Lab 2

The goal of this lab is to provide exercises that will help students reinforce and improve knowledge of Pandas library.

Note: For each exercise enclose your source code and provide a short description of your solution, e.g. in case you have used an existing Pandas function or written your own, explain briefly why/how the function helped you solve the exercise.

Exercise 1

In this exercise you will be working with the following data

```
exam_data = {'name': ['Anita', 'Adaleta', 'Lena', 'Ulrika', 'Mikael',
'Samuel', 'Simanthi', 'Kristian', 'Jusuf', 'Jonas'],
'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no',
'yes']}

labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']
```

a) Write a Pandas program to create and display a DataFrame from a specified dictionary data, which has the index labels.

The expected output:

```
       name  score  attempts qualify
a     Anita   12.5         1     yes
b      Lena   14.0         3      no
c   Adaleta   16.5         2     yes
d    Ulrika    NaN         3      no
e    Mikael    9.0         2      no
f    Samuel   20.0         3     yes
g  Simanthi   14.5         1     yes
h  Kristian    NaN         1      no
i     Jusuf    8.0         2      no
j     Jonas   19.0         1     yes
```

b) Write a Pandas program to get the first 3 rows of the dataframe created under part (a) of the exercise.

The expected output:

```
First three rows of the data frame:
      name  score  attempts qualify
a    Anita   12.5         1     yes
b     Lena   14.0         3      no
c  Adaleta   16.5         2     yes
```

c) Write a Pandas program to select the 'name' and 'score' columns from the dataframe created under part (a).

The expected output:

```
        name  score
a      Anita   12.5
b       Lena   14.0
c    Adaleta   16.5
d     Ulrika    NaN
e     Mikael    9.0
f     Samuel   20.0
g   Simanthi   14.5
h   Kristian    NaN
i      Jusuf    8.0
j      Jonas   19.0
```

d) Write a Pandas program to select the rows from the dataframe created under (a), where the number of attempts in the examination is less than 3 and score greater than 14.

```
        name  score  attempts qualify
c    Adaleta   16.5         2     yes
g   Simanthi   14.5         1     yes
j      Jonas   19.0         1     yes
```

e) Wrte a Pandas program to replace all NaN values with zeros in the dataframe created under (a). Print the dataframe before and after removing the null values.

Exercise 2

In this exercise we will redo the last exercise from the lab 1, but this time instead of reading the database.csv file into NumPy structure array, you will load data into Pandas dataframe object. Complete tasks a-d. Which data structure is more convenient, NumPy structured array or Pandas dataframe object to complete the tasks a-d below. Please motivate your answer.

a) Read in comma separated file database.csv using Pandas read_csv function and print out the dataframe header. What are the column headers? What is the shape of your dataframe? How many lines in the file are there (rows), how many columns?
b) Select from the dataframe created in (a) the following columns: Date, Depth, and Magnitude. Ensure that all the elements within the dataframe are properly formatted.
c) Write code to calculate minimum, maximum and mean Magnitude values. Print out the values.
d) Write a program that will calculate the total number of earthquakes for each year. Use dataframe bar plot function to plot the data.

Exercise 3

In this exercise we will examine a file that has information on individuals (contributors) who changed one or more source code files on a project. Thus, each line in the input file contains information on a changed file's unique revision number (id), a contributor's name, date of change, and the number of lines changed.

a) Read in the attached newfile.txt file using pandas read_csv file function. You should read in the above mentioned 4 columns. Note, you will need to slice the date to be of the format yyyy-mm-dd (we do not need the times).
b) Group all the contributions that belong to the same contributor (name) together, count the number of contributions for each contributor and sort the number of contributions from the highest to the lowest number.
c) Display the result from part (b) as a bar chart where x axis displays contributors names and the y axis displays the corresponding number of contributions.

Exercise 4

In this exercise you will be working with the earthquake data file. To start, we will need the loaded dataframe object, which you have already completed in the exercise 3, part (a) of this lab.

The goal of the exercise is to produce a graph as the one shown in figure 1.0 below:

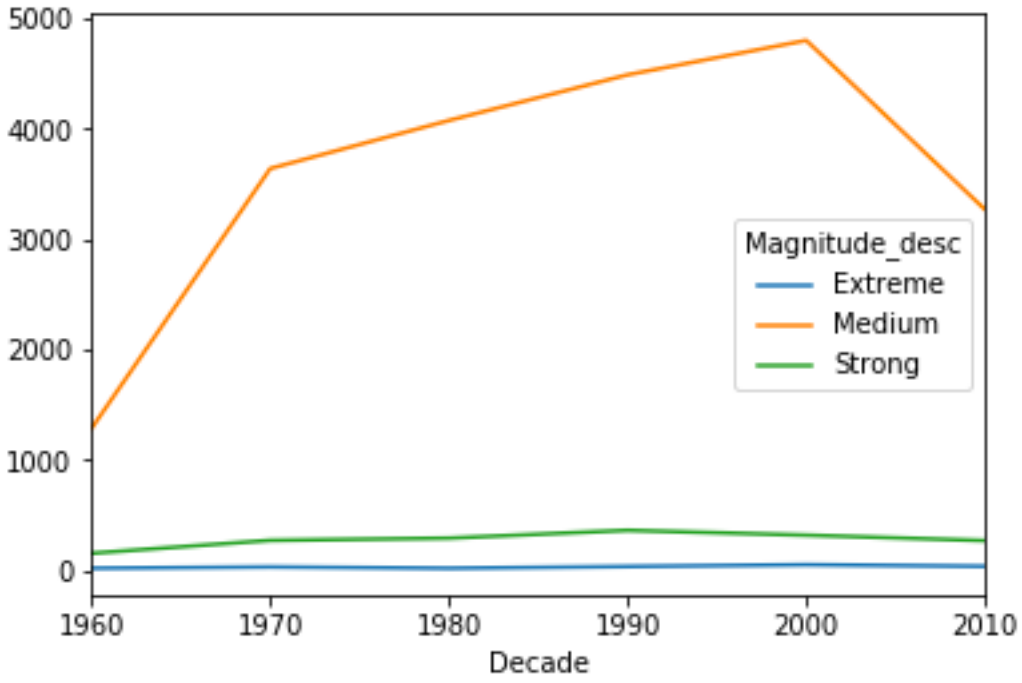

Figure 1.0

Note the graph in Figure 1.0 lists for each decade corresponding number of earthquakes categorized by the strength of their magnitude as Extreme, Medium, or Strong

Here are some tips on what needs to be done in order to complete the task:

a) Create a function that will add a new column ('Magnitude_desc') to your dataframe. The function ranks/describes earthquakes of magnitude less or equal to 6.5 as 'Medium' , those of magnitude greater than 6.5 and less or equal to 7.5 as 'Strong' and greater than 7.5 as 'Extreme'

b) Create a function that will convert year into decade and add a new column 'Decade' to your dataframe store this data.
c) The code provided here will create the graph from the Figure 1.0, but using groupby functionality. Your task is to produce the same graph, but using the pivot_table functionality.

```
eq_df.groupby(['Decade',
'Magnitude_desc'])['Magnitude_desc'].aggregate('count').unstac
k() eq_df.groupby(['Decade',
'Magnitude_desc'])['Magnitude_desc'].aggregate('count').unstac
k().plot() plt.ylabel('total earthquackes per year');
```