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

PIG is a data flow language, the key focus of Pig is manage the flow of data from input source to output store. The [pig latin](http://www.pluralsight.com/courses/pig-latin-getting-started) i'm talking about isn't made-up and will help you write mapreduce jobs-without using java. Pig latin is pig's language that allows developers to sort, join, parse, transform and calculate unstructured and semi-structured data in mapreduce all while using a language similar to sql versus java. [Pig](http://en.wikipedia.org/wiki/Pig_%28programming_tool%29) is an application that works on top of mapreduce, Yarn or Tez. Pig is written in Java and compiles Pig Latin scripts into to mapreduce jobs. Think of Pig as a compiler that takes Pig Latin scripts and transforms them into Java.

1. What are advantages of pig over MapReduce?
   1. Pig is application that runs on top of MapReduce and abstracts Java MapReduce jobs away from developers.
   2. Pig Latin uses a lot fewer lines of code than the Java MapReduce script.
   3. The Pig Latin script was is easier to read for someone without a Java background.
   4. MapReduce jobs can written in Pig Latin.
   5. Java is a great and powerful language, but it has a higher learning curve than something like Pig Latin. Therefore, using a higher-level language, like Pig Latin, enables many more developers/analysts to write MapReduce jobs.
   6. Ease of programming. It is trivial to achieve parallel execution of simple, "embarrassingly parallel" data analysis tasks. Complex tasks comprised of multiple interrelated data transformations are explicitly encoded as data flow sequences, making them easy to write, understand, and maintain.
   7. Optimization opportunities. The way in which tasks are encoded permits the system to optimize their execution automatically, allowing the user to focus on semantics rather than efficiency.
   8. Extensibility. Users can create their own functions to do special-purpose processing.
2. What is pig engine and what is its importance?

It acts as an interpreter between pig latin scripts and mapreduce jobs. It will create an environment like executing pig scripts into an mapreduce flows on parallel manner.

1. What are the modes of Pig execution?

You can run Apache Pig in two modes, namely, Local Mode and Mapreduce mode.

**Local Mode**

In this mode, all the files are installed and run from your local host and local file system. There is no need of Hadoop or HDFS. This mode is generally used for testing purpose.

**MapReduce Mode**

MapReduce mode is where we load or process the data that exists in the Hadoop File System (HDFS) using Apache Pig. In this mode, whenever we execute the Pig Latin statements to process the data, a MapReduce job is invoked in the back-end to perform a particular operation on the data that exists in the HDFS.

1. What is grunt shell in Pig?

Grunt shell− Also called as Interactive Mode. You can run Apache Pig in interactive mode using the Grunt shell. In this shell, you can enter the Pig Latin statements and get the output (using Dump operator). you can run your Pig scripts in the shell. In addition to that, there are certain useful shell and utility commands provided by the Grunt shell. Which helps to write and execute pig scripts.

1. What are the features of Pig Latin language?
   1. Rich set of operators − It provides many operators to perform operations like join, sort, filer, etc.
   2. Ease of programming − Pig Latin is similar to SQL and it is easy to write a Pig script if you are good at SQL.
   3. Optimization opportunities − The tasks in Apache Pig optimize their execution automatically, so the programmers need to focus only on semantics of the language.
   4. Extensibility − Using the existing operators, users can develop their own functions to read, process, and write data.
   5. UDF’s − Pig provides the facility to create User-defined Functions in other programming languages such as Java and invoke or embed them in Pig Scripts.
   6. Handles all kinds of data − Apache Pig analyzes all kinds of data, both structured as well as unstructured. It stores the results in HDFS.

A Pig Latin statement is an operator that takes a relation as input and produces another relation as output. (This definition applies to all Pig Latin operators except LOAD and STORE which read data from and write data to the file system.) Pig Latin statements can span multiple lines and must end with a semi-colon ( ; ). Pig Latin statements are generally organized in the following manner:

LOAD statement reads data from the file system.

series of "transformation" statements process the data.

STORE statement writes output to the file system; or, a DUMP statement displays output to the screen.

1. Is Pig latin commands case sensitive?

The names of Pig Latin functions are case sensitive. ... Keywords LOAD, USING, AS, GROUP, BY, FOREACH, GENERATE, and DUMP are case insensitive.

1. What is a data flow language?

Apache Pig[[1]](https://en.wikipedia.org/wiki/Pig_(programming_tool)#cite_note-mainpage-1) is a high-level platform for creating programs that run on [Apache Hadoop](https://en.wikipedia.org/wiki/Hadoop). The language for this platform is called Pig Latin.[[1]](https://en.wikipedia.org/wiki/Pig_(programming_tool)#cite_note-mainpage-1) Pig can execute its Hadoop jobs in MapReduce, Apache Tez, or [Apache Spark](https://en.wikipedia.org/wiki/Apache_Spark). Pig Latin abstracts the programming from the [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) MapReduce idiom into a notation which makes MapReduce programming high level, similar to that of [SQL](https://en.wikipedia.org/wiki/SQL) for [RDBMSs](https://en.wikipedia.org/wiki/RDBMS). Pig Latin can be extended using [User Defined Functions](https://en.wikipedia.org/wiki/User-defined_function) (UDFs) which the user can write in Java