coreservlets.com – Hadoop Course

Pig Intro

In this exercise, you will have a chance to practice developing Pig scripts in Grunt. You will also develop Pig script in its own script file.

Approx. Time: 60 minutes

Perform

1. Start Pig Grunt in Hadoop/MapReduce mode.

   Load records into a bag from:

   /training/exercises/pig/input1.txt

   The file contains two columns separated by a tab; be sure to create a schema where column one is of type in and column two is of type charrarray.

   Print the schema of the bag to screen.
   Dump Records to the screen; your output should look like this:

   ```
   (1,a)
   (2,b)
   (3,c)
   (4,d)
   ```

   Create another bag but limit the number of records to 2; print the bag to screen

   Exit Pig Grunt

2. Start Pig Grunt in Hadoop/MapReduce mode. /training/exercises/pig/input2.txt contains purchase records for fruits. Group these records by fruit and display results to the screen. Your output should look something like this:

   ```
   (apple,{(5,user5,apple),(3,user3,apple),(1,user1,apple)})
   (mango,{(9,user9,mango),(8,user8,mango),(7,user7,mango),(4,user4,mango)})
   (banana,{(6,user6,banana),(2,user2,banana)})
   ```

   Count the number of purchases for each fruit. Your output should look something like this:

   ```
   (apple,3)
   (mango,4)
   (banana,2)
   ```

   Exit Pig Grunt

3. Start Pig Grunt in Hadoop/MapReduce mode. Tokenize text in /training/exercises/pig/input3.txt and display 1 token per line. Your output should like this:

```
(1please)
(2tokenize)
(3and)
(4then)
(5flatten)
(6this)
(7text)
```

   Exit Pig Grunt

4. Implement and test pig script called *`MostOccuredTokens.pig`* which calculates the 5 most occurring tokens in `/training/data/hamlet.txt` text file. The script shall persist results to `/training/playArea/pig/mostOccuredTokens/` on HDFS. The script should be executed via command line:

```
$ cd $PLAY_AREA/pig/scripts
$ pig MostOccuredTokens.pig
```

   The result should look something like this:

```
$ hdfs dfs -cat /training/playArea/pig/mostOccuredTokens/part-r-00000
          the  970
          and  715
          of   667
          to   634
          I    535
```

   Implement the script in the Exercise project under

   `src/main/resources/pig/`

   Eclipse maven plugin will automatically copy the script under

   `$PLAY_AREA/pig/scripts/`

   If eclipse fails to automatically copy the script you can always execute `mvn package` command on Exercises project.

   HINT: You can use Grunt to develop your script and then capture all the statements in a single script

   HINT: You can create an intermediate bag that has a limited number of results (use LIMIT operator) and then dump the contents to the screen

   HINT: Don’t forget semicolons

Solution

1. Execute the following commands:

   ```
   $ pig
   grunt> records = LOAD '/training/exercises/pig/input1.txt' as (id:int, letter:chararray);
   grunt> describe records
   records: {id: int,letter: chararray}
   grunt> dump records
     (1,a)
     (2,b)
     (3,c)
     (4,d)
   grunt> lRecords = LIMIT records 2;
   grunt> dump lRecords
     (1,a)
     (2,b)
   grunt> quit
   ```

2. Execute the following commands:

   ```
   $ pig
   grunt> records = LOAD '/training/exercises/pig/input2.txt' as (id:int, user:chararray,
   fruit:chararray);
   grunt> byFruit = GROUP records BY fruit;
   grunt> dump byFruit;
     (apple,{(5,user5,apple),(3,user3,apple),(1,user1,apple)})
     (mango,{(9,user9,mango),(8,user8,mango),(7,user7,mango),(4,user4,mango)})
     (banana,{(6,user6,banana),(2,user2,banana)})
   grunt> numSoldByFruit = FOREACH byFruit GENERATE group, COUNT(records);
   grunt> dump numSoldByFruit;
     (apple,3)
     (mango,4)
     (banana,2)
   grunt> quit
   ```

3. Execute the following commands:

   ```
   grunt> linesOfText = LOAD '/training/exercises/pig/input3.txt' as (line:chararray);
   grunt> tokenBag = FOREACH linesOfText GENERATE TOKENIZE(line);
   grunt> dump tokenBag;
     ({(1please),(2tokenize),(3and)})
     ({(4then),(5flatten),(6this),(7text)})
   grunt> flatBag = FOREACH tokenBag GENERATE flatten($0);
   grunt> dump flatBag;
     (1please)
     (2tokenize)
     (3and)
     (4then)
     (5flatten)
     (6this)
     (7text)
   ```

4. The solutions script is located in the Solutions project

   `src/main/resources/pig/MostOccuredTokens.pig`

   To execute:

   ```
   $ cd $PLAY_AREA/pig/scripts-solutions
   $ pig MostOccuredTokens.pig
   ```