The Result should include link to github project

The project should include the following:

1. readMe-file - guides for setup (explaining about project and installations for bring up)
2. code written to meet the requirements - make sure to use code standards (pep8)
3. in Optional please provide query and results (preferred postman)

\*\* You can use  google for support

\*\* You can use docker

setup:

1. Establish db - you can use any database you prefer.
2. Write python code that creates a table of ads in your selected database.
3. Each ad record contains the following fields:

• Index – integer, row number sequence

• Id – string, unique identifier for the ad record

• Name – string, short descriptive name

(the Name should be random string between 5 -15 chars)

1. Fill the tables with 20,000 records
2. Add the setup steps and the DB credentials (configuration) to readMe file.

Sort Program -step 1:

1. write python  program that query data from the table
2. creates sorted list of names ( from all 20000 records)
3. store that list in new table named ‘RESULTS’ under column (‘Sorting-step1’)
4. store the time took from start of the process to the end in table named ‘RESULTS’ under column  (‘Sorting-step1\_Process\_time’) (in seconds)

Sort Program -step 2:

\*\*\* LIMITATION\*\*\* Your Program (service) can only Process up to 2000 records at a time

1. write python  program that query data from the table in chunks of 2000 each time
2. creates sorted list of names ( from all 20000 records)- (hint: sort merging sorted list)
3. store that list in new table named result under column (‘Sorting-step2’)
4. store the time took from start of the process to the end in table named ‘RESULTS’ under column  (‘Sorting-step2\_Process\_time’) (in seconds)

Sort Program -step 3:

\*\*\* LIMITATION\*\*\* Your Program (service) can only Process up to 2000 records at a time

\*\*\*LIMITATION\*\*\* We need to speed up time ( for that you can use multiple servers with 2000 records limitation

1. write python  program that query data from the table in chunks of 2000 each time
2. Think of way to parallel the processes and implement it
3. creates sorted list of names ( from all 20000 records)- (hint: sort merging sorted list)
4. store that list in new table named result under column (‘Sorting-step3’)
5. store the time took from start of the process to the end in table named ‘RESULTS’ under column  (‘Sorting-step3\_Process\_time’) (in seconds)

Optional:

implement API Server that exposes the “GetAds” method.

GetAds can takes arg  of TOP in the header and return the top Sorted names.

if no TOP value is defined the entire sorted list will be returned.

1. The data is based from sorting in section 1
2. the response should be in Json Format
3. Preferred Implementation using NodeJs