In [778]:

```
# Import dependencies
import pandas as pd
import numpy as np
import os
```

In [779]:

```
# define csv path
school_csv = ("raw_data/schools_complete.csv")
student_csv = ("raw_data/students_completee.csv")

# read school csv into pandas
school_df = pd.read_csv(school_csv)

# read student csv into pandas
student_df = pd.read_csv(student_csv)

# I renamed "name" to "school" so that I could merge the two csv files b
ased on the the common element "school
school_df.rename(columns = {'name': 'school'}, inplace = True)

merge_df = student_df.merge(school_df, how = 'left', on = 'school')
```

In [780]:

```
# merge student and school csv's together
merge_df = student_df.merge(school_df, how = 'left', on = 'school')

# merge_df
```

In [781]:

```
# District Summary
# made list of differnt schools
school_names = school_df['school'].unique()

# len provides the number of different schools
school_count = len(school_names)

# the number of students in each district
dist_student_count = school_df['size'].sum()

# the total number of students from student csv
tot_student = student_df['name'].count()

# the total budget of all the schools from sschool csv
tot_budget = school_df['budget'].sum()

# stats for the number of students who passed reading plus it's percenta
ge from the total number of students
pass_reading = student_df.loc[student_df['reading_score'] >= 70]['readin
g_score'].count()
per_pass_reading = pass_reading/tot_student

# stats for the number of students who passed math plus it's percentage
 from the tot # of students
pass_math = student_df.loc[student_df['math_score'] >= 70]['math_score']
.count()
per_pass_math = pass_math/tot_student

# mean math and reading scores
avg_math = student_df['math_score'].mean()
avg_reading = student_df['reading_score'].mean()

# overall passing rate
overall_pass = student_df[(student_df['math_score'] >= 70) & (student_df
['reading_score'] >= 70)]['name'].count()/tot_student

# a district summary table from the dictionary

district_summary = pd.DataFrame({"Total Schools": [school_count],
    "Total Students": [dist_student_count],
    "Total Budget": [tot_budget],
    "Average Reading Score": [avg_reading],
    "Average Math Score": [avg_math],
    "% Passing Reading":[per_pass_reading],
    "% Passing Math": [per_pass_math],
    "Overall Passing Rate": [overall_pass]})

# created different dataframe to change the order of columns
dist_sum = district_summary[["Total Schools",
                             "Total Students",
                             "Total Budget",
                             "Average Reading Score",
                             "Average Math Score",
                             '% Passing Reading',
                             '% Passing Math',
                             'Overall Passing Rate']]
```

```
# formatted the cells such that it is readable
dist_sum.style.format({"Total Budget": "${:,.2f}",
                       "Average Reading Score": "{:.1f}",
                       "Average Math Score": "{:.1f}",
                       "% Passing Math": "{:.1%}",
                       "% Passing Reading": "{:.1%}",
                       "Overall Passing Rate": "{:.1%}"})
```

Out[781]:

	Total Schools	Total Students	Total Budget	Average Reading Score	Average Math Score	% Passing Reading	% Passing Math	Overall Passing Rate
0	15	39170	$24,649,428.00	81.9	79.0	85.8%	75.0%	65.2%

In [782]:

```
# School Summary
# grouped schools by name
group_school = merge_df.set_index('school').groupby(['school'])


school_type = school_df.set_index('school')['type']
student_per_school = group_school['Student ID'].count()
school_budget = school_df.set_index('school')['budget']
student_budget = school_df.set_index('school')['budget']/school_df.set_i
ndex('school')['size']
avg_math = group_school['math_score'].mean()
avg_read = group_school['reading_score'].mean()

# percent passing math and reading
pass_math = merge_df[merge_df['math_score'] >= 70].groupby('school')['St
udent ID'].count()/student_per_school
pass_read = merge_df[merge_df['reading_score'] >= 70].groupby('school')[
'Student ID'].count()/student_per_school
overall = merge_df[(merge_df['reading_score'] >= 70) & (merge_df['math_s
core'] >= 70)].groupby('school')['Student ID'].count()/student_per_schoo
l

school_summ = pd.DataFrame({"School Type": school_type,
    "Total Students": student_per_school,
    "Per Student Budget": student_budget,
    "Total School Budget": school_budget,
    "Average Math Score": avg_math,
    "Average Reading Score": avg_read,
    '% Passing Math': pass_math,
    '% Passing Reading': pass_read,
    "Overall Passing Rate": overall})

school_summ = school_summ[['School Type',
                          'Total Students',
                          'Total School Budget',
                          'Per Student Budget',
                          'Average Math Score',
                          'Average Reading Score',
                          '% Passing Math',
                          '% Passing Reading',
                          'Overall Passing Rate']]

school_summ.style.format({'Total Students': '{:,}',
                          "Total School Budget": "${:,}",
                          "Per Student Budget": "${:.0f}",
                          'Average Math Score': "{:.1f}",
                          'Average Reading Score': "{:.1f}",
                          "% Passing Math": "{:.1%}",
                          "% Passing Reading": "{:.1%}",
                          "Overall Passing Rate": "{:.1%}"})
```

Out[782]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading
Bailey High School	District	4,976	$3,124,928	$628	77.0	81.0	66.7%	81.9%
Cabrera High School	Charter	1,858	$1,081,356	$582	83.1	84.0	94.1%	97.0%
Figueroa High School	District	2,949	$1,884,411	$639	76.7	81.2	66.0%	80.7%
Ford High School	District	2,739	$1,763,916	$644	77.1	80.7	68.3%	79.3%
Griffin High School	Charter	1,468	$917,500	$625	83.4	83.8	93.4%	97.1%
Hernandez High School	District	4,635	$3,022,020	$652	77.3	80.9	66.8%	80.9%
Holden High School	Charter	427	$248,087	$581	83.8	83.8	92.5%	96.3%
Huang High School	District	2,917	$1,910,635	$655	76.6	81.2	65.7%	81.3%
Johnson High School	District	4,761	$3,094,650	$650	77.1	81.0	66.1%	81.2%
Pena High School	Charter	962	$585,858	$609	83.8	84.0	94.6%	95.9%
Rodriguez High School	District	3,999	$2,547,363	$637	76.8	80.7	66.4%	80.2%
Shelton High School	Charter	1,761	$1,056,600	$600	83.4	83.7	93.9%	95.9%
Thomas High School	Charter	1,635	$1,043,130	$638	83.4	83.8	93.3%	97.3%

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading
Wilson High School	Charter	2,283	$1,319,574	$578	83.3	84.0	93.9%	96.5%
Wright High School	Charter	1,800	$1,049,400	$583	83.7	84.0	93.3%	96.6%

In [783]:

```
# sort schools by best performing schools in terms of overall passing rate
best_perform = school_summ.sort_values("Overall Passing Rate", ascending = False)
best_perform.head().style.format({'Total Students': '{:,}',
                           "Total School Budget": "${:,}",
                           "Per Student Budget": "${:.0f}",
                           "% Passing Math": "{:.1%}",
                           "% Passing Reading": "{:.1%}",
                           "Overall Passing Rate": "{:.1%}"})
```

Out[783]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	Ov Pa Ra
Cabrera High School	Charter	1,858	$1,081,356	$582	83.0619	83.9758	94.1%	97.0%	91
Thomas High School	Charter	1,635	$1,043,130	$638	83.4183	83.8489	93.3%	97.3%	90
Griffin High School	Charter	1,468	$917,500	$625	83.3515	83.8168	93.4%	97.1%	90
Wilson High School	Charter	2,283	$1,319,574	$578	83.2742	83.9895	93.9%	96.5%	90
Pena High School	Charter	962	$585,858	$609	83.8399	84.0447	94.6%	95.9%	90

In [784]:

```
# sort schools by worst performing schools in terms of overall passing r
ate
worst_perform = worst_perform.sort_values('Overall Passing Rate')
worst_perform.style.format({'Total Students': '{:,}',
                            "Total School Budget": "${:,}",
                            "Per Student Budget": "${:.0f}",
                            "% Passing Math": "{:.1%}",
                            "% Passing Reading": "{:.1%}",
                            "Overall Passing Rate": "{:.1%}"})
```

Out[784]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading
Rodriguez High School	District	3,999	$2,547,363	$637	76.8427	80.7447	66.4%	80.2%
Figueroa High School	District	2,949	$1,884,411	$639	76.7118	81.158	66.0%	80.7%
Huang High School	District	2,917	$1,910,635	$655	76.6294	81.1827	65.7%	81.3%
Hernandez High School	District	4,635	$3,022,020	$652	77.2898	80.9344	66.8%	80.9%
Johnson High School	District	4,761	$3,094,650	$650	77.0725	80.9664	66.1%	81.2%

In [785]:

```
# get average math scores by individual grade level
nine_math = student_df.loc[student_df['grade'] == '9th'].groupby('schoo
l')["math_score"].mean()
ten_math = student_df.loc[student_df['grade'] == '10th'].groupby('schoo
l')["math_score"].mean()
eleven_math = student_df.loc[student_df['grade'] == '11th'].groupby('sch
ool')["math_score"].mean()
twelve_math = student_df.loc[student_df['grade'] == '12th'].groupby('sch
ool')["math_score"].mean()

math_score = pd.DataFrame({"9th": nine_math,
        "10th": ten_math,
        "11th": eleven_math,
        "12th": twelve_math})

math_score = math_score[['9th', '10th', '11th', '12th']]
math_score.index.name = "School"

math_score.style.format({'9th': '{:.1f}',
                          "10th": '{:.1f}',
                          "11th": "{:.1f}",
                          "12th": "{:.1f}"})
```

Out[785]:

	9th	10th	11th	12th
School				
Bailey High School	77.1	77.0	77.5	76.5
Cabrera High School	83.1	83.2	82.8	83.3
Figueroa High School	76.4	76.5	76.9	77.2
Ford High School	77.4	77.7	76.9	76.2
Griffin High School	82.0	84.2	83.8	83.4
Hernandez High School	77.4	77.3	77.1	77.2
Holden High School	83.8	83.4	85.0	82.9
Huang High School	77.0	75.9	76.4	77.2
Johnson High School	77.2	76.7	77.5	76.9
Pena High School	83.6	83.4	84.3	84.1
Rodriguez High School	76.9	76.6	76.4	77.7
Shelton High School	83.4	82.9	83.4	83.8
Thomas High School	83.6	83.1	83.5	83.5
Wilson High School	83.1	83.7	83.2	83.0
Wright High School	83.3	84.0	83.8	83.6

In [786]:

```python
# get average reading scores by individual grade level

nine_read = students_df.loc[students_df['grade'] == '9th'].groupby('school')["reading_score"].mean()
ten_read = students_df.loc[students_df['grade'] == '10th'].groupby('school')["reading_score"].mean()
eleven_read = students_df.loc[students_df['grade'] == '11th'].groupby('school')["reading_score"].mean()
twelve_read = students_df.loc[students_df['grade'] == '12th'].groupby('school')["reading_score"].mean()


read_scores = pd.DataFrame({"9th": nine_read,
        "10th": ten_read,
        "11th": eleven_read,
        "12th": twelve_read})

read_scores = read_scores[['9th', '10th', '11th', '12th']]
read_scores.index.name = "School"

#format
read_scores.style.format({'9th': '{:.1f}',
                             "10th": '{:.1f}',
                             "11th": "{:.1f}",
                             "12th": "{:.1f}"})
```

Out[786]:

	9th	10th	11th	12th
School				
Bailey High School	81.3	80.9	80.9	80.9
Cabrera High School	83.7	84.3	83.8	84.3
Figueroa High School	81.2	81.4	80.6	81.4
Ford High School	80.6	81.3	80.4	80.7
Griffin High School	83.4	83.7	84.3	84.0
Hernandez High School	80.9	80.7	81.4	80.9
Holden High School	83.7	83.3	83.8	84.7
Huang High School	81.3	81.5	81.4	80.3
Johnson High School	81.3	80.8	80.6	81.2
Pena High School	83.8	83.6	84.3	84.6
Rodriguez High School	81.0	80.6	80.9	80.4
Shelton High School	84.1	83.4	84.4	82.8
Thomas High School	83.7	84.3	83.6	83.8
Wilson High School	83.9	84.0	83.8	84.3
Wright High School	83.8	83.8	84.2	84.1

In [787]:

```
# create individual bins based on spending ranges
bins = [0, 584.999, 614.999, 644.999, 674.999]
group_name = ['< $585', "$585 - 615", "$615 - 645", "$645 - $675"]
merge_df['spending_bins'] = pd.cut(merge_df['budget']/merge_df['size'],
bins, labels = group_name)

group_spending = merge_df.groupby('spending_bins')


avg_math = group_spending['math_score'].mean()
avg_read = group_spending['reading_score'].mean()
pass_math = merge_df[merge_df['math_score'] >= 70].groupby('spending_bin
s')['Student ID'].count()/group_spending['Student ID'].count()
pass_read = merge_df[merge_df['reading_score'] >= 70].groupby('spending_
bins')['Student ID'].count()/group_spending['Student ID'].count()
overall = merge_df[(merge_df['reading_score'] >= 70) & (merge_df['math_s
core'] >= 70)].groupby('spending_bins')['Student ID'].count()/group_spen
ding['Student ID'].count()


# dataframe to hold stats
group_by_spend_scores = pd.DataFrame({"Average Math Score": avg_math,
                                    "Average Reading Score": avg_read,
                                    '% Passing Math': pass_math,
                                    '% Passing Reading': pass_read,
                                    "Overall Passing Rate": overall})

# order the columns
group_by_spend_scores = group_by_spend_scores[["Average Math Score",
                                                "Average Reading Score",
                                                '% Passing Math',
                                                '% Passing Reading',
                                                "Overall Passing Rate"]]

group_by_spend_scores.index.name = "Per Student Budget"
group_by_spend_scores = group_by_spend_scores.reindex(group_name)

# format the columns
group_by_spend_scores.style.format({'Average Math Score': '{:.1f}',
                              'Average Reading Score': '{:.1f}',
                              '% Passing Math': '{:.1%}',
                              '% Passing Reading':'{:.1%}',
                              'Overall Passing Rate': '{:.1%}'})
```

Out[787]:

	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	Overall Passing Rate
Per Student Budget					
< $585	83.4	84.0	93.7%	96.7%	90.6%
$585 - 615	83.5	83.8	94.1%	95.9%	90.1%
$615 - 645	78.1	81.4	71.4%	83.6%	60.3%
645−**675**	77.0	81.0	66.2%	81.1%	53.5%

In [788]:

```
# create individual bins based on student count
bins = [0, 999, 1999, 99999999999]
group_name = ["Small (<1000)", "Medium (1000-2000)" , "Large (>2000)"]
merge_df['size_bins'] = pd.cut(merge_df['size'], bins, labels = group_na
me)

#group by student count
group_by_size_count = merge_df.groupby('size_bins')

#calculations
avg_math = group_by_size_count['math_score'].mean()
avg_read = group_by_size_count['math_score'].mean()
pass_math = merge_df[merge_df['math_score'] >= 70].groupby('size_bins')[
'Student ID'].count()/group_by_size_count['Student ID'].count()
pass_read = merge_df[merge_df['reading_score'] >= 70].groupby('size_bin
s')['Student ID'].count()/group_by_size_count['Student ID'].count()
overall = merge_df[(merge_df['reading_score'] >= 70) & (merge_df['math_s
core'] >= 70)].groupby('size_bins')['Student ID'].count()/group_by_size_
count['Student ID'].count()


# dataframe to hold stats
group_by_size_scores = pd.DataFrame({"Average Math Score": avg_math,
                                    "Average Reading Score": avg_read,
                                    '% Passing Math': pass_math,
                                    '% Passing Reading': pass_read,
                                    "Overall Passing Rate": overall})

# order columns
group_by_size_scores = group_by_size_scores[["Average Math Score",
                                            "Average Reading Score",
                                            '% Passing Math',
                                            '% Passing Reading',
                                            "Overall Passing Rate"]]

group_by_size_scores.index.name = "Total Students"
group_by_size_scores = group_by_size_scores.reindex(group_name)

#formating
group_by_size_scores.style.format({'Average Math Score': '{:.1f}',
                              'Average Reading Score': '{:.1f}',
                              '% Passing Math': '{:.1%}',
                              '% Passing Reading':'{:.1%}',
                              'Overall Passing Rate': '{:.1%}'})
```

Out[788]:

	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	Overall Passing Rate
Total Students					
Small (<1000)	83.8	83.8	94.0%	96.0%	90.1%
Medium (1000-2000)	83.4	83.4	93.6%	96.8%	90.6%
Large (>2000)	77.5	77.5	68.7%	82.1%	56.6%

In [789]:

```
# group by type of school
group_by_schooltype = merge_df.groupby("type")

#calculations
avg_math = group_by_schooltype['math_score'].mean()
avg_read = group_by_schooltype['math_score'].mean()
pass_math = merge_df[merge_df['math_score'] >= 70].groupby('type')['Stud
ent ID'].count()/group_by_schooltype['Student ID'].count()
pass_read = merge_df[merge_df['reading_score'] >= 70].groupby('type')['S
tudent ID'].count()/group_by_schooltype['Student ID'].count()
overall = merge_df[(merge_df['reading_score'] >= 70) & (merge_df['math_s
core'] >= 70)].groupby('type')['Student ID'].count()/group_by_schooltype
['Student ID'].count()

# df build
group_by_schooltype_scores = pd.DataFrame({
    "Average Math Score": avg_math,
    "Average Reading Score": avg_read,
    '% Passing Math': pass_math,
    '% Passing Reading': pass_read,
    "Overall Passing Rate": overall})

#reorder columns
group_by_schooltype_scores = group_by_schooltype_scores[["Average Math S
core",
                                                        "Average Reading
 Score",
                                                        '% Passing Math'
,
                                                        '% Passing Readi
ng',
                                                        "Overall Passing
 Rate"]]

group_by_schooltype_scores.index.name = "Type of School"


#formating
group_by_schooltype_scores.style.format({'Average Math Score': '{:.1f}',

                              'Average Reading Score': '{:.1f}',
                              '% Passing Math': '{:.1%}',
                              '% Passing Reading':'{:.1%}',
                              'Overall Passing Rate': '{:.1%}'})
```

Out[789]:

	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	Overall Passing Rate
Type of School					
Charter	83.4	83.4	93.7%	96.6%	90.6%
District	77.0	77.0	66.5%	80.9%	53.7%