Pig Latin

Case Sensitivity

- The names of relations and fields are case sensitive.
- The names of Pig Latin functions are case sensitive.
- All other Pig Latin keywords are case insensitive.

Case Sensitivity

In the example below

- The names of relations A, B, and C are case sensitive.
- The names of fields f1, f2, and f3 are case sensitive.
- Function names PigStorage and COUNT are case sensitive.
- Keywords LOAD, USING, AS, GROUP, BY, FOREACH, GENERATE, and DUMP are case insensitive. They can also be written as load, using, as, group, by, etc.
- In the FOREACH statement, the field in relation B is referred to by positional notation ($0).
- grunt> A = LOAD 'data' USING PigStorage() AS (f1:int, f2:int, f3:int);
- grunt> B = GROUP A BY f1;
- grunt> C = FOREACH B GENERATE COUNT ($0);
- grunt> DUMP C;

Identifiers

- In Pig, identifiers start with a letter and can be followed by any number of letters, digits, or underscores.

Valid identifiers:

- A A123 abc_123_BeX_

Invalid identifiers:

- _A123 abc_$ A!B

Pig Relation

- A Pig relation is a bag of tuples. A Pig relation is similar to a table in a relational database, where the tuples in the bag correspond to the rows in a table.
- Unlike a relational table, however, Pig relations don't require that every tuple contain the same number of fields or that the fields in the same position (column) have the same type.
- Also note that relations are unordered which means there is no guarantee that tuples are processed in any particular order.

Referencing Relations

- Relations are referred to by name (or alias). Names are assigned by you as part of the Pig Latin statement.
- In the example the name of the relation is A.

```
A = LOAD 'student' USING PigStorage() AS
(name:chararray, age:int, gpa:float);
 DUMP A;
 (John,18,4.0F)
 (Mary,19,3.8F)
 (Bill,20,3.9F)
 (Joe,18,3.8F)
```

Referencing Relations -aliases

- You an assign an alias to another alias. The new alias can be used in the place of the original alias to refer the original relation.

```
A = LOAD 'student' USING PigStorage() AS (name:chararray, age:int, gpa:float);
B = A;
DUMP B;
```

Referencing Fields –Positional Argument

- Fields are referred to by positional notation or by name .
- Positional notation is generated by the system. Positional notation is indicated with the dollar sign ($) and begins with zero (0); for example, $0, $1, $2.
- Names are assigned by you using schemas (or, in the case of the GROUP operator and some functions, by the system)

Positional Argument Example

- A = LOAD 'student' USING PigStorage() AS (name:chararray, age:int, gpa:float);
- X = FOREACH A GENERATE name,$2;
- DUMP X;
- (John,4.0F)
- (Mary,3.8F)
- (Bill,3.9F)
- (Joe,3.8F)

Complex Types in Pig

1	Tuple	A tuple is an ordered set of fields. **Example** : (raja, 30)
2	Bag	A bag is a collection of tuples. **Example** : {(raju,30),(Mohhammad,45)}
3	Map	A Map is a set of key-value pairs. **Example** : [‘name’#’Raju’, ‘age’#30]