Homework 03

Any Questions:
--Google!
--Discuss with peers, post questions on the class Piazza (https://piazza.com/class/j6o5l788o874i (https://piazza.com/class/j6o5l788o874i))
--Come to Office Hours on Tuesday 11am to 12 pm in Etcheverry 4176B.

Submission:
Submit on bcourses as directed in the assignment instructions.

```
In [2]: # Load required modules
        import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
```

Reading File

1) Read in a CSV file called 'data3.csv' into a dataframe called df.

Data description

- Data source: http://www.fao.org/nr/water/aquastat/data/query/index.html?* (http://www.fao.org/nr/water/aquastat/data/query/index.html?*) lang=en
- Data, units:
- GDP, current USD (CPI adjusted)
- NRI, mm/yr
- Population density, inhab/km^2
- Total area of the country, 1000 ha = 10km^2
- Total Population, unit 1000 inhabitants

2.1) Display the first 10 lines of the dataframe

2.1) Display the column names.

In [3]:

```
df = pd.read_csv('data3.csv')

print(df.head(10))
```

```
        Area  Area Id            Variable Name  Variable Id    Year  \
0  Argentina      9.0  Total area of the country       4100.0  1962.0   
1  Argentina      9.0  Total area of the country       4100.0  1967.0   
2  Argentina      9.0  Total area of the country       4100.0  1972.0   
3  Argentina      9.0  Total area of the country       4100.0  1977.0   
4  Argentina      9.0  Total area of the country       4100.0  1982.0   
5  Argentina      9.0  Total area of the country       4100.0  1987.0   
6  Argentina      9.0  Total area of the country       4100.0  1992.0   
7  Argentina      9.0  Total area of the country       4100.0  1997.0   
8  Argentina      9.0  Total area of the country       4100.0  2002.0   
9  Argentina      9.0  Total area of the country       4100.0  2007.0   

      Value Symbol  Other  
0  278040.0      E    NaN  
1  278040.0      E    NaN  
2  278040.0      E    NaN  
3  278040.0      E    NaN  
4  278040.0      E    NaN  
5  278040.0      E    NaN  
6  278040.0      E    NaN  
7  278040.0      E    NaN  
8  278040.0      E    NaN  
9  278040.0      E    NaN  
```

In [4]:

```
print(df.columns)
```

```
Index(['Area', 'Area Id', 'Variable Name', 'Variable Id', 'Year', 'Value',
       'Symbol', 'Other'],
      dtype='object')
```

Data Preprocessing

3.1) Create a mask of NAN values(i.e. apply .isnull on the dataframe). Inspect the mask for 'True' values, they denote NANs.

Hint: [You will notice that the last 8 rows and the last column ('Other') have NAN values.You can also use df.tail() to see the last lines.]

3.2) Now, we will try to get rid of the NaN valued rows and columns. Remove the bottom 8 rows from the dataframe. Also remove the column 'Other'.

```
In [5]: df.isnull()
```

```
Out[5]:
```

	Area	Area Id	Variable Name	Variable Id	Year	Value	Symbol	Other
0	False	False	False	False	False	False	False	True
1	False	False	False	False	False	False	False	True
2	False	False	False	False	False	False	False	True
3	False	False	False	False	False	False	False	True
4	False	False	False	False	False	False	False	True
5	False	False	False	False	False	False	False	True
6	False	False	False	False	False	False	False	True
7	False	False	False	False	False	False	False	True
8	False	False	False	False	False	False	False	True
9	False	False	False	False	False	False	False	True
10	False	False	False	False	False	False	False	True
11	False	False	False	False	False	False	False	True
12	False	False	False	False	False	False	False	True
13	False	False	False	False	False	False	False	True
14	False	False	False	False	False	False	False	True
15	False	False	False	False	False	False	False	True
16	False	False	False	False	False	False	False	True
17	False	False	False	False	False	False	False	True
18	False	False	False	False	False	False	False	True
19	False	False	False	False	False	False	False	True
20	False	False	False	False	False	False	False	True
21	False	False	False	False	False	False	False	True
22	False	False	False	False	False	False	False	True
23	False	False	False	False	False	False	False	True
24	False	False	False	False	False	False	False	True
25	False	False	False	False	False	False	False	True
26	False	False	False	False	False	False	False	True
27	False	False	False	False	False	False	False	True
28	False	False	False	False	False	False	False	True
29	False	False	False	False	False	False	False	True
...	...	...	...	...	...	...	...	...
368	False	False	False	False	False	False	False	True
369	False	False	False	False	False	False	False	True
370	False	False	False	False	False	False	False	True

	Area	Area Id	Variable Name	Variable Id	Year	Value	Symbol	Other
371	False	False	False	False	False	False	False	True
372	False	False	False	False	False	False	False	True
373	False	False	False	False	False	False	False	True
374	False	False	False	False	False	False	False	True
375	False	False	False	False	False	False	False	True
376	False	False	False	False	False	False	False	True
377	False	False	False	False	False	False	False	True
378	False	False	False	False	False	False	False	True
379	False	False	False	False	False	False	False	True
380	False	False	False	False	False	False	False	True
381	False	False	False	False	False	False	False	True
382	False	False	False	False	False	False	False	True
383	False	False	False	False	False	False	False	True
384	False	False	False	False	False	False	False	True
385	False	False	False	False	False	False	False	True
386	False	False	False	False	False	False	False	True
387	False	False	False	False	False	False	False	True
388	False	False	False	False	False	False	False	True
389	False	False	False	False	False	False	False	True
390	True	True	True	True	True	True	True	True
391	False	True	True	True	True	True	True	True
392	False	True	True	True	True	True	True	True
393	False	True	True	True	True	True	True	True
394	False	True	True	True	True	True	True	True
395	False	True	True	True	True	True	True	True
396	False	True	True	True	True	True	True	True
397	False	True	True	True	True	True	True	True

398 rows × 8 columns

In [6]:

```
df2 = df.iloc[:-8]
df3 = df2.dropna(axis=1,how="all")
df3
```

Out[6]:

	Area	Area Id	Variable Name	Variable Id	Year	Value	Symbol
0	Argentina	9.0	Total area of the country	4100.0	1962.0	2.780400e+05	E
1	Argentina	9.0	Total area of the country	4100.0	1967.0	2.780400e+05	E
2	Argentina	9.0	Total area of the country	4100.0	1972.0	2.780400e+05	E
3	Argentina	9.0	Total area of the country	4100.0	1977.0	2.780400e+05	E
4	Argentina	9.0	Total area of the country	4100.0	1982.0	2.780400e+05	E
5	Argentina	9.0	Total area of the country	4100.0	1987.0	2.780400e+05	E
6	Argentina	9.0	Total area of the country	4100.0	1992.0	2.780400e+05	E
7	Argentina	9.0	Total area of the country	4100.0	1997.0	2.780400e+05	E
8	Argentina	9.0	Total area of the country	4100.0	2002.0	2.780400e+05	E
9	Argentina	9.0	Total area of the country	4100.0	2007.0	2.780400e+05	E
10	Argentina	9.0	Total area of the country	4100.0	2012.0	2.780400e+05	E
11	Argentina	9.0	Total area of the country	4100.0	2014.0	2.780400e+05	E
12	Argentina	9.0	Total population	4104.0	1962.0	2.128800e+04	E
13	Argentina	9.0	Total population	4104.0	1967.0	2.293200e+04	E
14	Argentina	9.0	Total population	4104.0	1972.0	2.478300e+04	E
15	Argentina	9.0	Total population	4104.0	1977.0	2.687900e+04	E
16	Argentina	9.0	Total population	4104.0	1982.0	2.899400e+04	E
17	Argentina	9.0	Total population	4104.0	1987.0	3.132600e+04	E
18	Argentina	9.0	Total population	4104.0	1992.0	3.365500e+04	E
19	Argentina	9.0	Total population	4104.0	1997.0	3.583400e+04	E
20	Argentina	9.0	Total population	4104.0	2002.0	3.788900e+04	E
21	Argentina	9.0	Total population	4104.0	2007.0	3.997000e+04	E
22	Argentina	9.0	Total population	4104.0	2012.0	4.209500e+04	E
23	Argentina	9.0	Total population	4104.0	2015.0	4.341700e+04	E
24	Argentina	9.0	Population density	4107.0	1962.0	7.656000e+00	E
25	Argentina	9.0	Population density	4107.0	1967.0	8.248000e+00	E
26	Argentina	9.0	Population density	4107.0	1972.0	8.913000e+00	E
27	Argentina	9.0	Population density	4107.0	1977.0	9.667000e+00	E
28	Argentina	9.0	Population density	4107.0	1982.0	1.043000e+01	E
29	Argentina	9.0	Population density	4107.0	1987.0	1.127000e+01	E
...	...	...	...	...	...	...	...

	Area	Area Id	Variable Name	Variable Id	Year	Value	Symbol
360	United States of America	231.0	Population density	4107.0	1972.0	2.214000e+01	E
361	United States of America	231.0	Population density	4107.0	1977.0	2.317000e+01	E
362	United States of America	231.0	Population density	4107.0	1982.0	2.430000e+01	E
363	United States of America	231.0	Population density	4107.0	1987.0	2.549000e+01	E
364	United States of America	231.0	Population density	4107.0	1992.0	2.678000e+01	E
365	United States of America	231.0	Population density	4107.0	1997.0	2.834000e+01	E
366	United States of America	231.0	Population density	4107.0	2002.0	2.995000e+01	E
367	United States of America	231.0	Population density	4107.0	2007.0	3.132000e+01	E
368	United States of America	231.0	Population density	4107.0	2012.0	3.202000e+01	E
369	United States of America	231.0	Population density	4107.0	2015.0	3.273000e+01	E
370	United States of America	231.0	Gross Domestic Product (GDP)	4112.0	1962.0	6.050000e+11	E
371	United States of America	231.0	Gross Domestic Product (GDP)	4112.0	1967.0	8.620000e+11	E
372	United States of America	231.0	Gross Domestic Product (GDP)	4112.0	1972.0	1.280000e+12	E
373	United States of America	231.0	Gross Domestic Product (GDP)	4112.0	1977.0	2.090000e+12	E
374	United States of America	231.0	Gross Domestic Product (GDP)	4112.0	1982.0	3.340000e+12	E
375	United States of America	231.0	Gross Domestic Product (GDP)	4112.0	1987.0	4.870000e+12	E
376	United States of America	231.0	Gross Domestic Product (GDP)	4112.0	1992.0	6.540000e+12	E
377	United States of America	231.0	Gross Domestic Product (GDP)	4112.0	1997.0	8.610000e+12	E
378	United States of America	231.0	Gross Domestic Product (GDP)	4112.0	2002.0	1.100000e+13	E
379	United States of America	231.0	Gross Domestic Product (GDP)	4112.0	2007.0	1.450000e+13	E
380	United States of America	231.0	Gross Domestic Product (GDP)	4112.0	2012.0	1.620000e+13	E
381	United States of America	231.0	Gross Domestic Product (GDP)	4112.0	2015.0	1.790000e+13	E
382	United States of America	231.0	National Rainfall Index (NRI)	4472.0	1965.0	9.285000e+02	E

	Area	Area Id	Variable Name	Variable Id	Year	Value	Symbol
383	United States of America	231.0	National Rainfall Index (NRI)	4472.0	1969.0	9.522000e+02	E
384	United States of America	231.0	National Rainfall Index (NRI)	4472.0	1974.0	1.008000e+03	E
385	United States of America	231.0	National Rainfall Index (NRI)	4472.0	1981.0	9.492000e+02	E
386	United States of America	231.0	National Rainfall Index (NRI)	4472.0	1984.0	9.746000e+02	E
387	United States of America	231.0	National Rainfall Index (NRI)	4472.0	1992.0	1.020000e+03	E
388	United States of America	231.0	National Rainfall Index (NRI)	4472.0	1996.0	1.005000e+03	E
389	United States of America	231.0	National Rainfall Index (NRI)	4472.0	2002.0	9.387000e+02	E

390 rows × 7 columns

4.1) For our analysis we do not want all the columns in our dataframe. Lets drop all the redundant columns/ features.

Drop columns: Area Id, Variable Id, Symbol. Save the new dataframe as df1.

In [7]:
```
df1 = df3.drop(['Area Id', 'Variable Id', 'Symbol'],axis=1,)
df1
```

Out[7]:

	Area	Variable Name	Year	Value
0	Argentina	Total area of the country	1962.0	2.780400e+05
1	Argentina	Total area of the country	1967.0	2.780400e+05
2	Argentina	Total area of the country	1972.0	2.780400e+05
3	Argentina	Total area of the country	1977.0	2.780400e+05
4	Argentina	Total area of the country	1982.0	2.780400e+05
5	Argentina	Total area of the country	1987.0	2.780400e+05
6	Argentina	Total area of the country	1992.0	2.780400e+05
7	Argentina	Total area of the country	1997.0	2.780400e+05
8	Argentina	Total area of the country	2002.0	2.780400e+05
9	Argentina	Total area of the country	2007.0	2.780400e+05
10	Argentina	Total area of the country	2012.0	2.780400e+05
11	Argentina	Total area of the country	2014.0	2.780400e+05
12	Argentina	Total population	1962.0	2.128800e+04
13	Argentina	Total population	1967.0	2.293200e+04
14	Argentina	Total population	1972.0	2.478300e+04
15	Argentina	Total population	1977.0	2.687900e+04
16	Argentina	Total population	1982.0	2.899400e+04
17	Argentina	Total population	1987.0	3.132600e+04
18	Argentina	Total population	1992.0	3.365500e+04
19	Argentina	Total population	1997.0	3.583400e+04
20	Argentina	Total population	2002.0	3.788900e+04
21	Argentina	Total population	2007.0	3.997000e+04
22	Argentina	Total population	2012.0	4.209500e+04
23	Argentina	Total population	2015.0	4.341700e+04
24	Argentina	Population density	1962.0	7.656000e+00
25	Argentina	Population density	1967.0	8.248000e+00
26	Argentina	Population density	1972.0	8.913000e+00
27	Argentina	Population density	1977.0	9.667000e+00
28	Argentina	Population density	1982.0	1.043000e+01
29	Argentina	Population density	1987.0	1.127000e+01
...	...	...	...	...
360	United States of America	Population density	1972.0	2.214000e+01
361	United States of America	Population density	1977.0	2.317000e+01

	Area	Variable Name	Year	Value
362	United States of America	Population density	1982.0	2.430000e+01
363	United States of America	Population density	1987.0	2.549000e+01
364	United States of America	Population density	1992.0	2.678000e+01
365	United States of America	Population density	1997.0	2.834000e+01
366	United States of America	Population density	2002.0	2.995000e+01
367	United States of America	Population density	2007.0	3.132000e+01
368	United States of America	Population density	2012.0	3.202000e+01
369	United States of America	Population density	2015.0	3.273000e+01
370	United States of America	Gross Domestic Product (GDP)	1962.0	6.050000e+11
371	United States of America	Gross Domestic Product (GDP)	1967.0	8.620000e+11
372	United States of America	Gross Domestic Product (GDP)	1972.0	1.280000e+12
373	United States of America	Gross Domestic Product (GDP)	1977.0	2.090000e+12
374	United States of America	Gross Domestic Product (GDP)	1982.0	3.340000e+12
375	United States of America	Gross Domestic Product (GDP)	1987.0	4.870000e+12
376	United States of America	Gross Domestic Product (GDP)	1992.0	6.540000e+12
377	United States of America	Gross Domestic Product (GDP)	1997.0	8.610000e+12
378	United States of America	Gross Domestic Product (GDP)	2002.0	1.100000e+13
379	United States of America	Gross Domestic Product (GDP)	2007.0	1.450000e+13
380	United States of America	Gross Domestic Product (GDP)	2012.0	1.620000e+13
381	United States of America	Gross Domestic Product (GDP)	2015.0	1.790000e+13
382	United States of America	National Rainfall Index (NRI)	1965.0	9.285000e+02
383	United States of America	National Rainfall Index (NRI)	1969.0	9.522000e+02
384	United States of America	National Rainfall Index (NRI)	1974.0	1.008000e+03
385	United States of America	National Rainfall Index (NRI)	1981.0	9.492000e+02
386	United States of America	National Rainfall Index (NRI)	1984.0	9.746000e+02
387	United States of America	National Rainfall Index (NRI)	1992.0	1.020000e+03
388	United States of America	National Rainfall Index (NRI)	1996.0	1.005000e+03
389	United States of America	National Rainfall Index (NRI)	2002.0	9.387000e+02

390 rows × 4 columns

4.2) Display all the unique values in your new dataframe for column: Area, Variable Name, Year.

Note the Countries and the Metrics (ie.recorded variables) represented in your dataset.

Hint: Use .unique( ) method.

In [8]:
```
print(df1['Area'].unique())

print(df1['Variable Name'].unique())

print(df1['Year'].unique())
```

```
['Argentina' 'Australia' 'Germany' 'Iceland' 'Ireland' 'Sweden'
 'United States of America']
['Total area of the country' 'Total population' 'Population density'
 'Gross Domestic Product (GDP)' 'National Rainfall Index (NRI)']
[ 1962.  1967.  1972.  1977.  1982.  1987.  1992.  1997.  2002.  2007.
  2012.  2014.  2015.  1963.  1970.  1974.  1978.  1984.  1990.  1964.
  1981.  1985.  1996.  2001.  1969.  1973.  1979.  1993.  1971.  1975.
  1986.  1991.  1998.  2000.  1965.  1983.  1988.  1995.]
```

5) Convert the year column to pandas datetime.

Convert the 'Year' column string values to pandas datetime objects, where only the year is specified.
Hint:
df1['Year'] = pd.to_datetime(pd.Series(df1['Year']).astype(int),format='%Y').dt.year

Run df1.tail() to see if you get what you expect

```
In [9]: df1['Year']=pd.to_datetime(df1['Year'].astype(int),format='%Y').dt.year
        df1
```

Out[9]:

	Area	Variable Name	Year	Value
0	Argentina	Total area of the country	1962	2.780400e+05
1	Argentina	Total area of the country	1967	2.780400e+05
2	Argentina	Total area of the country	1972	2.780400e+05
3	Argentina	Total area of the country	1977	2.780400e+05
4	Argentina	Total area of the country	1982	2.780400e+05
5	Argentina	Total area of the country	1987	2.780400e+05
6	Argentina	Total area of the country	1992	2.780400e+05
7	Argentina	Total area of the country	1997	2.780400e+05
8	Argentina	Total area of the country	2002	2.780400e+05
9	Argentina	Total area of the country	2007	2.780400e+05
10	Argentina	Total area of the country	2012	2.780400e+05
11	Argentina	Total area of the country	2014	2.780400e+05
12	Argentina	Total population	1962	2.128800e+04
13	Argentina	Total population	1967	2.293200e+04
14	Argentina	Total population	1972	2.478300e+04
15	Argentina	Total population	1977	2.687900e+04
16	Argentina	Total population	1982	2.899400e+04
17	Argentina	Total population	1987	3.132600e+04
18	Argentina	Total population	1992	3.365500e+04
19	Argentina	Total population	1997	3.583400e+04
20	Argentina	Total population	2002	3.788900e+04
21	Argentina	Total population	2007	3.997000e+04
22	Argentina	Total population	2012	4.209500e+04
23	Argentina	Total population	2015	4.341700e+04
24	Argentina	Population density	1962	7.656000e+00
25	Argentina	Population density	1967	8.248000e+00
26	Argentina	Population density	1972	8.913000e+00
27	Argentina	Population density	1977	9.667000e+00
28	Argentina	Population density	1982	1.043000e+01
29	Argentina	Population density	1987	1.127000e+01
...	...	...	...	...
360	United States of America	Population density	1972	2.214000e+01
361	United States of America	Population density	1977	2.317000e+01

	Area	Variable Name	Year	Value
362	United States of America	Population density	1982	2.430000e+01
363	United States of America	Population density	1987	2.549000e+01
364	United States of America	Population density	1992	2.678000e+01
365	United States of America	Population density	1997	2.834000e+01
366	United States of America	Population density	2002	2.995000e+01
367	United States of America	Population density	2007	3.132000e+01
368	United States of America	Population density	2012	3.202000e+01
369	United States of America	Population density	2015	3.273000e+01
370	United States of America	Gross Domestic Product (GDP)	1962	6.050000e+11
371	United States of America	Gross Domestic Product (GDP)	1967	8.620000e+11
372	United States of America	Gross Domestic Product (GDP)	1972	1.280000e+12
373	United States of America	Gross Domestic Product (GDP)	1977	2.090000e+12
374	United States of America	Gross Domestic Product (GDP)	1982	3.340000e+12
375	United States of America	Gross Domestic Product (GDP)	1987	4.870000e+12
376	United States of America	Gross Domestic Product (GDP)	1992	6.540000e+12
377	United States of America	Gross Domestic Product (GDP)	1997	8.610000e+12
378	United States of America	Gross Domestic Product (GDP)	2002	1.100000e+13
379	United States of America	Gross Domestic Product (GDP)	2007	1.450000e+13
380	United States of America	Gross Domestic Product (GDP)	2012	1.620000e+13
381	United States of America	Gross Domestic Product (GDP)	2015	1.790000e+13
382	United States of America	National Rainfall Index (NRI)	1965	9.285000e+02
383	United States of America	National Rainfall Index (NRI)	1969	9.522000e+02
384	United States of America	National Rainfall Index (NRI)	1974	1.008000e+03
385	United States of America	National Rainfall Index (NRI)	1981	9.492000e+02
386	United States of America	National Rainfall Index (NRI)	1984	9.746000e+02
387	United States of America	National Rainfall Index (NRI)	1992	1.020000e+03
388	United States of America	National Rainfall Index (NRI)	1996	1.005000e+03
389	United States of America	National Rainfall Index (NRI)	2002	9.387000e+02

390 rows × 4 columns

Extract specific statistics from the preprocessed data:

6) Create a dataframe 'dftemp' to store rows where Area is Iceland.

In [10]:

```
dftemp = df1[df1['Area'].isin(['Iceland'])]
dftemp
```

Out[10]:

	Area	Variable Name	Year	Value
166	Iceland	Total area of the country	1962	1.030000e+04
167	Iceland	Total area of the country	1967	1.030000e+04
168	Iceland	Total area of the country	1972	1.030000e+04
169	Iceland	Total area of the country	1977	1.030000e+04
170	Iceland	Total area of the country	1982	1.030000e+04
171	Iceland	Total area of the country	1987	1.030000e+04
172	Iceland	Total area of the country	1992	1.030000e+04
173	Iceland	Total area of the country	1997	1.030000e+04
174	Iceland	Total area of the country	2002	1.030000e+04
175	Iceland	Total area of the country	2007	1.030000e+04
176	Iceland	Total area of the country	2012	1.030000e+04
177	Iceland	Total area of the country	2014	1.030000e+04
178	Iceland	Total population	1962	1.826000e+02
179	Iceland	Total population	1967	1.974000e+02
180	Iceland	Total population	1972	2.099000e+02
181	Iceland	Total population	1977	2.221000e+02
182	Iceland	Total population	1982	2.331000e+02
183	Iceland	Total population	1987	2.469000e+02
184	Iceland	Total population	1992	2.599000e+02
185	Iceland	Total population	1997	2.728000e+02
186	Iceland	Total population	2002	2.869000e+02
187	Iceland	Total population	2007	3.054000e+02
188	Iceland	Total population	2012	3.234000e+02
189	Iceland	Total population	2015	3.294000e+02
190	Iceland	Population density	1962	1.773000e+00
191	Iceland	Population density	1967	1.917000e+00
192	Iceland	Population density	1972	2.038000e+00
193	Iceland	Population density	1977	2.156000e+00
194	Iceland	Population density	1982	2.263000e+00
195	Iceland	Population density	1987	2.397000e+00
196	Iceland	Population density	1992	2.523000e+00
197	Iceland	Population density	1997	2.649000e+00
198	Iceland	Population density	2002	2.785000e+00

	Area	Variable Name	Year	Value
199	Iceland	Population density	2007	2.965000e+00
200	Iceland	Population density	2012	3.140000e+00
201	Iceland	Population density	2015	3.198000e+00
202	Iceland	Gross Domestic Product (GDP)	1962	2.849165e+08
203	Iceland	Gross Domestic Product (GDP)	1967	6.212260e+08
204	Iceland	Gross Domestic Product (GDP)	1972	8.465069e+08
205	Iceland	Gross Domestic Product (GDP)	1977	2.226539e+09
206	Iceland	Gross Domestic Product (GDP)	1982	3.232804e+09
207	Iceland	Gross Domestic Product (GDP)	1987	5.565384e+09
208	Iceland	Gross Domestic Product (GDP)	1992	7.138788e+09
209	Iceland	Gross Domestic Product (GDP)	1997	7.596126e+09
210	Iceland	Gross Domestic Product (GDP)	2002	9.161798e+09
211	Iceland	Gross Domestic Product (GDP)	2007	2.129384e+10
212	Iceland	Gross Domestic Product (GDP)	2012	1.419452e+10
213	Iceland	Gross Domestic Product (GDP)	2015	1.659849e+10
214	Iceland	National Rainfall Index (NRI)	1967	8.160000e+02
215	Iceland	National Rainfall Index (NRI)	1971	9.632000e+02
216	Iceland	National Rainfall Index (NRI)	1975	1.010000e+03
217	Iceland	National Rainfall Index (NRI)	1981	9.326000e+02
218	Iceland	National Rainfall Index (NRI)	1986	9.685000e+02
219	Iceland	National Rainfall Index (NRI)	1991	1.095000e+03
220	Iceland	National Rainfall Index (NRI)	1997	9.932000e+02
221	Iceland	National Rainfall Index (NRI)	1998	9.234000e+02

7) Print the years when the National Rainfall Index (NRI) was greater than 950 or less than 900 in Iceland. Use the dataframe you created in the previous question 'dftemp'.

In [29]:

```
a = list((dftemp['Year'][dftemp['Variable Name']=='National Rainfall Index (NRI)'
a.extend(list((dftemp['Year'][dftemp['Variable Name']=='National Rainfall Index (
print(a)
```

```
[1971, 1975, 1986, 1991, 1997, 1967]
```

In [ ]:

US statistics:

8) Get all the rows of df1 (preprocessed dataframe) area is United States of America

1) Create a new DataFrame called `df_usa` that only contains values where 'Area' is equal to 'United States of America'. Set the indices to be the 'Year' column (Use .set_index( ))

2) Pivot the DataFrame so that the unique 'Variable Name' entries becomes the column entries. The DataFrame values should be the ones in the the 'Value' column. Do this by running the three lines of code below:

`df_usa=df_usa.pivot(columns='Variable Name',values='Value')`

3) Display df_usa.head( ), rename new columns to ['GDP','NRI','PD','Area','Population']

4) Find `df_usa.isnull().sum()`.This gives us the number of NAN values in each column. Replace NAN values by 0, using `df_usa=df_usa.fillna(0)`. Again check `df_usa.isnull().sum()`.'

5) Calculate and print all the column averages and the column standard deviations.

In [30]:
```
df_usa = df1.loc[df['Area'].isin(['United States of America'])]
df_usa = df_usa.set_index('Year')
```

In [31]:
```
df_usa=df_usa.pivot(columns='Variable Name',values='Value')
df_usa.head()
df_usa = df_usa.rename(columns = {'Gross Domestic Product (GDP)':'GDP','National
df_usa = df_usa.rename(columns = {'Population density':'PD','Total area of the co
```

In [32]:
```
#Find df_usa.isnull().sum().This gives us the number of NAN values in each column
df_usa.isnull().sum()
```

Out[32]:
```
Variable Name
GDP            7
NRI           11
PD             7
Area           7
Population     7
dtype: int64
```

In [33]:

```
#Replace NAN values by 0, using df_usa=df_usa.fillna(0).
df_usa=df_usa.fillna(0)

#Again check df_usa.isnull().sum().'
df_usa.isnull().sum()
```

Out[33]:

```
Variable Name
GDP           0
NRI           0
PD            0
Area          0
Population    0
dtype: int64
```

In [34]:

```
print(df_usa.mean())
print(df_usa.std())
```

```
Variable Name
GDP           4.620895e+12
NRI           4.092737e+02
PD            1.670158e+01
Area          6.103147e+05
Population    1.615134e+05
dtype: float64
Variable Name
GDP           6.088656e+12
NRI           4.935515e+02
PD            1.355462e+01
Area          4.789482e+05
Population    1.313805e+05
dtype: float64
```

9) Use df_usa:

1: Multiply the Area by 10 (so instead of 1000 ha, the unit becomes 100 ha = 1km^2)

2: Create a new column in df_us called 'GDP/capita' and populate it with the calculated GDP per capita. Round the results to two decimal points.

3: Create a new column called 'PD2' (i.e. Population density 2). Calculate the Population density. Note: the units should be inhab/km^2 (see Data description above). Round the reults to two decimal point.

4: Find the maximum value and minimum value of the 'NRI' column in the US (using pandas methods). What years do the min and max values occur?

In [35]:

```
#9.1
df_usa['Area'] = df_usa['Area']*10
df_usa
```

Out[35]:

Variable Name	GDP	NRI	PD	Area	Population
Year					
1962	6.050000e+11	0.0	19.93	9629090.0	191861.0
1965	0.000000e+00	928.5	0.00	0.0	0.0
1967	8.620000e+11	0.0	21.16	9629090.0	203713.0
1969	0.000000e+00	952.2	0.00	0.0	0.0
1972	1.280000e+12	0.0	22.14	9629090.0	213220.0
1974	0.000000e+00	1008.0	0.00	0.0	0.0
1977	2.090000e+12	0.0	23.17	9629090.0	223091.0
1981	0.000000e+00	949.2	0.00	0.0	0.0
1982	3.340000e+12	0.0	24.30	9629090.0	233954.0
1984	0.000000e+00	974.6	0.00	0.0	0.0
1987	4.870000e+12	0.0	25.49	9629090.0	245425.0
1992	6.540000e+12	1020.0	26.78	9629090.0	257908.0
1996	0.000000e+00	1005.0	0.00	0.0	0.0
1997	8.610000e+12	0.0	28.34	9629090.0	272883.0
2002	1.100000e+13	938.7	29.95	9632030.0	288471.0
2007	1.450000e+13	0.0	31.32	9632030.0	301656.0
2012	1.620000e+13	0.0	32.02	9831510.0	314799.0
2014	0.000000e+00	0.0	0.00	9831510.0	0.0
2015	1.790000e+13	0.0	32.73	0.0	321774.0

In [40]:

```
#Create a new column in df_us called 'GDP/capita' and populate it with the calcul
#Round the results to two decimal points
#9.2
df_usa['GDP/Capita']=(df_usa['GDP']/df_usa['Population']).round(2)
df_usa
```

Out[40]:

Variable Name	GDP	NRI	PD	Area	Population	GDP/Capita
Year						
1962	6.050000e+11	0.0	19.93	9629090.0	191861.0	3153324.54
1965	0.000000e+00	928.5	0.00	0.0	0.0	NaN
1967	8.620000e+11	0.0	21.16	9629090.0	203713.0	4231443.26
1969	0.000000e+00	952.2	0.00	0.0	0.0	NaN
1972	1.280000e+12	0.0	22.14	9629090.0	213220.0	6003189.19
1974	0.000000e+00	1008.0	0.00	0.0	0.0	NaN
1977	2.090000e+12	0.0	23.17	9629090.0	223091.0	9368374.34
1981	0.000000e+00	949.2	0.00	0.0	0.0	NaN
1982	3.340000e+12	0.0	24.30	9629090.0	233954.0	14276310.73
1984	0.000000e+00	974.6	0.00	0.0	0.0	NaN
1987	4.870000e+12	0.0	25.49	9629090.0	245425.0	19843129.27
1992	6.540000e+12	1020.0	26.78	9629090.0	257908.0	25357879.55
1996	0.000000e+00	1005.0	0.00	0.0	0.0	NaN
1997	8.610000e+12	0.0	28.34	9629090.0	272883.0	31551983.82
2002	1.100000e+13	938.7	29.95	9632030.0	288471.0	38132082.60
2007	1.450000e+13	0.0	31.32	9632030.0	301656.0	48067997.98
2012	1.620000e+13	0.0	32.02	9831510.0	314799.0	51461408.71
2014	0.000000e+00	0.0	0.00	9831510.0	0.0	NaN
2015	1.790000e+13	0.0	32.73	0.0	321774.0	55629106.14

In [42]:

```
#9.3
df_usa['PD2'] = (((df_usa.loc[:,'Population'])*1000)/df_usa.loc[:,'Area']).round(
df_usa
```

Out[42]:

Variable Name	GDP	NRI	PD	Area	Population	GDP/Capita	PD2
Year							
1962	6.050000e+11	0.0	19.93	9629090.0	191861.0	3153324.54	19.930000
1965	0.000000e+00	928.5	0.00	0.0	0.0	NaN	NaN
1967	8.620000e+11	0.0	21.16	9629090.0	203713.0	4231443.26	21.160000
1969	0.000000e+00	952.2	0.00	0.0	0.0	NaN	NaN
1972	1.280000e+12	0.0	22.14	9629090.0	213220.0	6003189.19	22.140000
1974	0.000000e+00	1008.0	0.00	0.0	0.0	NaN	NaN
1977	2.090000e+12	0.0	23.17	9629090.0	223091.0	9368374.34	23.170000
1981	0.000000e+00	949.2	0.00	0.0	0.0	NaN	NaN
1982	3.340000e+12	0.0	24.30	9629090.0	233954.0	14276310.73	24.300000
1984	0.000000e+00	974.6	0.00	0.0	0.0	NaN	NaN
1987	4.870000e+12	0.0	25.49	9629090.0	245425.0	19843129.27	25.490000
1992	6.540000e+12	1020.0	26.78	9629090.0	257908.0	25357879.55	26.780000
1996	0.000000e+00	1005.0	0.00	0.0	0.0	NaN	NaN
1997	8.610000e+12	0.0	28.34	9629090.0	272883.0	31551983.82	28.340000
2002	1.100000e+13	938.7	29.95	9632030.0	288471.0	38132082.60	29.950000
2007	1.450000e+13	0.0	31.32	9632030.0	301656.0	48067997.98	31.320000
2012	1.620000e+13	0.0	32.02	9831510.0	314799.0	51461408.71	32.020000
2014	0.000000e+00	0.0	0.00	9831510.0	0.0	NaN	0.000000
2015	1.790000e+13	0.0	32.73	0.0	321774.0	55629106.14	inf

In [39]:

```
#9.4
max=df_usa['NRI'].max()
min=df_usa['NRI'].min()

#print(df_usa[df_usa['NRI'].isin([max])])
#print(df_usa[df_usa['NRI'].isin([min])])

print(df_usa.index[df_usa['NRI'].isin([max])].tolist())
print(df_usa.index[df_usa['NRI'].isin([min])].tolist())
```

```
[1992]
[1962, 1967, 1972, 1977, 1982, 1987, 1997, 2007, 2012, 2014, 2015]
```

Now, lets read another CSV file.

See https://www.quantshare.com/sa-43-10-ways-to-download-historical-stock-quotes-data-for-free (https://www.quantshare.com/sa-43-10-ways-to-download-historical-stock-quotes-data-for-free)

10 a) Show a 3 x 3 correlation matrix for Nike, Apple, and Disney stock prices for the month of July, 2017

In [57]:

```
dfg = pd.read_csv('https://www.google.com/finance/historical?output=csv&q=goog')
dfa = pd.read_csv('https://www.google.com/finance/historical?output=csv&q=aapl')

dfd= pd.read_csv('https://www.google.com/finance/historical?output=csv&q=dis')
dfn= pd.read_csv('https://www.google.com/finance/historical?output=csv&q=nke')

dfa.head()
# HINT: Convert 'Date' to datetime format in the datfarmes.
# Change indices of alldataframes to Date. Use Date indices to filter rows
# Create a new dataframe that stores values of 'Close' column from each dataframe
# Use the 'Close' Column of each companys stock data to find correlation using df


#df1['Year']=pd.to_datetime(df1['Year'].astype(int),format='%Y').dt.year
```

Out[57]:

	Date	Open	High	Low	Close	Volume
0	13-Sep-17	159.87	159.96	157.91	159.65	44813571
1	12-Sep-17	162.61	163.96	158.77	160.86	71714046
2	11-Sep-17	160.50	162.05	159.89	161.50	31580798
3	8-Sep-17	160.86	161.15	158.53	158.63	28611535
4	7-Sep-17	162.09	162.24	160.36	161.26	21928502

In [81]:

```
#print(pd.to_datetime(dfa['Date'].astype(int),format='%d-%m-%Y'))
dfa['Date']=pd.to_datetime(dfa['Date'],dayfirst=True,format=None)
dfg['Date']=pd.to_datetime(dfg['Date'],dayfirst=True,format=None)
dfd['Date']=pd.to_datetime(dfd['Date'],dayfirst=True,format=None)
dfn['Date']=pd.to_datetime(dfn['Date'],dayfirst=True,format=None)
```

In [84]:

```
dfa = dfa.set_index('Date')
dfg = dfg.set_index('Date')
dfd = dfd.set_index('Date')
dfn = dfn.set_index('Date')
```

In [126]:

```

df_close = pd.DataFrame({'Apple' : dfa['Close'],
                         'Google': dfg['Close'],
                         'Disney': dfd['Close'],
                         'Nike':dfn['Close']})
df_close

df_close.corr()
```

Out[126]:

	Apple	Disney	Google	Nike
Apple	1.000000	0.494258	0.898688	0.553415
Disney	0.494258	1.000000	0.350581	0.482196
Google	0.898688	0.350581	1.000000	0.415401
Nike	0.553415	0.482196	0.415401	1.000000

10b) Show the same correlation matrix but over different time periods,

i) the last 20 days ii) the last 80 days

In [141]:

```
print(df_close.head(20).corr())

print(df_close.head(80).corr())
```

```
           Apple    Disney    Google      Nike
Apple   1.000000  0.176193  0.690522 -0.375342
Disney  0.176193  1.000000 -0.356421  0.303717
Google  0.690522 -0.356421  1.000000 -0.384907
Nike   -0.375342  0.303717 -0.384907  1.000000
           Apple    Disney    Google      Nike
Apple   1.000000 -0.497856 -0.179233 -0.074875
Disney -0.497856  1.000000  0.409944  0.244481
Google -0.179233  0.409944  1.000000 -0.285185
Nike   -0.074875  0.244481 -0.285185  1.000000
```

11) Change the code so that it accepts a list of any stock symbols, ie ['NKE', 'APPL', 'DIS', ...] and creates a correlation matrix for the time period of the past 100 days

In [182]:

```
choices=[]
while True:
    x=input("Select stocks one at a time, type done to end: ")
    if x != 'done':
        choices.append(x)
    else:
        break
df_close[choices].head(100).corr()
```

```
Select stocks one at a time, type done to end: Google
Select stocks one at a time, type done to end: Apple
Select stocks one at a time, type done to end: Nike
Select stocks one at a time, type done to end: done
```

Out[182]:

	Google	Apple	Nike
Google	1.000000	0.074280	-0.197882
Apple	0.074280	1.000000	-0.064593
Nike	-0.197882	-0.064593	1.000000

In [ ]: