**PIG**

1. What is the difference between logical and physical plans?

Pig undergoes some steps when a Pig Latin Script is converted into MapReduce jobs. After performing the basic parsing and semantic checking, it produces a logical plan. The logical plan describes the logical operators that have to be executed by Pig during execution. After this, Pig produces a physical plan. The physical plan describes the physical operators that are needed to execute the script.

2. Does ‘ILLUSTRATE’ run MR job?

No, illustrate will not pull any MR, it will pull the internal data. On the console, illustrate will not do any job. It just shows output of each stage and not the final output.

3. Is the keyword ‘DEFINE’ like a function name?

Yes, the keyword ‘DEFINE’ is like a function name. Once you have registered, you have to define it. Whatever logic you have written in Java program, you have an exported jar and also a jar registered by you. Now the compiler will check the function in exported jar. When the function is not present in the library, it looks into your jar.

4. Is the keyword ‘FUNCTIONAL’ a User Defined Function (UDF)?

No, the keyword ‘FUNCTIONAL’ is not a User Defined Function (UDF). While using UDF, we have to override some functions. Certainly you have to do your job with the help of these functions only. But the keyword ‘FUNCTIONAL’ is a built-in function i.e a pre-defined function, therefore it does not work as a UDF.

5.Why do we need MapReduce during Pig programming?

Pig is a high-level platform that makes many Hadoop data analysis issues easier to execute. The language we use for this platform is: Pig Latin. A program written in Pig Latin is like a query written in SQL, where we need an execution engine to execute the query. So, when a program is written in Pig Latin, Pig compiler will convert the program into MapReduce jobs. Here, MapReduce acts as the execution engine.

6. Are there any problems which can only be solved by MapReduce and cannot be solved by PIG? In which kind of scenarios MR jobs will be more useful than PIG?

Let us take a scenario where we want to count the population in two cities. I have a data set and sensor list of different cities. I want to count the population by using one mapreduce for two cities. Let us assume that one is Bangalore and the other is Noida. So I need to consider key of Bangalore city similar to Noida through which I can bring the population data of these two cities to one reducer. The idea behind this is some how I have to instruct map reducer program – whenever you find city with the name ‘Bangalore‘ and city with the name ‘Noida’, you create the alias name which will be the common name for these two cities so that you create a common key for both the cities and it get passed to the same reducer. For this, we have to write custom partitioner.

In mapreduce when you create a ‘key’ for city, you have to consider ’city’ as the key. So, whenever the framework comes across a different city, it considers it as a different key. Hence, we need to use customized partitioner. There is a provision in mapreduce only, where you can write your custom partitioner and mention if city = bangalore or noida then pass similar hashcode. However, we cannot create custom partitioner in Pig. As Pig is not a framework, we cannot direct execution engine to customize the partitioner. In such scenarios, MapReduce works better than Pig.

7. Does Pig give any warning when there is a type mismatch or missing field?

No, Pig will not show any warning if there is no matching field or a mismatch. If you assume that Pig gives such a warning, then it is difficult to find in log file. If any mismatch is found, it assumes a null value in Pig.

8. What co-group does in Pig?

Co-group joins the data set by grouping one particular data set only. It groups the elements by their common field and then returns a set of records containing two separate bags. The first bag consists of the record of the first data set with the common data set and the second bag consists of the records of the second data set with the common data set.

9.Can we say cogroup is a group of more than 1 data set?

Cogroup is a group of one data set. But in the case of more than one data sets, cogroup will group all the data sets and join them based on the common field. Hence, we can say that cogroup is a group of more than one data set and join of that data set as well.

10. What does FOREACH do?

FOREACH is used to apply transformations to the data and to generate new data items. The name itself is indicating that for each element of a data bag, the respective action will be performed.

Syntax : FOREACH bagname GENERATE expression1, expression2, …..

The meaning of this statement is that the expressions mentioned after GENERATE will be applied to the current record of the data bag.

11. What is bag?

A bag is one of the data models present in Pig. It is an unordered collection of tuples with possible duplicates. Bags are used to store collections while grouping. The size of bag is the size of the local disk, this means that the size of the bag is limited. When the bag is full, then Pig will spill this bag into local disk and keep only some parts of the bag in memory. There is no necessity that the complete bag should fit into memory. We represent bags with “{}”.

12. what is differnce between pig and sql?  
PIG latin is procedural version of SQl.pig has certainly similarities,more difference from sql.sql is a query language for user asking question in query form.sql makes answer for given but dont tell how to answer the given question.suppose ,if user want to do multiple operations on tables,we have write maultiple queries and also use temporary table for storing,sql is support for subqueries but intermediate we have to use temporary tables,SQL users find subqueries confusing and difficult to form properly.using sub-queries creates an inside-out design where the first step in the data pipeline is the innermost query .pig is designed with a long series of data operations in mind, so there is no need to write the data pipeline in an inverted set of subqueries or to worry about storing data in temporary tables.

13. Why PIG ?  
1)Ease of programming  
2)Optimization opportunities.  
3)Extensibility

14. Advantages of Using PIG ?  
PIG can be treated as a higher level language  
1) Increases Programming Productivity

2) Decreases duplication of Effort

3) Opens the M/R Programming system to more uses

15. How PIG differs from MapReduce  
In mapreduce,groupby operation performed at reducer side and filter,projection can be implemented in the map phase.pig latin also provides standard-operation similar to mapreduce like orderby and filters,groupby..etc.we can analyze pig script and know data flows ans also early to find the error checking.pig Latin is much lower cost to write and maintain thanJava code for MapReduce.

16. How is PIG Useful For?  
In three categories,we can use pig .they are  
1)ETL data pipline  
2)Research on raw data  
3)Iterative processing  
Most common usecase for pig is data pipeline.Let us take one example, web based compaines gets the weblogs,so before storing data into warehouse,they do some operations on data like cleaning and aggeration operations..etc.i,e transformations on data.

17. What are the complex datatypes in pig?  
TUPLE:  
tuple have fixed length and it have collection datatypes.tuple containing multiple fields and also tuples are ordered.  
example, (hyd,500086) which containing two fields.  
BAG:  
A bag containing collection of tuples which are unordered,Bag constants are constructed using braces, with tuples in the bag separated by com-  
mas. For example, {(‘hyd’, 500086), (‘chennai’, 510071), (‘bombay’, 500185)}  
MAP:  
map in pig is chararray to data element mapping where element have pig data type including complex data type.  
example of map [‘city’#’hyd’,’pin’#500086]  
the above example city and pin are data elements(key) mapping to values

18. Whether pig latin language is case-sensitive or not?  
pig latin is some times not a case sensitive.let us see example,Load is equivalent to load.  
A=load ‘b’ is not equivalent to a=load ‘b’  
UDF are also case sensitive,count is not equivalent to COUNT.

19. What is the purpose of ‘DUMP’ keyword in pig?  
DUMP diaplay the output on the screen  
DUMP ‘processed’

20) what are relational operations in pig latin?  
they are  
A)FOR EACH  
B)ORDER BY  
C)FILTERS  
D)GROUP  
E)DISTINCT  
F)JOIN  
G)LIMIT

21.what is ‘GROUP’ keyword in pig scripts?  
The group statement collects together records with the same key.In SQL the group by clause creates a group that must feed directly into one or more aggregate functions. In PIG Latin there is no direct connection between group and aggregate functions.

22. what is ‘ORDERBY’ keyword in pig scripts?  
The order statement sorts your data for you, producing a total order of your output data.The syntax of order is similar to group. You indicate a key or set of keys by which you wish to order your data

23. What Is Difference Between Mapreduce and PIG ?

•In MR Need to write entire logic for operations like join,group,filter,sum etc ..  
•In PIG Bulit in functions are available  
•In MR Number of lines of code required is too much even for a simple functionality  
•In PIG 10 lines of pig latin equal to 200 lines of java  
•In MR Time of effort in coding is high  
•In PIG What took 4hrs to write in java took 15 mins in pig latin (approx)  
•In MRLess productivity  
•In PIG High Productivity

24. What is the function of CO-GROUP in PIG?  
CO-GROUP joins the data set by grouping one particular data set only. It groups the elements by their common field and then returns a set of records containing two separate bags. The first bag consists of the record of the first data set with the common data set and the second bag consists of the records of the second data set with the common data set.

25. Can we say co-group is a group of more than 1 data set?  
CO-GROUP is a group of one data set. But in the case of more than one data sets, co-group will group all the data sets and join them based on the common field. Hence, we can say that co-group is a group of more than one data set and join of that data set as well.

26. What does FOREACH do?  
FOREACH is used to apply transformations to the data and to generate new data items. The name itself is indicating that for each element of a data bag, the respective action will be performed.

Syntax : FOREACH bagname GENERATE expression1, expression2, …..  
The meaning of this statement is that the expressions mentioned after GENERATE will be applied to the current record of the data bag.