**Pig Components**

**Field:** Each piece of data is called field.

**Tuple:** Ordered set of fields, represented by "(" and ")".

**Bag:** Collection of tuple, represented by "{" and "}". Form RDBMS point of view, Bag is a table.

1. { (Sonoo, javatpoint, word, 4, 456.0), (yahoo, 1, happen), (1,esplanade,45.4) }

Pig Data Types

Apache Pig supports many data types. A list of Apache Pig Data Types with description and examples are given below.

|  |  |  |
| --- | --- | --- |
| **Type** | **Description** | **Example** |
| Int | Signed 32 bit integer | 2 |
| Long | Signed 64 bit integer | 15L or 15l |
| Float | 32 bit floating point | 2.5f or 2.5F |
| Double | 32 bit floating point | 1.5 or 1.5e2 or 1.5E2 |
| charArray | Character array | hello javatpoint |
| byteArray | BLOB(Byte array) |  |
| tuple | Ordered set of fields | (12,43) |
| bag | Collection f tuples | {(12,43),(54,28)} |
| map | collection of tuples | [open#apache] |

# Pig Example

**Use case:** Using Pig find the most occurred start letter.

**Solution:**

**Case 1:** Load the data into bag named "lines". The entire line is stuck to element line of type character array.

1. grunt> lines  = LOAD "/user/Desktop/data.txt" AS (line: chararray);

**Case 2:** The text in the bag lines needs to be tokenized this produces one word per row.

1. grunt>tokens = FOREACH lines GENERATE flatten(TOKENIZE(line))   As token: chararray

**Case 3:** To retain the first letter of each word type the below command .This commands uses substring method to take the first character.

1. grunt>letters = FOREACH tokens  GENERATE SUBSTRING(0,1)   as letter : chararray;

**Case 4:** Create a bag for unique character where the grouped bag will contain the same character for each occurrence of that character.

1. grunt>lettergrp = GROUP letters by letter;

**Case 5:** The number of occurrence is counted in each group.

1. grunt>countletter  = FOREACH  lettergrp  GENERATE group  , COUNT(letters);

**Case 6:** Arrange the output according to count in descending order using the commands below.

1. grunt>OrderCnt = ORDER countletter  BY  $1  DESC;

**Case 7:** Limit to One to give the result.

1. grunt> result  =LIMIT    OrderCnt    1;

**Case 8:** Store the result in HDFS . The result is saved in output directory under sonoo folder.

1. grunt> STORE   result   into 'home/sonoo/output';