

In [3]: `import pandas as pd`

In [25]: `#Q1 Create a pandas Series named "grades" as the following. Display the Series.
grades=pd.Series([90,85,79,92,93], index=['Jack', 'Joan', 'Max', 'Mike', 'Kevin'])
grades.index.name='Name'
grades=grades.rename('Exam Grades')
print(grades)`

Name
Jack 90
Joan 85
Max 79
Mike 92
Kevin 93
Name: Exam Grades, dtype: int64

In [27]: `#Q2 Highest, lowest, average
max_grade=grades.max()
min_grade=grades.min()
mean_grade=grades.mean()
print(max_grade,min_grade,mean_grade)`

93 79 87.8

In [30]: `#Q3 Display the names and grades where the grades are greater than or equal to the average.
grades[(grades>mean_grade)]`

Out[30]: Name
Jack 90
Mike 92
Kevin 93
Name: Exam Grades, dtype: int64

In [31]: `#Q4 Display Joan's and Jack's grades using one statement
grades[['Joan','Jack']]`

Out[31]: Name
Joan 85
Jack 90
Name: Exam Grades, dtype: int64

In [32]: `#Q5 Change Joan's grade to be 95 and display all grades
grades['Joan']=95
print(grades)`

Name
Jack 90
Joan 95
Max 79
Mike 92
Kevin 93
Name: Exam Grades, dtype: int64

In [35]: `#Q6 Create a DataFrame similar to the vehicles sales 2D-Array used in Assignment 1
data={
 'Q1': [1200,1300,1100],
 'Q2': [1400,1500,1200],
 'Q3': [800,1600,1000],
 'Q4': [1100,1000,1050]}
cities = ['Dallas','Houston','Austin']
frame = pd.DataFrame(data, index=cities)`

In [36]: `frame`

Out[36]:

	Q1	Q2	Q3	Q4
Dallas	1200	1400	800	1100
Houston	1300	1500	1600	1000
Austin	1100	1200	1000	1050

In [37]: `#Q7 Display the sales of Q3 and display the sales of Houston, respectively (in two statements).
frame['Q3']`

Out[37]: Dallas 800
Houston 1600
Austin 1000
Name: Q3, dtype: int64

In [40]: `frame.loc['Houston']`

Out[40]: Q1 1300
Q2 1500
Q3 1600
Q4 1000
Name: Houston, dtype: int64

In [81]: `#Q8 Add a Total column to the DataFrame. Add a name of "Quarter" to the columns and add a name of "City" to the index.
frame.index.name = 'City'
frame.columns.name = 'Quarter'
frame['Total'] = frame.sum(axis = 1)
frame`

Out[81]:

Quarter	Q1	Q2	Q3	Q4	Total
City					
Dallas	1200	1400	800	1100	4500
Houston	1300	1500	1600	1000	5400
Austin	1100	1200	1000	1050	4350

In [44]: `#Q9 Display the values, index and columns of sales.
print(frame.values)
print(frame.index)
print(frame.columns)`

[[1200 1400 800 1100 4500]
[1300 1500 1600 1000 5400]
[1100 1200 1000 1050 4350]]
Index(['Dallas', 'Houston', 'Austin'], dtype='object', name='City')
Index(['Q1', 'Q2', 'Q3', 'Q4', 'Total'], dtype='object', name='Quarter')

In [45]: `#Q10 Reindex the DataFrame's rows to the order: Austin, Dallas, Houston. This will create another copy of sales.
#The order of index in the original object does not change.
new_index=['Austin','Dallas','Houston']
frame.reindex(new_index)`

Out[45]:

Quarter	Q1	Q2	Q3	Q4	Total
City					
Austin	1100	1200	1000	1050	4350
Dallas	1200	1400	800	1100	4500
Houston	1300	1500	1600	1000	5400

In [46]: `frame`

Out[46]:

Quarter	Q1	Q2	Q3	Q4	Total
City					
Dallas	1200	1400	800	1100	4500
Houston	1300	1500	1600	1000	5400
Austin	1100	1200	1000	1050	4350

In [47]: `#Q11 Display the sales of Q2 and Q3.
frame[['Q2','Q3']]`

Out[47]:

Quarter	Q2	Q3
City		
Dallas	1400	800
Houston	1500	1600
Austin	1200	1000

In [53]: `#Q12 Display the rows where Q3 sales is greater than or equal to 1000
frame[(frame.Q3>=1000)]`

Out[53]:

Quarter	Q1	Q2	Q3	Q4	Total
City					
Houston	1300	1500	1600	1000	5400
Austin	1100	1200	1000	1050	4350

In [63]: `#Q13 Sort the DataFrame's values based on "Total", from large to small.
frame.sort_values(by=['Total'], ascending=False)`

Out[63]:

Quarter	Q1	Q2	Q3	Q4	Total
City					
Houston	1300	1500	1600	1000	5400
Dallas	1200	1400	800	1100	4500
Austin	1100	1200	1000	1050	4350

In [82]: `#Q14 Add another column named "Average" to sales.
frame['Average'] = frame.mean(axis=1)
frame`

Out[82]:

Quarter	Q1	Q2	Q3	Q4	Total	Average
City						
Dallas	1200	1400	800	1100	4500	1800.0
Houston	1300	1500	1600	1000	5400	2160.0
Austin	1100	1200	1000	1050	4350	1740.0

In [83]: `#Q15 Drop both 'Total' and 'Average' columns. Add a new row with an index of "San Antonio".
#The values of the new row is [1150, 1200, 900, 1000].
frame.drop(['Total','Average'], axis=1, inplace=True)
frame.loc['San Antonio']=[1150, 1200, 900, 1000]
frame`

Out[83]:

Quarter	Q1	Q2	Q3	Q4
City				
Dallas	1200	1400	800	1100
Houston	1300	1500	1600	1000
Austin	1100	1200	1000	1050
San Antonio	1150	1200	900	1000

In [84]: `#Q16 Transpose the DataFrame and display it.
frame.T`

Out[84]:

City	Dallas	Houston	Austin	San Antonio
Quarter				
Q1	1200	1300	1100	1150
Q2	1400	1500	1200	1200
Q3	800	1600	1000	900
Q4	1100	1000	1050	1000

In [88]: `#Q17 Show a summary of summary statistics in one shot about sales by using describe function.
frame.describe()`

Out[88]:

Quarter	Q1	Q2	Q3	Q4
count	4.000000	4.0	4.000000	4.000000
mean	1187.500000	1325.0	1075.000000	1037.500000
std	85.391256	150.0	359.397644	47.871355
min	1100.000000	1200.0	800.000000	1000.000000
25%	1137.500000	1200.0	875.000000	1000.000000
50%	1175.000000	1300.0	950.000000	1025.000000
75%	1225.000000	1425.0	1150.000000	1062.500000