Position: **Data Engineer Internship**

In all solutions provide the program's code and optionally the output.
All questions are quite open ended, interpret them as you see fit.
You can use any libraries you want.

Problem 1

With the attached "problem1.json" file, using python or JavaScript, replace all the empty string values with null/None. You don't have to save the output file anywhere. Optional: Write the program in both languages.

Problem 2

With the attached "problem2.ftr" pandas.DataFrame feather file, you are given a semi-parsed array of dates. The aim of this problem is to replace the dates not in the dd-mm-yyyy format as follows; Assume the array is chronologically sorted(strictly speaking, it is not). For the dates in the "n (maanden/weken)" or "Vandaag" or "6+ maaden" ago format, $n \in \mathbb{N}$, subtract the n (months/weeks) or Today or 6+ months ago respectively from the last previous date in the correct format (dd-mm-yyyy). Do this for the whole array. Finally, convert the array to either python's basic datetime format (datetime.datetime) or to a pandas timestamp (pandas.Timestamp).
Note; Dates parsed this way should not be considered being in the correct format when you convert the rest of the array.

For example

22-08-2019
7 weken
8 weken

becomes

22-08-2019
04-07-2019
27-06-2019

Problem 3

With the attached "problem3.sql" script, delete duplicate rows based on the "url" column (leaving only the row with the lowest ID), and update its column "count" to have the value of its highest duplicate row ID.

ID	First Name	Count	Url
1	A	10	www.A.com
2	B	21	www.B.com
3	C	12	www.C.com
4	D	31	www.D.com
5	A	13	www.A.com
6	D	18	www.D.com
7	A	5	www.A.com

Expected results;

ID	First Name	Count	Url
1	A	5	www.A.com
2	B	21	www.B.com
3	C	12	www.C.com
4	D	18	www.D.com

Problem 4

Scrape all the links to rooms listed in this url: https://kamernet.nl/huren/kamers-nederland. Then, scrape each individual room, gathering as much data as you can. You can optionally create an account and programmatically log-in with it to scrape the contact section too. Optional; Use scrapy.

Problem 5 (Optional)

Scrape all the links to rooms listed in this url: https://www.funda.nl/koop/heel-nederland/. Then, scrape each individual room, gathering as much data as you can. You can optionally create an account and programmatically login with it to scrape the contact section too. Optional; Use scrapy.
Note; This site has deployed multiple anti-scraping measures, this is a hard task.

Problem 6

Describe the design of a scraping pipeline for any of the previous 2 problems, uploading the results continuously in a database. Describe the schema, triggers and required routines of the database.
Note; This is an open-ended question.