**Experiment-7**

**Install and Run Pig then write Pig Latin scripts to sort, group, join, project, and filter your data.**

**7(a) Install and Run Pig on wordcount**

**7(b) Write Pig Latin scripts to sort, group, join, project, and filter your data.**

Phase:-1 Installation of Pig

1. Open cloudera on virtual box
2. Open browser
3. Click on Hue- Web application developed by cloudera
4. Login with use id and password as cloudera
5. Click on Manage HDFS
6. By default in user/cloudera folder
7. Create folder as Sec-A/Pig
8. Create file WordCount.txt- save it
9. Open terminal – write pig on command prompt to start pig

Phase: 2

1. Working with Grunt shell
2. Create word count application
3. Execute word count application
4. Accessing HDFS from grunt shell

**Step 1 : Start Grunt shell.**

    Open terminal and type pig

**Step 1A : Create a file at /user/cloudera/Sec-A/Pig/WordCount.txt with following content.**

I am learning Pig Using HadoopExam

I am learning Spark Using HadoopExam

I am learning Java Using HadoopExam

I am learning Hadoop Using HadoopExam

**Step 2 : Now load the file stored in hdfs (Space separated file)**

input1 = LOAD '/user/cloudera/Sec-A/Pig/WordCount.txt' AS (f1:chararray);

DUMP input1;

(I am learning Pig Using HadoopExam)

(I am learning Spark Using HadoopExam)

(I am learning Java Using HadoopExam)

(I am learning Hadoop Using HadoopExam)

**Step 3: flatten the words in each line**

wordsInEachLine = FOREACH input1 GENERATE flatten(TOKENIZE(f1)) as word;

DUMP wordsInEachLine;

**Step 4: Group the same words**

groupedWords = group wordsInEachLine by word;

dump groupedWords;

describe groupedWords;

**Step 5 : Now do the wordcount.**

countedWords = foreach groupedWords generate group, COUNT(wordsInEachLine);

dump countedWords;

**More About PigLatin** :

* Pig scripts can be a linear workflow (As shown above in word count example)
* Pig Scripts can have branching like multiple data inputs are joined (De-normalizing) and data splitting etc.
* In Pig latin scripts , you will not find if statements and for loop (This is simply a DAG : Direct Acyclic Graph)

**Grunt** : It is a shell, where we have been writing our Pig scripts. Generally production code will be written in a separate file. But while writing we want to test our scripts with test data, hence we will be using Grunt shell for prototyping our script.

**Remember :**

* It provides Tab completion of commands (Not file name as in shell scripts)
* Ctrl+D will help you to come out of Grunt

**Dump and Store**: Pig Latin will not execute scripts until it sees Dump or Store command , as we have done in our example.

**Accessing HDFS** : You can use hdfs commands inside Grunt shell as below

*> fs -ls*