**1. Why Map-reduce program is needed in Pig Programming?**

*When any PIG program is executed, internally it gets converted into MapReduce job and process the data.*

*We can also reuse java MapReduce programs within pig program.*

*Using ‘MAPREDUCE’ operator we can run native MapReduce jobs inside a Pig script.*

**2. What are advantages of pig over MapReduce?**

*1)Development time is more in MapReduce(writing mappers, reducers, compiling).*

*Pig scripts are easy to write and decreases development time.*

*2)Performing data set joins is very difficult in MapReduce.*

*In Pig, it is very simple.*

*3)Writing programs in map reduce requires exposure to Java.*

*Whereas Pig requires only basic SQL knowledge and no need of any java programming background*

**3. What is pig engine and what is its importance?**

*Pig engine converts Pig Latin scripts into series of MapReduce jobs.It acts as an interpreter between pig latin scripts and mapreduce jobs and it creates environment to execute scripts into series of mapreduce jobs in parallel manner.*

**4. What are the modes of Pig execution?**

*We can run pig in 2 modes:*

*1)local mode*

*syntax: pig -x local*

*2)MapReduce/Hadoop mode*

*syntax: pig (or) pig -x mapreduce*

**5. What is grunt shell in Pig?**

*It is an interactive shell where which we write and execute pig latin programs*.

**6. What are the features of Pig Latin language?**

*1)Pig processes data very fast*

*2)Ease of programming*

*3)Pig has rich set of operators*

*4)Pig is a language for parallel data processing*

*5)We can write User Defined Functions(UDF) in other languages and embed them in pig scripts.*

**7. Is Pig latin commands case sensitive?**

*Relations/field names and Function names are case sensitive, whereas keywords are case insensitive*

**8. What is a data flow language?**

*In a dataflow language, you have a stream of data which is passed from instruction to instruction to be processed*.*Data flow based applications are inherently parallel. Pig is a data flow language.*