```
In [169]:   # Dependencies and Setup
            import pandas as pd
            import numpy as np

            # File to Load (Remember to Change These)
            school_data_to_load = "Resources/schools_complete.csv"
            student_data_to_load = "Resources/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=["schoo
```

```
In [170]:   #Calculate the total number of schools
            total_schools = school_data.shape[0]
```

```
In [171]:   #Calculate the total number of students
            total_students = student_data.shape[0]
```

```
In [172]:   #Calculate the total budget $24,649,428.00
            total_budget = school_data['budget'].sum()
            total_budget = '${0:,.2f}'.format(total_budget)
```

```
In [173]:   #Calculate the average math score 78.985371
            avg_math_score = student_data['math_score'].mean()
            avg_math_score = '{0:.6f}'.format(avg_math_score)
```

```
In [174]:   #Calculate the average reading score 81.87784
            avg_reading_score = student_data['reading_score'].mean()
            avg_reading_score = '{0:.5f}'.format(avg_reading_score)
```

```
In [175]:   #Calculate the overall passing rate (overall average score), i.e. (avg. math scor
            overall_passing_rate = (float(avg_math_score) + float(avg_reading_score))/2
            overall_passing_rate = '{0:.6f}'.format(overall_passing_rate)
```

```
In [176]:   #Calculate the percentage of students with a passing math score (70 or greater) 7
            passing_math_students = len(student_data[student_data['math_score'] >= 70])
            percentage_passing_math_score = (passing_math_students/total_students) * 100
            percentage_passing_math_score = "{0:.6f}".format(percentage_passing_math_score)
```

```
In [177]:   #Calculate the percentage of students with a passing reading score (70 or greater
            passing_reading_students = len(student_data[student_data['reading_score'] >= 70])
            percentage_passing_reading_score = (passing_reading_students/total_students) * 10
            percentage_passing_reading_score = "{0:.6f}".format(percentage_passing_reading_sco
```

In [178]:
```python
#Create a dataframe to hold the above results
district_summary = [{'Total Schools': total_schools, 'Total Students': total_stud
                     'Average Math Score': avg_math_score, 'Average Reading Scor
                      '% Passing Math': percentage_passing_math_score, '% Passi
                      '% Overall Passing Rate' : overall_passing_rate }]
table_titles = ['Total Schools','Total Students', 'Total Budget', 'Average Math S
                '% Passing Reading', '% Overall Passing Rate']
district_summary_df = pd.DataFrame(district_summary)

#give the displayed data cleaner formatting
district_summary_df.reindex(columns = table_titles)
```

Out[178]:

	Total Schools	Total Students	Total Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing Rate
0	15	39170	$24,649,428.00	78.985371	81.87784	74.980853	85.805463	80.431606

In [179]:
```python
#School Summary
```

In [180]:
```python
schools_summary = school_data_complete.groupby(['school_name']).sum()
schools_summary = schools_summary[['reading_score','math_score']]
schools_summary = pd.merge(schools_summary, school_data, how="left", on=["school_
schools_summary['Per Student Budget'] = schools_summary['budget']/schools_summary
schools_summary['Average Math Score'] = schools_summary['math_score']/schools_sum
schools_summary['Average Reading Score'] = schools_summary['reading_score']/schoo
passing_Math=school_data_complete[school_data_complete['math_score'] >= 70].group
schools_summary['% Passing Math'] = (passing_Math.tolist()/schools_summary['size'
passing_Reading=school_data_complete[school_data_complete['reading_score'] >= 70]
schools_summary['% Passing Reading'] = (passing_Reading.tolist()/schools_summary[
schools_summary['% Overall Passing Rate'] = (schools_summary['% Passing Math'] +
schools_summary.rename(columns = {'type':'School Type', 'size' :'Total Students',
schools_summary = schools_summary.set_index(['school_name'])
schools_summary.index.name = ''
schools_summary = schools_summary.drop(columns=['reading_score', 'math_score', 'S
```

In [182]:
```python
#Sort and display the top five schools in overall passing rate
top_schools = schools_summary.nlargest(5, '% Overall Passing Rate')
#Formatting results
top_schools['Total School Budget'] = top_schools['Total School Budget'].map('${:,
top_schools['Per Student Budget'] = top_schools['Per Student Budget'].map('${:,.2
top_schools['% Passing Math'] = top_schools['% Passing Math'].map('{:.6f}'.format
top_schools['% Passing Reading'] = top_schools['% Passing Reading'].map('{:.6f}'.
top_schools['% Overall Passing Rate'] = top_schools['% Overall Passing Rate'].map
top_schools
```

Out[182]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% P
Cabrera High School	Charter	1858	$1,081,356.00	$582.00	83.061895	83.975780	94.133477	97.039828	95.
Thomas High School	Charter	1635	$1,043,130.00	$638.00	83.418349	83.848930	93.272171	97.308869	95.
Pena High School	Charter	962	$585,858.00	$609.00	83.839917	84.044699	94.594595	95.945946	95.
Griffin High School	Charter	1468	$917,500.00	$625.00	83.351499	83.816757	93.392371	97.138965	95.
Wilson High School	Charter	2283	$1,319,574.00	$578.00	83.274201	83.989488	93.867718	96.539641	95.

In [183]:
```python
#Sort and display the five worst-performing schools
bottom_schools = schools_summary.nsmallest(5, '% Overall Passing Rate')
#Formatting results
bottom_schools['Total School Budget'] = bottom_schools['Total School Budget'].map
bottom_schools['Per Student Budget'] = bottom_schools['Per Student Budget'].map('
bottom_schools['% Passing Math'] = bottom_schools['% Passing Math'].map('{:.6f}'.
bottom_schools['% Passing Reading'] = bottom_schools['% Passing Reading'].map('{:
bottom_schools['% Overall Passing Rate'] = bottom_schools['% Overall Passing Rate
bottom_schools
```

Out[183]:

	School Type	Total Students	Total School Budget	Per Student Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	
Rodriguez High School	District	3999	$2,547,363.00	$637.00	76.842711	80.744686	66.366592	80.220055	7
Figueroa High School	District	2949	$1,884,411.00	$639.00	76.711767	81.158020	65.988471	80.739234	7
Huang High School	District	2917	$1,910,635.00	$655.00	76.629414	81.182722	65.683922	81.316421	7
Johnson High School	District	4761	$3,094,650.00	$650.00	77.072464	80.966394	66.057551	81.222432	7
Ford High School	District	2739	$1,763,916.00	$644.00	77.102592	80.746258	68.309602	79.299014	7

In [184]:
```python
#Math Scores by Grade
```

In [185]:
```python
#Create a pandas series for each grade. Hint: use a conditional statement.
#Group each series by school
#Combine the series into a dataframe
math_scores = pd.DataFrame()
math_scores['9th'] = school_data_complete[school_data_complete['grade'] == '9th']
math_scores['10th'] = school_data_complete[school_data_complete['grade'] == '9th'
math_scores['11th'] = school_data_complete[school_data_complete['grade'] == '9th'
math_scores['12th'] = school_data_complete[school_data_complete['grade'] == '9th'
```

In [186]: ```
#Optional: give the displayed data cleaner formatting

math_scores.index.name = ''
math_scores
```

Out[186]:

|  | 9th | 10th | 11th | 12th |
|---|---|---|---|---|
| **Bailey High School** | 77.083676 | 77.083676 | 77.083676 | 77.083676 |
| **Cabrera High School** | 83.094697 | 83.094697 | 83.094697 | 83.094697 |
| **Figueroa High School** | 76.403037 | 76.403037 | 76.403037 | 76.403037 |
| **Ford High School** | 77.361345 | 77.361345 | 77.361345 | 77.361345 |
| **Griffin High School** | 82.044010 | 82.044010 | 82.044010 | 82.044010 |
| **Hernandez High School** | 77.438495 | 77.438495 | 77.438495 | 77.438495 |
| **Holden High School** | 83.787402 | 83.787402 | 83.787402 | 83.787402 |
| **Huang High School** | 77.027251 | 77.027251 | 77.027251 | 77.027251 |
| **Johnson High School** | 77.187857 | 77.187857 | 77.187857 | 77.187857 |
| **Pena High School** | 83.625455 | 83.625455 | 83.625455 | 83.625455 |
| **Rodriguez High School** | 76.859966 | 76.859966 | 76.859966 | 76.859966 |
| **Shelton High School** | 83.420755 | 83.420755 | 83.420755 | 83.420755 |
| **Thomas High School** | 83.590022 | 83.590022 | 83.590022 | 83.590022 |
| **Wilson High School** | 83.085578 | 83.085578 | 83.085578 | 83.085578 |
| **Wright High School** | 83.264706 | 83.264706 | 83.264706 | 83.264706 |

In [187]:
```
#Reading Score by Grade
#Create a pandas series for each grade.
#Group each series by school
#Combine the series into a dataframe

reading_scores = pd.DataFrame()
reading_scores['9th'] = school_data_complete[school_data_complete['grade'] == '9t
reading_scores['10th'] = school_data_complete[school_data_complete['grade'] == '9
reading_scores['11th'] = school_data_complete[school_data_complete['grade'] == '9
reading_scores['12th'] = school_data_complete[school_data_complete['grade'] == '9
#Optional: give the displayed data cleaner formatting
reading_scores.index.name = ''
reading_scores
```

Out[187]:

|  | 9th | 10th | 11th | 12th |
|---|---|---|---|---|
| **Bailey High School** | 81.303155 | 81.303155 | 81.303155 | 81.303155 |
| **Cabrera High School** | 83.676136 | 83.676136 | 83.676136 | 83.676136 |
| **Figueroa High School** | 81.198598 | 81.198598 | 81.198598 | 81.198598 |
| **Ford High School** | 80.632653 | 80.632653 | 80.632653 | 80.632653 |
| **Griffin High School** | 83.369193 | 83.369193 | 83.369193 | 83.369193 |
| **Hernandez High School** | 80.866860 | 80.866860 | 80.866860 | 80.866860 |
| **Holden High School** | 83.677165 | 83.677165 | 83.677165 | 83.677165 |
| **Huang High School** | 81.290284 | 81.290284 | 81.290284 | 81.290284 |
| **Johnson High School** | 81.260714 | 81.260714 | 81.260714 | 81.260714 |
| **Pena High School** | 83.807273 | 83.807273 | 83.807273 | 83.807273 |
| **Rodriguez High School** | 80.993127 | 80.993127 | 80.993127 | 80.993127 |
| **Shelton High School** | 84.122642 | 84.122642 | 84.122642 | 84.122642 |
| **Thomas High School** | 83.728850 | 83.728850 | 83.728850 | 83.728850 |
| **Wilson High School** | 83.939778 | 83.939778 | 83.939778 | 83.939778 |
| **Wright High School** | 83.833333 | 83.833333 | 83.833333 | 83.833333 |

In [188]:
```
#Scores by School Spending
```

In [189]:
```python
#Create a table that breaks down school performances based on average Spending Ra
#Average Math Score
#Average Reading Score
#% Passing Math
#% Passing Reading
#Overall Passing Rate (Average of the above two)

spending_bins = [0, 585, 615, 645, 675]
group_names = ["<$585", "$585-615", "$615-645", "$645-675"]
spending_bin = schools_summary.copy()
spending_bin['Spending Ranges (Per Student)'] = pd.cut(spending_bin['Per Student
spending_bin.drop(columns = ['Total Students','Total School Budget','Per Student
spending_bin.groupby(spending_bin['Spending Ranges (Per Student)']).mean()
```

Out[189]:

Spending Ranges (Per Student)	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing Rate
<$585	83.455399	83.933814	93.460096	96.610877	95.035486
$585-615	83.599686	83.885211	94.230858	95.900287	95.065572
$615-645	79.079225	81.891436	75.668212	86.106569	80.887391
$645-675	76.997210	81.027843	66.164813	81.133951	73.649382

In [190]:
```python
#Scores by School Size
```

In [191]:
```python
size_bins = [0, 1000, 2000, 5000]
group_names = ["Small (<1000)", "Medium (1000-2000)", "Large (2000-5000)"]
school_size_bin = schools_summary.copy()
school_size_bin['School Size'] = pd.cut(school_size_bin['Total Students'], size_b
school_size_bin.drop(columns = ['Total Students','Total School Budget','Per Stude
school_size_bin.groupby(school_size_bin['School Size']).mean()
```

Out[191]:

School Size	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing Rate
Small (<1000)	83.821598	83.929843	93.550225	96.099437	94.824831
Medium (1000-2000)	83.374684	83.864438	93.599695	96.790680	95.195187
Large (2000-5000)	77.746417	81.344493	69.963361	82.766634	76.364998

In [192]:
```python
#Scores by School Type
```

In [193]:
```python
school_type_bin = schools_summary.copy()
school_type_bin.drop(columns = ['Total Students', 'Total School Budget', 'Per Stu
school_type_bin = school_type_bin.rename(columns = {'type' : 'School Type'})
school_type_bin.groupby(school_type_bin['School Type']).mean()
```

Out[193]:

School Type	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing Rate
Charter	83.473852	83.896421	93.620830	96.586489	95.103660
District	76.956733	80.966636	66.548453	80.799062	73.673757

In [ ]: