**Social Media Project**

**Steps to achieve the output has been mentioned in the comments along with the code.**

***PROBLEM1 –TOP 10 MOST COMMONLY USED TAGS IN THE DATASET.***

**SOLUTION CODE:**

/\*Step -1 Loaded Social Media comma seperated file\*/

StackEx\_file = LOAD '/samples/Social\_Media.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'NO\_MULTILINE', 'NOCHANGE', 'SKIP\_INPUT\_HEADER') AS (uid:int, qid:int, i:int, qs:int, qt:chararray, tags:chararray, qvc:int, qac:int, aid:int, j:int, as:int, at:chararray);

/\*Step -2 For each record in file selected only "tag" column\*/

qus\_tags = FOREACH StackEx\_file GENERATE tags;

/\*Step 3 Since each qid has multiple tags associated with it, For each record in step 2 I have splitted/tokenized the tags\*/

split\_tags = FOREACH qus\_tags GENERATE FLATTEN(TOKENIZE(tags)) as splittag;

/\*Step 4 Grouped the data in step 3 using each tag as key\*/

grouped\_tags = GROUP split\_tags BY splittag;

/\*step 5: For each key in step 4 counted number of tags associated with it\*/

tag\_count = FOREACH grouped\_tags GENERATE group,COUNT(split\_tags) as tot\_tag;

/\*Step 6 Sorted data of step 5 in desc order \*/

sort\_tag = ORDER tag\_count BY tot\_tag DESC;

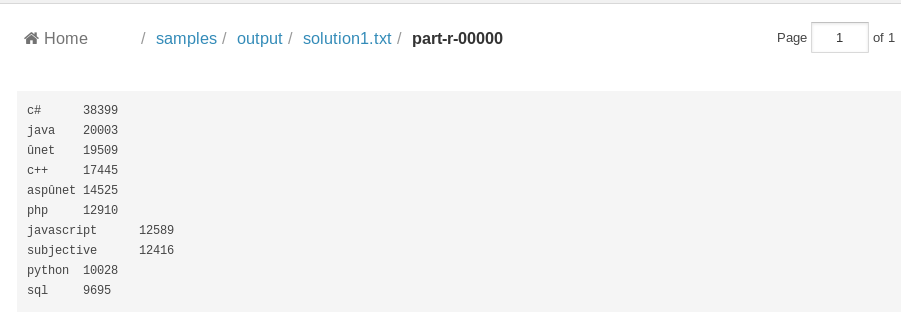
/\*Step 7 Selected only top 10 records from Step 6\*/

top\_10\_tag = limit sort\_tag 10;

/\*Stored the output in file\*/

store top\_10\_tag into '/samples/output/solution1.txt';

**Output Screenshot:**



**Output:**

c# 38399

java 20003

ûnet 19509

c++ 17445

aspûnet 14525

php 12910

javascript 12589

subjective 12416

python 10028

sql 9695

***PROBLEM 2: Average time to answer questions***

**SOLUTION CODE :**

/\*Step 1 - Loaded the file as a comma seperated values\*/

StackEx\_file = LOAD '/samples/Social\_Media.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'NO\_MULTILINE', 'NOCHANGE', 'SKIP\_INPUT\_HEADER') AS (uid:int, qid:int, i:int, qs:int, qt:long, tags:chararray, qvc:int, qac:int, aid:int, j:int, as:int, at:long);

/\*Step 2 - for each record in step 1 calculated the difference between the time at which question was posted and time at which it was answered for the first time \*/

ans\_time = FOREACH StackEx\_file GENERATE at-qt as timediff;

/\*Grouped the data in step 2 under 1 key and calculated the average using avg function in pig\*/

avg\_time = FOREACH (GROUP ans\_time ALL) GENERATE AVG(ans\_time.timediff);

/\*Stored the output in a file. Since epoch time is in seconds the final answer is in seconds\*/

store avg\_time into '/samples/output/solution2';store avg\_time into '/samples/output/solution2';

**Output Screenshot.**



**Output:**

133765.87433786143

**PROBLEM 3: NUMBER OF QUESTIONS WHICH GOT ANSWERED WITHIN ONE HOUR**

**SOLUTION CODE:**

/\*Step 1: Loaded the file as csv\*/

StackEx\_file = LOAD '/samples/Social\_Media.csv' USINGorg.apache.pig.piggybank.storage.CSVExcelStorage(',', 'NO\_MULTILINE', 'NOCHANGE', 'SKIP\_INPUT\_HEADER') AS (uid:int, qid:int, i:int, qs:int, qt:long, tags:chararray, qvc:int, qac:int, aid:int, j:int, as:int, at:long);

/\*Step 2:Filtered records from the file where the time difference between question posted and answer time was less than or equal to 1 hr. Since epoch time is in seconds i have compared with 3600 sec which is equivalent to 1 hr.\*/

flt\_rec = FILTER StackEx\_file BY (at-qt) <= 3600;

/\* step 3: Grouped the records in step 2 by question id\*/

grp\_rec = GROUP flt\_rec BY qid;

/\*Step 4: In step 3 we might get duplicates as there can be multiple answers given for a single question within an hour. Hence selecting only 1 record for each qid.\*/

rem\_dup = FOREACH grp\_rec { top\_rec = LIMIT flt\_rec 1;

GENERATE flatten(top\_rec); };

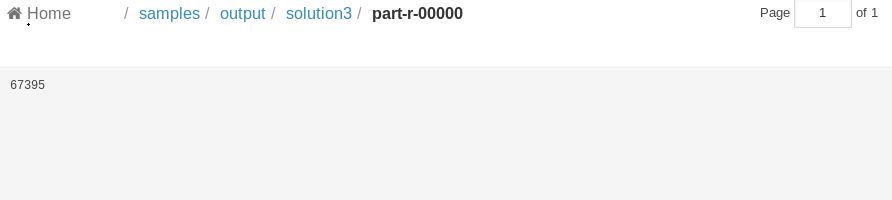
/\*Step 5: Counted number of records in step 4\*/

count\_rec = FOREACH (GROUP rem\_dup ALL) GENERATE COUNT(rem\_dup.qid) ;

/\*Step 6: stored the output in a file\*/

store count\_rec into '/samples/output/solution3';

**Output Screenshot:**



**Output:**

67395

***PROBLEM 4: TAGS OF QUESTIONS THAT GOT ANSWERED WITHIN AN HOUR.***

**SOLUTION CODE:**

/\*Step 1: Loaded the file as csv\*/

StackEx\_file = LOAD '/samples/Social\_Media.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'NO\_MULTILINE', 'NOCHANGE', 'SKIP\_INPUT\_HEADER')

AS (uid:int, qid:int, i:int, qs:int, qt:long, tags:chararray, qvc:int, qac:int, aid:int, j:int, as:int, at:long);

/\*Step 2: Filtered records from the file where the time difference between question posted and answer time was less than or equal to 1 hr. Since epoch time is in seconds i have compared with 3600 sec which is equivalent to 1 hr.\*/

flt\_rec = FILTER StackEx\_file BY (at-qt) <= 3600;

/\* step 3: Grouped the records in step 2 by question id\*/

grp\_rec = GROUP flt\_rec BY qid;

/\*Step 4: In step 3 we might get duplicates as there can be multiple answers given for a single question within an hour. Hence selecting only 1 record for each qid.\*/

rem\_dup = FOREACH grp\_rec {

top\_rec = LIMIT flt\_rec 1;

GENERATE flatten(top\_rec);

};

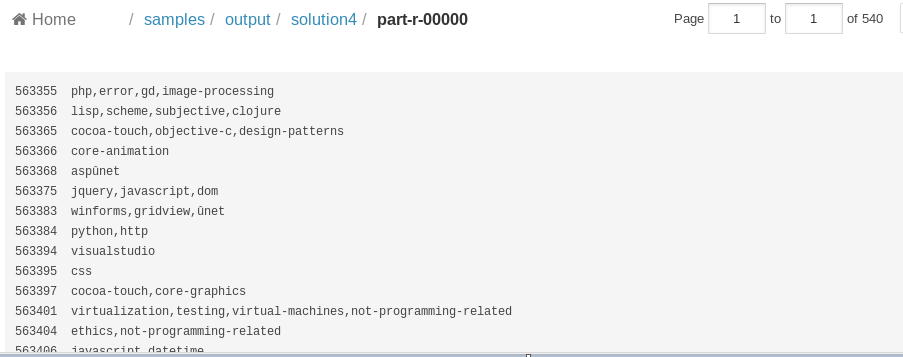
/\*For each record in step 4 i have selected qid and tags associated with that question\*/

tag\_rec = FOREACH rem\_dup GENERATE qid,tags;

/\*stored the output in a file\*/

store tag\_rec into '/samples/output/solution4';

**Output Screesnhot:**



**Since the output is big I have attached the output file below:**

****