Problem Set 2

The second problem set focuses on data scraping and data wrangling. Your solution should be composed of a well-structured R script which should provide the designated functions. Besides the functions the code should be directly runnable or at least sufficiently well documented (working directory, path settings) to be executed.

This problem set is **due on June 17th by 18.00** through the Wuecampus upload functionality.

1. To get started with web scraping we leverage the website http://quotes.toscrape.com/ . This page offers a collection of inspiring quotes and is particular designed to train your web scraping skills.
    a. Use rvest to scrape the first 10 quotes. Return a data frame with 3 columns (author, tags,quotes).
    b. Use your function to collect all 100 quotes from the website and store them in a data frame (**quotes**).
    c. Additionally, we want to collect more information on the authors. Therefore, your next task is to scrape the URLs of each author's about-page.
    d. Write a function to scrape the content of an about-page returning a data frame (**authorDetails**) with 3 columns (author, description, bornDate). Apply the function to scrape all about-pages.
    e. Next, your task is to analyze the collected data. Leverage your data wrangling skills to perform the following tasks:
        i. The authorDetails data frame stores the information on the birth data in one column (bornDate). Transform the data frame to store the information on the day, month and year in distinct columns. How many authors where born in the 19th century (1800-1899)?
        ii. Transform and summarize the quotes data set to answer the following questions:
            1. Which author has the most quotes on the website?
            2. How many quotes does an author have on average?
            3. Find all quotes that use the tag "life"
        iii. Join both data frames (you may need to transform keys first)

2. Your second task is to analyze the deals posted on https://www.mydealz.de/ . This page offers a platform for users to share and vote on offers from online as well as brick and mortar stores.
    a. First, use rvest to scrape at least the 1000 latest deals. In order to minimize server traffic, you should scrape all the relevant information from the starting page and do not need to follow the deep links to the particular deals. Your function should return a data frame (**deals**) with 5 columns (title, temperature, author, deep link, number of comments).
    b. Having collected the data set you will probably notice that there are some arcane symbols in the title and temperature columns. Clean both columns by changing the encoding from 'latin1' to 'ASCII' using the iconv function. Additionally, remove unneeded whitespaces, line breaks and tabulators from the data frame.
    c. Use your data wrangling skills to answer the following questions:
        i. What share of the posted deals is voted hot (over 0 degrees)? What share is voted cold (below 0 degrees)? What is the average temperature of a deal?
        ii. On average, do hot or cold deals have more comments?
        iii. Which author posted the most deals? How many deals did an author post on average?
        iv. Based on the title, what share of deals is about Xiaomi products?