Activities: Pandas 1

1. Download the file 'Salaries.csv' that contains salary data from municipal employees in San Francisco, California (one of the most expensive cities in the world). This file is so big that it might not fit into your favorite csv-reader such as excel or numbers. You can find the file on Kaggle.

2. **Create a Pandas Dataframe:** Pandas has a function pandas.read_csv that will read a .csv file and make assumptions on the data types read. It will use a default index and take the column names from the .csv file. The file contains real data and as is typical for real data, it has missing values. We need to use "converters" in order to make columns that contain numerical data fill in the missing values with Numpy NaN values (numpy.NaN).
   1. Write a function `my_converter(x)` that tries to convert `x` into a float value and returns the value and if the conversion fails, returns `numpy.NaN`.
   2. Determine the names of the columns that should have numerical data in them by looking at the beginning of the file by writing a python file `get_head()` that writes out the first 10 lines of the file.
   3. Write a function that returns a Pandas dataframe by using the `pandas.read_csv` function. You need to use the converter-parameter, that contains a dictionary of values, as in

```
pd.read_csv('Salaries.csv',
            converters = {'BasePay': my_converter}
           )
```

3. Use the Pandas dataframe in order to find the mean base pay and the mean total pay benefits in the data set.

4. Find all records where the base pay exceeds 300,000 dollars. Print out the Name, Year, Pay, and Job-title.

5. Find the means of all numerical values. (The Pandas dataframe method .mean( ) sets the skipna parameter to True, so that NaN values are skipped.)

6. We now want to find out who is not getting paid a lot.
   1. Find the smallest total pay benefits in the dataframe.
   2. Since we want to find the record, we would like to use argmin, but argmin is being deprecated, so we will use `idxmin( )` instead.
   3. Use the `.iloc` method in order to find the row of the data frame with this minimum total pay benefits.
   4. Print out the name and job title of this employee.
   5. Repeat the same with base pay.

7. Find the employees that have Captain in the job title. Notice that some titles are all in capital letters, some mixed, and some lower case only. This is a case for using the apply method on a Pandas series.