1. Does Pig differ from MapReduce? If yes, how?

Yes, Pig differs from MapReduce because, in MapReduce, the group by operation is performed at reducer side and filter, and also in the map phase the projection is implemented. Pig Latin provides the operations that are similar to MapReduce, such as groupby, orderby, and filters. We can analyze the Pig script and data flow to find the error checking. Pig Latin is lower in cost to write and maintain compared to MapReduce Java code.

2. Explain the uses of Map Reduce in Pig.

- Apache Pig programs are written in Pig Latin query language which is similar to the SQL query language. To execute this queries, there requires an execution engine. The Pig engine enables to convert the queries into MapReduce jobs and thus MapReduce acts as the execution engine and is designed to run the programs as per the requirements.
- Pigs' operators are using Hadoops' API depending upon the configurations the job is executed in local mode or Hadoop cluster. Pig is never passes any outputs to Hadoop instead set the inputs and data locations for map-reduce.
- Pig Latin provides a set of standard Data-processing operations, such as join, filter, group by, order by, union, etc which are mapped to do the map-reduce tasks. A Pig Latin script describes a (DAG) directed acyclic graph, where the edges are data flows and the nodes are operators that process the data.

3. Explain the uses of PIG.

We can use Pig in three categories, they are

1. ETL data pipeline : It helps to populate our data warehouse. Pig can pipeline the data to an external application, it will wait until it's finished, so that it has receive the processed data and continue from there. It is the most common use case for Pig.
2. Research on raw data.
3. Iterative processing.

4. Name the scalar data type and complex data types in Pig.

The scalar data types in pig are int, float, double, long, chararray, and bytearray.

The complex data types in Pig are map, tuple, and bag.

Map: The data element with the data type chararray where element has pig data type include complex data type

Example- [city'#'bang','pin'#560001]

In this city and pin are data element mapping to values.

Tuple : It is a collection of data types and it has fixed length. Tuple is having multiple fields and these are ordered.

Bag : It is a collection of tuples, but it is unordered, tuples in the bag are separated by comma

Example: {('Bangalore', 560001),('Mysore',570001),('Mumbai',400001)

5. State the usage of ‘filters’, ‘group’ , ‘orderBy’, ‘distinct’ keywords in pig scripts.

Filters : Filters has the similar functionality as where clause in SQL. Filters contain predicate and if it evaluates true for a given record, then that record will be passed down the pipeline. Otherwise, it will not predicate the results and thus contains different operators like ==,>=, <=,!=.so,== and != which is been applied in creating maps and tuples.

A= load ‘inputs’ as (name,address)
B=filter A by symbol matches ‘CM.*';

GroupBy : The group statement collects various records with the same key. In SQL database GroupBy creates a group which feeds directly to one or more aggregate functions. But in Pig Latin has no direct connection between group and aggregate functions.

Input 2 = load ‘daily’ as(exchanges,stocks);
grpds = group input2 by stocks;

Order : The Order statement sorts the data producing a total order of output data. The Order syntax is similar to Group. Give a key or set of keys to order your data as per requirement. The following are the examples for the same:

Input 2 = load ‘daily’ as(exchanges,stocks);
grpds = order input2 by exchanges;

Distinct : The distinct statement is very simple to understand and implement. It removes duplicate records and the original data will be secured. It is implemented only on entire records, not on individual fields. Consider the below examples which explains the same:

Input 2 = load ‘daily’ as(exchanges,stocks);
grpds = distinct exchanges;

6. Explain the LOAD keyword in Pig script.

Load helps to load data from the file system. It is a relational operator

In the first step in data-flow language we need to mention the input, which is completed by using ‘load’ keyword.

The LOAD syntax is

LOAD ‘mydata’ [USING function] [AS schema];
Example- A = LOAD ‘intellipaat.txt’;
A = LOAD ‘intellipaat.txt’ USINGPigStorage(‘\t’);

7. What are the relation operations in Pig? Explain any two with examples.

The relational operations in Pig:

foreach, order by, filters, group, distinct, join, limit.foreach: It takes a set of expressions and applies them to all records in the data pipeline to the next operator.A =LOAD ‘input’ as (emp_name :charrarray, emp_id : long, emp_add : chararray, phone : chararray, preferences : map []);B = foreach A generate emp_name, emp_id;Filters: It contains a predicate and it allows us to select which records will be retained in our data pipeline.

Syntax: alias = FILTER alias BY expression; //Alias indicates the name of the relation, By indicates required keyword and the expression has Boolean.
Example: M = FILTER N BY F5 == 4;

8. Does Pig support multi-line commands?

Yes, pig supports both single line and multi-line commands. In single line command it executes the data, but it doesn't store in the file system, but in multiple lines commands it stores the data into '/output';/* , so it can store the data in HDFS.

9. Explain different execution modes available in Pig.

Three different execution modes available in Pig they are,

1. Interactive mode or Grunt mode.
2. Batch mode or Script mode.

Embedded mode
Interactive mode or grunt mode: Pig's interactive shell is known as grunt shell. If no file is specified to run in Pig it will start.
grunt> run scriptfile.pig

3. grunt> exec scriptfile.pig
   Batch mode or Script mode : Pig executes the specified commands in the script file.
   Embedded mode : We can embed Pig programs in Java and we can run the programs from Java.

10. What are the exception handling operators in Pig script?

Following operators are used for handling the exception in pig script.

DUMP : It helps to display the results on screen.

DESCRIBE : It helps to display the schema of aparticular relation.

ILLUSTRATE : It helps to display step by step execution of a sequence of pig statements

EXPLAIN : It helps to display the execution plan for Pig Latin statements.

11. Differentiate between the physical plan and logical plan in Pig script.

Both plans are created while to execute the pig script.

Physical plan : It is a series of MapReduce jobs while creating the physical plan.It's divided into three physical operators such as Local Rearrange, Global Rearrange, and package. It illustrates the physical operators Pig will use to execute the script without referring to how they will execute in MapReduce Loading and storing functions are resolved in physical plan.

Example- A: Load(/emp:PigStorage(' '))

Logical plan : The Logical plan is a plan which is created for each line in the Pig scripts. It is produced after semantic checking and basic parsing. With every line, the logical plan for that particular program becomes extended and larger because each and every statement has its own logical plan.Loading and storing function are not resolved in logical plan.

Example: X: (Name: LOLoad schema:
emp_id#36:bytearray,emp_name#37:bytearray,city#38:bytearray,salary#39:bytearray)Required Fields:null

12. Is Pig script case sensitive?

Pig script is both case sensitive and case insensitive. For example, in user defined functions, the field name, and relations are case sensitive ,i.e., INTELLIPAAT is not same as intellipaat or M=load 'test' is not same as m=load 'test'. And Pig script keywords are case insensitive i.e., LOAD is same as a load.

13. Highlight the difference between group and Cogroup operators in Pig.

Both the operators can work with one or more relations. Group and Cogroup operators are identical. Group operator collects all records with the same key. Cogroup is a combination of group and join, it is a generalization of a group instead of collecting records of one input depends on a key, it collects records of n inputs based on a key. At a time we can Cogroup upto 127 relations.

14. What is the function of UNION and SPLIT operators? Give examples.

Union operator helps to merge the contents of two or more relations.

Syntax: grunt> Relation_name3 = UNION Relation_name1, Relation_name2
Example: grunt> INTELLIPAAT = UNION intellipaat_data1.txt intellipaat_data2.txt

SPLIT operator helps to divide the contents of two or more relations.

Syntax: grunt> SPLIT Relationa1_name INTO Relationa2_name IF (condition1), Relation2_name (condition2);
Example: SPLIT student_details into student_details1 if marks<35, student_details2 if (8590);

15. How does the Pig platform handle relational systems data?

There are two ways Pig can work with relational datasets.

1. Load relational data directly into the Hadoop framework, where Pig can access it.
2. Using database connectors, Pig can load data directly from a relational database system and we can access it.

16. What are the drawbacks of Pig?

Some of the drawbacks of Pig are:

1. Pig is not really a convenient option for real-time use cases.
2. Pig does not prove to be useful when you need to fetch single record from a huge dataset.
3. Since it works on MapReduce, it works in batches.

17. Mention the common features in Pig and Hive.

The common features in Both Hive and Pig are

1. Internally both are converted the commands into MapReduce.
2. Both the technologies provide high-level abstractions.
3. Both do not support low-latency queries.
4. Both do not support OLAP or OLTP.

18. Differentiate between Pig Latin and Pig Engine.

Pig Latin is scripting language like Perl for searching huge data sets and it is made up of a series of transformations and operations that are applied to the input data to produce data.

Pig engine is an environment to execute the Pig Latin programs. It converts Pig Latin operators into a series of MapReduce jobs.

19. What are all stats classes in the org.apache.pig.tools.pigstats package?

Stat classes are in the package

- PigStats
- JobStats
- OutputStats
- InputStats.