PSG COLLEGE OF TECHNOLOGY
Department of Applied Mathematics and Computational Sciences
IX MSc Theoretical Computer Science- DM lab
Problem Sheet 2

You can't make visualizations without data, and data coming from the real world is notoriously messy. In this assignment you're going to practice the basics of accessing data from various sources, data parsing/cleaning and manipulation, and doing some preliminary analysis on the data you collect.

1. Working with Files

During the first portion of this lab, you'll download the UCI Zoo Dataset and write code to

convert the raw data into a more usable and human/machine-readable CSV format. The raw UCI

Zoo Dataset looks like this:

aardvark,1,0,0,1,0,0,1,1,1,1,0,0,4,0,0,1,1
antelope,1,0,0,1,0,0,0,1,1,1,0,0,4,1,0,1,1
bass,0,0,1,0,0,1,1,1,1,0,0,1,0,1,0,0,4

This is certainly machine-readable, but it's difficult for a human to tell what's going on. Of course there are advantages to storing data in compact machine-readable formats when the data is large, but this data is not large at all. The goal is to write code (likely Python, but not necessarily) to convert it to a readable CSV. Note that there are no column headers in the data above, so it's a bit unclear what all those 1s and 0s are telling you without cross-referencing another file. Specifically, you'd like the output to look like:

animalname,hair,feathers,eggs,milk,airborne,aquatic,predator,toothed,...
aardvark,true,false,false,true,false,false,true,true,...

To accomplish this, you will need the zoo.data and zoo.names files. Notice that the zoo.names file is relatively unstructured. The information you want is continued in the Attribute Information in item number 7 of this file.
After the data is loaded, loop through the rows and convert each data value using the data types you stored previously: Boolean becomes true or false, Numeric stays numeric, etc. Afterwards, print the attribute names array in a comma-separated format, followed by the new dataset (also comma-separated) to a CSV file.

You can write your conversion program in any language you like: Python, Processing, Node.js, R,Java, C, perl, etc.

2. Data from the Web

For this assignment, write several small python programs to scrape simple HTML data from several websites. Use Python with the following libraries:

- Beautiful Soup 4 (makes it easier to pull data out of HTML and XML documents)
- Requests (for handling HTTP requests from python)
- lxml (XML and HTML parser)

For example if you are collecting data from Wikipedia "List of..." pages. In the following example, Wikipedia List of Nobel Laureates is used.

Use the above tools to scrape the winners, the year they won, and the URL of their individual Wiki page. Combine this into a single data table as shown below.

	winner_name	subject	year	url
0	Wilhelm Röntgen	Physics	1901	/wiki/Wilhelm_R%C3%B6ntgen
1	Jacobus Henricus van 't Hoff	Chemistry	1901	/wiki/Jacobus_Henricus_van_%27t_Hoff
2	Emil Adolf von Behring	Physiology or Medicine	1901	/wiki/Emil_Adolf_von_Behring
3	Sully Prudhomme	Literature	1901	/wiki/Sully_Prudhomme
4	Henry Dunant	Peace	1901	/wiki/Henry_Dunant

3. Analysis

After you got data, see if you can identify any interesting trends in either one. For example, are there any interesting historic trends in the winners of Nobel Prizes (e.g. during WWI and WWII)? Are there any relationships between attributes in the Zoo dataset (e.g. hair vs. feathers and milk vs. eggs)? Explore! You may use any analytical tools you like (pandas, R, matlab, ...).