Load local file to pig bag

cust = load '/home/hduser/Downloads/custs.txt' using PigStorage(',') AS (custid,firstname,lastname,age:long,profession);

---------------------------------------------------------------------------------------------------------------------

describe bag structure

describe cust;

cust: {custid: bytearray,firstname: bytearray,lastname: bytearray,age: long,profession: bytearray}

---------------------------------------------------------------------------------------------------------------------

Display bag data

dump custs;

---------------------------------------------------------------------------------------------------------------------

Specific criteria data

teacher\_bag = filter cust by profession=='Teacher';

teacher\_bag: {custid: bytearray,firstname: bytearray,lastname: bytearray,age: long,profession: bytearray}

---------------------------------------------------------------------------------------------------------------------

select limited record

amt = limit cust 10;

describe amt;

amt: {custid: bytearray,firstname: bytearray,lastname: bytearray,age: long,profession: bytearray}

---------------------------------------------------------------------------------------------------------------------

Group bag by some column

GroupByProfession = GROUP cust by profession;

describe GroupByProfession;

GroupByProfession: {group: bytearray,cust: {(custid: bytearray,firstname: bytearray,lastname: bytearray,age: long,profession: bytearray)}}

--Count on group by data

countbyprofession = FOREACH GroupByProfession GENERATE group as profession, COUNT(cust) as headcount;

---------------------------------------------------------------------------------------------------------------------

--order By Clause on grouped data

orderbycount = order countbyprofession by $1 desc;

$1 second column.

---------------------------------------------------------------------------------------------------------------------

--Store data into local fileSystem

store orderbycount into '/home/hduser/Downloads/pigdump' using PigStorage(',');

---------------------------------------------------------------------------------------------------------------------

--Ranking - Returns each tuple with the rank within a relation.

cust = load '/home/hduser/Downloads/custs.txt' using PigStorage(',') as (custid, firstname, lastname, age:long, profession);

GroupByProfession = GROUP cust by profession;

countbyprofession = FOREACH GroupByProfession GENERATE group as profession, COUNT(cust) as headcount;

--describe countbyprofession;

--countbyprofession: {profession: bytearray,headcount: long}

ranked = rank countbyprofession;

--describe ranked;

--ranked: {rank\_countbyprofession: long,profession: bytearray,headcount: long}

filterrank = filter ranked by rank\_countbyprofession >10 and rank\_countbyprofession < 20;

(11,Writer,95)

(12,Athlete,196)

(13,Chemist,206)

(14,Teacher,189)

(15,Designer,204)

(16,Musician,204)

(17,Reporter,199)

(18,Architect,202)

(19,Carpenter,180)

---------------------------------------------------------------------------------------------------------------------

**Example 1** – Find top 20 customer who have made the maximum total purchase

**Sample Record -**

00000000,06-26-2011,4007024,040.33,Exercise & Fitness,Cardio Machine Accessories,Clarksville,Tennessee,credit

loading

txn = load '/home/hduser/Downloads/txns1.txt' using PigStorage(',') as (txnid,date,custid,amount:double,category,product,city,state,type);

Grouping

groupbycustomer = GROUP txn by custid;

Calculate total sale by each customer

salecount = FOREACH groupbycustomer GENERATE group AS custid, ROUND\_TO( SUM(txn.amount),2) as Total\_Sale;

Ordering by desc

orderbysalecount = order salecount by $1 desc;

Fetching top 20 Records

top20 = limit orderbysalecount 10;

Join customer and Transaction

Top20Cuatomer = JOIN top20 by $0, cust by $0;

top20 = FOREACH Top20Cuatomer GENERATE $0,$3,$4,$5,$6,$1;

---------------------------------------------------------------------------------------------------------------------

**Example 2** Find Total sales, total cash sales with % and total credit sales with %.

**Sample Record -**

00000000,06-26-2011,4007024,040.33,Exercise & Fitness,Cardio Machine Accessories,Clarksville,Tennessee,credit

txn = load '/home/hduser/Downloads/txns1.txt' using PigStorage(',') as (txnid,date,custid,amount:double,category,product,city,state,type);

groupbytype = GROUP txn by type;

totalbytype = FOREACH groupbytype generate group as Type, SUM(txn.amount) as Total;

groupall = GROUP totalbytype all;

totalsale = FOREACH groupall generate ROUND\_TO(SUM(totalbytype.Total),2) as TotalAmount;

final = FOREACH totalbytype generate $0, $1 , ROUND\_TO(($1 \* 100 )/totalsale.TotalAmount,2);

---------------------------------------------------------------------------------------------------------------------

**Example 3** Track customer whose age is less than 50 and total purchase done more than USD 500.

**Sample Record –** Transaction

00000000,06-26-2011,4007024,040.33,Exercise & Fitness,Cardio Machine Accessories,Clarksville,Tennessee,credit

**Sample Record –** Customer

4000001,Kristina,Chung,55,Pilot

Load both tables

txn = load '/home/hduser/Downloads/txns1.txt' using PigStorage(',') as (txnid,date,custid,amount:double,category,product,city,state,type);

cust = load '/home/hduser/Downloads/custs.txt' using PigStorage(',') as (custid, firstname, lastname, age:long, profession);

groupbyid = GROUP txn by custid;

totalamt = FOREACH groupbyid generate group, ROUND\_TO(SUM(txn.amount),2) as Total;

finaltxn = filter totalamt by Total > 500;

finalcust = filter cust by age < 50;

txnjoincust = JOIN finaltxn by $0, finalcust by $0;

final = FOREACH txnjoincust generate $0, $3, $4, $5, $6, $1;

store final into '/home/hduser/Downloads/custtxn' using PigStorage(',');

--store final into '/home/hduser/Downloads/custtxn' using BinStorage();

pig -x local custtxn.pig ///to run pig script

--------------------------------------------------------------------------------------------

**Example 3** Analyze monthly retail trade report for the US market

A US‐based online retailer wants to launch a new product category and wants to understand the potential growth areas and areas that have stagnated over a period of time. It wants to use this information to ensure its product focus is aligned to opportunities that will grow over the next 5–7 years. The customer has also provided pointers to the data set you can use. The following are the goals for the project:

**Sample Record**

4411,Automobile and other motor vehicle dealers,62306,63801,63027,60592,60492,61345,59995,60075,61360,61017,59479,58207

bag2000 = load '/home/hduser/Downloads/Pig Retail/2000.txt' using PigStorage(',') as (id, name, jan:double, feb:double, mar:double, apr:double, may:double, jun:double, jul:double, aug:double, sep:double, oct:double, nov:double, dec:double);

bag2001 = load '/home/hduser/Downloads/Pig Retail/2001.txt' using PigStorage(',') as (id, name, jan:double, feb:double, mar:double, apr:double, may:double, jun:double, jul:double, aug:double, sep:double, oct:double, nov:double, dec:double);

bag2002 = load '/home/hduser/Downloads/Pig Retail/2002.txt' using PigStorage(',') as (id, name, jan:double, feb:double, mar:double, apr:double, may:double, jun:double, jul:double, aug:double, sep:double, oct:double, nov:double, dec:double);

bag2000 = FOREACH bag2000 generate $0,$1,($2+$3+$4+$5+$6+$7+$8+$9+$10+$11+$12+$13);

bag2001 = FOREACH bag2001 generat$0,$1,($2+$3+$4+$5+$6+$7+$8+$9+$10+$11+$12+$13);

bag2002 = FOREACH bag2002 generate $0,$1,($2+$3+$4+$5+$6+$7+$8+$9+$10+$11+$12+$13);

finalbag = join bag2000 by $0, bag2001 by $0, bag2002 by $0;

final = FOREACH finalbag generate $0, $1, $2, $5, $8;

growth = FOREACH final generate $0,$1,$2,$3,$4,ROUND\_TO(($3-$2)\*100/$2,2) as f\_cycle, ROUND\_TO(($4-$3)\*100/$3,2) as s\_cycle;

growthfinal = FOREACH growth generate $0,$1,$2,$3,$4,$5,$6, ROUND\_TO(($5 + $6)/2,2) as avggrowthrate;

**‐ Analyze the entire data set and arrive at products that have experienced a consolidated yearly avg growth of 10% or more in sales since 2000**.

outputfinal = filter growthfinal by avggrowthrate >= 10;

**‐ Analyze the entire data set and arrive at products that have experienced a consolidated yearly avg drop of 5% or more since 2000..**

outputfinal1 = filter growthfinal by avggrowthrate <= -5;

**- Find top 5 products and bottom 5 products of overall sales for 3 years.**

top = order growthfinal by avggrowthrate desc;

top5 = limit top 5;

bottom = order growthfinal by avggrowthrate;

bottom5 = limit bottom 5;

---------------------------------------------------------------------------------------------------------------------

**Example 4** Find the list of authors whose name starts with ‘J’ and price of their book is greater than $200.

**Sample Record**

File 1

100,200,10

File 1

10,John

book\_info = load ‘Book\_info.txt’ using PigStorage(‘,’) as (Book\_Id:int, Price:int, Author\_ID:int);

book\_info\_filtered = filter book\_info by Price >200;

author\_info = load ‘Author\_info.txt’ using PigStorage(‘,’) as (Author\_ID:int, Author\_name);

autho**Sample Record**

File 1r\_info\_filtered =FILTER author\_info by INDEXOF(Author\_name,’J’,0)==0;

book\_author\_info = join book\_info\_filtered by Author\_ID, author\_info\_filtered by Author\_ID;

STORE book\_author\_info into ‘/store\_info’ using PigStorage(‘,’);

---------------------------------------------------------------------------------------------------------------------

**Example 5** Find word count using pig script.

-- Command to run through local mode using parameter

-- pig -x local -p input=/home/hduser/Downloads/wordCount.txt wordcount.pig;

-- pig -x local -p input=/home/hduser/Downloads/wordCount.txt -f wordcount.pig;

-- pig -x local -p input=/home/hduser/Downloads/wordCount.txt -p output=/home/hduser/Downloads/Count.txt wordcount.pig;

-- pig -x local -p input=/home/hduser/Downloads/wordCount.txt -p myword=anita wordcount.pig;

--command to run in hdfs mode

-- pig -p input=/home/hduser/Downloads/wordCount.txt -f wordcount.pig

--command to run in local mode with parameter file

-- pig -x local -param\_file /home/hduser/Downloads/parameter.txt wordcount.pig

word = load '$input' using TextLoader() as (lines:chararray);

--transform = foreach word generate TOKENIZE(lines) as word;

--({(Anita),(Hyderabad),(Smita)})

--transform: {word: {tuple\_of\_tokens: (token: chararray)}}

transform = foreach word generate FLATTEN(TOKENIZE(lines)) as word;

--(Anita)

--(Hyderabad)

--(Smita)

--transform: {word: chararray}

transform = foreach transform generate TRIM(LOWER(REPLACE(word, '[\\p{Punct},\\p{Cntrl}]',''))) as word;

--if need to find specific word

--transform = FILTER transform by word == '$myword';

groupbyword = group transform by word;

--(anita,{(anita),(anita),(anita),(anita),(anita)})

--groupbyword: {group: chararray,transform: {(word: chararray)}}

countofword = FOREACH groupbyword generate group, COUNT(transform);

--(anita,5)

--countofword: {group: chararray,long}

dump countofword;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

--illustrate transform;

The **illustrate** operator gives you the step-by-step execution of a sequence of statements.

-----------------------------------------------

| word | lines:chararray |

------------------------------------------------

| | Rajesh Pune |

------------------------------------------------

------------------------------------------------

| transform | word:chararray |

------------------------------------------------

| | Rajesh |

| | Pune |

------------------------------------------------

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Example 5** Calculate average medical claim by each user.

**Sample Record**

amy hr 8000

jack hr 7500

medical = load '/home/hduser/Downloads/medical.txt' using PigStorage(',') as (Name,Dept,claim:double);

groupbyname = GROUP medical by Name;

--(tim,{(tim,TS,2750.0),(tim,TS,3500.0),(tim,TS,4750.0)})

--(jack,{(jack,hr,7500.0)})

avgclaim = FOREACH groupbyname generate group, AVG(medical.claim);