

Week 2 Quiz

Bryan Gibson - brg2130

```
In [1]: # initialize with import pandas as pd
import pandas as pd

# set plots to print in notebook: %matplotlib inline
%matplotlib inline
```

```
In [2]: # use pandas to read in 'country_electricity_by_region.csv' with index_col = 'country_code' using .read_csv()
df = pd.read_csv('../data/country_electricity_by_region.csv', index_col='country_code')
```

```
In [3]: # print out dataframe info using .info()
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 217 entries, ABW to ZWE
Data columns (total 14 columns):
short_name                217 non-null object
region                    217 non-null object
income_group              217 non-null object
access_to_electricity     217 non-null float64
gdp                       193 non-null float64
population_density        215 non-null float64
population_total          216 non-null float64
unemployment              113 non-null float64
region_europe             217 non-null int64
region_latin_america_and_caribbean 217 non-null int64
region_middle_east_and_north_africa 217 non-null int64
region_north_america      217 non-null int64
region_south_asia         217 non-null int64
region_subsaharan_africa  217 non-null int64
dtypes: float64(5), int64(6), object(3)
memory usage: 25.4+ KB
```

```
In [4]: # print out the row with index label 'USA' using .loc[]
df.loc['USA']
```

```
Out[4]: short_name      United States
        region         North America
        income_group    High income
        access_to_electricity  100
        gdp              1.61775e+13
        population_density    34.8255
        population_total    3.18563e+08
        unemployment        6.17
        region_europe        0
        region_latin_america_and_caribbean  0
        region_middle_east_and_north_africa  0
        region_north_america    1
        region_south_asia       0
        region_subsaharan_africa  0
        Name: USA, dtype: object
```

```
In [5]: # print out the column with label 'region' using .loc[]
df.loc[:, 'region']
```

```
Out[5]: country_code
        ABW      Latin America & Caribbean
        AFG              South Asia
        AGO      Sub-Saharan Africa
        ALB      Europe & Central Asia
        AND      Europe & Central Asia
        ...
        XKX      Europe & Central Asia
        YEM      Middle East & North Africa
        ZAF      Sub-Saharan Africa
        ZMB      Sub-Saharan Africa
        ZWE      Sub-Saharan Africa
        Name: region, Length: 217, dtype: object
```

```
In [6]: # print out the rows with label 'SWE' and 'CAN' and column with label
        'region' using .loc[]
df.loc[['SWE', 'CAN'], 'region']
```

```
Out[6]: country_code
        SWE      Europe & Central Asia
        CAN      North America
        Name: region, dtype: object
```

```
In [7]: # print out the first 5 rows and first 3 columns using iloc[]
df.iloc[:5,:3]
```

Out[7]:

	short_name	region	income_group
country_code			
ABW	Aruba	Latin America & Caribbean	High income
AFG	Afghanistan	South Asia	Low income
AGO	Angola	Sub-Saharan Africa	Lower middle income
ALB	Albania	Europe & Central Asia	Upper middle income
AND	Andorra	Europe & Central Asia	High income

```
In [8]: # print out the last 5 rows of the 'gdp' column using .iloc[] and .loc[]
df.iloc[-5:].loc[:, 'gdp']
```

Out[8]:

country_code	
XKX	6.565321e+09
YEM	2.890029e+10
ZAF	4.130233e+11
ZMB	2.531884e+10
ZWE	1.437213e+10

Name: gdp, dtype: float64

```
In [9]: # print out the summary stats for columns 'gdp' and 'unemployment'
#        for all rows with region == 'Europe & Central Asia'
#        using .loc[] and describe()
df.loc[df.region == 'Europe & Central Asia', ['gdp', 'unemployment']].describe()
```

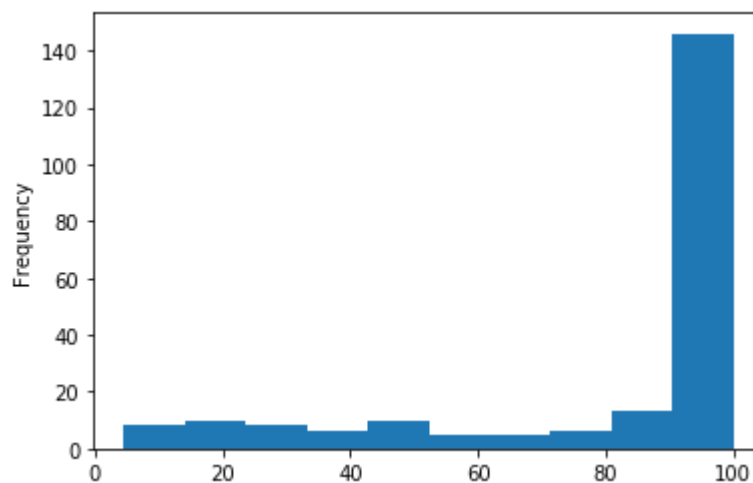
Out[9]:

	gdp	unemployment
count	5.100000e+01	48.000000
mean	4.307250e+11	11.044625
std	7.865990e+11	7.444054
min	2.287264e+09	0.492000
25%	2.028514e+10	6.045000
50%	6.314945e+10	8.825000
75%	4.324500e+11	13.357500
max	3.634080e+12	35.150002

```
In [10]: # create a bar chart of counts by 'region' using .value_counts() and
         .plot.bar()
         df.loc[:, 'region'].value_counts()
```

```
Out[10]: Europe & Central Asia      58
         Sub-Saharan Africa         48
         Latin America & Caribbean  42
         East Asia & Pacific         37
         Middle East & North Africa  21
         South Asia                  8
         North America              3
         Name: region, dtype: int64
```

```
In [11]: # print out a distribution plot of 'access_to_electricity' using plo
         t.hist()
         df.access_to_electricity.plot.hist();
```



```
In [12]: # when completed,
         # make sure you've replaced [Name] and [UNI] in the first cell and
         # filename
         # use Print Preview, Print-> Save to pdf
         # and post pdf to GradeScope
```