Hadoop Developer Training – Lab Hand Book

Pig Lab: Export and Import of data

Directory Name	Description
Downloads	Contains all Installable for Hadoop, Hive and Pig
Lab	For all lab activities
Lab/hdfs	For configuring hdfs related contents
Lab/mapred	For configuring mapred related contents
Lab/software	Folder for installing Hadoop, Hive, Pig and Sqoop
Lab/data	Input files for Lab Exercises
Lab/programs	For all Map Reduce Programs

Pig configuration

- **untar Pig Jar file**
  - Go to lab/software
  - Untar Pig files into software folder
    tar –xvf../../downloads/ pig-0.9.2.tar

- **Set up .bash_profile**

  - Open .bash_profile file under home directory/ **home/notroot**
    Enter the following settings
    Export PIG_INSTALL=/home/notroot/lab/software/pig-x.y.z
    Export PATH=$PATH:$PIG_INSTALL/bin

  - Save and exit .bash_profile

  - Run following command
    . .bash_profile

  - Verify whether variable are defined or not by typing export at command prompt

- **Set the following values in the $install-folder/conf/pig.properties file**

  fs.default.name=hdfs://localhost/
  mapred.job.tracker=localhost:8021

- **Check if Pig is running**

  - **Run pig and verify if enters pig grunt shell**

    pig

Lab 11: Pig Programming

A. Load Customer records

```
cust = LOAD 'input/custs' using PigStorage(',') AS ( custid:chararray,
firstname:chararray, lastname:chararray, age:long, profession:chararray);
```

B. Select only 100 records

```
amt = LIMIT cust 100;
dump amt;
```

C. Group customer records by profession

```
groupbyprofession = GROUP cust BY profession;
```

D. Count no of customers by profession

```
countbyprofession = FOREACH groupbyprofession GENERATE group, COUNT ( cust
);
dump countbyprofession;
```

E. Load transaction records

```
txn = LOAD 'input/txns' using PigStorage(',') AS ( txnid:chararray, date:chararray,
custid:chararray, amount:double, category:chararray, product:chararray,
city:chararray, state:chararray, type:chararray);
```

F. Group transactions by customer

```
txnbycust = group txn by custid;
```

G. Sum total amount spent by each customer

```
spendbycust = foreach txnbycust generate group, SUM( txn.amount );
```

H. Order the customer records beginning from highest spender

```
custorder = order spendbycust by $1 desc;
```

I. Select only top 100 customers

```
top100cust = limit custorder 100;
```

J. Join the transactions with customer details

```
top100join = join top100cust by $0, cust by $0;
describe top100join;
```

K. Select the required fields from the join for final output

```
top100 = foreach top100join generate $0, $3, $4, $5, $6, $1;
describe top100;
```

L. Dump the final output

```
dump top100;
```