## Introduction: Special thanks to: http://chrisalbon.com/ for sharing the dataset and materials. Step 1. Import the necessary libraries In [1]: import pandas as pd Step 2. Create the DataFrame with the following values: In [2]: raw\_data = {'regiment': ['Nighthawks', 'Nighthawks', 'Nighthawks', 'Dragoons', 'Dragoons', 'Dragoons', 'Dragoons', 'Scouts', ' 'company': ['1st', '1st', '2nd', '1st', '1st', '2nd', '2nd', '1st', '1st', '2nd', '2nd'], 'name': ['Miller', 'Jacobson', 'Ali', 'Milner', 'Cooze', 'Jacon', 'Ryaner', 'Sone', 'Sloan', 'Piger', 'Riani', 'Ali'], 'preTestScore': [4, 24, 31, 2, 3, 4, 24, 31, 2, 3, 2, 3], 'postTestScore': [25, 94, 57, 62, 70, 25, 94, 57, 62, 70, 62, 70]} Step 3. Assign it to a variable called regiment. Don't forget to name each column In [3]: regiment = pd.DataFrame(raw\_data, columns = raw\_data.keys()) Out[3]: regiment company name preTestScore postTestScore 0 Nighthawks Miller 25 1st 94 1 Nighthawks 1st Jacobson 24 2 Nighthawks 2nd Ali 31 57 2 62 3 Nighthawks 2nd Milner Dragoons 1st Cooze 3 70 25 Dragoons 1st Jacon Dragoons 2nd Ryaner 24 94 Dragoons 31 57 2nd Sone Scouts 1st Sloan 2 62 9 Scouts Piger 70 1st 10 Scouts 2nd Riani 2 62 70 11 Scouts 2nd Step 4. What is the mean preTestScore from the regiment Nighthawks? In [4]: regiment.groupby("regiment").mean() Out[4]: preTestScore postTestScore regiment

In [5]: regiment.groupby("company").describe().transpose() company 1st 2nd Out[5]:

6.000000

2.000000

3.000000

6.000000

6.666667 15.500000 8.524475 14.652645

3.500000 13.500000

4.000000 29.250000

24.000000 31.000000

57.666667 67.000000

Step 5. Present general statistics by company

61.5

59.5

66.0

6.000000

2.000000

2.250000

6.000000

15.50

15.25

2.50

count

std

25%

**50**% 75%

count

**Dragoons** Nighthawks

**Scouts** 

preTestScore

postTestScore

regiment

Scouts

Out[9]:

Regiment

	std	27.485754	14.057027		
	min	25.000000	57.000000		
	25%	34.250000	58.250000		
	50%	66.000000	62.000000		
	75%	70.000000	68.000000		
	max	94.000000	94.000000		
Step 6. What is the mean of each company's preTestScore?					
In [6]:	regiment.groupby("d	company").	preTestSo	core.mean()	
Out[6]:	company 1st 6.666667 2nd 15.500000 Name: preTestScore,	dtype: fl	Loat64		

regiment.groupby(["regiment","company"]).preTestScore.mean()

2.5

2.5

regiment.groupby(['regiment', 'company']).mean()

27.5

3.5 Dragoons 1st 2nd 27.5 Nighthawks 1st 2nd

Step 7. Present the mean preTestScores grouped by regiment and company

Step 8. Present the mean preTestScores grouped by regiment and company without heirarchical indexing In [8]: regiment.groupby(['regiment', 'company']).preTestScore.mean().unstack() Out[8]: company 1st 2nd regiment **Dragoons** 3.5 27.5

Nighthawks 14.0 16.5

**Scouts** 2.5 2.5

company

Name: preTestScore, dtype: float64

1st 2nd

preTestScore postTestScore regiment company **Dragoons** 1st 3.5 47.5

75.5

Step 9. Group the entire dataframe by regiment and company

**Nighthawks** 14.0 1st 59.5 2nd 16.5 59.5 2.5 66.0 **Scouts** 1st 2.5 2nd 66.0 Step 10. What is the number of observations in each regiment and company In [10]: regiment.groupby(['regiment', 'company']).size() regiment company Dragoons 2 2 1st 2nd Nighthawks 1st

2

2 2

regiment company name preTestScore postTestScore

Cooze

Jacon

Sone

Miller

Milner

Ali

Jacobson

2nd Ryaner

1st Miller

1st

1st

2nd

1st

1st

2nd

2nd

regiment company

2nd

1st

2nd

Scouts

Out[11]:

dtype: int64

0 Nighthawks

Dragoons

Dragoons

Dragoons

Dragoons

Nighthawks

Nighthawks

Nighthawks

3 Nighthawks

Nighthawks

5

1

2

2nd

In [11]: regiment.head(1)

25

70

25

94

57

25

94

57

62

Step 11. Iterate over a group and print the name and the whole data from the regiment

In [12]:	<pre>group = regiment.groupby("regiment") for x, group_name in group: # groupby object returns 2 obj/ tuples     print(x)     print(group_name)     print("\n")</pre>				
	Dragoons regiment company name preTestScore postTestScore				

3

4

24

name preTestScore postTestScore

4

24

31

Scouts name preTestScore postTestScore regiment company 1st Sloan Scouts 62 3 70 9 Scouts 1st Piger 2 62 10 Scouts 2nd Riani 11 Scouts Ali 70