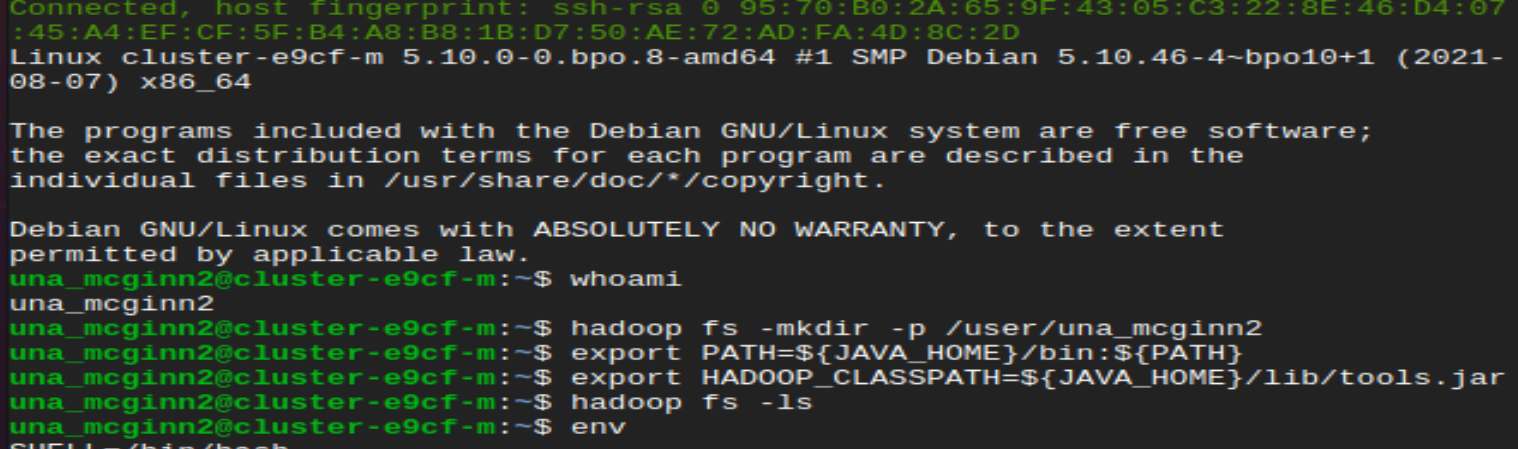
**Supporting Documentation: Task 2 and 3**

**Student: Úna McGinn**

**CA675 Assignment 1**

**Task 2:**

**Load Data to Pig:**

-- Setup new directory-- The data was loaded to the Pig Data folder.

hadoop fs -put /home/una\_mcginn2/QueryResults1.csv /PigData

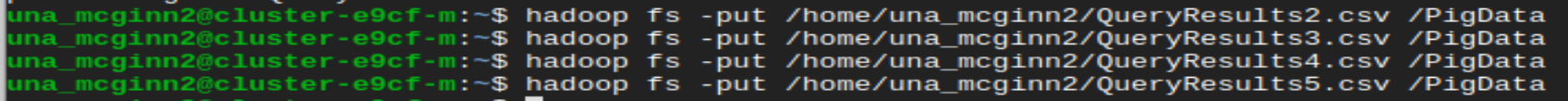
hadoop fs -put /home/una\_mcginn2/QueryResults2.csv /PigData

hadoop fs -put /home/una\_mcginn2/QueryResults3.csv /PigData

hadoop fs -put /home/una\_mcginn2/QueryResults4.csv /PigData

hadoop fs -put /home/una\_mcginn2/QueryResults5.csv /PigData





REGISTER /usr/lib/pig/piggybank.jar;

**Source:** <https://stackoverflow.com/questions/31460632/reading-a-csv-file-in-pig>

**Code in Pig:**

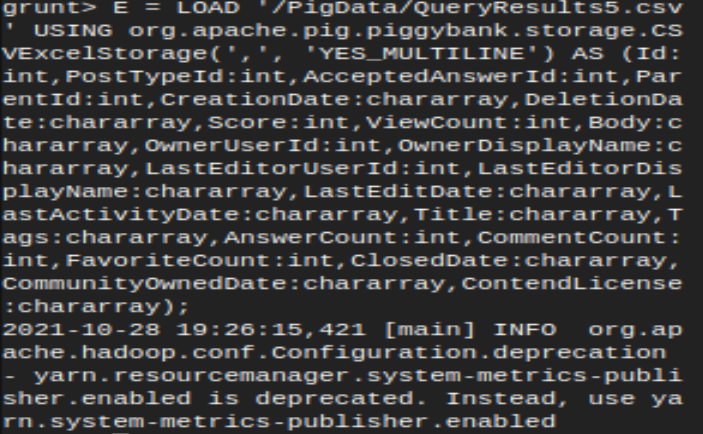
A = LOAD '/PigData/QueryResults1.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'YES\_MULTILINE') AS (Id:int,PostTypeId:int,AcceptedAnswerId:int,ParentId:int,CreationDate:chararray,DeletionDate:chararray,Score:int,ViewCount:int,Body:chararray,OwnerUserId:int,OwnerDisplayName:chararray,LastEditorUserId:int,LastEditorDisplayName:chararray,LastEditDate:chararray,LastActivityDate:chararray,Title:chararray,Tags:chararray,AnswerCount:int,CommentCount:int,FavoriteCount:int,ClosedDate:chararray,CommunityOwnedDate:chararray,ContendLicense:chararray);

B = LOAD '/PigData/QueryResults2.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'YES\_MULTILINE') AS (Id:int,PostTypeId:int,AcceptedAnswerId:int,ParentId:int,CreationDate:chararray,DeletionDate:chararray,Score:int,ViewCount:int,Body:chararray,OwnerUserId:int,OwnerDisplayName:chararray,LastEditorUserId:int,LastEditorDisplayName:chararray,LastEditDate:chararray,LastActivityDate:chararray,Title:chararray,Tags:chararray,AnswerCount:int,CommentCount:int,FavoriteCount:int,ClosedDate:chararray,CommunityOwnedDate:chararray,ContendLicense:chararray);

C = LOAD '/PigData/QueryResults3.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'YES\_MULTILINE') AS (Id:int,PostTypeId:int,AcceptedAnswerId:int,ParentId:int,CreationDate:chararray,DeletionDate:chararray,Score:int,ViewCount:int,Body:chararray,OwnerUserId:int,OwnerDisplayName:chararray,LastEditorUserId:int,LastEditorDisplayName:chararray,LastEditDate:chararray,LastActivityDate:chararray,Title:chararray,Tags:chararray,AnswerCount:int,CommentCount:int,FavoriteCount:int,ClosedDate:chararray,CommunityOwnedDate:chararray,ContendLicense:chararray);

D = LOAD '/PigData/QueryResults4.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'YES\_MULTILINE') AS (Id:int,PostTypeId:int,AcceptedAnswerId:int,ParentId:int,CreationDate:chararray,DeletionDate:chararray,Score:int,ViewCount:int,Body:chararray,OwnerUserId:int,OwnerDisplayName:chararray,LastEditorUserId:int,LastEditorDisplayName:chararray,LastEditDate:chararray,LastActivityDate:chararray,Title:chararray,Tags:chararray,AnswerCount:int,CommentCount:int,FavoriteCount:int,ClosedDate:chararray,CommunityOwnedDate:chararray,ContendLicense:chararray);

E = LOAD '/PigData/QueryResults5.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'YES\_MULTILINE') AS (Id:int,PostTypeId:int,AcceptedAnswerId:int,ParentId:int,CreationDate:chararray,DeletionDate:chararray,Score:int,ViewCount:int,Body:chararray,OwnerUserId:int,OwnerDisplayName:chararray,LastEditorUserId:int,LastEditorDisplayName:chararray,LastEditDate:chararray,LastActivityDate:chararray,Title:chararray,Tags:chararray,AnswerCount:int,CommentCount:int,FavoriteCount:int,ClosedDate:chararray,CommunityOwnedDate:chararray,ContendLicense:chararray);



**Data Cleaning:**

-- The data is cleaned, accounting for line breaks, new lines, carriage returns and commas.

A1 = FOREACH A GENERATE Id, PostTypeId, AcceptedAnswerId, ParentId, CreationDate, DeletionDate, Score, ViewCount, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Body,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Body, OwnerUserId, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(OwnerDisplayName,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS OwnerDisplayName, LastEditorUserId, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(LastEditorDisplayName,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS LastEditorDisplayName, LastEditDate, LastActivityDate, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Title,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Title, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Tags,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Tags, AnswerCount, CommentCount, FavoriteCount, ClosedDate, CommunityOwnedDate, ContendLicense;

B2 = FOREACH B GENERATE Id, PostTypeId, AcceptedAnswerId, ParentId, CreationDate, DeletionDate, Score, ViewCount, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Body,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Body, OwnerUserId, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(OwnerDisplayName,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS OwnerDisplayName, LastEditorUserId, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(LastEditorDisplayName,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS LastEditorDisplayName, LastEditDate, LastActivityDate, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Title,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Title, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Tags,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Tags, AnswerCount, CommentCount, FavoriteCount, ClosedDate, CommunityOwnedDate, ContendLicense;

C3 = FOREACH C GENERATE Id, PostTypeId, AcceptedAnswerId, ParentId, CreationDate, DeletionDate, Score, ViewCount, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Body,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Body, OwnerUserId, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(OwnerDisplayName,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS OwnerDisplayName, LastEditorUserId, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(LastEditorDisplayName,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS LastEditorDisplayName, LastEditDate, LastActivityDate, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Title,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Title, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Tags,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Tags, AnswerCount, CommentCount, FavoriteCount, ClosedDate, CommunityOwnedDate, ContendLicense;

D4 = FOREACH D GENERATE Id, PostTypeId, AcceptedAnswerId, ParentId, CreationDate, DeletionDate, Score, ViewCount, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Body,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Body, OwnerUserId, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(OwnerDisplayName,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS OwnerDisplayName, LastEditorUserId, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(LastEditorDisplayName,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS LastEditorDisplayName, LastEditDate, LastActivityDate, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Title,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Title, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Tags,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Tags, AnswerCount, CommentCount, FavoriteCount, ClosedDate, CommunityOwnedDate, ContendLicense;

E5= FOREACH E GENERATE Id, PostTypeId, AcceptedAnswerId, ParentId, CreationDate, DeletionDate, Score, ViewCount, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Body,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Body, OwnerUserId, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(OwnerDisplayName,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS OwnerDisplayName, LastEditorUserId, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(LastEditorDisplayName,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS LastEditorDisplayName, LastEditDate, LastActivityDate, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Title,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Title, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(Tags,'\\n',''),'\\r',''),'\\r\\n',''),'<br>',''),',',' ') AS Tags, AnswerCount, CommentCount, FavoriteCount, ClosedDate, CommunityOwnedDate, ContendLicense;

---

**Source:** <https://www.simplilearn.com/tutorials/hadoop-tutorial/pig>

<https://www.educba.com/pig-commands/>

-- Union the subsets to make one dataset F

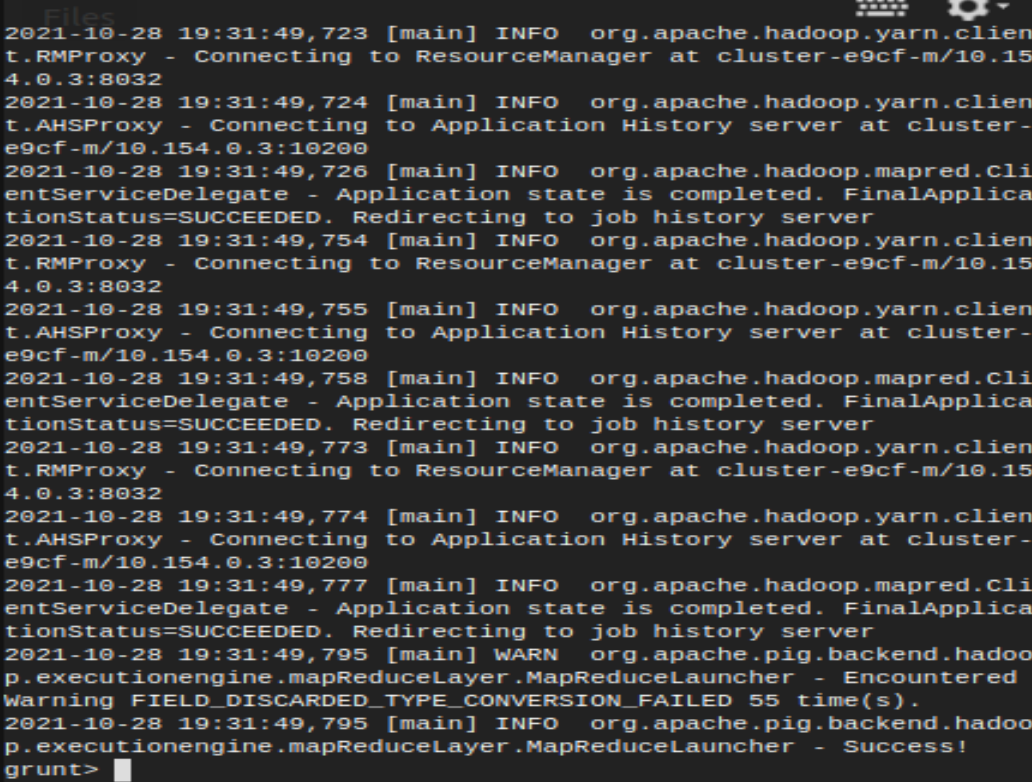
F = UNION A1, B2, C3, D4, E5;

G = FOREACH F GENERATE Id, Title, PostTypeId, Score;

Dump G;

-- Store the clean data from F to Hive directory previously created

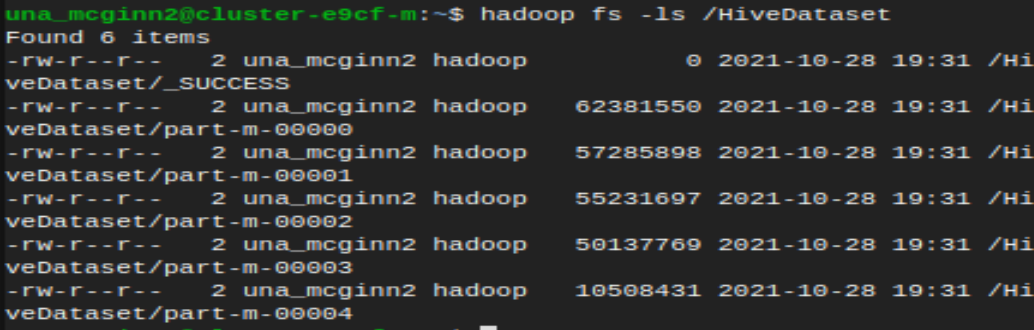
Store F INTO '/HiveDataset' USING PigStorage(',');



**In Local:**

-- Check if file was stored

hadoop fs -ls /HiveDataset



**Source:** <https://spark.apache.org/docs/3.1.1/sql-ref-syntax-ddl-create-table-hiveformat.html>

**Go to Hive:**

**Code:**

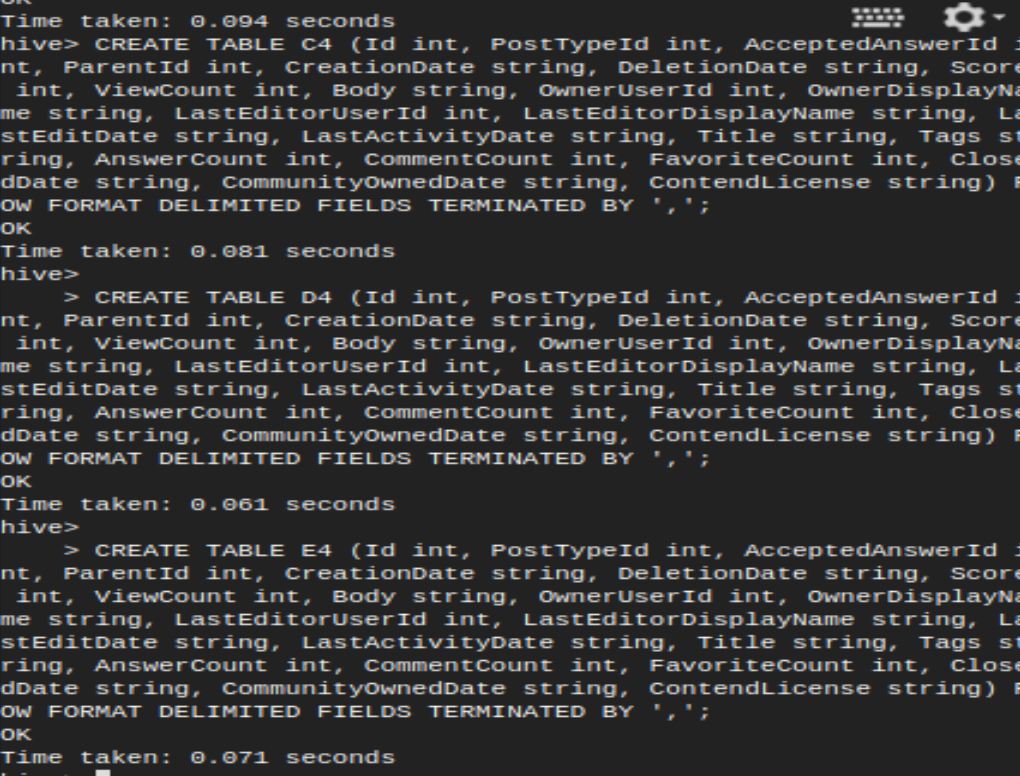
CREATE TABLE A4 (Id int, PostTypeId int, AcceptedAnswerId int, ParentId int, CreationDate string, DeletionDate string, Score int, ViewCount int, Body string, OwnerUserId int, OwnerDisplayName string, LastEditorUserId int, LastEditorDisplayName string, LastEditDate string, LastActivityDate string, Title string, Tags string, AnswerCount int, CommentCount int, FavoriteCount int, ClosedDate string, CommunityOwnedDate string, ContendLicense string) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' TBLPROPERTIES('skip.header.line.count'='1');

CREATE TABLE B4 (Id int, PostTypeId int, AcceptedAnswerId int, ParentId int, CreationDate string, DeletionDate string, Score int, ViewCount int, Body string, OwnerUserId int, OwnerDisplayName string, LastEditorUserId int, LastEditorDisplayName string, LastEditDate string, LastActivityDate string, Title string, Tags string, AnswerCount int, CommentCount int, FavoriteCount int, ClosedDate string, CommunityOwnedDate string, ContendLicense string) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',';

CREATE TABLE C4 (Id int, PostTypeId int, AcceptedAnswerId int, ParentId int, CreationDate string, DeletionDate string, Score int, ViewCount int, Body string, OwnerUserId int, OwnerDisplayName string, LastEditorUserId int, LastEditorDisplayName string, LastEditDate string, LastActivityDate string, Title string, Tags string, AnswerCount int, CommentCount int, FavoriteCount int, ClosedDate string, CommunityOwnedDate string, ContendLicense string) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',';

CREATE TABLE D4 (Id int, PostTypeId int, AcceptedAnswerId int, ParentId int, CreationDate string, DeletionDate string, Score int, ViewCount int, Body string, OwnerUserId int, OwnerDisplayName string, LastEditorUserId int, LastEditorDisplayName string, LastEditDate string, LastActivityDate string, Title string, Tags string, AnswerCount int, CommentCount int, FavoriteCount int, ClosedDate string, CommunityOwnedDate string, ContendLicense string) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',';

CREATE TABLE E4 (Id int, PostTypeId int, AcceptedAnswerId int, ParentId int, CreationDate string, DeletionDate string, Score int, ViewCount int, Body string, OwnerUserId int, OwnerDisplayName string, LastEditorUserId int, LastEditorDisplayName string, LastEditDate string, LastActivityDate string, Title string, Tags string, AnswerCount int, CommentCount int, FavoriteCount int, ClosedDate string, CommunityOwnedDate string, ContendLicense string) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',';



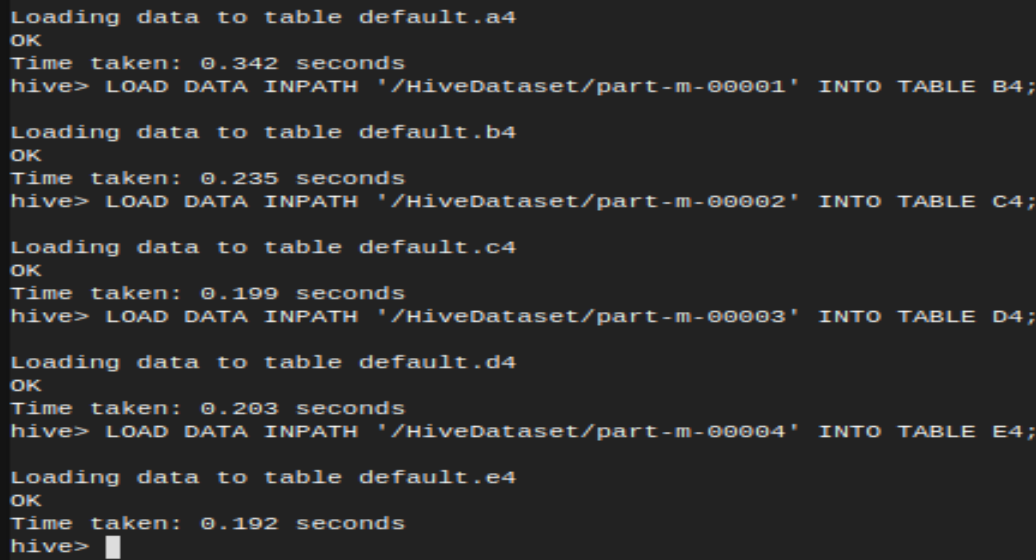
LOAD DATA INPATH '/HiveDataset/part-m-00000' INTO TABLE A4;

LOAD DATA INPATH '/HiveDataset/part-m-00001' INTO TABLE B4;

LOAD DATA INPATH '/HiveDataset/part-m-00002' INTO TABLE C4;

LOAD DATA INPATH '/HiveDataset/part-m-00003' INTO TABLE D4;

LOAD DATA INPATH '/HiveDataset/part-m-00004' INTO TABLE E4;



-- Create table F4 as resultant table of Union of A4, B4, C4, D4 and E4

create table F4 as

select \* From A4 union all

select \* From B4 union all

select \* From C4 union all

select \* From D4 union all

select \* From E4;

-- Remove records where Body is not equal to ‘Body’

insert overwrite table F4

select \* from F4

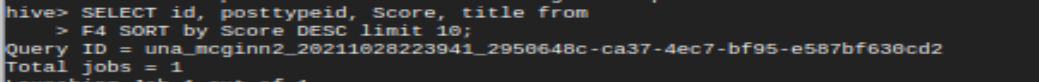
where Body <> 'Body';

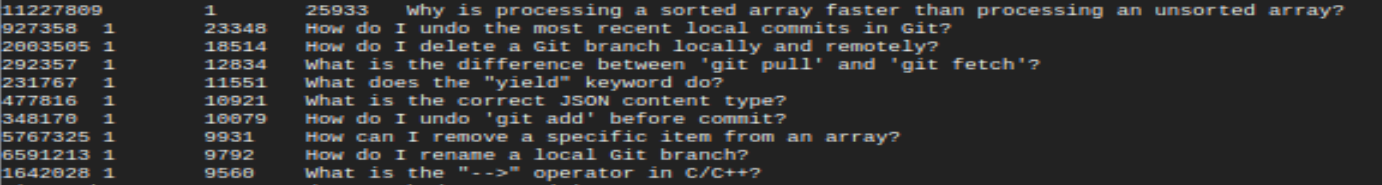
**Task 3:**

-- The top 10 posts by score

SELECT id, posttypeid, Score, title from

F4 SORT by Score DESC limit 10;





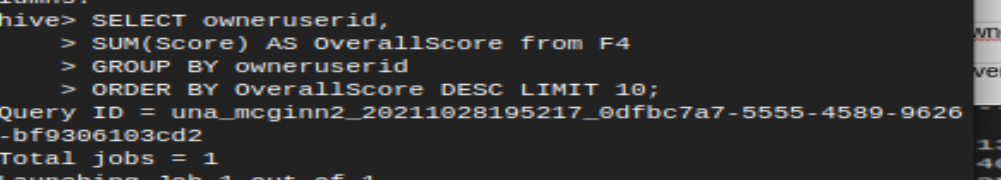
-- The top 10 users by post score

SELECT owneruserid,

SUM(Score) AS OverallScore from F4

GROUP BY owneruserid

ORDER BY OverallScore DESC LIMIT 10;





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

SELECT DISTINCT(OwnerUserId) From F4

WHERE (Body REGEXP 'cloud') OR (Title REGEXP 'cloud');

