# IST769 Homework 5 Submission

## Basic Information

Your Name: Srihari Busam   
Your SUID: sbusam  
Your Email: sbusam@syr.edu  
Date Due: 11/11/2021  
Homework #: 6

## QUESTIONS:

1. Load the comma-delimited HDFS dataset at **clickstream/iplookup** into a relation with an explicit schema. Use filter logic to remove the first row (which contains a header) then sort the output by IP and dump a comma-delimited data set to **clickstream/iplookup\_noheader.** Record all of your Pig commands required to complete your transformation.
2. Write Pig commands to produce a count of IP Addresses by state codes, sorted by the count with highest values first, like this:  
   (CA, 10)  
   (NY, 4)  
   (VA, 2)  
   Etc…  
   Record all your Pig commands required to complete your transformation.
3. Use pig to load the web log files from **clickstream/logs** using the following schema:   
   reqdate:chararray, reqtime:chararray, x1:int, method:chararray, uri:chararray, x2:int ,x3:int, x4:int ,ipaddress:chararray, useragent:chararray, filter any rows which begin with a “#” (these are header rows and should be removed, then writes out the reqdate, reqtime, method, uri, ipaddress and useragent columns to a tab-delimited data set in HDFS **clickstream/logs\_noheader**. HINT: The data is space delimited.
4. Use hive to create two external tables for the **clickstream/logs\_noheader** and **clickstream/iplookup\_noheader** files you created in the previous steps. These tables should be named **weblogs** and **iplookup** respectively and should be placed in the **clickstream** database. Be sure to record all HQL steps to complete the operations.
5. Write an HQL query to display the name of the city and the number of HTTP requests from that city (NOTE: each row in the web logs is an HTTP request). Order the output so cities with the most requests are at the top. If you complete the query correctly, you should see Syracuse has 272-page requests and Los Angeles has 24.

## ANSWERS:

### **ANSWER 1:**

|  |
| --- |
| CODE |
| iplookup = LOAD 'clickstream/iplookup/ip\_lookup.csv' USING PigStorage(',') AS (IP:chararray, Country:chararray,State:chararray, City:chararray, ApproxLat:float, ApproxLng: float);  iplookup\_no\_header = FILTER iplookup BY IP != 'IP';  iplookup\_no\_header\_ordered = ORDER iplookup\_no\_header BY IP ASC;  -- Dump the output to check  DUMP iplookup\_no\_header\_ordered  -- store the output to hdfs  STORE iplookup\_no\_header\_ordered INTO 'clickstream/iplookup\_noheader' USING PigStorage(',');  -- check the output  fs -cat clickstream/iplookup\_noheader/\* |

Screenshot of output of file written:

Text

Description automatically generated

ANSWER2:

|  |
| --- |
| CODE |
| ---Loading the data  iplookup = LOAD 'clickstream/iplookup/ip\_lookup.csv' USING PigStorage(',') AS (IP:chararray, Country:chararray,State:chararray, City:chararray, ApproxLat:float, ApproxLng: float);  iplookup\_no\_header = FILTER iplookup BY IP != 'IP';  -- aggregation operations  group\_data = GROUP iplookup\_no\_header BY State;  DESCRIBE group\_data  group\_data\_aggregate = FOREACH group\_data GENERATE group as State, COUNT(iplookup\_no\_header.IP) AS Total;  group\_data\_aggregate\_ordered = ORDER group\_data\_aggregate BY Total DESC;  DUMP group\_data\_aggregate\_ordered; |

Text

Description automatically generated

ANSWER3:

|  |
| --- |
| CODE |
| -- Load data  logs = LOAD 'clickstream/logs/\*' USING PigStorage(' ') AS (reqdate:chararray, reqtime:chararray, x1:int, method:chararray, uri:chararray, x2:int, x3:int, x4:int, ipaddress:chararray, useragent:chararray);  -- filter logs that do not contain first field(reqdata) the # sign"  logs\_filtered = FILTER logs BY NOT(reqdate MATCHES '#.\*');  -- Create projection of the data  log\_filtered\_projected = FOREACH logs\_filtered GENERATE reqdate, reqtime, method, uri, ipaddress, useragent;  -- troubleshoot. dump 20 items  limit\_data = LIMIT log\_filtered\_projected 20;  DUMP limit\_data  -- store the data  STORE log\_filtered\_projected INTO 'clickstream/logs\_noheader' USING PigStorage('\t');  -- get top rows of data from HDFS using  -- hdfs dfs -cat clickstream/logs\_noheader/\* | head |

Screenshot: Dump few rows in the final file wrtoe to hdfs

Text

Description automatically generated

ANSWER 4:

|  |
| --- |
| CODE |
| CREATE DATABASE clickstream;  USE clickstream;  CREATE EXTERNAL TABLE iplookup (ip STRING,country STRING,state STRING,city STRING,approx\_lat FLOAT,approx\_lng FLOAT)  ROW FORMAT DELIMITED  FIELDS TERMINATED BY ','  STORED AS TEXTFILE  LOCATION 'hdfs://quickstart.cloudera:8020/user/cloudera/clickstream/iplookup\_noheader';  CREATE EXTERNAL TABLE weblogs (reqdate DATE, reqtime STRING, method STRING, uri STRING, ipaddress STRING, useragent STRING)  ROW FORMAT DELIMITED  FIELDS TERMINATED BY '\t'  STORED AS TEXTFILE  LOCATION 'hdfs://quickstart.cloudera:8020/user/cloudera/clickstream/logs\_noheader'; |

**Screenshot iplookup table:**

Graphical user interface

Description automatically generated with low confidence

Screenshot web\_logs table:

Text

Description automatically generated

ANSWER 5:

|  |
| --- |
| CODE |
| SELECT ip.city, count(\*) AS total\_requests  FROM weblogs wl JOIN iplookup ip  ON (wl.ipaddress == ip.ip)  GROUP BY ip.city  ORDER BY total\_requests DESC; |

**Query screenshot:**

Text

Description automatically generated with medium confidence