

# Introduction to Python 090718

Dashboard ► My courses ► Python 090718 ► 21 September - 27 September ► Exercise 2

<b>Started on</b>	Sunday, 30 September 2018, 5:17 PM
<b>State</b>	Finished
<b>Completed on</b>	Sunday, 30 September 2018, 6:24 PM
<b>Time taken</b>	1 hour 7 mins
<b>Grade</b>	<b>0.00</b> out of 1.00 ( <b>0%</b> )

## Information

Load the earthquake data in QuakeData.csv into a DataFrame, and use it to answer the following questions:

**For each question submit code and answer**, either in the textbox or by submitting a file (at the bottom). Later you will be asked to do further analysis, **so save your work.**


When you are satisfied with your answer for the questions, click **Finish attempt.** To see the model answer, click **Submit all and finish.**

## Question 1

Not answered

Marked out of  
1.00

- a. What is the median earthquake magnitude?
- b. What is the correlation between magnitude and depth?

 Jjenkins\_Week 3\_Exercise #1\_and\_#2.ipynb

You've already worked with this dataset in last week's homework. Now you just need to use pandas on it.

One piece of good news is that the dates are well-formatted enough for pandas to parse on its own -- no need to write a special date parser like we did for the movie data!

In [20]:

```
quakes = pandas.read_csv("QuakeData.csv", parse_dates=[0])
```

In [21]:

```
quakes.dtypes
```

a)

Finding the median is just a matter of the *median()* method on the Magnitude column.

In [22]:

```
quakes.Magnitude.median()
```

b)

The `.corr()` method on the entire DataFrame will give you the correlations between all columns, including the ones you care about:

In [23]:

```
quakes.corr()
```

Or:

Each column also has its own `.corr(...)` method, which takes another column or Series as an argument and finds the correlation between them:

In [24]:

```
quakes.Depth.corr(quakes.Magnitude)
```

## ◀ Exercise 1

Jump to...



End of Week Assignment 3 - due at 02:00 AM ET, TUESDAY, OCTOBER 2, 2018 ▶