### **Pig Latin Basics:**

Assume the people\_data.csv file has the following structure:

id,name,age,city\_id

1,John,28,101

2,Alice,22,102

3,Bob,35,101

4,Charlie,30,103

And the cities\_data.csv file looks like:

city\_id,city\_name

101,New York

102,San Francisco

103,Los Angeles

**1. Loading Data:**

-- Load people data

people\_data = LOAD 'people\_data.csv' USING PigStorage(',') AS (id:int, name:chararray, age:int, city\_id:int);

-- Load cities data

cities\_data = LOAD 'cities\_data.csv' USING PigStorage(',') AS (city\_id:int, city\_name:chararray);

**2. Filtering Data:**

-- Filter people aged 30 and above

filtered\_people = FILTER people\_data BY age >= 30;

-- Filter cities with names containing 'Los'

filtered\_cities = FILTER cities\_data BY city\_name matches '.\*Los.\*';

**3. Grouping Data:**

-- Group people by city\_id

grouped\_people = GROUP people\_data BY city\_id;

**4. Joining Data:**

-- Join people and cities data on city\_id

joined\_data = JOIN people\_data BY city\_id, cities\_data BY city\_id;

**5. Storing Data:**

-- Store the filtered people data

STORE filtered\_people INTO 'output/filtered\_people';

-- Store the joined data

STORE joined\_data INTO 'output/joined\_data';

### **Pig Data Types:**

Pig supports various data types. Here's a quick overview:

* **int:** Integer
* **long:** Long integer
* **float:** Single-precision floating-point
* **double:** Double-precision floating-point
* **chararray:** Character array (string)
* **bytearray:** Binary data
* **boolean:** Boolean

### **Download the Data**:

For this example, let's consider a simple dataset named sample\_data.csv with the following structure:

name,age,city,salary

John,25,New York,50000

Alice,30,San Francisco,60000

Bob,28,Los Angeles,55000

Charlie,35,Chicago,70000

You can download this data or use your own dataset.

### **Create Your Pig Latin Script**:

Let's create a Pig Latin script (example\_script.pig) that loads the data, performs some transformations, and stores the results.

-- Load data

data = LOAD 'sample\_data.csv' USING PigStorage(',') AS (name:chararray, age:int, city:chararray, salary:double);

-- Filter data for people aged 30 and above

filtered\_data = FILTER data BY age >= 30;

-- Calculate average salary

grouped\_data = GROUP filtered\_data ALL;

avg\_salary = FOREACH grouped\_data GENERATE AVG(filtered\_data.salary) AS avg\_salary;

-- Display the results

DUMP avg\_salary;

-- Store the filtered data

STORE filtered\_data INTO 'output/filtered\_data';

### **Save and Execute the Script:**

Save the script file (example\_script.pig). To execute it, use the following command:

pig example\_script.pig

### **Diagnostic Operators:**

#### input\_data.csv:

csv

name,age,city

John,28,New York

Alice,22,San Francisco

Bob,35,Los Angeles

Charlie,30,Chicago

#### people\_data.csv:

csv

id,name,city\_id

1,John,101

2,Alice,102

3,Bob,103

4,Charlie,101

#### cities\_data.csv:

csv

city\_id,city\_name

101,New York

102,San Francisco

103,Los Angeles

104,Chicago

#### input\_data\_A.csv:

csv

id,name

101,John

102,Alice

103,Bob

104,Charlie

#### input\_data\_B.csv:

csv

id,city

105,New York

106,San Francisco

107,Los Angeles

108,Chicago

-- Diagnostic Operators example

A = LOAD 'input\_data.csv' USING PigStorage(',') AS (name:chararray, age:int, city:chararray);

-- Display the contents of relation A

DUMP A;

-- Show the schema of relation A

DESCRIBE A;

### **Grouping and Joining:**

-- Grouping and Joining example

people = LOAD 'people\_data.csv' USING PigStorage(',') AS (id:int, name:chararray, city\_id:int);

cities = LOAD 'cities\_data.csv' USING PigStorage(',') AS (city\_id:int, city\_name:chararray);

-- Group people by city\_id

grouped\_people = GROUP people BY city\_id;

-- Join people and cities data on city\_id

joined\_data = JOIN grouped\_people BY group, cities BY city\_id;

-- Display the joined data

DUMP joined\_data;

### **Combining & Splitting:**

-- Combining & Splitting example

A = LOAD 'input\_data\_A.csv' USING PigStorage(',') AS (id:int, name:chararray);

B = LOAD 'input\_data\_B.csv' USING PigStorage(',') AS (id:int, city:chararray);

-- Combining

C = UNION A, B;

-- Splitting

SPLIT C INTO D IF id < 100, E IF id >= 100;

-- Display the split data

DUMP D;

DUMP E;

### **Filtering**:

-- Filtering example

data = LOAD 'input\_data.csv' USING PigStorage(',') AS (name:chararray, age:int, city:chararray);

-- Filter data for people aged 25 and above

filtered\_data = FILTER data BY age >= 25;

-- Display the filtered data

DUMP filtered\_data;

### **Sorting:**

-- Sorting example

data = LOAD 'input\_data.csv' USING PigStorage(',') AS (name:chararray, age:int, city:chararray);

-- Sort data by age in descending order

sorted\_data = ORDER data BY age DESC;

-- Display the sorted data

DUMP sorted\_data;