

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 [39]: # Dependencies and Setup
import pandas as pd

# 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=["school_id", "school_name"])
school_data_complete.head()
```

Out[39]:

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

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 [40]: total_schools = len(school_data_complete["School ID"].unique())
total_students = school_data_complete["student_name"].count()
total_budget = school_data_complete["budget"].unique().sum()
average_math_score = school_data_complete["math_score"].mean()
average_reading_score = school_data_complete["reading_score"].mean()

reading_score_count = school_data_complete["reading_score"].count()
passing_reading_score = school_data_complete.loc[(school_data_complete["reading_score"] > 70)]
passing_reading_score_count = passing_reading_score["reading_score"].count()
percent_passing_reading = passing_reading_score_count/reading_score_count * 100

math_score_count = school_data_complete["math_score"].count()
passing_math_score = school_data_complete.loc[(school_data_complete["math_score"] > 70)]
passing_math_score_count = passing_math_score["math_score"].count()
percent_passing_math = passing_math_score_count/math_score_count * 100

school_data_complete["overall_average_score"] = (school_data_complete["reading_score"] +
school_data_complete["math_score"]) / 2

overall_average_score_count = school_data_complete["overall_average_score"].count()
passing_average_score = school_data_complete.loc[(school_data_complete["overall_average_score"] > 70)]
passing_average_score_count = passing_average_score["overall_average_score"].count()
overall_passing_rate = passing_average_score_count/overall_average_score_count * 100
#print(total_schools)
#print(total_students)
#print(total_budget)
#print(average_math_score)
#print(average_reading_score)
#print(percent_passing_reading)
#print(percent_passing_math)
#print(overall_passing_rate)

district_summary_table = pd.DataFrame({"Total Schools": total_schools,
                                     "Total Students": total_students,
                                     "Total Budget":total_budget,
                                     "Average Math Score":average_math_score,
                                     "Average Reading Score":average_reading_score,
                                     "% Passing Math":percent_passing_math,
                                     "% Passing Reading":percent_passing_reading,
                                     "% Overall Passing Rate":overall_passing_rate},index=[0])
district_summary_table["Total Students"] =district_summary_table["Total Students"]
district_summary_table["Total Budget"] =district_summary_table["Total Budget"].m

district_summary_table.head()

```

Out[40]:

	Total Schools	Total Students	Total Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	% Overall Passing Rate
0	15	39,170	\$24,649,428.00	78.985371	81.87784	74.980853	85.805463	89.387286

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 [41]: by_school = school_data_complete.set_index('school_name').groupby(['school_name',
school_types = school_data.set_index('school_name')['type']

stu_per_sch = by_school['Student ID'].count()

sch_budget = school_data.set_index('school_name')['budget']

stu_budget = school_data.set_index('school_name')['budget']/school_data.set_index('school_name')['Student ID'].count()

avg_math = by_school['math_score'].mean()

avg_read = by_school['reading_score'].mean()

# passing score
pass_math = school_data_complete[school_data_complete['math_score'] >= 70].groupby('school_name')['math_score'].mean()
pass_read = school_data_complete[school_data_complete['reading_score'] >= 70].groupby('school_name')['reading_score'].mean()
overall = school_data_complete[(school_data_complete['reading_score'] >= 70) & (school_data_complete['math_score'] >= 70)].groupby('school_name')['reading_score'].mean().mean()

# Create new data frame

sch_summary = pd.DataFrame({
    "School Type": school_types,
    "Total Students": stu_per_sch,
    "Per Student Budget": stu_budget,
    "Total School Budget": sch_budget,
    "Average Math Score": avg_math,
    "Average Reading Score": avg_read,
    "% Passing Math": pass_math,
    "% Passing Reading": pass_read,
    "Overall Passing Rate": overall
})
sch_summary.sort_values("Overall Passing Rate", ascending=False)

sch_summary.head()

```

Out[41]:

	School Type	Total Students	Per Student Budget	Total School Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	Overall Passing Rate
Bailey High School	District	4976	628.0	3124928	77.048432	81.033963	0.666801	0.819333	0.5464
Cabrera High School	Charter	1858	582.0	1081356	83.061895	83.975780	0.941335	0.970398	0.9133
Figueroa High School	District	2949	639.0	1884411	76.711767	81.158020	0.659885	0.807392	0.5320
Ford High School	District	2739	644.0	1763916	77.102592	80.746258	0.683096	0.792990	0.5428

	School Type	Total Students	Per Student Budget	Total School Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	Overall Passing Rate
Griffin High School	Charter	1468	625.0	917500	83.351499	83.816757	0.933924	0.971390	0.9059

Bottom Performing Schools (By Passing Rate)

- Sort and display the five worst-performing schools

```
In [42]: sch_summary.sort_values("Overall Passing Rate", ascending=True)

sch_summary.head()
```

Out[42]:

	School Type	Total Students	Per Student Budget	Total School Budget	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	Overall Passing Rate
Bailey High School	District	4976	628.0	3124928	77.048432	81.033963	0.666801	0.819333	0.546423
Cabrera High School	Charter	1858	582.0	1081356	83.061895	83.975780	0.941335	0.970398	0.913348
Figueroa High School	District	2949	639.0	1884411	76.711767	81.158020	0.659885	0.807392	0.532045
Ford High School	District	2739	644.0	1763916	77.102592	80.746258	0.683096	0.792990	0.542899
Griffin High School	Charter	1468	625.0	917500	83.351499	83.816757	0.933924	0.971390	0.905995

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 [43]: school_9th_math = school_data_complete[school_data_complete['grade']=='9th'].gro
school_10th_math = school_data_complete[school_data_complete['grade']=='10th'].g
school_11th_math = school_data_complete[school_data_complete['grade']=='11th'].g
school_12th_math = school_data_complete[school_data_complete['grade']=='12th'].g

# Create new data frame
grade_math_score={
    '9th':school_9th_math,
    '10th':school_10th_math,
    '11th':school_11th_math,
    '12th':school_12th_math,
}

math_score_grade = pd.DataFrame(grade_math_score)
math_score_grade.index.name = None
math_score_grade
```

Out[43]:

	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

Reading Score by Grade

- Perform the same operations as above for reading scores

```
In [44]: school_9th_reading = school_data_complete[school_data_complete['grade']=='9th'].score
school_10th_reading = school_data_complete[school_data_complete['grade']=='10th'].score
school_11th_reading = school_data_complete[school_data_complete['grade']=='11th'].score
school_12th_reading = school_data_complete[school_data_complete['grade']=='12th'].score

# Create new data frame
reading_score={
    '9th':school_9th_reading,
    '10th':school_10th_reading,
    '11th':school_11th_reading,
    '12th':school_12th_reading
}

reading_score_grade = pd.DataFrame(reading_score)
reading_score_grade.index.name = None
reading_score_grade
```

Out[44]:

	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

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 [45]: # 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 [46]: # Create new data frame
score_spending = sch_summary.loc[:, ['Average Math Score',
                                     'Average Reading Score', '% Passing Math',
                                     '% Passing Reading', 'Overall Passing Rate',]]
score_spending['Spending Ranges (Per Student)'] = pd.cut(sch_summary['Per Student'],
                                                         spending_bins,
                                                         labels=group_names)
score_spending = score_spending.groupby('Spending Ranges (Per Student)').mean()
score_spending.head()
```

Out[46]:

	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	Overall Passing Rate
Spending Ranges (Per Student)					
<\$585	83.455399	83.933814	0.934601	0.966109	0.903695
\$585-615	83.599686	83.885211	0.942309	0.959003	0.902163
\$615-645	79.079225	81.891436	0.756682	0.861066	0.661121
\$645-675	76.997210	81.027843	0.661648	0.811340	0.535269

Scores by School Size

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

```
In [47]: # 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 [48]: # Create new data frame

score_size = sch_summary.loc[:,['Average Math Score',
                                'Average Reading Score', '% Passing Math',
                                '% Passing Reading', 'Overall Passing Rate',]]
score_size['School Size'] = pd.cut(sch_summary['Total Students'], size_bins, labels=
score_size = score_size.groupby('School Size').mean()
score_size.head()
```

Out[48]:

	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	Overall Passing Rate
School Size					
Small (<1000)	83.821598	83.929843	0.935502	0.960994	0.898839
Medium (1000-2000)	83.374684	83.864438	0.935997	0.967907	0.906215
Large (2000-5000)	77.746417	81.344493	0.699634	0.827666	0.582860

Scores by School Type

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

```
In [49]: # Create new data frame

score_type = sch_summary[['School Type', 'Average Math Score',
                            'Average Reading Score', '% Passing Math',
                            '% Passing Reading', 'Overall Passing Rate',]]
score_type = score_type.groupby('School Type').mean()
score_type.head()
```

Out[49]:

	Average Math Score	Average Reading Score	% Passing Math	% Passing Reading	Overall Passing Rate
School Type					
Charter	83.473852	83.896421	0.936208	0.965865	0.904322
District	76.956733	80.966636	0.665485	0.807991	0.536722

In []: