Data Engineer Coding Challenge

Assessment Criteria

Technical

- Assess candidate's ability to write complex SQL statements
- Assess candidate's ability to build a ETL workflow in Python

Documentation

- Assess whether code was appropriately documented in the form appropriate to implementation language

Testing / Sanity Checks

- Assess whether candidate included tests to explain and reinforce design of code

Coding Challenge

Question 1

This is a snapshot of a pageview activity log table

ID	User_ID	Page_ID	Visit_Date	Visit_Time
1	1	54	2018-01-01	11:54:34
2	1	55	2018-01-01	11:55:10
3	1	56	2018-01-02	13:11:12
4	1	55	2018-01-02	17:10:08
5	1	56	2018-01-02	17:12:45
6	2	55	2018-01-01	10:25:18
7	2	55	2018-01-01	17:30:12
8	2	55	2018-01-01	17:45:57
9	3	54	2018-01-02	00:00:12

10	3	56	2018-01-02	00:03:22
11	3	55	2018-01-02	01:20:11
12	3	56	2018-01-02	01:40:09

Write an SQL statement to find the total number of user sessions each page has each day.
(A user session is defined as continuous activity on a site where each activity is within 10 mins of each other.)

Link to the pageviews table csv file:
https://s3-ap-southeast-1.amazonaws.com/ms-data-coding-challenge/SamplePageviews.csv

Expected result:

Page_ID	Visit_Date	Total_User_Sessions
54	2018-01-01	1
54	2018-01-02	1
55	2018-01-01	4
55	2018-01-02	2
56	2018-01-02	4

Question 2

This is a link to a sample order dataset:
https://s3-ap-southeast-1.amazonaws.com/ms-data-coding-challenge/SampleOrders.csv

Using the dataset above, we want you to do a basket analysis in SQL to find out what are the items that are frequently purchased together with our bestseller products (products that have had the highest number of orders)

You will need to:

- **Create an SQL query will show a list of products frequently purchased with the top 10 bestsellers.** *The top 10 bestsellers should appear in the 'ProductA' column of your resultset and the products frequently purchased with it should appear on the 'ProductB' column.*
- **Your query should only consider product pairs that meet the following conditions:**
    - Support >= 0.2
    - Confidence >= 0.6

    - Lift ratio > 1
- **The result set should return the following fields:**
    - *ProductA*
    - *ProductB*
    - *Occurrences*
    - *Support*
    - *Confidence*
    - *LiftRatio*
- **Please ensure that your resultset contains no duplicate pairs**

Tip: *Google "Market Basket Analysis" for resources to help explain what support, confidence and lift ratio means*

Question 3

The link below directs you to a log file that is generated by our A/B test platform which helps us to randomly assign visitors to experiment groups when they visit pages that have an A/B test experiment running on them. The log file contains all the messages generated by the platform.

Every time a visitor is assigned to an experiment group, the following log entry gets created

```
{
      "Level":"info",
      "msg":"Request Number is : 79, hence assigned to control for MS-002",
      "time":"2018-03-16T04:09:04Z"
}
```

This log message tells us that a visitor was assigned to the "Control" group in experiment "MS-002"

Link to log file: **https://s3-ap-southeast-1.amazonaws.com/ms-data-coding-challenge/lighthouse-logs.log**

What we want to know from the data in the logs are:

A. What are the total number users assigned to the "Test" and "Control" groups in each experiment?
B. Which day had the highest number of user group assignments per experiment?

You'll need to:

- **Write an SQL script to create a data table to store the visitor assignment data**
- **Write a python script to extract all the visitor assignment log messages from the log file and store it in the data table you created**
- **Write the SQL queries that will help answer the questions A and B above.**

Submission Instructions

1) Create a public repository in any popular version control service (Github / Bitbucket etc)
2) Upload your answer scripts and packages to the public repo
3) Create a readme file that describes which scripts contain the answers to each question in the coding challenge
4) Share the link to the repo with us