Lab Assignment 2: How to Load CSV, ASCII, and other data into Python

DS 6001: Practice and Application of Data Science

Instructions

Please answer the following questions as completely as possible using text, code, and the results of code as needed. Format your answers in a Jupyter notebook. To receive full credit, make sure you address every part of the problem, and make sure your document is formatted in a clean and professional way.

There are 11 data files attached to this lab assignment, with different extensions. First, download all of these data files, and save them in the same folder on your local machine. Your task in the following questions is to load each file into Python correctly, so that you can begin the process of data cleaning. If the variable names are included in the file, use those names to name the columns. If the variable names are not included, use these names in order:

In [1]:

```
column_names = ["Country", "Happiness score", "Whisker-high", "Whisker-low",
  "Dystopia (1.92) + residual", "Explained by: GDP per capita",
  "Explained by: Social support", "Explained by: Healthy life expectancy",
  "Explained by: Freedom to make life choices", "Explained by: Generosity",
  "Explained by: Perceptions of corruption" ]
```

If you loaded the data correctly, it will look like `data_clean.csv` , which is also attached to this lab.

Problem 0

Import the libraries you will need. Then write code to change the working directory to the folder in which you saved the data files, run the code displayed above to create the `column_names` list, load `data_clean.csv` , and display the output of the `.info()` method of `data_clean` . (1 point)

In [2]:

```
# Import the libraries
import os
import numpy as np
import pandas as pd
```

In [3]:

```
# Changing the working directory
os.chdir('lab data/')
```

In [4]:
```
# Check the data directory
!ls
```

```
data1.csv      data2.txt      data5.csv      data8.dta
data10.xpt     data3.txt      data6.dat      data9.sav
data11.txt     data4.txt      data7.xlsx     data_clean.csv
```

In [5]:
```
# Load clean dataset
data_clean = pd.read_csv('data_clean.csv')
```

In [6]:
```
# Create the list of column names
column_names = list(data_clean.columns)
type(column_names)
```

Out[6]:
```
list
```

In [7]:
```
data_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column                                      Non-Null Count  Dtype
---  ------                                      --------------  -----
 0   Country                                     156 non-null    object
 1   Happiness score                             156 non-null    float6
4
 2   Whisker-high                                156 non-null    float6
4
 3   Whisker-low                                 156 non-null    float6
4
 4   Dystopia (1.92) + residual                  156 non-null    float6
4
 5   Explained by: GDP per capita                156 non-null    float6
4
 6   Explained by: Social support                156 non-null    float6
4
 7   Explained by: Healthy life expectancy       156 non-null    float6
4
 8   Explained by: Freedom to make life choices  156 non-null    float6
4
 9   Explained by: Generosity                    156 non-null    float6
4
 10  Explained by: Perceptions of corruption     156 non-null    float6
4
dtypes: float64(10), object(1)
memory usage: 13.5+ KB
```

Problem 1

Load `data1.csv` . Use the tools we discussed in class to decide whether the data file loaded correctly, and include that code in your lab report. In one or two sentences, describe how you decided on the right combination of parameters needed to load the data. (1 point)

Answer 1: after skipping the first 2 rows, Finland is the first observation as expected - this does match the clean template:

In [8]:
```
# Loading and inspecting `data1`
data1 = pd.read_csv('data1.csv')
data1.head(3)
```

Out[8]:

	Source: The World Happiness Report (2018), The Sustainable Development Solutions Network (SDSN)	Unnamed: 1	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	Ur
0	URL: http://worldhappiness.report/ed/2018	NaN	NaN	NaN	NaN	NaN	
1	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	E b
2	Finland	7.632	7.695	7.569	2.595	1.305	

In [9]:
```
# Need to skip firts 2 rows as it is technical description of the data,
 not the data per se
data1 = pd.read_csv('data1.csv', skiprows=2)
data1.head(3)
```

Out[9]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Explaine b Freedo to mal li choic
0	Finland	7.632	7.695	7.569	2.595	1.305	1.592	0.874	0.68
1	Norway	7.594	7.657	7.530	2.383	1.456	1.582	0.861	0.68
2	Denmark	7.555	7.623	7.487	2.370	1.351	1.590	0.868	0.68

In [10]:
```
data_clean.head(3)
```

Out[10]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Explaine b Freedo to mal li choic
0	Finland	7.632	7.695	7.569	2.595	1.305	1.592	0.874	0.68
1	Norway	7.594	7.657	7.530	2.383	1.456	1.582	0.861	0.68
2	Denmark	7.555	7.623	7.487	2.370	1.351	1.590	0.868	0.68

Problem 2

Load `data2.txt` . Use the tools we discussed in class to decide whether the data file loaded correctly, and include that code in your lab report. In one or two sentences, describe how you decided on the right combination of parameters needed to load the data. (1 point)

Answer 2: again selected rows 0, 1, and 3 needed to be skipped. The result matches the template

In [11]:
```
data2 = pd.read_csv('data2.txt')
data2.head(5)
```

Out[11]:

	Source: The World Happiness Report (2018), The Sustainable Development Solutions Network (SDSN)	Unnamed: 1	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	Ur
0	URL: http://worldhappiness.report/ed/2018	NaN	NaN	NaN	NaN	NaN	
1	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	E b
2	/The following countries comprise the "very ha...	NaN	NaN	NaN	NaN	NaN	
3	Finland	7.632	7.695	7.569	2.595	1.305	
4	Norway	7.594	7.657	7.530	2.383	1.456	

In [13]:

```python
data2 = pd.read_csv('data2.txt', skiprows=[0, 1, 3])
data2.head(3)
```

Out[13]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Explaine b Freedo to mal li choic
0	Finland	7.632	7.695	7.569	2.595	1.305	1.592	0.874	0.68
1	Norway	7.594	7.657	7.530	2.383	1.456	1.582	0.861	0.68
2	Denmark	7.555	7.623	7.487	2.370	1.351	1.590	0.868	0.68

Problem 3

Load `data3.txt`. Use the tools we discussed in class to decide whether the data file loaded correctly, and include that code in your lab report. In one or two sentences, describe how you decided on the right combination of parameters needed to load the data. (1 point)

Answer 3: this was tap delimited text file - use `sep='\t'` and also skip the first 2 rows.

In [16]:

```python
data3 = pd.read_csv('data3.txt')
data3.head(3)
```

Out[16]:

	Source: The World Happiness Report (2018), The Sustainable Development Solutions Network (SDSN)\t\t\t\t\t\t\t\t\t\t
0	URL: http://worldhappiness.report/ed/2018\t\t\...
1	Country\tHappiness score\tWhisker-high\tWhiske...
2	Finland\t7.632\t7.695\t7.569\t2.595\t1.305\t1....

In [17]:

```python
data3 = pd.read_csv('data3.txt', sep='\t', skiprows=2)
data3.head(3)
```

Out[17]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Explaine b Freedo to mal li choic
0	Finland	7.632	7.695	7.569	2.595	1.305	1.592	0.874	0.68
1	Norway	7.594	7.657	7.530	2.383	1.456	1.582	0.861	0.68
2	Denmark	7.555	7.623	7.487	2.370	1.351	1.590	0.868	0.68

Problem 4

Load `data4.txt` . Use the tools we discussed in class to decide whether the data file loaded correctly, and include that code in your lab report. In one or two sentences, describe how you decided on the right combination of parameters needed to load the data. (1 point)

Answer 4: different delimiter was used and column names were missing - pass dollar sign to `sep=` and a list of `column_names` to `names`.

In [18]:
```
data4 = pd.read_csv('data4.txt')
data4.head(3)
```

Out[18]:

	Finland7.6327.6957.5692.5951.3051.5920.8740.6810.1920.393
0	Norway7.5947.6577.5302.3831.4561.582$0.8...
1	Denmark7.5557.6237.4872.3701.3511.590$0....
2	Iceland7.4957.5937.3982.4261.3431.644$0....

In [19]:
```
data4 = pd.read_csv('data4.txt', sep='$', names=column_names)
data4.head(3)
```

Out[19]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Explaine b Freedo to mal li choic
0	Finland	7.632	7.695	7.569	2.595	1.305	1.592	0.874	0.68
1	Norway	7.594	7.657	7.530	2.383	1.456	1.582	0.861	0.68
2	Denmark	7.555	7.623	7.487	2.370	1.351	1.590	0.868	0.68

Problem 5

Load `data5.csv` . Use the tools we discussed in class to decide whether the data file loaded correctly, and include that code in your lab report. In one or two sentences, describe how you decided on the right combination of parameters needed to load the data. (1 point)

Answer 5: this file had extra 2 rows at the bottom - used skip footer and explicitly called original python parsing engine to avoid the warning.

In [20]:
```
data5 = pd.read_csv('data5.csv')
data5.head(3)
```

Out[20]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Explaine b Freedo to mal li choic
0	Finland	7.632	7.695	7.569	2.595	1.305	1.592	0.874	0.68
1	Norway	7.594	7.657	7.530	2.383	1.456	1.582	0.861	0.68
2	Denmark	7.555	7.623	7.487	2.370	1.351	1.590	0.868	0.68

In [21]:
```
data5.columns == column_names
```

Out[21]:
```
array([ True,  True,  True,  True,  True,  True,  True,  True,  True,
        True,  True])
```

In [22]:
```
len(data5)
```

Out[22]:
```
158
```

In [23]:
```
len(data_clean)
```

Out[23]:
```
156
```

In [24]:
```
data5.tail()
```

Out[24]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Expl by: S su
153	South Sudan	3.254	3.385	3.123	1.691	0.337	
154	Central African Republic	3.083	3.227	2.939	2.487	0.024	
155	Burundi	2.905	3.074	2.735	1.752	0.091	
156	Source: The World Happiness Report (2018), The...	NaN	NaN	NaN	NaN	NaN	
157	URL: http://worldhappiness.report/ed/2018	NaN	NaN	NaN	NaN	NaN	

In [25]:
```
data_clean.tail()
```

Out[25]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Expla Free to m cho
151	Yemen	3.355	3.448	3.262	1.106	0.442	1.073	0.343	0
152	Tanzania	3.303	3.414	3.193	0.628	0.455	0.991	0.381	0
153	South Sudan	3.254	3.385	3.123	1.691	0.337	0.608	0.177	0
154	Central African Republic	3.083	3.227	2.939	2.487	0.024	0.000	0.010	0
155	Burundi	2.905	3.074	2.735	1.752	0.091	0.627	0.145	0

In [26]:
```
data5 = pd.read_csv('data5.csv', skipfooter=2, engine='python')
data5.tail(3)
```

Out[26]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Expla Free to m cho
153	South Sudan	3.254	3.385	3.123	1.691	0.337	0.608	0.177	0
154	Central African Republic	3.083	3.227	2.939	2.487	0.024	0.000	0.010	0
155	Burundi	2.905	3.074	2.735	1.752	0.091	0.627	0.145	0

In [27]:
```
len(data_clean) == len(data5)
```

Out[27]:
```
True
```

Problem 6

Load `data6.dat`. Use the tools we discussed in class to decide whether the data file loaded correctly, and include that code in your lab report. In one or two sentences, describe how you decided on the right combination of parameters needed to load the data. (1 point)

Answer 6: all looks good when using `pd.read_csv()` - checking with comparing output of `.info()` method.

In [32]:
```
data6 = pd.read_csv('data6.dat')
data6.head(3)
```

Out[32]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Explaine b Freedo to mak li choice
0	Finland	7.632	7.695	7.569	2.595	999.000	999.000	999.0	0.68
1	Norway	7.594	7.657	7.530	999.000	999.000	1.582	999.0	0.68
2	Denmark	7.555	7.623	7.487	2.370	1.351	1.590	999.0	0.68

In [33]:
```
data6.tail(3)
```

Out[33]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Expla Free to m cho
153	South Sudan	3.254	999.000	3.123	1.691	0.337	999.0	0.177	0
154	Central African Republic	3.083	3.227	2.939	2.487	0.024	0.0	0.010	0
155	Burundi	2.905	3.074	999.000	1.752	0.091	999.0	0.145	0

In [34]:
```
data6.columns == data_clean.columns
```

Out[34]:
```
array([ True,  True,  True,  True,  True,  True,  True,  True,  True,
        True,  True])
```

```
In [35]: data6.info() == data_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column                                      Non-Null Count  Dtype
---  ------                                      --------------  -----
 0   Country                                     156 non-null    object
 1   Happiness score                             156 non-null    float6
4
 2   Whisker-high                                156 non-null    float6
4
 3   Whisker-low                                 156 non-null    float6
4
 4   Dystopia (1.92) + residual                  156 non-null    float6
4
 5   Explained by: GDP per capita                156 non-null    float6
4
 6   Explained by: Social support                156 non-null    float6
4
 7   Explained by: Healthy life expectancy       156 non-null    float6
4
 8   Explained by: Freedom to make life choices  156 non-null    float6
4
 9   Explained by: Generosity                    156 non-null    float6
4
 10  Explained by: Perceptions of corruption     156 non-null    float6
4
dtypes: float64(10), object(1)
memory usage: 13.5+ KB
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column                                      Non-Null Count  Dtype
---  ------                                      --------------  -----
 0   Country                                     156 non-null    object
 1   Happiness score                             156 non-null    float6
4
 2   Whisker-high                                156 non-null    float6
4
 3   Whisker-low                                 156 non-null    float6
4
 4   Dystopia (1.92) + residual                  156 non-null    float6
4
 5   Explained by: GDP per capita                156 non-null    float6
4
 6   Explained by: Social support                156 non-null    float6
4
 7   Explained by: Healthy life expectancy       156 non-null    float6
4
 8   Explained by: Freedom to make life choices  156 non-null    float6
4
 9   Explained by: Generosity                    156 non-null    float6
4
 10  Explained by: Perceptions of corruption     156 non-null    float6
4
dtypes: float64(10), object(1)
memory usage: 13.5+ KB
```

```
Out[35]: True
```

Problem 7

Load `data7.xlsx` , which is an Excel file. Keep only the sheet named “Data”. Use the tools we discussed in class to decide whether the data file loaded correctly, and include that code in your lab report. In one or two sentences, describe how you decided on the right combination of parameters needed to load the data. (2 points)

Answer 7: after having used parameter `sheet_name='Data'` , all looks good as confirmed by `.info()` method comparison above.

```
In [36]: data7 = pd.read_excel('data7.xlsx', sheet_name='Data')
         data7.head(3)
```

Out[36]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Explaine b Freedo to mal li choic
0	Finland	7.632	7.695	7.569	2.595	1.305	1.592	0.874	0.68
1	Norway	7.594	7.657	7.530	2.383	1.456	1.582	0.861	0.68
2	Denmark	7.555	7.623	7.487	2.370	1.351	1.590	0.868	0.68

In [37]:

```
data7.info() == data_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column                                      Non-Null Count  Dtype  
---  ------                                      --------------  -----  
 0   Country                                     156 non-null    object 
 1   Happiness score                             156 non-null    float6
4
 2   Whisker-high                                156 non-null    float6
4
 3   Whisker-low                                 156 non-null    float6
4
 4   Dystopia (1.92) + residual                  156 non-null    float6
4
 5   Explained by: GDP per capita                156 non-null    float6
4
 6   Explained by: Social support                156 non-null    float6
4
 7   Explained by: Healthy life expectancy       156 non-null    float6
4
 8   Explained by: Freedom to make life choices  156 non-null    float6
4
 9   Explained by: Generosity                    156 non-null    float6
4
 10  Explained by: Perceptions of corruption     156 non-null    float6
4
dtypes: float64(10), object(1)
memory usage: 13.5+ KB
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column                                      Non-Null Count  Dtype  
---  ------                                      --------------  -----  
 0   Country                                     156 non-null    object 
 1   Happiness score                             156 non-null    float6
4
 2   Whisker-high                                156 non-null    float6
4
 3   Whisker-low                                 156 non-null    float6
4
 4   Dystopia (1.92) + residual                  156 non-null    float6
4
 5   Explained by: GDP per capita                156 non-null    float6
4
 6   Explained by: Social support                156 non-null    float6
4
 7   Explained by: Healthy life expectancy       156 non-null    float6
4
 8   Explained by: Freedom to make life choices  156 non-null    float6
4
 9   Explained by: Generosity                    156 non-null    float6
4
 10  Explained by: Perceptions of corruption     156 non-null    float6
4
dtypes: float64(10), object(1)
memory usage: 13.5+ KB
```

```
Out[37]: True
```

Problem 8

Load `data8.dta` , which is a Stata 13 file. Use the tools we discussed in class to decide whether the data file loaded correctly, and include that code in your lab report. In one or two sentences, describe how you decided on the right combination of parameters needed to load the data. (2 points)

Answer 8: after having used `pd.read_stata` , need to specify required column names but providing the list of names `column_names` .

```
In [38]: data8 = pd.read_stata('data8.dta')
         data8.head(2)
```

Out[38]:

	country	happinessscore	whiskerhigh	whiskerlow	dystopia192residual	explainedbygdppercapit
0	Finland	7.632	7.695	7.569	2.595	1.30
1	Norway	7.594	7.657	7.530	2.383	1.45

```
In [39]: data8.columns = column_names
         data8.head(2)
```

Out[39]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Explaine by Freedor to mak lif choice
0	Finland	7.632	7.695	7.569	2.595	1.305	1.592	0.874	0.68
1	Norway	7.594	7.657	7.530	2.383	1.456	1.582	0.861	0.68

```
In [40]: data8.info() == data_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column                                      Non-Null Count  Dtype
---  ------                                      --------------  -----
 0   Country                                     156 non-null    object
 1   Happiness score                             156 non-null    float3
2
 2   Whisker-high                                156 non-null    float3
2
 3   Whisker-low                                 156 non-null    float3
2
 4   Dystopia (1.92) + residual                  156 non-null    float3
2
 5   Explained by: GDP per capita                156 non-null    float3
2
 6   Explained by: Social support                156 non-null    float3
2
 7   Explained by: Healthy life expectancy       156 non-null    float3
2
 8   Explained by: Freedom to make life choices  156 non-null    float3
2
 9   Explained by: Generosity                    156 non-null    float3
2
 10  Explained by: Perceptions of corruption     156 non-null    float3
2
dtypes: float32(10), object(1)
memory usage: 8.5+ KB
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column                                      Non-Null Count  Dtype
---  ------                                      --------------  -----
 0   Country                                     156 non-null    object
 1   Happiness score                             156 non-null    float6
4
 2   Whisker-high                                156 non-null    float6
4
 3   Whisker-low                                 156 non-null    float6
4
 4   Dystopia (1.92) + residual                  156 non-null    float6
4
 5   Explained by: GDP per capita                156 non-null    float6
4
 6   Explained by: Social support                156 non-null    float6
4
 7   Explained by: Healthy life expectancy       156 non-null    float6
4
 8   Explained by: Freedom to make life choices  156 non-null    float6
4
 9   Explained by: Generosity                    156 non-null    float6
4
 10  Explained by: Perceptions of corruption     156 non-null    float6
4
dtypes: float64(10), object(1)
memory usage: 13.5+ KB
```

```
Out[40]: True
```

Problem 9

Load `data9.sav` , which is an SPSS file. Use the tools we discussed in class to decide whether the data file loaded correctly, and include that code in your lab report. In one or two sentences, describe how you decided on the right combination of parameters needed to load the data. (2 points)

Answer 9: I had to install additional library `pyreadstat` and restart the kernel to refresh `pandas` import.

```
In [108]: !pip install pyreadstat
```

```
Requirement already satisfied: pyreadstat in /Users/dmitrymikhaylov/op
t/anaconda3/lib/python3.8/site-packages (1.1.4)
Requirement already satisfied: pandas>=1.2.0 in /Users/dmitrymikhaylov/
opt/anaconda3/lib/python3.8/site-packages (from pyreadstat) (1.4.0)
Requirement already satisfied: numpy>=1.18.5 in /Users/dmitrymikhaylov/
opt/anaconda3/lib/python3.8/site-packages (from pandas>=1.2.0->pyreadst
at) (1.18.5)
Requirement already satisfied: pytz>=2020.1 in /Users/dmitrymikhaylov/o
pt/anaconda3/lib/python3.8/site-packages (from pandas>=1.2.0->pyreadsta
t) (2020.1)
Requirement already satisfied: python-dateutil>=2.8.1 in /Users/dmitrym
ikhaylov/opt/anaconda3/lib/python3.8/site-packages (from pandas>=1.2.0-
>pyreadstat) (2.8.1)
Requirement already satisfied: six>=1.5 in /Users/dmitrymikhaylov/opt/a
naconda3/lib/python3.8/site-packages (from python-dateutil>=2.8.1->pand
as>=1.2.0->pyreadstat) (1.15.0)
WARNING: You are using pip version 21.1.3; however, version 21.3.1 is a
vailable.
You should consider upgrading via the '/Users/dmitrymikhaylov/opt/anaco
nda3/bin/python -m pip install --upgrade pip' command.
```

```
In [43]: data9 = pd.read_spss('data9.sav')
         data9.head()
```

Out[43]:

	country	happiness	whiskerhigh	whiskerlow	dystopia	gdpPC	socsupport	lifeexp	lifechc
0	Finland	7.632	7.695	7.569	2.595	1.305	1.592	0.874	0.
1	Norway	7.594	7.657	7.530	2.383	1.456	1.582	0.861	0.
2	Denmark	7.555	7.623	7.487	2.370	1.351	1.590	0.868	0.
3	Iceland	7.495	7.593	7.398	2.426	1.343	1.644	0.914	0.
4	Switzerland	7.487	7.570	7.405	2.320	1.420	1.549	0.927	0.

In [44]:
```
data9.info() == data_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   country      156 non-null    object 
 1   happiness    156 non-null    float64
 2   whiskerhigh  156 non-null    float64
 3   whiskerlow   156 non-null    float64
 4   dystopia     156 non-null    float64
 5   gdpPC        156 non-null    float64
 6   socsupport   156 non-null    float64
 7   lifeexp      156 non-null    float64
 8   lifechoice   156 non-null    float64
 9   generous     156 non-null    float64
 10  corrupt      156 non-null    float64
dtypes: float64(10), object(1)
memory usage: 13.5+ KB
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column                                      Non-Null Count  Dtype  
---  ------                                      --------------  -----  
 0   Country                                     156 non-null    object 
 1   Happiness score                             156 non-null    float6
4
 2   Whisker-high                                156 non-null    float6
4
 3   Whisker-low                                 156 non-null    float6
4
 4   Dystopia (1.92) + residual                  156 non-null    float6
4
 5   Explained by: GDP per capita                156 non-null    float6
4
 6   Explained by: Social support                156 non-null    float6
4
 7   Explained by: Healthy life expectancy       156 non-null    float6
4
 8   Explained by: Freedom to make life choices  156 non-null    float6
4
 9   Explained by: Generosity                    156 non-null    float6
4
 10  Explained by: Perceptions of corruption     156 non-null    float6
4
dtypes: float64(10), object(1)
memory usage: 13.5+ KB
```

Out[44]:
```
True
```

Problem 10

Load `data10.xpt` , which is a SAS file. Use the tools we discussed in class to decide whether the data file loaded correctly, and include that code in your lab report. In one or two sentences, describe how you decided on the right combination of parameters needed to load the data. (If some of the country names display as `b'Finland'` , don't worry aout that.) (2 points)

Answer 10: I could not pass desired column names to `pd.read_sas()` therefore assigning names manually in a separate step.

In [48]:
```
data10 = pd.read_sas('data10.xpt')
data10.head()
```

Out[48]:

	COUNTRY	HAPPINES	WHISKERH	WHISKERL	DYSTOPIA	EXPLAINE	EXPLAIN2	EXPLAIN
0	b'Finland'	7.632	7.695	7.569	2.595	1.305	1.592	0.87
1	b'Norway'	7.594	7.657	7.530	2.383	1.456	1.582	0.86
2	b'Denmark'	7.555	7.623	7.487	2.370	1.351	1.590	0.86
3	b'Iceland'	7.495	7.593	7.398	2.426	1.343	1.644	0.91
4	b'Switzerland'	7.487	7.570	7.405	2.320	1.420	1.549	0.92

In [49]:
```
data10.columns = column_names
```

In [50]:

```
data10.info() == data_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column                                      Non-Null Count  Dtype  
---  ------                                      --------------  -----  
 0   Country                                     156 non-null    object 
 1   Happiness score                             156 non-null    float6
4
 2   Whisker-high                                156 non-null    float6
4
 3   Whisker-low                                 156 non-null    float6
4
 4   Dystopia (1.92) + residual                  156 non-null    float6
4
 5   Explained by: GDP per capita                156 non-null    float6
4
 6   Explained by: Social support                156 non-null    float6
4
 7   Explained by: Healthy life expectancy       156 non-null    float6
4
 8   Explained by: Freedom to make life choices  156 non-null    float6
4
 9   Explained by: Generosity                    156 non-null    float6
4
 10  Explained by: Perceptions of corruption     156 non-null    float6
4
dtypes: float64(10), object(1)
memory usage: 13.5+ KB
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column                                      Non-Null Count  Dtype  
---  ------                                      --------------  -----  
 0   Country                                     156 non-null    object 
 1   Happiness score                             156 non-null    float6
4
 2   Whisker-high                                156 non-null    float6
4
 3   Whisker-low                                 156 non-null    float6
4
 4   Dystopia (1.92) + residual                  156 non-null    float6
4
 5   Explained by: GDP per capita                156 non-null    float6
4
 6   Explained by: Social support                156 non-null    float6
4
 7   Explained by: Healthy life expectancy       156 non-null    float6
4
 8   Explained by: Freedom to make life choices  156 non-null    float6
4
 9   Explained by: Generosity                    156 non-null    float6
4
 10  Explained by: Perceptions of corruption     156 non-null    float6
4
dtypes: float64(10), object(1)
memory usage: 13.5+ KB
```

```
Out[50]: True
```

Problem 11

Please load the `data11.txt` file, which is a fixed width file. The columns are defined as follows:

Variable	Width	Start	End
Country	24	1	24
Happiness score	5	25	29
Whisker-high	5	30	34
Whisker-low	5	35	39
Dystopia (1.92) + residual	5	40	44
Explained by: GDP per capita	5	45	49
Explained by: Social support	5	50	54
Explained by: Healthy life expectancy	5	55	59
Explained by: Freedom to make life choices	5	60	64
Explained by: Generosity	5	65	69
Explained by: Perceptions of corruption	5	70	74

Then save the this loaded data frame as a CSV file on your local machine. Be sure to use a unique filename so as not to overwrite any existing files. (5 points)

Answer 11: this fixed width text file required widths of the columns that were provided manually via `widths` list; also names for the columns were provided in `column_names`.

```
In [53]: widths = [24, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5]
         data11 = pd.read_fwf('data11.txt', widths=widths, names=column_names)
         data11.head(3)
```

Out[53]:

	Country	Happiness score	Whisker-high	Whisker-low	Dystopia (1.92) + residual	Explained by: GDP per capita	Explained by: Social support	Explained by: Healthy life expectancy	Explaine b Freedo to mal li choic
0	Finland	7.632	7.695	7.569	2.595	1.305	1.592	0.874	0.68
1	Norway	7.594	7.657	7.530	2.383	1.456	1.582	0.861	0.68
2	Denmark	7.555	7.623	7.487	2.370	1.351	1.590	0.868	0.68

In [54]:

```
data11.info() == data_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column                                      Non-Null Count  Dtype  
---  ------                                      --------------  -----  
 0   Country                                     156 non-null    object 
 1   Happiness score                             156 non-null    float6
4
 2   Whisker-high                                156 non-null    float6
4
 3   Whisker-low                                 156 non-null    float6
4
 4   Dystopia (1.92) + residual                  156 non-null    float6
4
 5   Explained by: GDP per capita                156 non-null    float6
4
 6   Explained by: Social support                156 non-null    float6
4
 7   Explained by: Healthy life expectancy       156 non-null    float6
4
 8   Explained by: Freedom to make life choices  156 non-null    float6
4
 9   Explained by: Generosity                    156 non-null    float6
4
 10  Explained by: Perceptions of corruption     156 non-null    float6
4
dtypes: float64(10), object(1)
memory usage: 13.5+ KB
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 11 columns):
 #   Column                                      Non-Null Count  Dtype  
---  ------                                      --------------  -----  
 0   Country                                     156 non-null    object 
 1   Happiness score                             156 non-null    float6
4
 2   Whisker-high                                156 non-null    float6
4
 3   Whisker-low                                 156 non-null    float6
4
 4   Dystopia (1.92) + residual                  156 non-null    float6
4
 5   Explained by: GDP per capita                156 non-null    float6
4
 6   Explained by: Social support                156 non-null    float6
4
 7   Explained by: Healthy life expectancy       156 non-null    float6
4
 8   Explained by: Freedom to make life choices  156 non-null    float6
4
 9   Explained by: Generosity                    156 non-null    float6
4
 10  Explained by: Perceptions of corruption     156 non-null    float6
4
dtypes: float64(10), object(1)
memory usage: 13.5+ KB
```

```
Out[54]: True
```