Cloud Based Song recommendation

Team LA ®

Tushar N, Divyanshi K, Prasann V, Naman A

Requirement:

To enable the running code to make realtime changes to an existing Database of songs and to use this modified database for further execution of the the same code and file.

Steps Accomplished:

1. Enabled a google sheet api request
2. Downloaded the credentials
3. Initiated a google sheet using our Songs_LA.csv and granted editing rights using credentials
4. Learnt how to convert this sheet into a usable pandas dataframe and make recommendations to this copied Database
5. After the user types “exit”, this edited database is copied back to the google sheet and the information is stored.

Bugs faced:

1. It is dificult to coordinate corresponding changes in a dataframe and a google sheet **simultaneously**, hence all changes are made once the programme ends with exit code 0.
2. This means if the programme crashes in between, the change is not stored, even if the code ran properly for some time.
3. For a smooth editing of the sheet, the commands of the api require an in-depth knowledge of json which were, in our case, tediously solved using unfriendly and lousy methods.
4. The above point means that: After operating on the dataframe during the running of the program, it is required to delete the entire content of the google sheet and copy the modified dataframe back again.
5. Even the dataframe cannot be copied as such. It can only be appended row-by-row, which we thought of doing with loops.

After a long and tiring session of coding, we solved these bugs, only to encounter the most irritating of all:

The google sheet api allows only 100 sec of access to a google sheet per execution of a program. This means, from the moment we request an editing right, our code should make all changes within 100 secs of runtime which, you see, is impossible to resolve.

Conclusion:

This certain task was a wonderful approach to making a cloud-based song recommendation system but we lacked the precise skills required to achieve this certain task. However, the challenge was as gripping as any and we are still happy to have learned yet another approach to APIs.