Name :- Hashmeetsingh Obhan
Student ID :- 21262793
Email ID :- hashmeetsingh.obhan2@mail.dcu.ie

Link for the Git Repository:-
https://github.com/Hashmeetsingh/CloudTechnologies

Description of the dataset:-
The data in the dataset comes from a social media app like Facebook or Instagram. It has several key columns, such as Id, Body, Score, UserName, and Tags, that provide detailed information on the user.

Task 1:- Get data from Stack Exchange (Data Acquisition/Collection)

- To acquire the data I used the Data Explorer feature of the StackExchange system using following link to run the below 4 queries.

LINK:- https://data.stackexchange.com/stackoverflow/query/new

Technology used:- Used SQL like queries to fetch the data from Data Explorer feature of the StackExchange system.

Queries:-
SELECT * FROM (SELECT ROW_NUMBER() OVER (ORDER BY VIEWCOUNT DESC) AS RowNumber, Id, PostTypeId, AcceptedAnswerId, ParentId, CreationDate, DeletionDate, Score, ViewCount, OwnerUserId, OwnerDisplayName, LastEditorUserId, LastEditorDisplayName, LastEditDate, LastActivityDate, Title, Tags, AnswerCount, CommentCount, FavoriteCount, ClosedDate, CommunityOwnedDate, ContentLicense FROM POSTS WHERE VIEWCOUNT IS NOT NULL) AS POSTSTABLE WHERE RowNumber BETWEEN 1 AND 50001

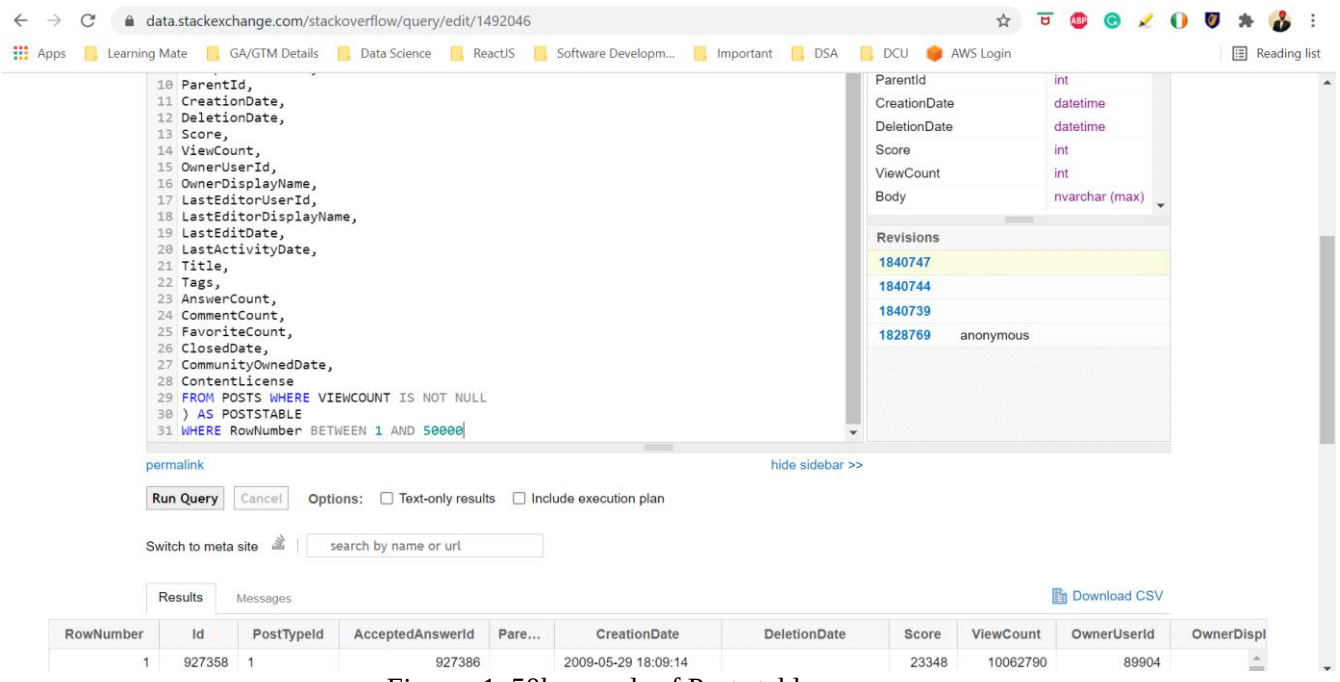

Figure:-1. 50k records of Posts table

SELECT * FROM (SELECT ROW_NUMBER() OVER (ORDER BY VIEWCOUNT DESC) AS RowNumber, Id, PostTypeId, AcceptedAnswerId, ParentId, CreationDate, DeletionDate, Score, ViewCount, OwnerUserId, OwnerDisplayName, LastEditorUserId, LastEditorDisplayName, LastEditDate, LastActivityDate, Title, Tags, AnswerCount, CommentCount, FavoriteCount, ClosedDate, CommunityOwnedDate, ContentLicense FROM POSTS WHERE VIEWCOUNT IS NOT NULL) AS POSTSTABLE WHERE RowNumber BETWEEN 50001 AND 100000

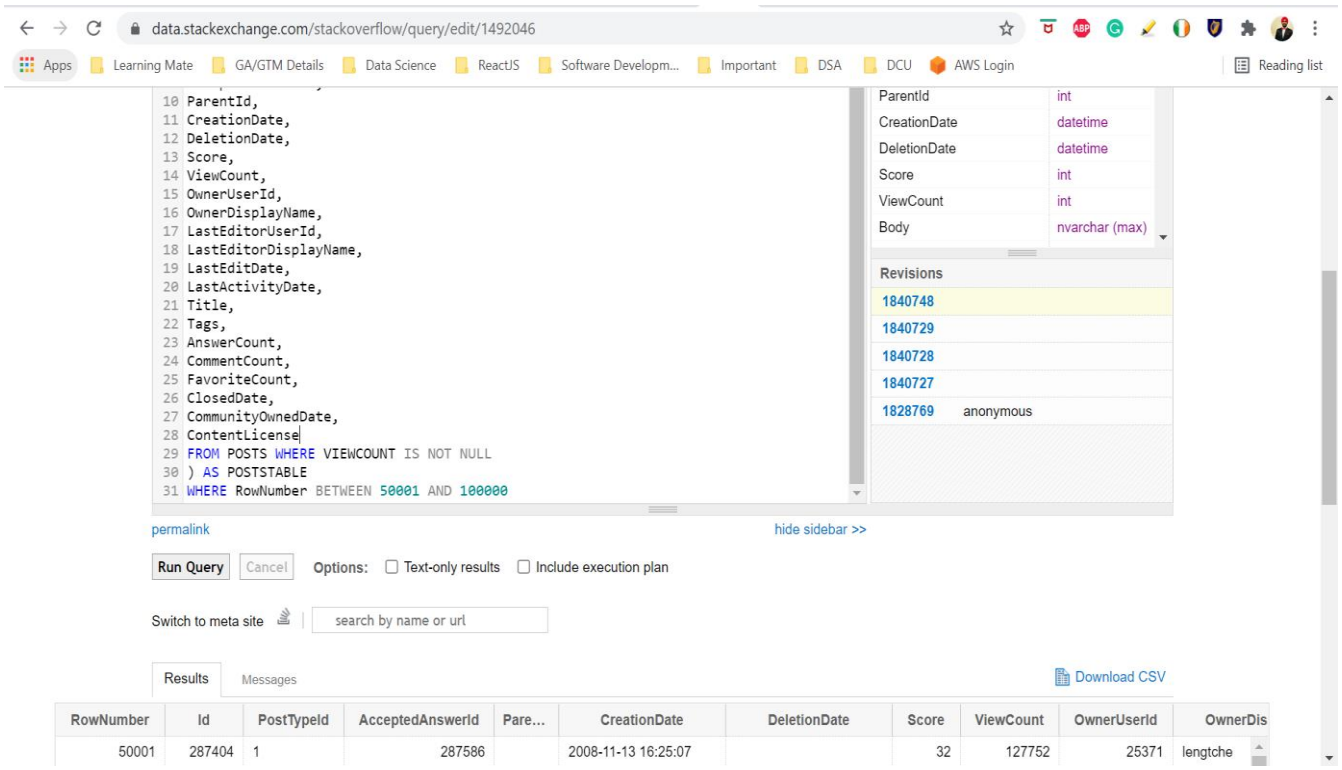

Figure:-2. 50k records of Posts table

SELECT * FROM (SELECT ROW_NUMBER() OVER (ORDER BY VIEWCOUNT DESC) AS RowNumber, Id, PostTypeId, AcceptedAnswerId, ParentId, CreationDate, DeletionDate, Score, ViewCount, OwnerUserId, OwnerDisplayName, LastEditorUserId, LastEditorDisplayName, LastEditDate, LastActivityDate, Title, Tags, AnswerCount, CommentCount, FavoriteCount, ClosedDate, CommunityOwnedDate, ContentLicense FROM POSTS WHERE VIEWCOUNT IS NOT NULL) AS POSTSTABLE WHERE RowNumber BETWEEN 100001 AND 150000

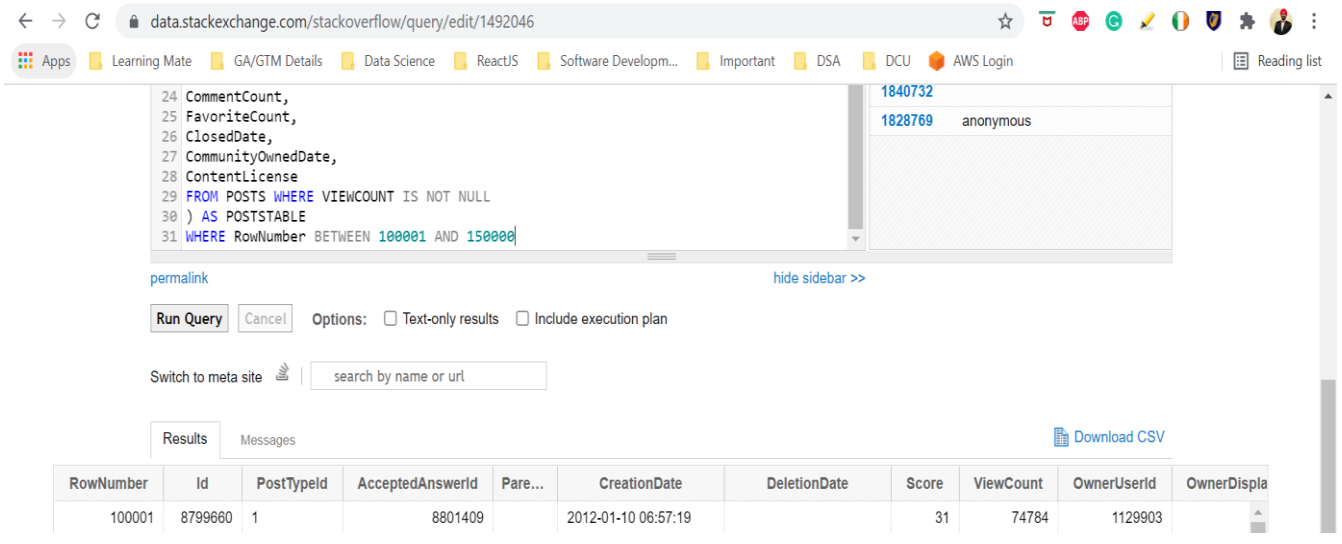

Figure:-3. 50k records of Posts table

SELECT * FROM (SELECT ROW_NUMBER() OVER (ORDER BY VIEWCOUNT DESC) AS RowNumber, Id, PostTypeId, AcceptedAnswerId, ParentId, CreationDate, DeletionDate, Score, ViewCount, OwnerUserId, OwnerDisplayName, LastEditorUserId, LastEditorDisplayName, LastEditDate, LastActivityDate, Title, Tags, AnswerCount, CommentCount, FavoriteCount, ClosedDate, CommunityOwnedDate, ContentLicense FROM POSTS WHERE VIEWCOUNT IS NOT NULL) AS POSTSTABLE WHERE RowNumber BETWEEN 150001 AND 200000

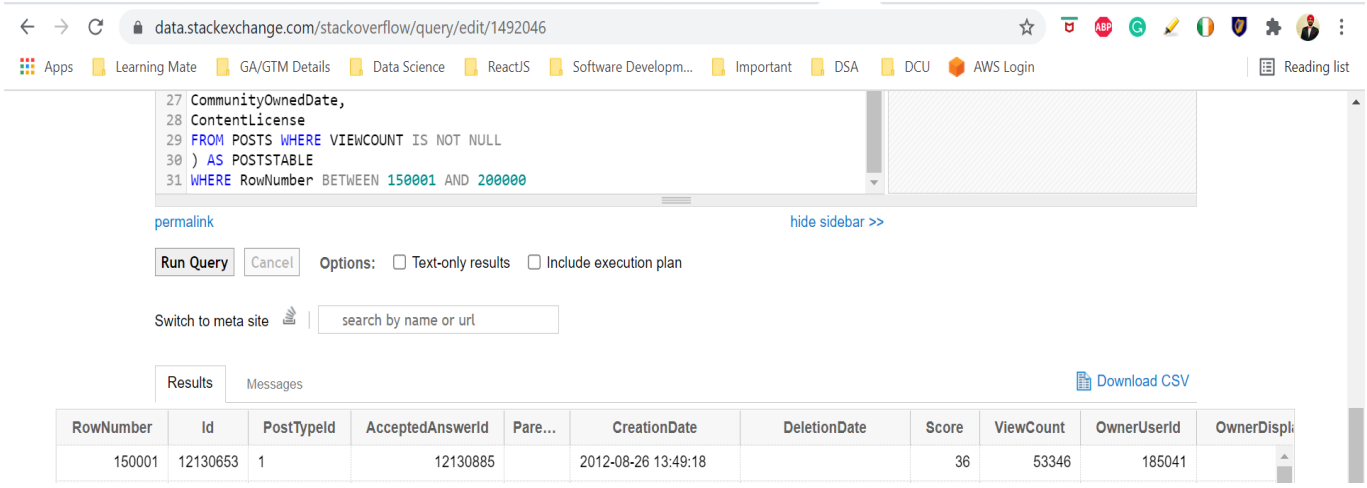

Figure:-4. 50k records of Posts table

- Following the execution of the above four queries, I downloaded the CSV files containing 50000 records each, and then combined them into a single file with 200000 records. After that I cleaned "**PostBodyData**" and "**PostTitle**" columns of that file to get the optimized result.

  Technology used:- I used Python to clean the data as it is quite easy to clean the dataset using python and reduces complexity as well. **(6) (7) (8)**

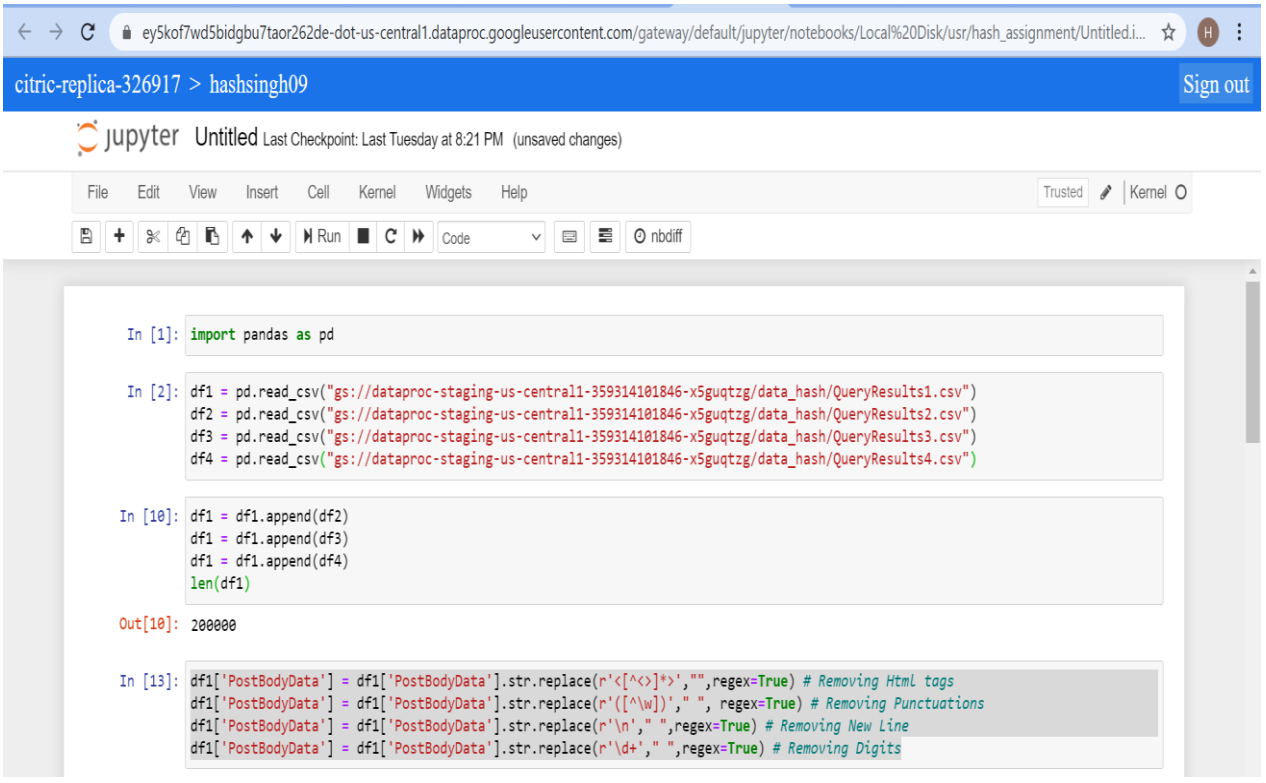

Figure:-5. Merging 4 dataset files into 1 single file

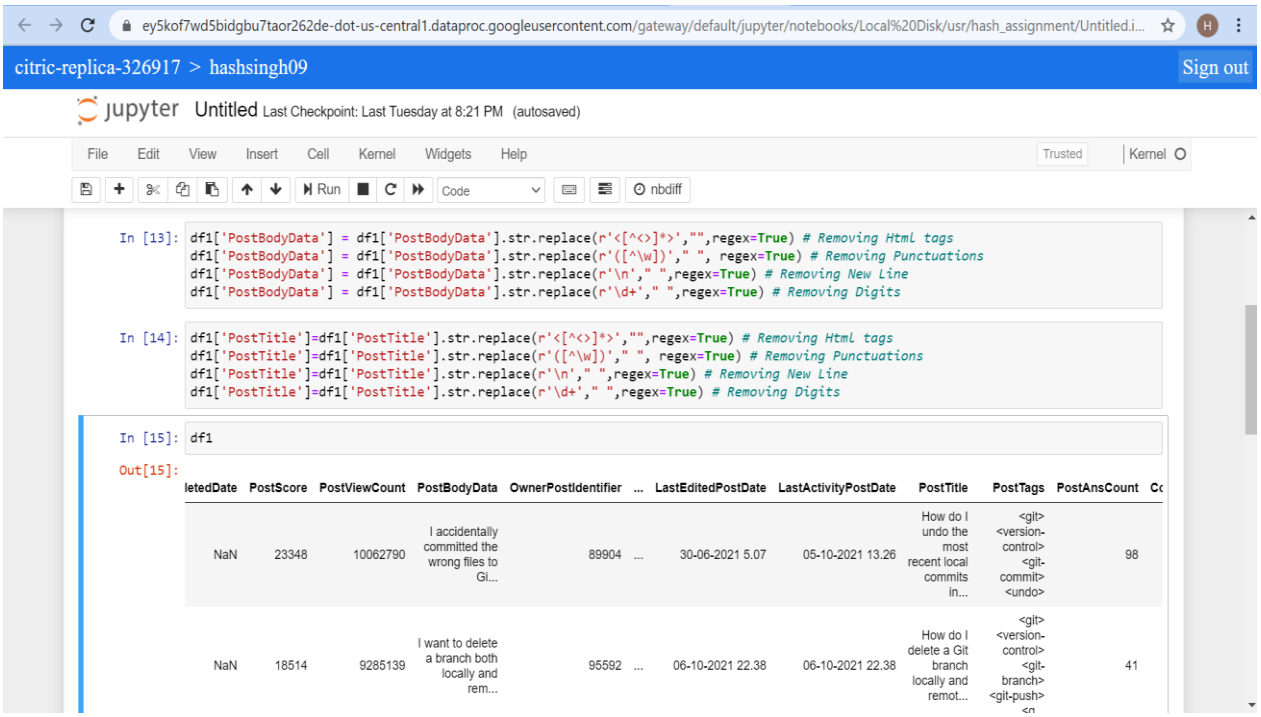

Figure:-6. Cleaning "PostBodyData" and "PostTitle" columns of the dataset

- I performed all the tasks on Google Cloud Platform (GCP) environment. Firstly, I created an instance on GCP having 1 namenode and 2 workers (2 datanodes).
- Then I uploaded the CSV file having 200000 records on GCP cloud storage as shown in below image.

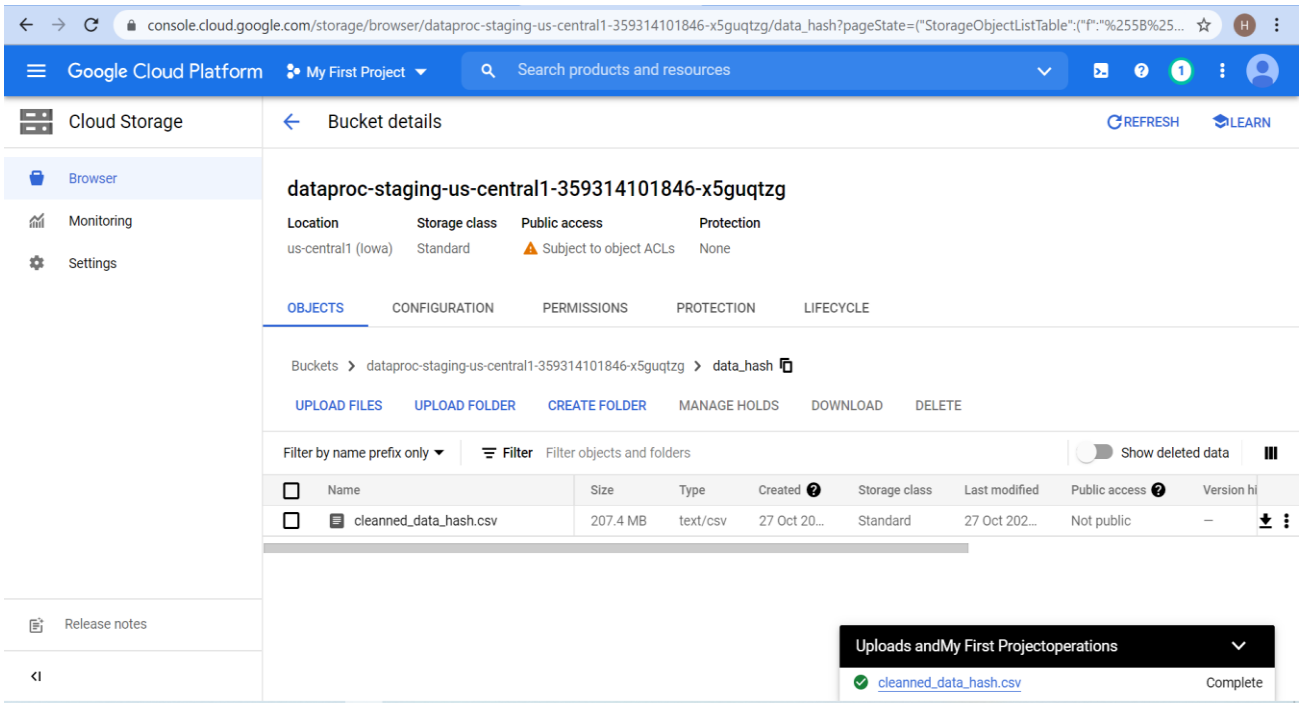

Figure:-7. Successfully uploaded the cleaned data file to GCP cloud

Task 2:- Load data into chosen cloud technology (Hive)

- Once uploading the file on GCP is completed I created a table on HIVE, which is pre-installed on GCP. Then I loaded the data in that table using the query. After that I created a View which has a similar structure as that of table where in I typecast few columns.

Technology used:- I have used HIVE to load the data into table. This is because the HIVE is comparatively easy to implement than Pig and Mapreduce due to which there are less efforts required for development. Another reason being using HIVE is I am quite familiar with SQL which made it easier to use HIVE. **(5)**

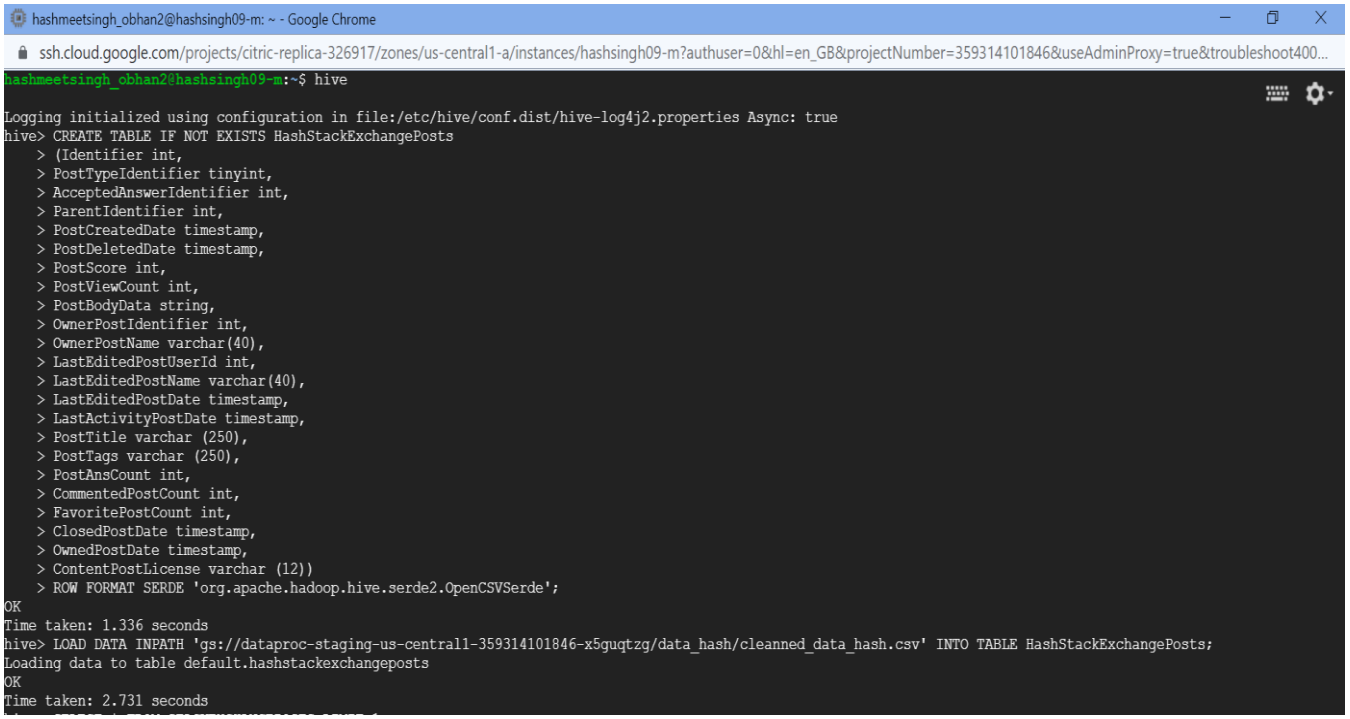

Figure:-8. Successfully loaded the data into table using HIVE

Task 3:- Run the Query data using Hive

Technology used:- Similarly like Task 2, I have used HIVE to execute the all the task 3 queries as they are similar to SQL like query language and easy to implement.

3.1. The top 10 posts by score

- To get the result I executed the below query on HIVE (GCP)
  - SELECT Identifier, PostTitle, PostScore from HashStackExchangeView ORDER BY PostScore DESC LIMIT 10;

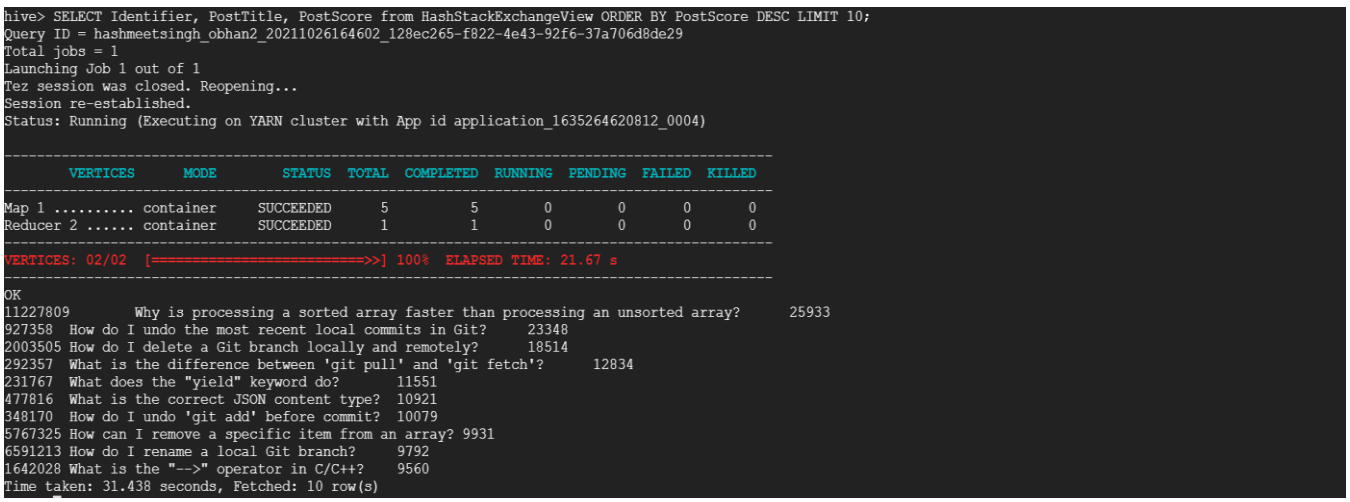

Figure:-9. Successfully executed the top 10 posts by score

3.2. The top 10 users by post score

- To get the result I executed the below query on HIVE (GCP)
  - SELECT OwnerPostIdentifier,OwnerPostName,sum(PostScore) as PostScore from HashStackExchangeView GROUP BY OwnerPostIdentifier, OwnerPostName ORDER BY PostScore DESC LIMIT 10;

```
hive> SELECT OwnerPostIdentifier,OwnerPostName,sum(PostScore) as PostScore from HashStackExchangeView GROUP BY OwnerPostIdentifier, OwnerPostName ORDER BY PostScore
 DESC LIMIT 10
    > ;
Query ID = hashmeetsingh_obhan2_20211026170055_b66d3402-a3b5-4fef-903b-6f2efe0828df
Total jobs = 1
Launching Job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1635264620812_0005)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      5          5        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
Reducer 3 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 03/03  [==========================>>] 100%  ELAPSED TIME: 24.27 s
--------------------------------------------------------------------------------
OK
87234   GManNickG       37672
4883    readonly        28817
9951    e-satis 26878
6068    pupeno  25944
89904   Hamza Yerlikaya 24024
51816   Joan Venge      23763
49153   Ali     20203
179736  TIMEX   19603
95592   Matthew Rankin  19479
63051   flybywire       19362
Time taken: 34.873 seconds, Fetched: 10 row(s)
```

Figure:-10. Successfully executed the top 10 users by post score

3.3. The number of distinct users, who used the word “cloud” in one of their posts

- Similarly like above 2 queries, I executed the below query on HIVE (GCP) to obtain the result.
  - SELECT COUNT(DISTINCT OwnerPostIdentifier) as TotalDistinctUsers FROM HashStackExchangeView WHERE PostTitle LIKE '% cloud %' OR PostBodyData LIKE '% cloud %';

```
hive> SELECT COUNT(DISTINCT OwnerPostIdentifier) as TotalDistinctUsers FROM HashStackExchangeView WHERE PostTitle LIKE '% cloud %' OR PostBodyData LIKE '% cloud %';

Query ID = hashmeetsingh_obhan2_20211026171011_7364cfc4-8390-4cfc-bf13-15f2016f8cd8
Total jobs = 1
Launching Job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1635264620812_0006)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      5          5        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
Reducer 3 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 03/03  [==========================>>] 100%  ELAPSED TIME: 23.64 s
--------------------------------------------------------------------------------
OK
248
Time taken: 33.976 seconds, Fetched: 1 row(s)
```

Figure:-11. Number of distinct users who used the word “cloud” in their posts Body/Title column

Task 4:- Calculate the per-user TF-IDF of the top 10 terms for each of the top 10 users.

Technology used:- This time I have used Python to find the top 10 terms for each of the 10 users as it was easy to develop and implement the code in python as compared to HIVE and Pig. Also, the code is much more efficient using Python and able to get the optimum result.

Explanation of the code Snippet:-
Firstly I installed the required libraries and packages. Then I establish a connection with HIVE database by passing the necessary parameters for authorization to the connection string. After that I found the top 10 user details based on their posts score. Once done with this, I searched for OwnerPostIdentifier in details that I fetched earlier for 10 users. Last but not the least, I used sklearn library to find the TF_IDF of each user. Refer Figure-9,10 and 11 which describes the implementation to find the TF_IDF of the top 10 terms for each of the 10 users. **(4)**

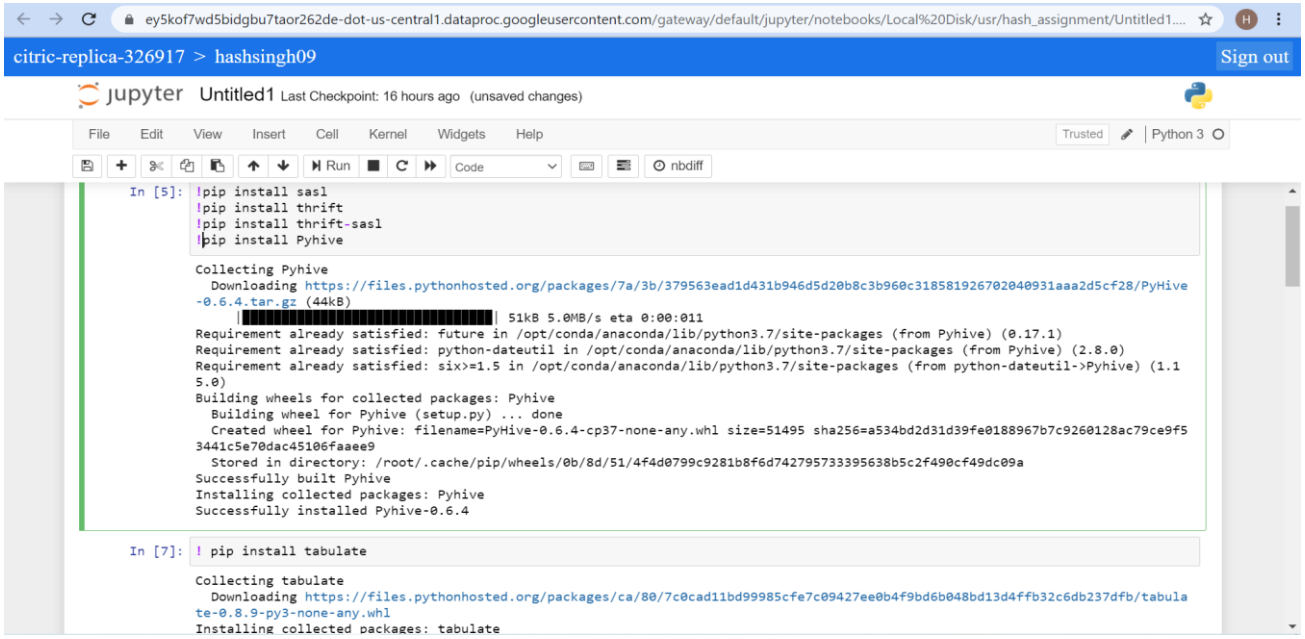

Figure:-12. Installation of necessary library and packages needed to execute the code

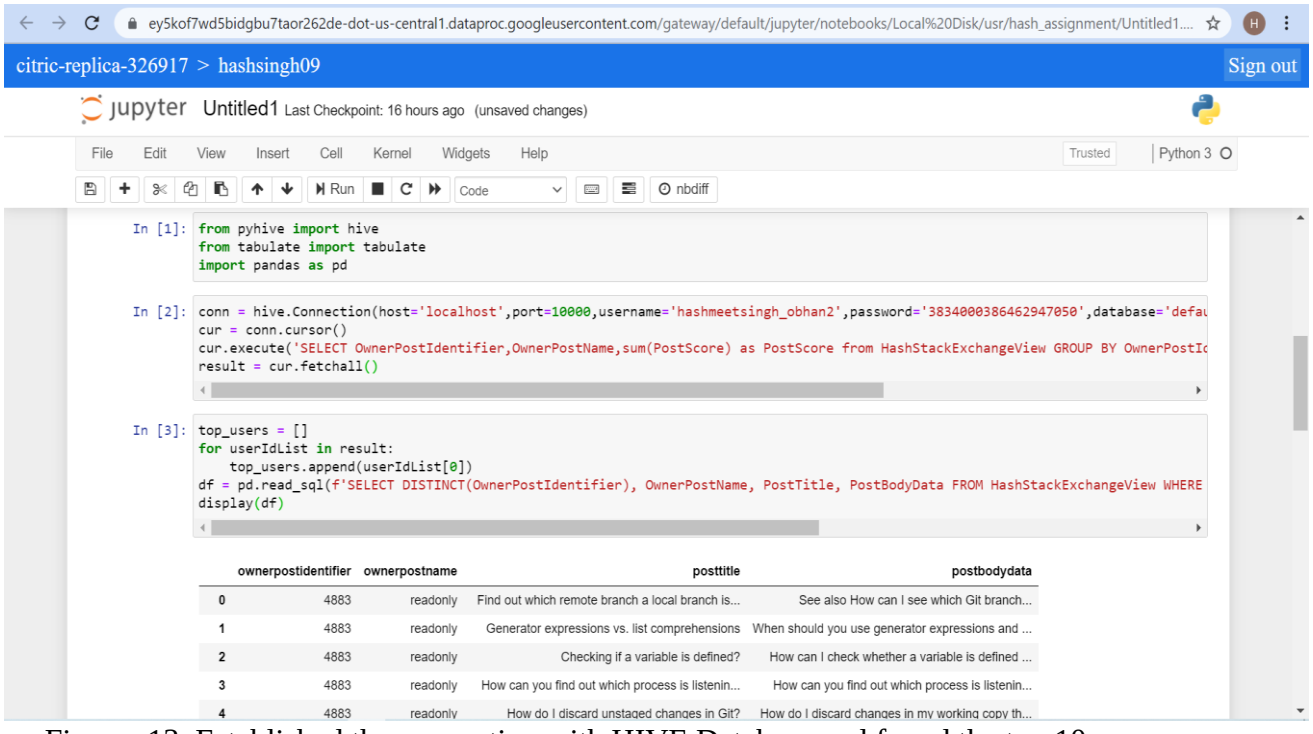

Figure:-13. Established the connection with HIVE Database and found the top 10 users

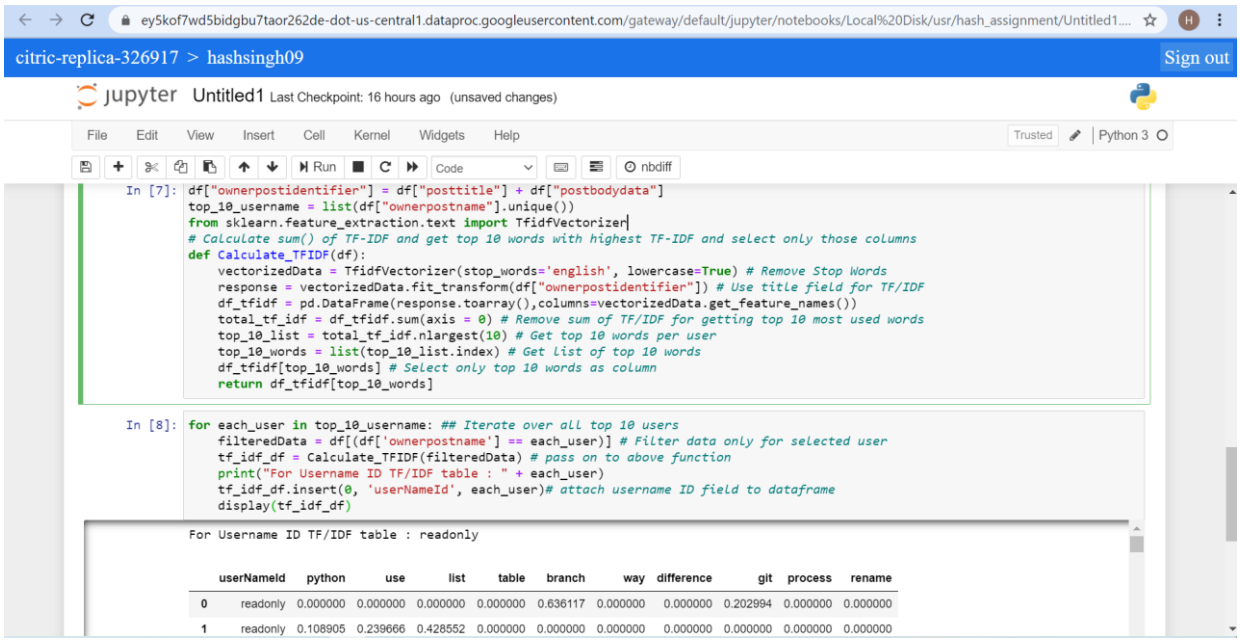

Figure:-14. Successfully executed the top 10 terms for each of the 10 users

APPENDIX

References:-

1. https://phoenixnap.com/kb/install-hive-on-ubuntu
2. https://phoenixnap.com/kb/install-hadoop-ubuntu
3. https://www.guru99.com/file-permissions.html
4. https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html
5. https://www.projectpro.io/article/mapreduce-vs-pig-vs-hive/163
6. https://stackoverflow.com/questions/45999415/removing-html-tags-in-pandas
7. https://www.pythondaddy.com/python/how-to-remove-punctuation-from-a-dataframe-in-pandas-and-python/
8. https://stackoverflow.com/questions/41719259/how-to-remove-numbers-from-string-terms-in-a-pandas-dataframe