Assignment: Data Engineer

- *Please complete the following questions and upload the solutions (python code or Jupyter notebooks) into your github account. You may use external libraries whenever necessary.*
- *Please share the github link to the solutions on completion. The duration for completion of assignments is 1 week.*

Q1. Write a function in python to sum up a given set of numbers other than itself
Input: An array of *n* integers `nums`,
Output: An array `output` such that `output[i]` is equal to the sum of all the elements of `nums` except `nums[i]`.
For example, given `[1,2,3,4]`, return `[9,8,7,6]`.

Q2. Sales Data Exploration and Analysis (code in python)

a) Write code to download the following Kaggle dataset:
Weekly Sales Transaction Data: https://www.kaggle.com/crawford/weekly-sales-transactions
b) Identify the best performing product (based on volume)
c) Identify the most promising product (emerging product)
d) Identify the worst performing product on a biweekly basis
e) Identify outliers from the data and output the corresponding week numbers

Q3. Jobposts Data Exploration and Analysis (code in python)

a) Reuse code from Q2 to download the following Kaggle dataset:
Jobposts Data: https://www.kaggle.com/madhab/jobposts/
b) Extract the following fields from the *jobpost* column:
    1. Job Title
    2. Position Duration
    3. Position Location
    4. Job Description
    5. Job Responsibilities
    6. Required Qualifications
    7. Remuneration
    8. Application Deadline
    9. About Company
c) Identify the company with the most number of job ads in the past 2 years
d) Identify the month with the largest number of job ads over the years
e) Find median, mean, min and max values for each product
f) Clean text and generate new text from Job Responsibilities column: The new text shall not contain any stop words, and the plural words shall be converted into singular words.
g) Store the results in a new Dataframe/SQL table

Q4. String similarity (code in python):

a) Download test.csv from https://www.kaggle.com/rishisankineni/text-similarity/data
b) Load the data to a Spark/Pandas data frame
c) Calculate similarity between *description_x* and *description_y* and store resultant scores in a new column
d) Parallelise the matching process (bonus)

End