

# Exercise Practice #1

This exercise will present a simple walkthrough of the Python script structure and what could be done when using Python for the purposes of data analysis and data preparation. Python is an interpreted programming language that has many packages designed primarily for data analysis and preparation. Data munging/wrangling is made simpler when using Python compared to R. And Python has many packages that can help you do the task, following is the list of the major Python packages that will be used in this class: 1. Pandas 2. numpy 3. cPickle 4. SQLAlchemy 5. PyMongo

```
In [1]: # General syntax to import specific functions in a library:  
##from (library) import (specific library function)  
  
from pandas import DataFrame, read_csv
```

```
In [2]: #General syntax to import a library but no functions:  
##import (library) as (give the library a nickname/alias)  
  
import matplotlib.pyplot as plt  
import pandas as pd  
import sys # needed to determine Python version number
```

```
In [3]: print 'Python version ' + sys.version  
print 'Pandas version ' + pd.__version__
```

```
Python version 2.7.11 | 64-bit | (default, Jun 11 2016, 03:41:56)  
[GCC 4.2.1 Compatible Apple LLVM 6.0 (clang-600.0.57)]  
Pandas version 0.19.0
```

```
In [4]: #####
##### 1. Create Data #####
#####

CityNames = ['Atlanta','Houston','Columbus','SanFrancisco','LosAngeles']
StateNames = ['GA','TX','OH','CA','CA']
DaysSpent = [4, 5, 3, 7, 2]
YearVisited = [2017, 2013, 2016, 2016, 2010]

VacationDataSet = zip(CityNames ,StateNames, DaysSpent, YearVisited)

df = pd.DataFrame(data = VacationDataSet, columns=['CityNames', 'StateNames', 'DaysSpent', 'YearVisited'])

#Sanity test: what is in the dataframe object
df
```

```
Out[4]:
```

	CityNames	StateNames	DaysSpent	YearVisited
0	Atlanta	GA	4	2017
1	Houston	TX	5	2013
2	Columbus	OH	3	2016
3	SanFrancisco	CA	7	2016
4	LosAngeles	CA	2	2010

```
In [5]: #####
##### 2. Write Data to CSV File #####
#####

df.to_csv('VacationHistory.csv',index=False,header=False)
```

```
In [6]: #####
##### 3. Read Data from CSV File #####
#####
#Location = r'C:\bader\nu\420\ExercisePractices\ExercisePractice1\Vaca
tionHistory.csv'
Location = r'/Users/Zeeshan/Desktop/PREDICT 420/Sync Session1-2/Exerci
sePractice1/VacationHistory.csv'
df = pd.read_csv(Location, names=['CityNames', 'StateNames', 'DaysSpent', 'YearVisited'])
```

```

In [7]: #####
#####
4. Prepare Data from #####
# The data we have consists of 'CityNames' , 'StateNames' , 'DaysSpent' ,
'YearVisited' ####
# We already know that we have 5 records and none of the records are m
issing (non-null values). ###
#####

df.dtypes

```

```

Out[7]: CityNames      object
StateNames    object
DaysSpent      int64
YearVisited    int64
dtype: object

```

```

In [8]: df

```

```

Out[8]:

```

	CityNames	StateNames	DaysSpent	YearVisited
0	Atlanta	GA	4	2017
1	Houston	TX	5	2013
2	Columbus	OH	3	2016
3	SanFrancisco	CA	7	2016
4	LosAngeles	CA	2	2010

```
In [9]: #####
#####
##### 5. Analyze Data #####
#####

Sorted = df.sort(['YearVisited'], ascending=False)
Sorted
```

```
/Users/Zeeshan/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-
packages/ipykernel/__main__.py:5: FutureWarning: sort(columns=....
) is deprecated, use sort_values(by=.....)
```

Out[9]:

	CityNames	StateNames	DaysSpent	YearVisited
0	Atlanta	GA	4	2017
2	Columbus	OH	3	2016
3	SanFrancisco	CA	7	2016
1	Houston	TX	5	2013
4	LosAngeles	CA	2	2010

```
In [10]: Sorted.head(2)
```

Out[10]:

	CityNames	StateNames	DaysSpent	YearVisited
0	Atlanta	GA	4	2017
2	Columbus	OH	3	2016

```
In [11]: Sorted = df.sort(['DaysSpent'], ascending=False)
Sorted
```

```
/Users/Zeeshan/Library/Enthought/Canopy_64bit/User/lib/python2.7/site-
packages/ipykernel/__main__.py:1: FutureWarning: sort(columns=....
) is deprecated, use sort_values(by=.....)
if __name__ == '__main__':
```

Out[11]:

	CityNames	StateNames	DaysSpent	YearVisited
3	SanFrancisco	CA	7	2016
1	Houston	TX	5	2013
0	Atlanta	GA	4	2017
2	Columbus	OH	3	2016
4	LosAngeles	CA	2	2010

In [12]: `Sorted.tail(2)`

Out[12]:

	CityNames	StateNames	DaysSpent	YearVisited
2	Columbus	OH	3	2016
4	LosAngeles	CA	2	2010

In [13]: `df['DaysSpent'].max()`

Out[13]: 7

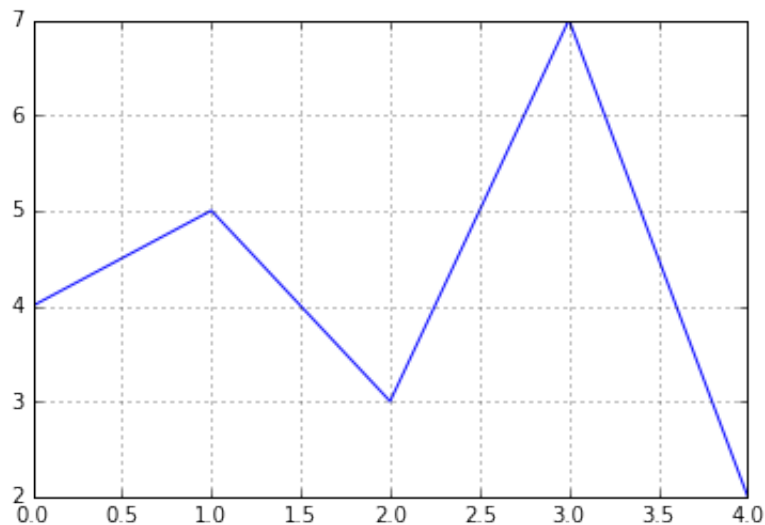
In [14]:

```
#####
#####
##### 6. Present Data #####
#####

%matplotlib inline

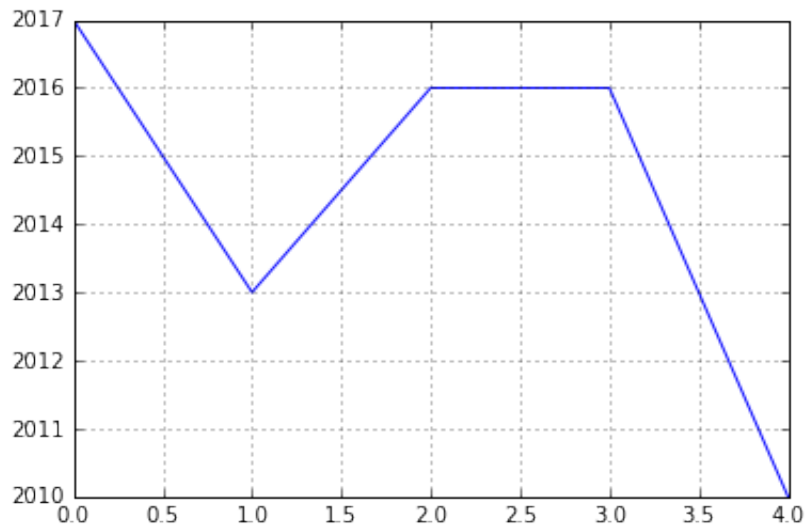
df['DaysSpent'].plot()

ax = plt.gca()
ax.grid(True)
ax.get_yaxis().get_major_formatter().set_useOffset(False)
```



```
In [15]: df['YearVisited'].plot()

ax = plt.gca()
ax.grid(True)
ax.get_yaxis().get_major_formatter().set_useOffset(False)
```



## Deliverable :

Update and run this script to include YOUR PERSONAL vactaion history; update the script to include your personal list of cities, states, days spent, and year visited

In [ ]: