Reem Almijmaj

ID: 3217747723

DSCI 551

HW1

Why only one round trip (send request and receive response) is permitted to the Firebase server?

1) **requests.put('https://dsci551-a7a0e-default-rtdb.firebaseio.com/.json', js)**

requests: it is one of the HTTP libraries that allow sending requests to (and retrieved data from) specified URLs using python.

put: It is one of the requests methods that add the data which is in json format to the specified URL. If the URI refers to an already existing resource, it modifies that data, but if the URI does not point to an existing resource, then the firebase server creates the resource with that URI

url: the URL I used is a firebase real time database link and I added .json to make it possible to add my json (file format) data into the firebase database.

js: my variable name that has the data in json file format, I pass this parameter to the firebase database.

I send **one** request to the firebase that has **all** the data filtering **commands** to get my response (data) faster so I can improve Firebase Realtime Database performance in my program and enhance my program reliability. Requesting data from the firebase takes time, so it is better to mention all the data filter parameter in one line of code. In addition, because there is & sign that allow me to add as many filtering commands as I want, so I took advantage of this and use it in my string URL to filter the data the way I want, so I only have to communicate with the firebase once as firebase has limitation for how many requests a script can have to access the firebase database data.

2) **requests.get('https://dsci551-a7a0e-default-rtdb.firebaseio.com/.json?&orderBy="Churn"&equalTo="Yes"&print=pretty')**

requests: it is one of the HTTP libraries that allow sending requests to (and retrieved data from) specified URLs using python.

get: it is one of the requests methods that sends request to the specified URL to get and fetch the data.

url: the URL I used is a firebase real time database link and I added .json to make it possible to add my json (file format) data into the firebase database. I also added ? to allow me to add query parameters. And & is used to link every query with rest of the URL string.

orderBy: it is one of the filtering data query parameters that has value Churn which is a way to filter the data by child key. I had to use orderBy Churn because I want to use the attribute equalTo which specify a specific value for churn. I Also, can combine orderBy with any of the other parameters.

equalTo: it is one of the filtering data query parameters that allows me to choose arbitrary value point and the query has to satisfy this condition and only fetch points that its value is equal to the specified value. For example, I used “Yes” because I want the query to retrieve data that their Churn value is Yes. [Churn has either yes or no value].

print: This attribute has value pretty which meant printing the response(data) that you fetch from the firebase in a nice format that is easier for human to read.

I send **one** request to the firebase that has **all** the data filtering **commands** to get my response (data) faster so I can improve Firebase Realtime Database performance in my program and enhance my program reliability. Requesting data from the firebase takes time, so it is better to mention all the data filter parameter in one line of code. . In addition, because there is & sign that allow me to add as many filtering commands as I want, so I took advantage of this and use it in my string URL to filter the data the way I want, so I only have to communicate with the firebase once as firebase has limitation for how many requests a script can have to access the firebase database data.

3) **requests.get ('https://dsci551-a7a0e-default-rtdb.firebaseio.com/.json?orderBy="tenure"&print=pretty&startAt={}'.format(int(monthsNum)))**

requests: it is one of the HTTP libraries that allow sending requests to (and retrieved data from) specified URLs using python.

get: it is one of the requests methods that sends request to the specified URL to get and fetch the data.

url: the URL I used is a firebase real time database link and I added .json to make it possible to add my json (file format) data into the firebase database. I also added ? to allow me to add query parameters. And & is used to link every query with rest of the URL string.

orderBy: it is one of the filtering data query parameters that has value tenure which is a way to filter the data by child key. The output will be order based on the tenure value in an ascending order. I Also, can combine orderBy with any of the other parameters.

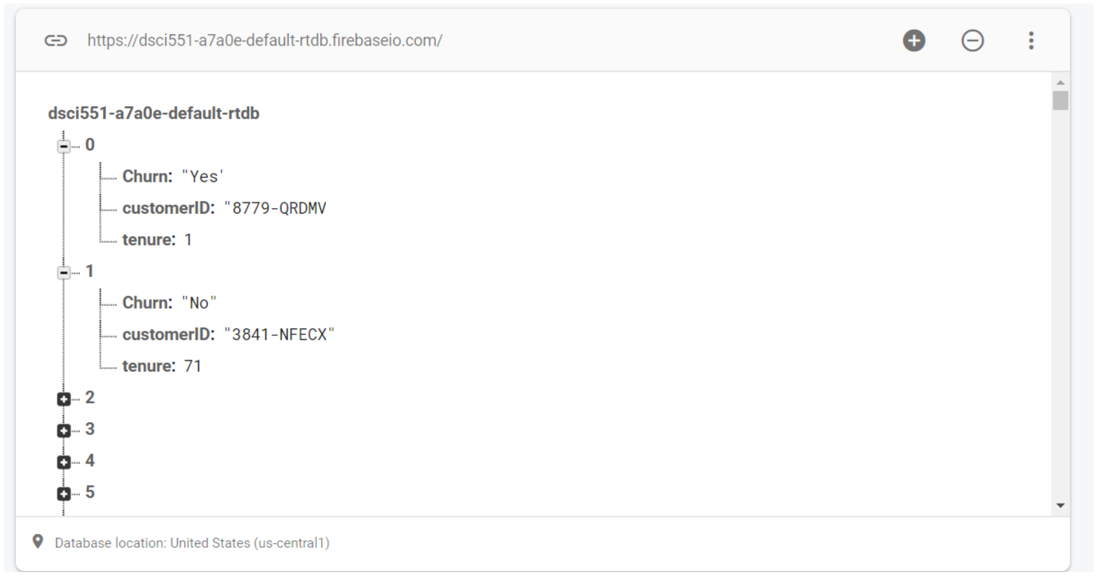
print: This attribute has value pretty which meant printing the response(data) that you fetch from the firebase in a nice format that is easier for human to read.

startAt: it is one of the filtering data query parameters that used to filter data based on arbitrary choice start points for my queries. For example, if the user enters 10, the query will retrieve the data that their tenure is starting from 10 and above.

format(int(monthsNum): Function format helps me to put the number that the user has enter into the specified {} place inside the URL string. MonthsNum is the variable name that receive an input from the user and because the input function receives any input as sting, I needed to transform it into integer.

I send **one** request to the firebase that has **all** the data filtering **commands** to get my response (data) faster so I can improve Firebase Realtime Database performance in my program and enhance my program reliability. Requesting data from the firebase takes time, so it is better to mention all the data filter parameter in one line of code. In addition, because there is & sign that allow me to add as many filtering commands as I want, so I took advantage of this and use it in my string URL to filter the data the way I want, so I only must communicate with the firebase once as firebase has limitation for how many requests a script can have to access the firebase database data.

A screenshot of the Firebase, showing the structure of the database:



https://dsci551-a7a0e-default-rtdb.firebaseio.com/

```
dsci551-a7a0e-default-rtdb
  0
    Churn: "Yes'
    customerID: "8779-QRDMV
    tenure: 1
  1
    Churn: "No"
    customerID: "3841-NFECX"
    tenure: 71
  2
  3
  4
  5
```

Database location: United States (us-central1)