# Homework #1: Firebase and JSON

## Due: February14, Friday (11:59pm)

## 100 points

Consider the “world” database that you are using for your project. The data of 3 tables in the database are provided to you in CSV files: country.csv, city.csv, and countrylanguage.csv. In this homework, we ask to implement a simplified version (since data are provided to you here) of import.py as stated in the project description, and a command line search interface (instead of GUI in the project). Here are details. The homework should be completed using python 3.

1. [60 points] Write a Python program “load.py”.

Execution format: python load.py country.csv city.csv countrylanguage.csv

The program will do two things:

* + Convert the data in the provided 3 CSV files into JSON and load them into Firebase (e.g., under the nodes: /country, /city, and /countrylanguage). You can use Python “requests” package in your program.
  + Create an inverted index in Firebase (e.g., in the /index node) that lists words and their occurrences in the database. For example, the word “america” appears in the value of the “Continent” attribute of the first row (whose code is “ABW”) of the “country” table.

You can assume that words in the attribute values are delimited by white spaces. You only need to consider textual attributes (whose data types are char or varchar).

1. [40 points] Write a Python script “search.py” that takes a list of keywords and returns primary key values of tuples which contain at least one keyword in the values of its attributes. For example, searching for “America” will return:

{Country: [“ABW”, “AIA”, …]

City:[ … (if the keyword also appears in some city tuples)]}

Note that search is NOT case-sensitive. Primary keys for each table are listed below.

|  |  |  |  |
| --- | --- | --- | --- |
|  | city | country | countrylanguage |
| primary key | **ID** | **Code** | **Language** |

Also note that the tuples that contain more search keywords should be listed first. For example, tuples that contain “South America” should be returned after tuples containing “North America”, if the search query is “north america”. The return will be a json object in following format:

{

‘america’ : {

‘ID’ : [SORTED\_VALUES],

‘Code’ : [SORTED\_VALUES],

‘Language’ : [SORTED\_VALUES]

},

‘kabul’: {...},

...}

The keywords may appear in multiple attributes of a tuple and a keyword will have no more than two words in grading your program.

The search program should utilize the index built above. You should assume the data and your code locate in the same folder.

Execution format: python search.py <keywords>

**Submission**: Submit a compressed file [FirstName]\_[LastName]\_HW1.zip, which only contains load.py and search.py file.