading score
avg_reading = school_data_complete['reading_score'].mean()
avg_reading
```

Out[6]:
```
81.87784018381414
```

In [7]:
```
# Overall average score
overall_avg = (school_data_complete['math_score'].mean() \
     + school_data_complete['reading_score'].mean()) / 2
overall_avg
```

Out[7]:
```
80.43160582078121
```

In [8]:
```
# Percentage of passing math (70 or greater)
math_pass = school_data_complete[school_data_complete['math_score'] >= 70] \
```

```
    .shape[0] / school_data_complete.shape[0] * 100
math_pass
```

Out[8]:
```
74.9808526933878
```

In [9]:
```
# Percentage of passing reading (70 or greater)
read_pass = school_data_complete[school_data_complete['reading_score'] >= 70] \
    .shape[0] / school_data_complete.shape[0] * 100
read_pass
```

Out[9]:
```
85.80546336482001
```

In [10]:
```
# Overall passing rate
overall_pass = (math_pass + read_pass) / 2
overall_pass
```

Out[10]:
```
80.39315802910392
```

In [11]:
```
# District Summary Overview Table
district_summary = pd.DataFrame({'Number of Schools':[school_count],
                                 'Number of Students':[student_count],
                                 'Total Budget':[total_budget],
                                 'Avg Math Score':[avg_math],
                                 'Avg Reading Score':[avg_reading],
                                 'Overall Avg Score':[overall_avg],
                                 'Math Pass Rate':[math_pass],
                                 'Reading Pass Rate':[read_pass],
                                 'Overall Pass Rate': [overall_pass]})
district_summary
```

Out[11]:

	Number of Schools	Number of Students	Total Budget	Avg Math Score	Avg Reading Score	Overall Avg Score	Math Pass Rate	Reading Pass Rate	Overall Pass Rate
0	15	39170	82932329558	78.985371	81.87784	80.431606	74.980853	85.805463	80.393158

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 [62]:
```
#  Sort and display the top five schools in overall passing rate
top_school = school_summary.sort_values('overall_pass_rate', ascending=False)\
    .reset_index()
top_school.iloc[:, [1, 2, -1]].head(5)
```

Out[62]:

	school_name	type	overall_pass_rate
0	Cabrera High School	Charter	95.586652
1	Thomas High School	Charter	95.290520
2	Pena High School	Charter	95.270270
3	Griffin High School	Charter	95.265668
4	Wilson High School	Charter	95.203679

In [26]:
```
# Calculate total school budget
table0 = school_data_complete.groupby(['school_name', 'budget', 'type']).count().reset_i
    .sort_values('school_name', ascending=True).iloc[:,0:4]
table0
```

Out[26]:

	school_name	budget	type	Student ID
0	Bailey High School	3124928	District	4976
1	Cabrera High School	1081356	Charter	1858
2	Figueroa High School	1884411	District	2949
3	Ford High School	1763916	District	2739
4	Griffin High School	917500	Charter	1468
5	Hernandez High School	3022020	District	4635
6	Holden High School	248087	Charter	427
7	Huang High School	1910635	District	2917
8	Johnson High School	3094650	District	4761
9	Pena High School	585858	Charter	962
10	Rodriguez High School	2547363	District	3999
11	Shelton High School	1056600	Charter	1761
12	Thomas High School	1043130	Charter	1635
13	Wilson High School	1319574	Charter	2283
14	Wright High School	1049400	Charter	1800

In [29]:
```
# Calculate per student budget
table1 = table0
table1['per_student_budget'] = table1['budget'] / table1['Student ID']
table1
```

Out[29]:

	school_name	budget	type	Student ID	per_student_budget
0	Bailey High School	3124928	District	4976	628.0
1	Cabrera High School	1081356	Charter	1858	582.0
2	Figueroa High School	1884411	District	2949	639.0
3	Ford High School	1763916	District	2739	644.0

4	Griffin High School	917500	Charter	1468	625.0
5	Hernandez High School	3022020	District	4635	652.0
6	Holden High School	248087	Charter	427	581.0
7	Huang High School	1910635	District	2917	655.0
8	Johnson High School	3094650	District	4761	650.0
9	Pena High School	585858	Charter	962	609.0
10	Rodriguez High School	2547363	District	3999	637.0
11	Shelton High School	1056600	Charter	1761	600.0
12	Thomas High School	1043130	Charter	1635	638.0
13	Wilson High School	1319574	Charter	2283	578.0
14	Wright High School	1049400	Charter	1800	583.0

In [14]:
```
# Calculate the avg math and reading score
table2 = school_data_complete.groupby(['school_name']).sum(numeric_only=True)\
    .reset_index().sort_values('school_name', ascending=True).iloc[:,[0,2,3]]
table2['avg_reading_score'] = table2['reading_score'] / table1['Student ID']
table2['avg_math_score'] = table2['math_score'] / table1['Student ID']
table2 = table2.iloc[:,[0,3,4]]
table2
```

Out[14]:

	school_name	avg_reading_score	avg_math_score
0	Bailey High School	81.033963	77.048432
1	Cabrera High School	83.975780	83.061895
2	Figueroa High School	81.158020	76.711767
3	Ford High School	80.746258	77.102592
4	Griffin High School	83.816757	83.351499
5	Hernandez High School	80.934412	77.289752
6	Holden High School	83.814988	83.803279
7	Huang High School	81.182722	76.629414
8	Johnson High School	80.966394	77.072464
9	Pena High School	84.044699	83.839917
10	Rodriguez High School	80.744686	76.842711
11	Shelton High School	83.725724	83.359455
12	Thomas High School	83.848930	83.418349
13	Wilson High School	83.989488	83.274201
14	Wright High School	83.955000	83.682222

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

In [15]:
```
# Find the total counts of math result
table3 = school_data_complete.groupby(['school_name', 'math_score']).count()\
    .reset_index().sort_values('school_name', ascending=True).iloc[:,:]

# Find the counts for math result in each school that pass 70 or higher
table3['math_pass'] = 0
```

```
table3.loc[table3['math_score'] >= 70, 'math_pass'] = 1
table3['math_pass_count'] = table3['Student ID'] * table3['math_pass']

table3 = table3.groupby(['school_name']).sum()\
    .reset_index().sort_values('school_name', ascending=True).iloc[:,[0,2,-1]]

# Calculate the math passing rate
table3['math_pass_rate'] = table3['math_pass_count'] / table3['Student ID']\
    * 100
table3
```

Out[15]:

	school_name	Student ID	math_pass_count	math_pass_rate
0	Bailey High School	4976	3318	66.680064
1	Cabrera High School	1858	1749	94.133477
2	Figueroa High School	2949	1946	65.988471
3	Ford High School	2739	1871	68.309602
4	Griffin High School	1468	1371	93.392371
5	Hernandez High School	4635	3094	66.752967
6	Holden High School	427	395	92.505855
7	Huang High School	2917	1916	65.683922
8	Johnson High School	4761	3145	66.057551
9	Pena High School	962	910	94.594595
10	Rodriguez High School	3999	2654	66.366592
11	Shelton High School	1761	1653	93.867121
12	Thomas High School	1635	1525	93.272171
13	Wilson High School	2283	2143	93.867718
14	Wright High School	1800	1680	93.333333

In [16]:

```
# Find the total counts of read result
table4 = school_data_complete.groupby(['school_name', 'reading_score']).count()\
    .reset_index().sort_values('school_name', ascending=True).iloc[:,:]

# Find the counts for read result in each school that pass 70 or higher
table4['reading_pass'] = 0
table4.loc[table4['reading_score'] >= 70, 'reading_pass'] = 1
table4['reading_pass_count'] = table4['Student ID'] * table4['reading_pass']

table4 = table4.groupby(['school_name']).sum()\
    .reset_index().sort_values('school_name', ascending=True).iloc[:,[0,2,-1]]

# Calculate the read passing rate
table4['reading_pass_rate'] = table4['reading_pass_count'] / table4['Student ID']\
    * 100
table4
```

Out[16]:

	school_name	Student ID	reading_pass_count	reading_pass_rate
0	Bailey High School	4976	4077	81.933280
1	Cabrera High School	1858	1803	97.039828
2	Figueroa High School	2949	2381	80.739234
3	Ford High School	2739	2172	79.299014

4	Griffin High School	1468	1426	97.138965
5	Hernandez High School	4635	3748	80.862999
6	Holden High School	427	411	96.252927
7	Huang High School	2917	2372	81.316421
8	Johnson High School	4761	3867	81.222432
9	Pena High School	962	923	95.945946
10	Rodriguez High School	3999	3208	80.220055
11	Shelton High School	1761	1688	95.854628
12	Thomas High School	1635	1591	97.308869
13	Wilson High School	2283	2204	96.539641
14	Wright High School	1800	1739	96.611111

In [17]:
```
# Calculate the overall passing rate (average of the math and reading passing rate)
table5 = table4
table5['math_pass_rate'] = table3['math_pass_rate']
table5['overall_pass_rate'] = (table3['math_pass_rate'] + \
    table5['reading_pass_rate']) / 2
table5.iloc[:,[0, 3, 4, 5]]
```

Out[17]:

	school_name	reading_pass_rate	math_pass_rate	overall_pass_rate
0	Bailey High School	81.933280	66.680064	74.306672
1	Cabrera High School	97.039828	94.133477	95.586652
2	Figueroa High School	80.739234	65.988471	73.363852
3	Ford High School	79.299014	68.309602	73.804308
4	Griffin High School	97.138965	93.392371	95.265668
5	Hernandez High School	80.862999	66.752967	73.807983
6	Holden High School	96.252927	92.505855	94.379391
7	Huang High School	81.316421	65.683922	73.500171
8	Johnson High School	81.222432	66.057551	73.639992
9	Pena High School	95.945946	94.594595	95.270270
10	Rodriguez High School	80.220055	66.366592	73.293323
11	Shelton High School	95.854628	93.867121	94.860875
12	Thomas High School	97.308869	93.272171	95.290520
13	Wilson High School	96.539641	93.867718	95.203679
14	Wright High School	96.611111	93.333333	94.972222

In [36]:
```
# Merge above tables
table6 = table1
table6 = pd.merge(table1, table2, how="left", on=["school_name", "school_name"])
table6
```

Out[36]:

	school_name	budget	type	Student ID	per_student_budget	avg_reading_score	avg_math_score
0	Bailey High School	3124928	District	4976	628.0	81.033963	77.048432
1	Cabrera High	1081356	Charter	1858	582.0	83.975780	83.061895

	School						
2	Figueroa High School	1884411	District	2949	639.0	81.158020	76.711767
3	Ford High School	1763916	District	2739	644.0	80.746258	77.102592
4	Griffin High School	917500	Charter	1468	625.0	83.816757	83.351499
5	Hernandez High School	3022020	District	4635	652.0	80.934412	77.289752
6	Holden High School	248087	Charter	427	581.0	83.814988	83.803279
7	Huang High School	1910635	District	2917	655.0	81.182722	76.629414
8	Johnson High School	3094650	District	4761	650.0	80.966394	77.072464
9	Pena High School	585858	Charter	962	609.0	84.044699	83.839917
10	Rodriguez High School	2547363	District	3999	637.0	80.744686	76.842711
11	Shelton High School	1056600	Charter	1761	600.0	83.725724	83.359455
12	Thomas High School	1043130	Charter	1635	638.0	83.848930	83.418349
13	Wilson High School	1319574	Charter	2283	578.0	83.989488	83.274201
14	Wright High School	1049400	Charter	1800	583.0	83.955000	83.682222

In [51]:
```
# School Summary Overview Table
school_summary = pd.merge(table6, table5, how="left", on=["school_name", "school_name"])
school_summary = school_summary.iloc[:, [0, 2, 3, 1, 4, 5, 6, 9, 10, 11]]
school_summary.rename(columns = {'Student ID_x':'students'}, inplace = True)
school_summary
```

Out[51]:

	school_name	type	students	budget	per_student_budget	avg_reading_score	avg_math_score	reading
0	Bailey High School	District	4976	3124928	628.0	81.033963	77.048432	
1	Cabrera High School	Charter	1858	1081356	582.0	83.975780	83.061895	
2	Figueroa High School	District	2949	1884411	639.0	81.158020	76.711767	
3	Ford High School	District	2739	1763916	644.0	80.746258	77.102592	
4	Griffin High School	Charter	1468	917500	625.0	83.816757	83.351499	
5	Hernandez High School	District	4635	3022020	652.0	80.934412	77.289752	
6	Holden High School	Charter	427	248087	581.0	83.814988	83.803279	
7	Huang High School	District	2917	1910635	655.0	81.182722	76.629414	
8	Johnson High School	District	4761	3094650	650.0	80.966394	77.072464	
9	Pena High School	Charter	962	585858	609.0	84.044699	83.839917	

10	Rodriguez High School	District	3999	2547363	637.0	80.744686	76.842711	
11	Shelton High School	Charter	1761	1056600	600.0	83.725724	83.359455	
12	Thomas High School	Charter	1635	1043130	638.0	83.848930	83.418349	
13	Wilson High School	Charter	2283	1319574	578.0	83.989488	83.274201	
14	Wright High School	Charter	1800	1049400	583.0	83.955000	83.682222	

Bottom Performing Schools (By Passing Rate)

- Sort and display the five worst-performing schools

In [64]:
```
#  Sort and display the worst five schools in overall passing rate
bottom_school = school_summary.sort_values('overall_pass_rate', ascending=True)\
    .reset_index()
bottom_school.iloc[:, [1, 2, -1]].head(5)
```

Out[64]:

	school_name	type	students	budget	per_student_budget	avg_reading_score	avg_math_score	reading
0	Bailey High School	District	4976	3124928	628.0	81.033963	77.048432	
1	Cabrera High School	Charter	1858	1081356	582.0	83.975780	83.061895	
2	Figueroa High School	District	2949	1884411	639.0	81.158020	76.711767	
3	Ford High School	District	2739	1763916	644.0	80.746258	77.102592	
4	Griffin High School	Charter	1468	917500	625.0	83.816757	83.351499	
5	Hernandez High School	District	4635	3022020	652.0	80.934412	77.289752	
6	Holden High School	Charter	427	248087	581.0	83.814988	83.803279	
7	Huang High School	District	2917	1910635	655.0	81.182722	76.629414	
8	Johnson High School	District	4761	3094650	650.0	80.966394	77.072464	
9	Pena High School	Charter	962	585858	609.0	84.044699	83.839917	
10	Rodriguez High School	District	3999	2547363	637.0	80.744686	76.842711	
11	Shelton High School	Charter	1761	1056600	600.0	83.725724	83.359455	
12	Thomas High School	Charter	1635	1043130	638.0	83.848930	83.418349	
13	Wilson High School	Charter	2283	1319574	578.0	83.989488	83.274201	

14	Wright High School	Charter	1800	1049400		583.0	83.955000	83.682222

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 [ ]:
```
# Create table that lists the average math score for each school of each grade level.
```

In [ ]:
```
# Calculate the average math score for 9th grade in each school
```

In [ ]:
```
# Calculate the average math score for 10th grade in each school
```

In [ ]:
```
# Calculate the average math score for 11th grade in each school
```

In [ ]:
```
# Calculate the average math score for 12th grade in each school
```

Reading Score by Grade

- Perform the same operations as above for reading scores

In [ ]:
```
# Create table that lists the average reading score for each school of each grade level.
```

In [ ]:
```
# Calculate the average reading score for 9th grade in each school
```

In [ ]:
```
# Calculate the average reading score for 10th grade in each school
```

In [ ]:
```
# Calculate the average reading score for 11th grade in each school
```

In [ ]:
```
# Calculate the average reading score for 12th grade in each school
```

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 [ ]:
```
# 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 [ ]:
```
# Create a new column to show budget per student in each row
```

In [ ]:
```
# Create a new column to define the spending ranges per student
```

In [ ]:
```
# Calculate the average math score within each spending range
```

In [ ]:
```
# Calculate the percentage passing rate for math in each spending range
```

In [ ]:
```
# Calculate the percentage passing rate for reading in each spending range
```

In [ ]:
```
# Calculate the percentage overall passing rate in each spending range
```

Scores by School Size

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

In [ ]:
```
# 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 [ ]:
```
# Create a new column for the bin groups
```

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

In [ ]:
```
# math_pass_size
```

In [ ]:
```
# read_pass_size
```

In [ ]:
```
# Calculate the overall passing rate for different school size
```

Scores by School Type

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

In [ ]:
```
# Create bins and groups, school type {'Charter', 'District'}
```

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

In [ ]:
```
# math pass size
```

In [ ]:
```
# reading pass size
```

In [ ]:
```
# Calculate the overall passing rate
```