


```
In [178]: #Create a dataframe to hold the above results
district_summary = [{'Total Schools': total_schools, 'Total Students': total_students,
                    'Average Math Score': avg_math_score, 'Average Reading Score': avg_reading_score,
                    '% Passing Math': percentage_passing_math_score, '% Passing Reading': percentage_passing_reading_score,
                    '% Overall Passing Rate' : overall_passing_rate }]
table_titles = ['Total Schools', 'Total Students', 'Total Budget', 'Average Math Score', 'Average Reading Score',
                '% Passing Math', '% 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]: #School Summary
```

```
In [180]: 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_name"])
schools_summary['Per Student Budget'] = schools_summary['budget']/schools_summary['size']
schools_summary['Average Math Score'] = schools_summary['math_score']/schools_summary['size']
schools_summary['Average Reading Score'] = schools_summary['reading_score']/schools_summary['size']
passing_Math=school_data_complete[school_data_complete['math_score'] >= 70].groupby('school_name').sum()
schools_summary['% Passing Math'] = (passing_Math.tolist()/schools_summary['size'])
passing_Reading=school_data_complete[school_data_complete['reading_score'] >= 70].groupby('school_name').sum()
schools_summary['% Passing Reading'] = (passing_Reading.tolist()/schools_summary['size'])
schools_summary['% Overall Passing Rate'] = (schools_summary['% Passing Math'] + schools_summary['% Passing Reading'])/2
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]: #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('${:,}
top_schools['Per Student Budget'] = top_schools['Per Student Budget'].map('${:,}
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	% F
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]: #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]: #Math Scores by Grade
```

```
In [198]: #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'] == '10th']
math_scores['11th'] = school_data_complete[school_data_complete['grade'] == '11th']
math_scores['12th'] = school_data_complete[school_data_complete['grade'] == '12th']
```

In [203]: *#Optional: give the displayed data cleaner formatting*

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

Out[203]:

	9th	10th	11th	12th
Bailey High School	77.083676	76.996772	77.515588	76.492218
Cabrera High School	83.094697	83.154506	82.765560	83.277487
Figueroa High School	76.403037	76.539974	76.884344	77.151369
Ford High School	77.361345	77.672316	76.918058	76.179963
Griffin High School	82.044010	84.229064	83.842105	83.356164
Hernandez High School	77.438495	77.337408	77.136029	77.186567
Holden High School	83.787402	83.429825	85.000000	82.855422
Huang High School	77.027251	75.908735	76.446602	77.225641
Johnson High School	77.187857	76.691117	77.491653	76.863248
Pena High School	83.625455	83.372000	84.328125	84.121547
Rodriguez High School	76.859966	76.612500	76.395626	77.690748
Shelton High School	83.420755	82.917411	83.383495	83.778976
Thomas High School	83.590022	83.087886	83.498795	83.497041
Wilson High School	83.085578	83.724422	83.195326	83.035794
Wright High School	83.264706	84.010288	83.836782	83.644986

```

In [204]: #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'] == '9th']
reading_scores['10th'] = school_data_complete[school_data_complete['grade'] == '10th']
reading_scores['11th'] = school_data_complete[school_data_complete['grade'] == '11th']
reading_scores['12th'] = school_data_complete[school_data_complete['grade'] == '12th']
#Optional: give the displayed data cleaner formatting
reading_scores.index.name = ''
reading_scores

```

Out[204]:

	9th	10th	11th	12th
Bailey High School	81.303155	80.907183	80.945643	80.912451
Cabrera High School	83.676136	84.253219	83.788382	84.287958
Figueroa High School	81.198598	81.408912	80.640339	81.384863
Ford High School	80.632653	81.262712	80.403642	80.662338
Griffin High School	83.369193	83.706897	84.288089	84.013699
Hernandez High School	80.866860	80.660147	81.396140	80.857143
Holden High School	83.677165	83.324561	83.815534	84.698795
Huang High School	81.290284	81.512386	81.417476	80.305983
Johnson High School	81.260714	80.773431	80.616027	81.227564
Pena High School	83.807273	83.612000	84.335938	84.591160
Rodriguez High School	80.993127	80.629808	80.864811	80.376426
Shelton High School	84.122642	83.441964	84.373786	82.781671
Thomas High School	83.728850	84.254157	83.585542	83.831361
Wilson High School	83.939778	84.021452	83.764608	84.317673
Wright High School	83.833333	83.812757	84.156322	84.073171

```

In [188]: #Scores by School Spending

```

```
In [189]: #Create a table that breaks down school performances based on average Spending Rate
#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]:

	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing Rate
Spending Ranges (Per Student)					
<\$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]: #Scores by School Size

```
In [191]: 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]:

	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing Rate
School Size					
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]: #Scores by School Type

```
In [193]: 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]:

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

```
In [ ]:
```