LESSON PLAN

LESSON #1

Aim: Data reading and selection part 1

Objective: After a lesson on reading and selection of data, students will be able to use Python to perform the same procedures on two other datasets.

Do Now: What is a data science? https://www.youtube.com/watch?v=oMBE2TeH2no

Standards: 9-12. IC.7 Impacts of Computing, Career Paths

9-12. CT.2 Computational Thinking, Data Analysis, and Visualization

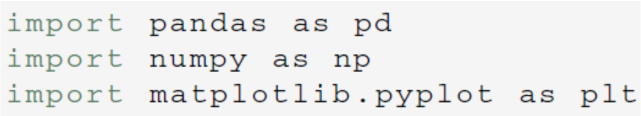
9-12. CT.3 Computational Thinking, Data Analysis, and Visualization

Mini-Lesson:

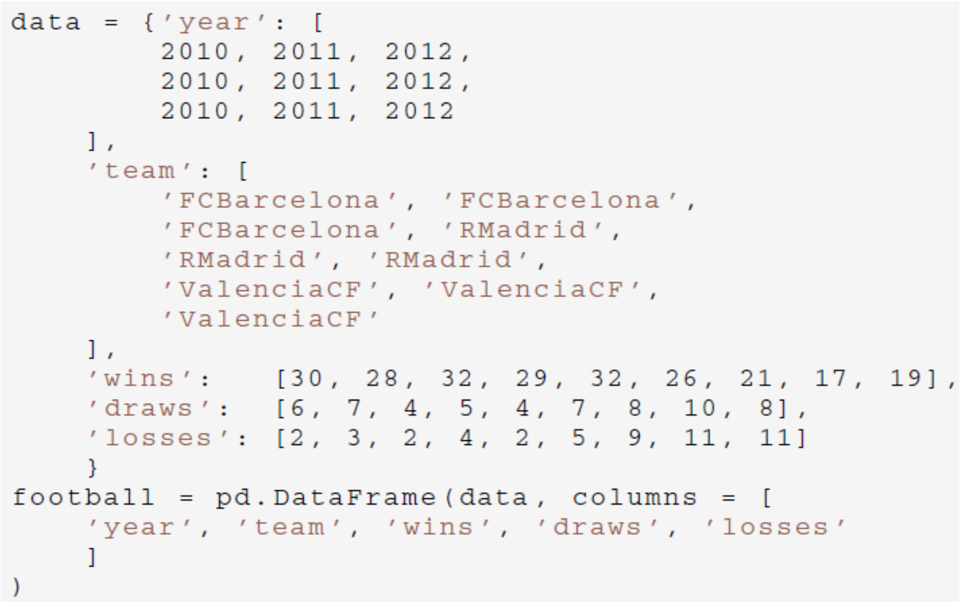
1. Creating a data frame object
   Code Below: use a pandas constructor with a dictionary of list arguments
   Why pandas? Pandas offers several useful functions such as: aggregation, manipulation, and transformation.
2. Reading the Open Government Data from Eurostat
   a. Data downloaded and stored as: educ_figdp_1_Data.csv
   b. Download in same directory as Python. Open file, then read file
   c. Viewing top and bottom of data frame
   d. Viewing summary of the data frame
3. How to select a subset of data from the data frame?
   a. Selecting one column
   b. Selecting a subset of rows
   c. Selecting a subset of columns and rows
4. Conclusion and summary
   Discussion:
   » What are some of the problems or challenges you encountered?
   » How did you resolve them?
   » What did you learn from this lesson?
   » Do you have any lingering questions on today's lesson or data science in general?

DATA FRAME OBJECT CODE

Import

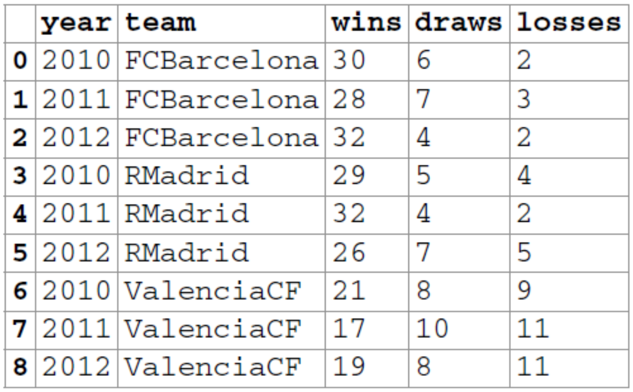


```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```



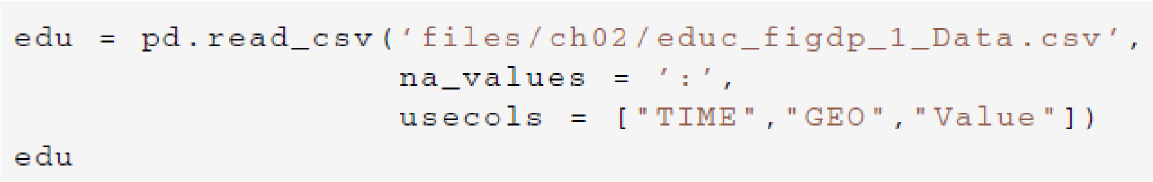
```
data = {’year’: [
          2010, 2011, 2012,
          2010, 2011, 2012,
          2010, 2011, 2012
     ],
     ’team’: [
          ’FCBarcelona’, ’FCBarcelona’,
          ’FCBarcelona’, ’RMadrid’,
          ’RMadrid’, ’RMadrid’,
          ’ValenciaCF’, ’ValenciaCF’,
          ’ValenciaCF’
     ],
     ’wins’:   [30, 28, 32, 29, 32, 26, 21, 17, 19],
     ’draws’:  [6, 7, 4, 5, 4, 7, 8, 10, 8],
     ’losses’: [2, 3, 2, 4, 2, 5, 9, 11, 11]
     }
football = pd.DataFrame(data, columns = [
     ’year’, ’team’, ’wins’, ’draws’, ’losses’
     ]
)
```

Output



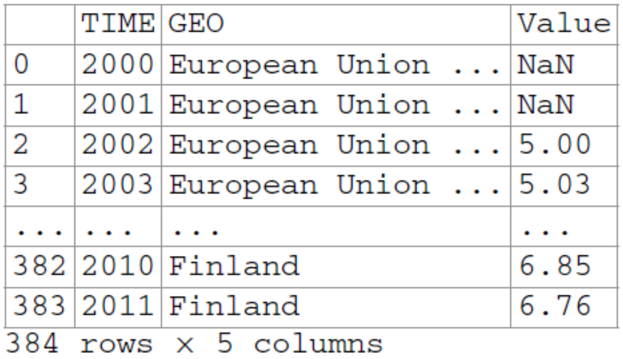
	year	team	wins	draws	losses
0	2010	FCBarcelona	30	6	2
1	2011	FCBarcelona	28	7	3
2	2012	FCBarcelona	32	4	2
3	2010	RMadrid	29	5	4
4	2011	RMadrid	32	4	2
5	2012	RMadrid	26	7	5
6	2010	ValenciaCF	21	8	9
7	2011	ValenciaCF	17	10	11
8	2012	ValenciaCF	19	8	11

READING THE OPEN GOVERNMENT DATA



```
edu = pd.read_csv(’files/ch02/educ_figdp_1_Data.csv’,
                  na_values = ’:’,
                  usecols = ["TIME","GEO","Value"])
edu
```

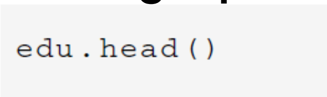
Output



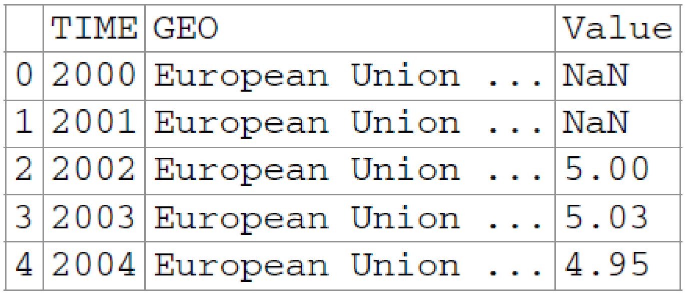
	TIME	GEO	Value
0	2000	European Union ...	NaN
1	2001	European Union ...	NaN
2	2002	European Union ...	5.00
3	2003	European Union ...	5.03
...	...	...	...
382	2010	Finland	6.85
383	2011	Finland	6.76

384 rows × 5 columns

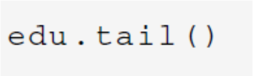
Viewing top and bottom of data frame



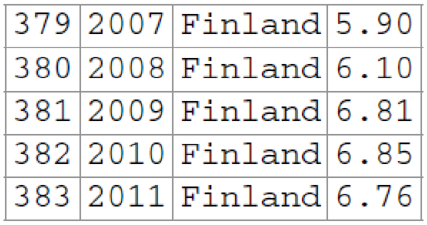
```
edu.head()
```



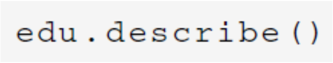
	TIME	GEO	Value
0	2000	European Union ...	NaN
1	2001	European Union ...	NaN
2	2002	European Union ...	5.00
3	2003	European Union ...	5.03
4	2004	European Union ...	4.95



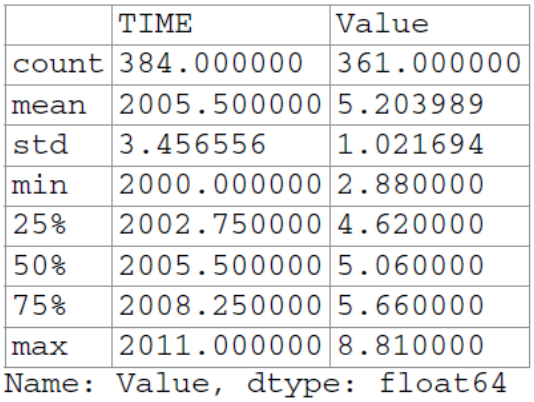
```
edu.tail()
```



379	2007	Finland	5.90
380	2008	Finland	6.10
381	2009	Finland	6.81
382	2010	Finland	6.85
383	2011	Finland	6.76



```
edu.describe()
```

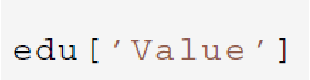


	TIME	Value
count	384.000000	361.000000
mean	2005.500000	5.203989
std	3.456556	1.021694
min	2000.000000	2.880000
25%	2002.750000	4.620000
50%	2005.500000	5.060000
75%	2008.250000	5.660000
max	2011.000000	8.810000

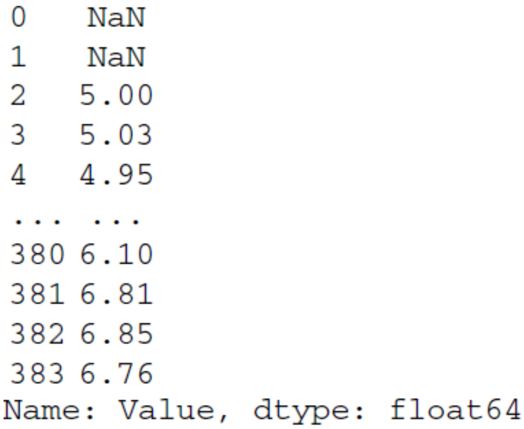
```
Name: Value, dtype: float64
```

SELECTING DATA

Selecting one column

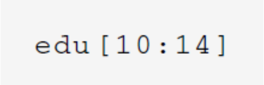


```
edu['Value']
```

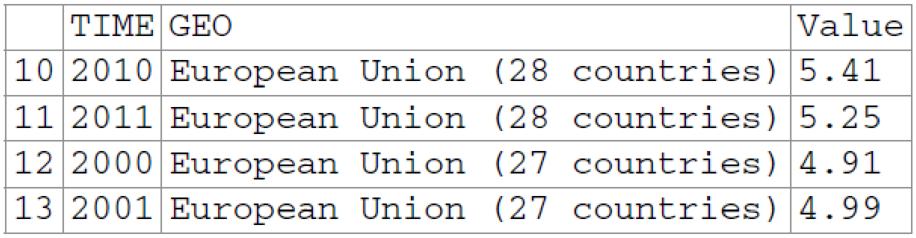


```
0    NaN
1    NaN
2   5.00
3   5.03
4   4.95
... ...
380 6.10
381 6.81
382 6.85
383 6.76
Name: Value, dtype: float64
```

Selecting a subset of rows or slicing

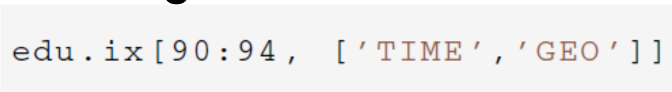


```
edu[10:14]
```

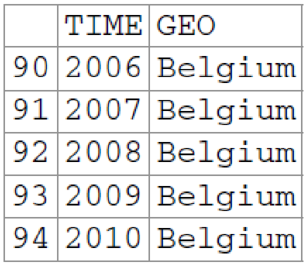


	TIME	GEO	Value
10	2010	European Union (28 countries)	5.41
11	2011	European Union (28 countries)	5.25
12	2000	European Union (27 countries)	4.91
13	2001	European Union (27 countries)	4.99

Selecting a subset of columns and rows



```
edu.ix[90:94, ['TIME','GEO']]
```



	TIME	GEO
90	2006	Belgium
91	2007	Belgium
92	2008	Belgium
93	2009	Belgium
94	2010	Belgium