2.4 due.

DSCI 551 – Spring 2022

Homework #1: Firebase, JSON, and Data Modeling

Deadline: February 4, Friday (100 points)

Consider managing customer churn data in Firebase realtime database. The data are stored in a CSV file, with 7044 rows and 21 columns. You can find the details about the data set at Kaggle web site: https://www.kaggle.com/blastchar/telco-customer-churn. You can also download the data set from the web site (archive.zip containing WA_Fn-UseC_-Telco-Customer-Churn.csv).

To reduce the amount of data to be handled, this homework will only consider customers who are senior citizens (1142 of them).

① pd.read_csv
② seniorcitizen =1
③ 上传到database

① 转换成Json文件：df.to_json
② url ="filebase database url"
response=request.put(url, df)

Tasks:

relative path

1. [40 points] Write a python script "load.py" which load the rows for the above senior customers to your database.
   Execution format:

   python3 load.py → 在terminal中运行

   You can assume the WA_Fn-UseC_-Telco-Customer-Churn.csv file is stored at the same directory where you execute your script.

1.5 rules修改：给churned添加index

2. [30 points] Write a Python script "churn.py" to find the first k (senior) customers who has churned. Only need to return IDs of first k customers (ordered by their IDs).
   Execution format:

   python3 churn.py <k>

   For example:

   python3 churn.py 10

   will return IDs of first 10 customers who have churned.

① k = sys.argv[1]
② df_churned = request.get("url?orderBy=churn&EqualTo="Yes"
&limitToFirst =
df_churned.json()

3. [30 points] Write a Python script "tenure.py" to find out how many customers who have used the service for at least k months.
   Execution format:

   python3 tenure.py <k>

   For example,

   python3 tenure.py 10

orderBy="tenure"&startAt=k

Requirements:

- For each query in both patterns, only one round trip (send request and receive response) is permitted to the Firebase server.
- You should not download entire database to answer the query. √
- You should create indexes in Firebase console that allow the above programs to execute without errors. √

Permitted libraries: pandas, requests, json, and other common Python libraries (e.g., sys). Do not use firebase-admin, firebase python libraries.

Submissions: 3个py scripts + explaining document + JSON dump + screenshot

- Above 3 scripts.
- Prepend your full name to the script name, e.g., John_Smith_load.py, so on.
- A document (word/pdf) explaining why your program sends only one request to Firebase for each query. ?
- A JSON dump of your Firebase database for this app.
- A screenshot of your Firebase, showing the structure of your database.
- Submit online. See syllabus for late penalty!

Checklist for Submission :

1. Name your folder and zip LASTNAME_FIRSTNAME_HWX. Your submission should **be a zip (not rar)** file AND **unzipping it would have all your files (no folders)**. **Do not include csv files**. Notice that your submission should have **capital first and last names.** Example: TANEJA_DAKSH_HW1.
2. **DO NOT return anything we didn't ask for**. For example, "please enter XXX: ___". Please no. We have given you the EXACT output format. Please just follow them.
3. Use **ONLY relative path.** You should assume your scripts will be run in the directory where the scripts are at. For example, no 'C:\\homework1\...' or '/Users/blabla/....'
4. Make sure that you are able to run the code according to the execution format mentioned above in the questions.
5. Double-check your files before submitting them. Please use python3 to complete the homework and try to maintain the python version as 3.7. Do not use any libraries other than the ones specified in the handout. You can use EC2 to test your code, and python 3.7 is preinstalled on EC2.
6. You can submit it multiple times on DEN but only the latest attempt will be graded.