

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 [272]:

```
# 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_name"])
school_data_complete.head()

# school_data_complete.count()
```

Out[272]:

| | Student ID | student_name | gender | grade | school_name | reading_score | math_score | School ID | type | size | budget |
|---|------------|-------------------|--------|-------|-------------------|---------------|------------|-----------|----------|------|---------|
| 0 | 0 | Paul Bradley | M | 9th | Huang High School | 66 | 79 | 0 | District | 2917 | 1910635 |
| 1 | 1 | Victor Smith | M | 12th | Huang High School | 94 | 61 | 0 | District | 2917 | 1910635 |
| 2 | 2 | Kevin Rodriguez | M | 12th | Huang High School | 90 | 60 | 0 | District | 2917 | 1910635 |
| 3 | 3 | Dr. Richard Scott | M | 12th | Huang High School | 67 | 58 | 0 | District | 2917 | 1910635 |
| 4 | 4 | Bonnie Ray | F | 9th | Huang High School | 97 | 84 | 0 | District | 2917 | 1910635 |

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 [273]:

```
# Create a District Summary
school = len(school_data_complete['school_name'].unique())
students = len(school_data_complete['Student ID'].unique())
budget = school_data_complete['budget'].sum()
maths = school_data_complete['math_score'].mean()
reading = school_data_complete['reading_score'].mean()
all_mean = (maths + reading) / 2
student_good_in_math = school_data_complete[school_data_complete['math_score'] >= 70].shape[0]
percent_good_in_math = (student_good_in_math*100)/ students
student_good_in_r = school_data_complete[school_data_complete['reading_score'] >= 70].shape[0]
percent_good_in_r = (student_good_in_r*100)/ students

numbers = [school, students, budget, maths, reading, all_mean, student_good_in_math, percent_good_in_math, student_good_in_r, percent_good_in_r]
names = ['school', 'students', 'budget', 'maths', 'reading', 'all_mean', 'student_good_in_math', 'percent_good_in_math', 'student_good_in_r', 'percent_good_in_r']
data = {'Name': names,
        'Numbers': numbers}
df = pd.DataFrame(data)
print(df)

def print_elems(s):
    for idx, names in enumerate(df['Name']):
        if names == s:
            print(df['Numbers'][idx])
```

| | Name | Numbers |
|---|----------------------|--------------|
| 0 | school | 1.500000e+01 |
| 1 | students | 3.917000e+04 |
| 2 | budget | 8.293233e+10 |
| 3 | maths | 7.898537e+01 |
| 4 | reading | 8.187784e+01 |
| 5 | all_mean | 8.043161e+01 |
| 6 | student_good_in_math | 2.937000e+04 |
| 7 | percent_good_in_math | 7.498085e+01 |
| 8 | student_good_in_r | 3.361000e+04 |
| 9 | percent_good_in_r | 8.580546e+01 |

In [274]:

```
# Total number of schools
print_elems('school')
```

15.0

In [275]:

```
# Total number of students
print_elems('students')
```

39170.0

In [276]:

```
# Total budget
print_elems('budget')
```

82932329558.0

In [277]:

```
# Average math score
print_elems('maths')
```

```
78.98537145774827
```

In [278]:

```
# Average reading score
print_elems('reading')
```

```
81.87784018381414
```

In [279]:

```
# Overall average score
print_elems('all_mean')
```

```
80.43160582078121
```

In [280]:

```
# Percentage of passing math (70 or greater)
print_elems('percent_good_in_math')
```

```
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

In [281]:

```
def create_dict(type, school_name, name):
    TT = []
    for i in type:
        TT.append(dict(zip([name], [i])))
    return TT

def add_elem_dict(TT, av_math):
    a = list(av_math[0].keys())[0]
    for idx, i in enumerate(TT):
        i[a] = av_math[idx][a]
    return TT

def create_tab_school_type():
    school_name = school_data_complete['school_name'].unique()
    TT = create_dict(school_name, school_name, "School")
    TT = add_elem_dict(TT, TT)
    type = []
    for i in school_name:
        id = []
        id.append(school_data_complete.loc[school_data_complete["school_name"] == i]['Student ID'].iloc[0])
        t = school_data_complete.loc[school_data_complete["school_name"] == i]['type'][[id[0]]]
```

```

        type.append(t.iloc[0])
    type = create_dict(type, school_name, "Type")
    TT = add_elem_dict(TT, type)
    return TT, school_name

def average_f(TT, school_name, name, tab_name):
    average_math = school_data_complete.groupby('school_name').mean()[name]
    av_math = []
    for i in school_name:
        av_math.append(average_math[i])
    av_math = create_dict(av_math, school_name, tab_name)
    TT = add_elem_dict(TT, av_math)
    return TT

def number_of_students(TT):
    students = list(school_data_complete["school_name"].value_counts(sort=False))
    students = create_dict(students, school_name, "Students")
    TT = add_elem_dict(TT, students)
    return TT, students

def nb_budget(TT, students):
    budget = list(school_data_complete.groupby('school_name', sort=False).sum()['budget'
    ])
    for idx, i in enumerate(budget):
        budget[idx] /= students[idx]['Students']
    budget_per_student = []
    for idx, i in enumerate(budget):
        # print(i, students)
        budget_per_student.append(i/students[idx]['Students'])
    budget = create_dict(budget, school_name, "Budget")
    TT = add_elem_dict(TT, budget)
    budget_per_student = create_dict(budget_per_student, school_name, "Budget per student
    ")
    TT = add_elem_dict(TT, budget_per_student)
    return TT

def passing_m(TT, school_name, name, tab_name, students):
    good_m = []
    for idx, i in enumerate(school_name):
        good_m.append(len(school_data_complete[(school_data_complete[name] >= 70) & (sch
ool_data_complete['school_name'] == i)]) / students[idx]['Students'] * 100)
    good_m = create_dict(good_m, school_name, tab_name)
    TT = add_elem_dict(TT, good_m)
    return TT

TT, school_name = create_tab_school_type()
TT, students = number_of_students(TT)
TT = average_f(TT, school_name, 'math_score', 'Average Math')
TT = average_f(TT, school_name, 'reading_score', 'Average Reading')
TT = nb_budget(TT, students)
TT = passing_m(TT, school_name, 'math_score', "Percent passing Math", students)
TT = passing_m(TT, school_name, 'reading_score', "Percent passing Reading", students)
df_schools = pd.DataFrame(TT)
df_schools['Passing rate'] = (df_schools['Percent passing Math'] + df_schools['Percent pa
ssing Reading']) / 2
df_schools

```

Out[281]:

| | School | Type | Students | Average Math | Average Reading | Budget | Budget per student | Percent passing Math | Percent passing Reading | Passing rate |
|---|-----------------------|----------|----------|--------------|-----------------|-----------|--------------------|----------------------|-------------------------|--------------|
| 0 | Huang High School | District | 2917 | 76.629414 | 81.182722 | 1910635.0 | 655.0 | 65.683922 | 81.316421 | 73.500171 |
| 1 | Figueroa High School | District | 2949 | 76.711767 | 81.158020 | 1884411.0 | 639.0 | 65.988471 | 80.739234 | 73.363852 |
| 2 | Shelton High School | Charter | 1761 | 83.359455 | 83.725724 | 1056600.0 | 600.0 | 93.867121 | 95.854628 | 94.860875 |
| 3 | Hernandez High School | District | 4635 | 77.289752 | 80.934412 | 3022020.0 | 652.0 | 66.752967 | 80.862999 | 73.807983 |

| High School | | | | | | | | | | | |
|-------------|-----------------------|----------|----------|--------------|-----------------|-----------|--------------------|----------------------|-------------------------|--------------|--|
| 4 | Griffin High School | Type | Students | Average Math | Average Reading | Budget | Budget Per student | Percent passing Math | Percent passing Reading | Passing rate | |
| 5 | Wilson High School | Charter | 2283 | 83.274201 | 83.989488 | 1319574.0 | 578.0 | 93.867718 | 96.539641 | 95.203679 | |
| 6 | Cabrera High School | Charter | 1858 | 83.061895 | 83.975780 | 1081356.0 | 582.0 | 94.133477 | 97.039828 | 95.586652 | |
| 7 | Bailey High School | District | 4976 | 77.048432 | 81.033963 | 3124928.0 | 628.0 | 66.680064 | 81.933280 | 74.306672 | |
| 8 | Holden High School | Charter | 427 | 83.803279 | 83.814988 | 248087.0 | 581.0 | 92.505855 | 96.252927 | 94.379391 | |
| 9 | Pena High School | Charter | 962 | 83.839917 | 84.044699 | 585858.0 | 609.0 | 94.594595 | 95.945946 | 95.270270 | |
| 10 | Wright High School | Charter | 1800 | 83.682222 | 83.955000 | 1049400.0 | 583.0 | 93.333333 | 96.611111 | 94.972222 | |
| 11 | Rodriguez High School | District | 3999 | 76.842711 | 80.744686 | 2547363.0 | 637.0 | 66.366592 | 80.220055 | 73.293323 | |
| 12 | Johnson High School | District | 4761 | 77.072464 | 80.966394 | 3094650.0 | 650.0 | 66.057551 | 81.222432 | 73.639992 | |
| 13 | Ford High School | District | 2739 | 77.102592 | 80.746258 | 1763916.0 | 644.0 | 68.309602 | 79.299014 | 73.804308 | |
| 14 | Thomas High School | Charter | 1635 | 83.418349 | 83.848930 | 1043130.0 | 638.0 | 93.272171 | 97.308869 | 95.290520 | |

Top Performing Schools (By Passing Rate)

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

In [282]:

```
# Sort and display the top five schools in overall passing rate
top = df_schools.sort_values('Passing rate', ascending=False).head().iloc[0:, [0, -1]]
top
```

Out[282]:

| | School | Passing rate |
|----|---------------------|--------------|
| 6 | Cabrera High School | 95.586652 |
| 14 | Thomas High School | 95.290520 |
| 9 | Pena High School | 95.270270 |
| 4 | Griffin High School | 95.265668 |
| 5 | Wilson High School | 95.203679 |

In [283]:

```
# Calculate total school budget
df_schools.iloc[0:, [0, 5]]
```

Out[283]:

| | School | Budget |
|---|-----------------------|-----------|
| 0 | Huang High School | 1910635.0 |
| 1 | Figueroa High School | 1884411.0 |
| 2 | Shelton High School | 1056600.0 |
| 3 | Hernandez High School | 3022020.0 |
| 4 | Griffin High School | 917500.0 |

| | | |
|----|-----------------------|-----------|
| 5 | Wilson High School | 1319574.0 |
| 6 | Cabrera High School | 1081356.0 |
| 7 | Bailey High School | 3124928.0 |
| 8 | Holden High School | 248087.0 |
| 9 | Pena High School | 585858.0 |
| 10 | Wright High School | 1049400.0 |
| 11 | Rodriguez High School | 2547363.0 |
| 12 | Johnson High School | 3094650.0 |
| 13 | Ford High School | 1763916.0 |
| 14 | Thomas High School | 1043130.0 |

In [284]:

```
# Calculate per student budget
df_schools.iloc[0:, [0, 6]]
```

Out[284]:

| | School | Budget per student |
|----|-----------------------|--------------------|
| 0 | Huang High School | 655.0 |
| 1 | Figueroa High School | 639.0 |
| 2 | Shelton High School | 600.0 |
| 3 | Hernandez High School | 652.0 |
| 4 | Griffin High School | 625.0 |
| 5 | Wilson High School | 578.0 |
| 6 | Cabrera High School | 582.0 |
| 7 | Bailey High School | 628.0 |
| 8 | Holden High School | 581.0 |
| 9 | Pena High School | 609.0 |
| 10 | Wright High School | 583.0 |
| 11 | Rodriguez High School | 637.0 |
| 12 | Johnson High School | 650.0 |
| 13 | Ford High School | 644.0 |
| 14 | Thomas High School | 638.0 |

In [285]:

```
# Cacluate the avg math and reading score
average = df_schools.iloc[0:, [0, 3, 4]]
average["average"] = (df_schools['Average Math'] + df_schools['Average Reading']) / 2
average
```

Out[285]:

| | School | Average Math | Average Reading | average |
|---|-----------------------|--------------|-----------------|-----------|
| 0 | Huang High School | 76.629414 | 81.182722 | 78.906068 |
| 1 | Figueroa High School | 76.711767 | 81.158020 | 78.934893 |
| 2 | Shelton High School | 83.359455 | 83.725724 | 83.542589 |
| 3 | Hernandez High School | 77.289752 | 80.934412 | 79.112082 |
| 4 | Griffin High School | 83.351499 | 83.816757 | 83.584128 |
| 5 | Wilson High School | 83.274201 | 83.989488 | 83.631844 |
| 6 | Cabrera High School | 83.061805 | 83.075700 | 83.518752 |

| 6 | Cabrera High School | 83.001893 | 83.979780 | 83.518837 |
|----|-----------------------|-----------|-----------|-----------|
| 7 | Bailey High School | 77.048432 | 81.033963 | 79.041198 |
| 8 | Holden High School | 83.803279 | 83.814988 | 83.809133 |
| 9 | Pena High School | 83.839917 | 84.044699 | 83.942308 |
| 10 | Wright High School | 83.682222 | 83.955000 | 83.818611 |
| 11 | Rodriguez High School | 76.842711 | 80.744686 | 78.793698 |
| 12 | Johnson High School | 77.072464 | 80.966394 | 79.019429 |
| 13 | Ford High School | 77.102592 | 80.746258 | 78.924425 |
| 14 | Thomas High School | 83.418349 | 83.848930 | 83.633639 |

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

In [286]:

```
# Find the total counts of math result
count_of_math_r = df_schools.iloc[0:, [0, 2]]
print('count_of_math_r')
print(count_of_math_r)
# Find the counts for math result in each school that pass 70 or higher
good_m = []
for idx, i in enumerate(school_name):
    good_m.append(len(school_data_complete[(school_data_complete['math_score'] >= 70) &
(school_data_complete['school_name'] == i)]))
print('good_m')
print(good_m)

# Calculate the math passing rate
df_schools.iloc[0:, [0, -3]]
```

```
count_of_math_r
School  Students
0      Huang High School      2917
1    Figueroa High School      2949
2    Shelton High School      1761
3  Hernandez High School      4635
4    Griffin High School      1468
5    Wilson High School      2283
6    Cabrera High School      1858
7    Bailey High School      4976
8    Holden High School       427
9      Pena High School       962
10   Wright High School      1800
11 Rodriguez High School      3999
12   Johnson High School      4761
13     Ford High School      2739
14   Thomas High School      1635
good_m
[1916, 1946, 1653, 3094, 1371, 2143, 1749, 3318, 395, 910, 1680, 2654, 3145, 1871, 1525]
```

Out[286]:

| | School | Percent passing Math |
|---|-----------------------|----------------------|
| 0 | Huang High School | 65.683922 |
| 1 | Figueroa High School | 65.988471 |
| 2 | Shelton High School | 93.867121 |
| 3 | Hernandez High School | 66.752967 |
| 4 | Griffin High School | 93.392371 |
| 5 | Wilson High School | 93.867718 |
| 6 | Cabrera High School | 94.133477 |
| 7 | Bailey High School | 66.680064 |
| 8 | Holden High School | 62.505955 |

| | School | Percent passing Math |
|----|-----------------------|----------------------|
| 9 | Pena High School | 94.594595 |
| 10 | Wright High School | 93.333333 |
| 11 | Rodriguez High School | 66.366592 |
| 12 | Johnson High School | 66.057551 |
| 13 | Ford High School | 68.309602 |
| 14 | Thomas High School | 93.272171 |

In [287]:

```
# Find the total counts of read result
count_of_reading_r = df_schools.iloc[0:, [0, 2]]
print('count_of_reading_r')
print(count_of_reading_r)

# Find the counts for read result in each school that pass 70 or higher
good_r = []
for idx, i in enumerate(school_name):
    good_r.append(len(school_data_complete[(school_data_complete['reading_score'] >= 70)
& (school_data_complete['school_name'] == i)]))
print('good_r')
print(good_r)

# Calculate the read passing rate
df_schools.iloc[0:, [0, -2]]
```

```
count_of_reading_r
School  Students
0      Huang High School      2917
1    Figueroa High School      2949
2      Shelton High School      1761
3    Hernandez High School      4635
4      Griffin High School      1468
5      Wilson High School      2283
6    Cabrera High School      1858
7      Bailey High School      4976
8      Holden High School       427
9        Pena High School       962
10     Wright High School      1800
11  Rodriguez High School      3999
12    Johnson High School      4761
13      Ford High School      2739
14    Thomas High School      1635
good_r
[2372, 2381, 1688, 3748, 1426, 2204, 1803, 4077, 411, 923, 1739, 3208, 3867, 2172, 1591]
```

Out[287]:

| | School | Percent passing Reading |
|----|-----------------------|-------------------------|
| 0 | Huang High School | 81.316421 |
| 1 | Figueroa High School | 80.739234 |
| 2 | Shelton High School | 95.854628 |
| 3 | Hernandez High School | 80.862999 |
| 4 | Griffin High School | 97.138965 |
| 5 | Wilson High School | 96.539641 |
| 6 | Cabrera High School | 97.039828 |
| 7 | Bailey High School | 81.933280 |
| 8 | Holden High School | 96.252927 |
| 9 | Pena High School | 95.945946 |
| 10 | Wright High School | 96.611111 |
| 11 | Rodriguez High School | 80.220055 |

| 12 | Johnson High School | Percent passing Reading |
|----|---------------------|-------------------------|
| 13 | Ford High School | 79.299014 |
| 14 | Thomas High School | 97.308869 |

In [288]:

```
# Calculate the overall passing rate (average of the math and reading passing rate)
l = list(df_schools.iloc[0:, [0, -1]]['Passing rate'])
m = 0
for i in l:
    m += i
m /= len(df_schools.iloc[0:, [0, -1]])
print("OVERHALL for all schools", m)
df_schools.iloc[0:, [0, -1]]
```

OVERHALL for all schools 85.10303857542577

Out[288]:

| | School | Passing rate |
|----|-----------------------|--------------|
| 0 | Huang High School | 73.500171 |
| 1 | Figueroa High School | 73.363852 |
| 2 | Shelton High School | 94.860875 |
| 3 | Hernandez High School | 73.807983 |
| 4 | Griffin High School | 95.265668 |
| 5 | Wilson High School | 95.203679 |
| 6 | Cabrera High School | 95.586652 |
| 7 | Bailey High School | 74.306672 |
| 8 | Holden High School | 94.379391 |
| 9 | Pena High School | 95.270270 |
| 10 | Wright High School | 94.972222 |
| 11 | Rodriguez High School | 73.293323 |
| 12 | Johnson High School | 73.639992 |
| 13 | Ford High School | 73.804308 |
| 14 | Thomas High School | 95.290520 |

In []:

In []:

In []:

Bottom Performing Schools (By Passing Rate)

- Sort and display the five worst-performing schools

In [289]:

```
# Sort and display the worst five schools in overall passing rate
worst = df_schools.sort_values('Passing rate').head().iloc[0:, [0, -1]]
worst.sort_values('Passing rate', ascending=[False])
```

Out [289]:

| | School | Passing rate |
|----|-----------------------|--------------|
| 13 | Ford High School | 73.804308 |
| 12 | Johnson High School | 73.639992 |
| 0 | Huang High School | 73.500171 |
| 1 | Figueroa High School | 73.363852 |
| 11 | Rodriguez High School | 73.293323 |

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 [290]:

```
# Create table that lists the average math score for each school of each grade level.
school_data_complete.groupby(['school_name', 'grade']).mean(['reading_score', 'math_score']).iloc[0:, [2]]
```

Out [290]:

| | | math_score |
|-----------------------|-------|------------|
| school_name | grade | |
| Bailey High School | 10th | 76.996772 |
| | 11th | 77.515588 |
| | 12th | 76.492218 |
| | 9th | 77.083676 |
| Cabrera High School | 10th | 83.154506 |
| | 11th | 82.765560 |
| | 12th | 83.277487 |
| | 9th | 83.094697 |
| Figueroa High School | 10th | 76.539974 |
| | 11th | 76.884344 |
| | 12th | 77.151369 |
| | 9th | 76.403037 |
| Ford High School | 10th | 77.672316 |
| | 11th | 76.918058 |
| | 12th | 76.179963 |
| | 9th | 77.361345 |
| Griffin High School | 10th | 84.229064 |
| | 11th | 83.842105 |
| | 12th | 83.356164 |
| | 9th | 82.044010 |
| Hernandez High School | 10th | 77.337408 |

| school_name | 11th | 77.136029 |
|-----------------------|------|-----------|
| | 12th | 77.186567 |
| | | |
| Holden High School | 9th | 77.438495 |
| | 10th | 83.429825 |
| | 11th | 85.000000 |
| | 12th | 82.855422 |
| Huang High School | 9th | 83.787402 |
| | 10th | 75.908735 |
| | 11th | 76.446602 |
| | 12th | 77.225641 |
| Johnson High School | 9th | 77.027251 |
| | 10th | 76.691117 |
| | 11th | 77.491653 |
| | 12th | 76.863248 |
| Pena High School | 9th | 77.187857 |
| | 10th | 83.372000 |
| | 11th | 84.328125 |
| | 12th | 84.121547 |
| Rodriguez High School | 9th | 83.625455 |
| | 10th | 76.612500 |
| | 11th | 76.395626 |
| | 12th | 77.690748 |
| Shelton High School | 9th | 76.859966 |
| | 10th | 82.917411 |
| | 11th | 83.383495 |
| | 12th | 83.778976 |
| Thomas High School | 9th | 83.420755 |
| | 10th | 83.087886 |
| | 11th | 83.498795 |
| | 12th | 83.497041 |
| Wilson High School | 9th | 83.590022 |
| | 10th | 83.724422 |
| | 11th | 83.195326 |
| | 12th | 83.035794 |
| Wright High School | 9th | 83.085578 |
| | 10th | 84.010288 |
| | 11th | 83.836782 |
| | 12th | 83.644986 |
| | 9th | 83.264706 |

In [291]:

```
# Calculate the average math score for 9th grade in each school
school_data_complete[school_data_complete['grade'] == '9th'].groupby(['school_name', 'grade']).mean(['reading_score', 'math_score']).iloc[0:, [2]]
```

Out[291]:

math_score

| | | math_score |
|-----------------------|-------|------------|
| school_name | grade | |
| school_name | grade | |
| Bailey High School | 9th | 77.083676 |
| Cabrera High School | 9th | 83.094697 |
| Figueroa High School | 9th | 76.403037 |
| Ford High School | 9th | 77.361345 |
| Griffin High School | 9th | 82.044010 |
| Hernandez High School | 9th | 77.438495 |
| Holden High School | 9th | 83.787402 |
| Huang High School | 9th | 77.027251 |
| Johnson High School | 9th | 77.187857 |
| Pena High School | 9th | 83.625455 |
| Rodriguez High School | 9th | 76.859966 |
| Shelton High School | 9th | 83.420755 |
| Thomas High School | 9th | 83.590022 |
| Wilson High School | 9th | 83.085578 |
| Wright High School | 9th | 83.264706 |

In [292]:

```
# Calculate the average math score for 10th grade in each school
school_data_complete[school_data_complete['grade'] == '10th'].groupby(['school_name', 'grade']).mean(['reading_score', 'math_score']).iloc[0:, [2]]
```

Out[292]:

| | | math_score |
|-----------------------|-------|------------|
| school_name | grade | |
| Bailey High School | 10th | 76.996772 |
| Cabrera High School | 10th | 83.154506 |
| Figueroa High School | 10th | 76.539974 |
| Ford High School | 10th | 77.672316 |
| Griffin High School | 10th | 84.229064 |
| Hernandez High School | 10th | 77.337408 |
| Holden High School | 10th | 83.429825 |
| Huang High School | 10th | 75.908735 |
| Johnson High School | 10th | 76.691117 |
| Pena High School | 10th | 83.372000 |
| Rodriguez High School | 10th | 76.612500 |
| Shelton High School | 10th | 82.917411 |
| Thomas High School | 10th | 83.087886 |
| Wilson High School | 10th | 83.724422 |
| Wright High School | 10th | 84.010288 |

In [293]:

```
# Calculate the average math score for 11th grade in each school
school_data_complete[school_data_complete['grade'] == '11th'].groupby(['school_name', 'grade']).mean(['reading_score', 'math_score']).iloc[0:, [2]]
```

Out[293]:

| | | math_score |
|-----------------------|-------|------------|
| school_name | grade | |
| Bailey High School | 11th | 77.515588 |
| Cabrera High School | 11th | 82.765560 |
| Figueroa High School | 11th | 76.884344 |
| Ford High School | 11th | 76.918058 |
| Griffin High School | 11th | 83.842105 |
| Hernandez High School | 11th | 77.136029 |
| Holden High School | 11th | 85.000000 |
| Huang High School | 11th | 76.446602 |
| Johnson High School | 11th | 77.491653 |
| Pena High School | 11th | 84.328125 |
| Rodriguez High School | 11th | 76.395626 |
| Shelton High School | 11th | 83.383495 |
| Thomas High School | 11th | 83.498795 |
| Wilson High School | 11th | 83.195326 |
| Wright High School | 11th | 83.836782 |

In [294]:

```
# Calculate the average math score for 12th grade in each school
school_data_complete[school_data_complete['grade'] == '12th'].groupby(['school_name', 'grade']).mean(['reading_score', 'math_score']).iloc[0:, [2]]
```

Out[294]:

| | | math_score |
|-----------------------|-------|------------|
| school_name | grade | |
| Bailey High School | 12th | 76.492218 |
| Cabrera High School | 12th | 83.277487 |
| Figueroa High School | 12th | 77.151369 |
| Ford High School | 12th | 76.179963 |
| Griffin High School | 12th | 83.356164 |
| Hernandez High School | 12th | 77.186567 |
| Holden High School | 12th | 82.855422 |
| Huang High School | 12th | 77.225641 |
| Johnson High School | 12th | 76.863248 |
| Pena High School | 12th | 84.121547 |
| Rodriguez High School | 12th | 77.690748 |
| Shelton High School | 12th | 83.778976 |
| Thomas High School | 12th | 83.497041 |
| Wilson High School | 12th | 83.035794 |
| Wright High School | 12th | 83.644986 |

Reading Score by Grade

- Perform the same operations as above for reading scores

In [295]:

```
# Create table that lists the average reading score for each school of each grade level.
school_data_complete.groupby(['school_name', 'grade']).mean(['reading_score', 'math_score']).iloc[0:, [1]]
```

Out[295]:

| | | reading_score |
|-----------------------|-------|---------------|
| school_name | grade | |
| Bailey High School | 10th | 80.907183 |
| | 11th | 80.945643 |
| | 12th | 80.912451 |
| | 9th | 81.303155 |
| Cabrera High School | 10th | 84.253219 |
| | 11th | 83.788382 |
| | 12th | 84.287958 |
| | 9th | 83.676136 |
| Figueroa High School | 10th | 81.408912 |
| | 11th | 80.640339 |
| | 12th | 81.384863 |
| | 9th | 81.198598 |
| Ford High School | 10th | 81.262712 |
| | 11th | 80.403642 |
| | 12th | 80.662338 |
| | 9th | 80.632653 |
| Griffin High School | 10th | 83.706897 |
| | 11th | 84.288089 |
| | 12th | 84.013699 |
| | 9th | 83.369193 |
| Hernandez High School | 10th | 80.660147 |
| | 11th | 81.396140 |
| | 12th | 80.857143 |
| | 9th | 80.866860 |
| Holden High School | 10th | 83.324561 |
| | 11th | 83.815534 |
| | 12th | 84.698795 |
| | 9th | 83.677165 |
| Huang High School | 10th | 81.512386 |
| | 11th | 81.417476 |
| | 12th | 80.305983 |
| | 9th | 81.290284 |
| Johnson High School | 10th | 80.773431 |
| | 11th | 80.616027 |
| | 12th | 81.227564 |
| | 9th | 81.260714 |
| Pena High School | 10th | 83.612000 |
| | 11th | 84.335938 |

| | 12th | 84.591160 |
|-----------------------|-------|---------------|
| | 9th | 83.807273 |
| school_name | grade | reading_score |
| Rodriguez High School | 10th | 80.629808 |
| | 11th | 80.864811 |
| | 12th | 80.376426 |
| | 9th | 80.993127 |
| Shelton High School | 10th | 83.441964 |
| | 11th | 84.373786 |
| | 12th | 82.781671 |
| | 9th | 84.122642 |
| Thomas High School | 10th | 84.254157 |
| | 11th | 83.585542 |
| | 12th | 83.831361 |
| | 9th | 83.728850 |
| Wilson High School | 10th | 84.021452 |
| | 11th | 83.764608 |
| | 12th | 84.317673 |
| | 9th | 83.939778 |
| Wright High School | 10th | 83.812757 |
| | 11th | 84.156322 |
| | 12th | 84.073171 |
| | 9th | 83.833333 |

In [296]:

```
# Calculate the average reading score for 9th grade in each school
school_data_complete[school_data_complete['grade'] == '9th'].groupby(['school_name', 'grade']).mean(['reading_score', 'math_score']).iloc[0:, [1]]
```

Out[296]:

| | | reading_score |
|-----------------------|-------|---------------|
| school_name | grade | |
| Bailey High School | 9th | 81.303155 |
| Cabrera High School | 9th | 83.676136 |
| Figueroa High School | 9th | 81.198598 |
| Ford High School | 9th | 80.632653 |
| Griffin High School | 9th | 83.369193 |
| Hernandez High School | 9th | 80.866860 |
| Holden High School | 9th | 83.677165 |
| Huang High School | 9th | 81.290284 |
| Johnson High School | 9th | 81.260714 |
| Pena High School | 9th | 83.807273 |
| Rodriguez High School | 9th | 80.993127 |
| Shelton High School | 9th | 84.122642 |
| Thomas High School | 9th | 83.728850 |
| Wilson High School | 9th | 83.939778 |
| Wright High School | 9th | 83.833333 |

In [297]:

```
# Calculate the average reading score for 10th grade in each school
school_data_complete[school_data_complete['grade'] == '10th'].groupby(['school_name', 'grade']).mean(['reading_score', 'math_score']).iloc[0:, [1]]
```

Out[297]:

| | | reading_score |
|-----------------------|-------|---------------|
| school_name | grade | |
| Bailey High School | 10th | 80.907183 |
| Cabrera High School | 10th | 84.253219 |
| Figueroa High School | 10th | 81.408912 |
| Ford High School | 10th | 81.262712 |
| Griffin High School | 10th | 83.706897 |
| Hernandez High School | 10th | 80.660147 |
| Holden High School | 10th | 83.324561 |
| Huang High School | 10th | 81.512386 |
| Johnson High School | 10th | 80.773431 |
| Pena High School | 10th | 83.612000 |
| Rodriguez High School | 10th | 80.629808 |
| Shelton High School | 10th | 83.441964 |
| Thomas High School | 10th | 84.254157 |
| Wilson High School | 10th | 84.021452 |
| Wright High School | 10th | 83.812757 |

In [298]:

```
# Calculate the average reading score for 11th grade in each school
school_data_complete[school_data_complete['grade'] == '11th'].groupby(['school_name', 'grade']).mean(['reading_score', 'math_score']).iloc[0:, [1]]
```

Out[298]:

| | | reading_score |
|-----------------------|-------|---------------|
| school_name | grade | |
| Bailey High School | 11th | 80.945643 |
| Cabrera High School | 11th | 83.788382 |
| Figueroa High School | 11th | 80.640339 |
| Ford High School | 11th | 80.403642 |
| Griffin High School | 11th | 84.288089 |
| Hernandez High School | 11th | 81.396140 |
| Holden High School | 11th | 83.815534 |
| Huang High School | 11th | 81.417476 |
| Johnson High School | 11th | 80.616027 |
| Pena High School | 11th | 84.335938 |
| Rodriguez High School | 11th | 80.864811 |
| Shelton High School | 11th | 84.373786 |
| Thomas High School | 11th | 83.585542 |
| Wilson High School | 11th | 83.764608 |
| Wright High School | 11th | 84.156322 |

In [299]:

```
# Calculate the average reading score for 12th grade in each school
school_data_complete[school_data_complete['grade'] == '12th'].groupby(['school_name', 'grade']).mean(['reading_score', 'math_score']).iloc[0:, [1]]
```

Out[299]:

| | | reading_score |
|-----------------------|-------|---------------|
| school_name | grade | |
| Bailey High School | 12th | 80.912451 |
| Cabrera High School | 12th | 84.287958 |
| Figueroa High School | 12th | 81.384863 |
| Ford High School | 12th | 80.662338 |
| Griffin High School | 12th | 84.013699 |
| Hernandez High School | 12th | 80.857143 |
| Holden High School | 12th | 84.698795 |
| Huang High School | 12th | 80.305983 |
| Johnson High School | 12th | 81.227564 |
| Pena High School | 12th | 84.591160 |
| Rodriguez High School | 12th | 80.376426 |
| Shelton High School | 12th | 82.781671 |
| Thomas High School | 12th | 83.831361 |
| Wilson High School | 12th | 84.317673 |
| Wright High School | 12th | 84.073171 |

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 [300]:

```
# 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 [301]:

```
# Create a new column to show budget per student in each row
df_schools.iloc[0:, [0, -4]]
```

Out[301]:

| | School | Budget per student |
|---|----------------------|--------------------|
| 0 | Huang High School | 655.0 |
| 1 | Figueroa High School | 639.0 |
| 2 | Shelton High School | 600.0 |

| | | |
|----|-----------------------|-------|
| 3 | Hernandez High School | 652.0 |
| 4 | Griffin High School | 625.0 |
| 5 | Wilson High School | 578.0 |
| 6 | Cabrera High School | 582.0 |
| 7 | Bailey High School | 628.0 |
| 8 | Holden High School | 581.0 |
| 9 | Pena High School | 609.0 |
| 10 | Wright High School | 583.0 |
| 11 | Rodriguez High School | 637.0 |
| 12 | Johnson High School | 650.0 |
| 13 | Ford High School | 644.0 |
| 14 | Thomas High School | 638.0 |

In [302]:

```
# Create a new column to define the spending ranges per student
df_schools['Spending range'] = pd.cut(x=df_schools['Budget per student'], bins=spending_
bins, labels=group_names)
df_schools
```

Out[302]:

| | School | Type | Students | Average Math | Average Reading | Budget | Budget per student | Percent passing Math | Percent passing Reading | Passing rate | Spending range |
|----|-----------------------|----------|----------|--------------|-----------------|-----------|--------------------|----------------------|-------------------------|--------------|----------------|
| 0 | Huang High School | District | 2917 | 76.629414 | 81.182722 | 1910635.0 | 655.0 | 65.683922 | 81.316421 | 73.500171 | \$645-675 |
| 1 | Figueroa High School | District | 2949 | 76.711767 | 81.158020 | 1884411.0 | 639.0 | 65.988471 | 80.739234 | 73.363852 | \$615-645 |
| 2 | Shelton High School | Charter | 1761 | 83.359455 | 83.725724 | 1056600.0 | 600.0 | 93.867121 | 95.854628 | 94.860875 | \$585-615 |
| 3 | Hernandez High School | District | 4635 | 77.289752 | 80.934412 | 3022020.0 | 652.0 | 66.752967 | 80.862999 | 73.807983 | \$645-675 |
| 4 | Griffin High School | Charter | 1468 | 83.351499 | 83.816757 | 917500.0 | 625.0 | 93.392371 | 97.138965 | 95.265668 | \$615-645 |
| 5 | Wilson High School | Charter | 2283 | 83.274201 | 83.989488 | 1319574.0 | 578.0 | 93.867718 | 96.539641 | 95.203679 | <\$585 |
| 6 | Cabrera High School | Charter | 1858 | 83.061895 | 83.975780 | 1081356.0 | 582.0 | 94.133477 | 97.039828 | 95.586652 | <\$585 |
| 7 | Bailey High School | District | 4976 | 77.048432 | 81.033963 | 3124928.0 | 628.0 | 66.680064 | 81.933280 | 74.306672 | \$615-645 |
| 8 | Holden High School | Charter | 427 | 83.803279 | 83.814988 | 248087.0 | 581.0 | 92.505855 | 96.252927 | 94.379391 | <\$585 |
| 9 | Pena High School | Charter | 962 | 83.839917 | 84.044699 | 585858.0 | 609.0 | 94.594595 | 95.945946 | 95.270270 | \$585-615 |
| 10 | Wright High School | Charter | 1800 | 83.682222 | 83.955000 | 1049400.0 | 583.0 | 93.333333 | 96.611111 | 94.972222 | <\$585 |
| 11 | Rodriguez High School | District | 3999 | 76.842711 | 80.744686 | 2547363.0 | 637.0 | 66.366592 | 80.220055 | 73.293323 | \$615-645 |
| 12 | Johnson High School | District | 4761 | 77.072464 | 80.966394 | 3094650.0 | 650.0 | 66.057551 | 81.222432 | 73.639992 | \$645-675 |
| 13 | Ford High School | District | 2739 | 77.102592 | 80.746258 | 1763916.0 | 644.0 | 68.309602 | 79.299014 | 73.804308 | \$615-645 |
| 14 | Thomas High School | Charter | 1635 | 83.418349 | 83.848930 | 1043130.0 | 638.0 | 93.272171 | 97.308869 | 95.290520 | \$615-645 |

In [303]:

```
# Calculate the average math score within each spending range
df_schools.groupby(['Spending range']).mean(['Average Math']).iloc[0:, [1]]
```

Out[303]:

| Average Math | |
|----------------|-----------|
| Spending range | |
| <\$585 | 83.455399 |
| \$585-615 | 83.599686 |
| \$615-645 | 79.079225 |
| \$645-675 | 76.997210 |

In [304]:

```
# Calculate the percentage passing rate for math in each spending range
df_schools.groupby(['Spending range']).mean(['Percent passing Math']).iloc[0:, [5]]
```

Out[304]:

| Percent passing Math | |
|----------------------|-----------|
| Spending range | |
| <\$585 | 93.460096 |
| \$585-615 | 94.230858 |
| \$615-645 | 75.668212 |
| \$645-675 | 66.164813 |

In [305]:

```
# Calculate the percentage passing rate for reading in each spending range
df_schools.groupby(['Spending range']).mean(['Percent passing Reading']).iloc[0:, [6]]
```

Out[305]:

| Percent passing Reading | |
|-------------------------|-----------|
| Spending range | |
| <\$585 | 96.610877 |
| \$585-615 | 95.900287 |
| \$615-645 | 86.106569 |
| \$645-675 | 81.133951 |

In [306]:

```
# Calculate the percentage overall passing rate in each spending range
df_schools.groupby(['Spending range']).mean(['Passing rate']).iloc[0:, [7]]
```

Out[306]:

| Passing rate | |
|----------------|-----------|
| Spending range | |
| <\$585 | 95.035486 |
| \$585-615 | 95.065572 |
| \$615-645 | 80.887391 |
| \$645-675 | 73.649382 |

Scores by School Size

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

In [307]:

```
# 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 [308]:

```
# Create a new column for the bin groups
school_data_complete
list(school_data_complete.groupby('school_name', sort=False).mean()['size'])
df_schools['size'] = list(school_data_complete.groupby('school_name', sort=False).mean()
['size'])
df_schools['size'] = pd.cut(x=df_schools['size'], bins=size_bins, labels=group_names)
df_schools
```

Out[308]:

| | School | Type | Students | Average Math | Average Reading | Budget | Budget per student | Percent passing Math | Percent passing Reading | Passing rate | Spending range | Size |
|----|-----------------------|----------|----------|--------------|-----------------|-----------|--------------------|----------------------|-------------------------|--------------|----------------|--------------------|
| 0 | Huang High School | District | 2917 | 76.629414 | 81.182722 | 1910635.0 | 655.0 | 65.683922 | 81.316421 | 73.500171 | \$645-675 | Large (2000-5000) |
| 1 | Figueroa High School | District | 2949 | 76.711767 | 81.158020 | 1884411.0 | 639.0 | 65.988471 | 80.739234 | 73.363852 | \$615-645 | Large (2000-5000) |
| 2 | Shelton High School | Charter | 1761 | 83.359455 | 83.725724 | 1056600.0 | 600.0 | 93.867121 | 95.854628 | 94.860875 | \$585-615 | Medium (1000-2000) |
| 3 | Hernandez High School | District | 4635 | 77.289752 | 80.934412 | 3022020.0 | 652.0 | 66.752967 | 80.862999 | 73.807983 | \$645-675 | Large (2000-5000) |
| 4 | Griffin High School | Charter | 1468 | 83.351499 | 83.816757 | 917500.0 | 625.0 | 93.392371 | 97.138965 | 95.265668 | \$615-645 | Medium (1000-2000) |
| 5 | Wilson High School | Charter | 2283 | 83.274201 | 83.989488 | 1319574.0 | 578.0 | 93.867718 | 96.539641 | 95.203679 | <\$585 | Large (2000-5000) |
| 6 | Cabrera High School | Charter | 1858 | 83.061895 | 83.975780 | 1081356.0 | 582.0 | 94.133477 | 97.039828 | 95.586652 | <\$585 | Medium (1000-2000) |
| 7 | Bailey High School | District | 4976 | 77.048432 | 81.033963 | 3124928.0 | 628.0 | 66.680064 | 81.933280 | 74.306672 | \$615-645 | Large (2000-5000) |
| 8 | Holden High School | Charter | 427 | 83.803279 | 83.814988 | 248087.0 | 581.0 | 92.505855 | 96.252927 | 94.379391 | <\$585 | Small (<1000) |
| 9 | Pena High School | Charter | 962 | 83.839917 | 84.044699 | 585858.0 | 609.0 | 94.594595 | 95.945946 | 95.270270 | \$585-615 | Small (<1000) |
| 10 | Wright High School | Charter | 1800 | 83.682222 | 83.955000 | 1049400.0 | 583.0 | 93.333333 | 96.611111 | 94.972222 | <\$585 | Medium (1000-2000) |
| 11 | Rodriguez High School | District | 3999 | 76.842711 | 80.744686 | 2547363.0 | 637.0 | 66.366592 | 80.220055 | 73.293323 | \$615-645 | Large (2000-5000) |
| 12 | Johnson High School | District | 4761 | 77.072464 | 80.966394 | 3094650.0 | 650.0 | 66.057551 | 81.222432 | 73.639992 | \$645-675 | Large (2000-5000) |

| | | | | | | | | | | | | | |
|----|--------------------|----------|----------|--------------|-----------------|-----------|--------------------|----------------------|-------------------------|--------------|----------------|--|--|
| 13 | Ford High School | District | 2739 | 77.7 | 80.4 | 17639 | 644.0 | 68.3 | 79.2 | 73.3 | | | |
| | | Type | Students | Average Math | Average Reading | Budget | Budget per student | Percent passing Math | Percent passing Reading | Passing rate | Spending range | | |
| 14 | Thomas High School | Charter | 1635 | 83.418349 | 83.848930 | 1043130.0 | 638.0 | 93.272171 | 97.308869 | 95.290520 | \$615-645 | | |
| | | | | | | | | | | | | | |

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

In [309]:

```
# math_pass_size
df_schools.groupby(['size']).mean(['Percent passing Math']).iloc[0:, [5]]
```

Out[309]:

| Percent passing Math | |
|----------------------|-----------|
| size | |
| Small (<1000) | 93.550225 |
| Medium (1000-2000) | 93.599695 |
| Large (2000-5000) | 69.963361 |

In [310]:

```
# read_pass_size
df_schools.groupby(['size']).mean(['Percent passing Reading']).iloc[0:, [6]]
```

Out[310]:

| Percent passing Reading | |
|-------------------------|-----------|
| size | |
| Small (<1000) | 96.099437 |
| Medium (1000-2000) | 96.790680 |
| Large (2000-5000) | 82.766634 |

In [311]:

```
# Calculate the overall passing rate for different school size
df_schools.groupby(['size']).mean(['Passing rate']).iloc[0:, [7]]
```

Out[311]:

| Passing rate | |
|--------------------|-----------|
| size | |
| Small (<1000) | 94.824831 |
| Medium (1000-2000) | 95.195187 |
| Large (2000-5000) | 76.364998 |

Scores by School Type

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

In [312]:

```
# Create bins and groups, school type {'Charter', 'District'}
df_schools.groupby(['Type']).mean(['Passing rate']).iloc[0:, [5, 6, 7]]
```

Out[312]:

Out[312]:

| | Percent passing Math | Percent passing Reading | Passing rate |
|----------|----------------------|-------------------------|--------------|
| Type | | | |
| Charter | 93.620830 | 96.586489 | 95.103660 |
| District | 66.548453 | 80.799062 | 73.673757 |

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

In [313]:

```
# math pass size
print(len(school_data_complete[(school_data_complete['math_score'] >= 70])))
```

29370

In [314]:

```
# reading pass size
print(len(school_data_complete[(school_data_complete['reading_score'] >= 70)]))
```

33610

In [315]:

```
# Calculate the overall passing rate
print(len(school_data_complete[(school_data_complete['math_score'] >= 70) | (school_data_complete['reading_score'] >= 70)]))
```

37452