**Core Pig Interview Questions and Answers**

**What is Pig?**

Pig is the programming language, supported by the Hadoop framework, to process large data sets present in HDFS. Pig uses Pig Latin language for processing.

**Pig is a scripting language** comes from **Pig Latin Language,**Which lives on top of MapReduce. In 2006, **Yahoo has developed a Pig Latin language** and donated to the Apache Software Foundation as an open source project. **Pig Latin**is a **ProcedureOriented Language,**Means the program execution happens in sequence or in data flow manner. The ability of **procedural extension of Pig language** makes it highly recommendable for **ETL (Extract Transform Load).**Pig can also be used as an **Ad-Hocdata analysis.**

**Pig Latin –** a simple yet powerful high-level data flow language similar to SQL that executes MapReduce jobs. Pig Latin is also called as “Pig”.

**Pig Engine**– parses, optimizes and automatically executes PigLatin scripts as a series of MapReduce jobs on a Hadoop cluster.

Pig can be used with structured and semi-structure data.

**The pig was developed** based on a philosophy, which is that Pigs can eat anything, live anywhere, can be easily controlled and modified by the user.

**Why there is need of Pig language?**

The Big Data processing has made many advances since it was developed. The MapReduce programming has a modest design which breaks work down and recombines it in a series of parallelizable operations making it incredibly scalable. Since MapReduce expects hardware failures, it can run on inexpensive commodity hardware, sharply lowering the cost of a computing cluster. However, although MapReduce puts parallel programming within reach of most professional software engineers**, developing MapReduce jobs** isn’t easy:

1. They require the programmer to think in terms of “map” and “reduce”
2. N-stage jobs can be difficult to manage.
3. Common operations (such as filters, projections, and joins) and rich data types require custom code.

Thus, Apache Pig was developed. Which automates the MapReduce low-level details handling and provide a high-level MapReduce architecture to the users.

**How does Pig work?**

Every time a Pig Latin statement or the Pig script is executed, inside it is transformed into a Map Reduce program and run above HDFS.

**Pig Architecture Interview Questions**

**Explain Pig Architecture?**

To convert Pig Latin statements to MapReduce code, Pig Architecture comes into action. It will compile and execute a pig script.

1. Logical Plan
2. Physical Plan
3. MapReduce Plan

**What is the logical plan in pig architecture?**

In Logical plan help us to find us find the syntax error.

**What is the Physical plan in pig architecture?**

Physical plan is responsible for converting operators to Physical Plan.

**What is the MapReduce plan in pig architecture?**

In MapReduce than the output of Physical plan is converted into an actual MapReduce program. Which then executed across the Hadoop Cluster.

**What are the different modes available in Pig?**

Two modes are available in the pig.

1. Local Mode (Runs on localhost filesystem)
2. MapReduce Mode (Runs on Hadoop Cluster)

**What are the different execution mode available in Pig?**

There are 3 modes of execution available in pig

1. Interactive Mode (Also known as Grunt Mode)
2. Batch Mode
3. Embedded Mode

**What is grunt shell?**

Grant shell is a pig shell, which we use to write commands for execution that interacts with HDFS.

**What are the advantages of pig language?**

1. **The pig is easy to learn:** Pig is easy to learn, it overcomes the need for writing complex MapReduce programs to some extent. Pig works in a step by step manner. So it is easy to write, and even better, it is easy to read.
2. **It can handle heterogeneous data:** Pig can handle all types of data – structured, semi-structured, or unstructured.
3. **Pig is Faster:**Pig’s multi-query approach combines certain types of operations together in a single pipeline, reducing the number of times data is scanned.
4. **Pig does more with less:**Pig provides the common data operations (filters, joins, ordering, etc.) And nested data types (e.g. Tuples, bags, and maps) which can be used in processing data.
5. **Pig is Extensible:** Pig is easily extensible by UDFs – including Python, Java, JavaScript, and Ruby so you can use them to load, aggregate and analysis. Pig insulates your code from changes to the Hadoop Java API.

**Pig Latin Scripting Interview Questions**

**What are the different pig data types?**

Following are the data types supported by pig Latin language

1. Primitive data type
2. Complex data type

**What is UDF in Pig?**

Thus, **Pig provides support to allow writing User Defined Functions (UDFs)**as a way to stipulate custom processing.

Pig UDFs can presently be implemented in Java, Python, JavaScript, Ruby and Groovy. The most far-reaching support is provided for Java functions. You can customize all parts of the processing, including data load/store, column transformation, and aggregation. Java functions are also additional efficient because they are implemented in the same language as Pig and because additional interfaces are supported. Such as the Algebraic Interface and the Accumulator Interface. Limited support is provided for Python, JavaScript, Ruby and Groovy functions.

**What are the basic steps to writing a UDF Function in Pig?**

1. Define a class that extends EvalFunc<Class DataType>
2. Override the public String exec (Tuple input) method and add requisite business logic
3. Create Jar file for that class
4. Register JarFilename;
5. Write The Pig Script which uses the UDF created (UDF should be referred using fully qualified path i.e. PackageName.Classname)
6. Execute the Pig script

**What are the primitive data types in pig?**

Following are the primitive data types in pig

1. Int
2. Long
3. Float
4. Double
5. Char array
6. Byte array

**What are the complex data types in pig?**

Below are the complex data types in pig

1. Map
2. Tuple
3. Bag

**Field:**

A field is a piece of data. It can be referred as a cell in a table. Example:

{(0110), (”Peter”), (”London”), (25)…}

**Tuple:**

A tuple is an ordered set of fields. It can be referred as a Record as in SQL. Example:

(0110,”Peter”,”London”, 25)

**Bag:**

A bag is a collection of tuples. Example:

{(0110,”Peter”,”London”, 25), (0112,”Paul”,”New York”, 22), (0156,”Prakash”,”Bangalore”, 35)}

**What is Map data type in pig?**

The map data type concept is similar to the associative array concept. Which stores a character array and its key value pair. The map is the complex data type in pig language. If you do not declare the map data type, then the default would be a byte array.

**Syntax**[key #value key1#value key2#value……]

**What is Tuple data type in pig?**

**A tuple**is a complex data type in pig Latin language. The concept of the tuple is similar to the Collections. The length of the tuple is fixed and it is sorted. Each data value in a tuple is called a field.

**Syntax**(v1, v2, v3….)

**What is bag data type in pig?**

The **bag data type**worked as a container for tuples and other bags. It is a complex data type in pig latin language.

**Syntax**(tuple1, tuple2….)

**Is pig allowed nesting of data types?**

**Yes,**We can do nesting of data type pig. A bag is a collection of tuples.

**What is the Relation in Pig?**

**In Pig, Data is stored in Relations**. 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. Pig relations don’t require that every tuple contains the same number of fields or that the fields in the same position have the same type.

**Example Pig Latin Script:**

A = LOAD **‘PigInterview’** USING PigStorage() AS (name: char array, age: int, percent: float);  
DUMP A;  
(A, 10, 87)  
(B, 12, 10)  
(C, 15, 40)  
(D, 11, 50)

In overhead example- “A “is a relation which is a bag of **‘PigInterview’** details (collection of tuples for ‘PigInterview’ details).Each Tuple containing fields – name, age and percent.

**LOAD statement** is used to load data stored in file **‘PigInterview’**. The statement followed by “AS” denote the schema for relation “A”.

DUMP Statement is used to print the data stored in Relation “A”

**What are the different functions available in pig latin language?**

Pig latin has many data analysis functions. Below are some of the categorization

1. Math Functions
2. Eval Functions
3. String Functions

**What are the different math functions available in pig?**

Below are most commonly used**math** **pig functions**

1. ABS
2. ACOS
3. EXP
4. LOG
5. ROUND
6. CBRT
7. RANDOM
8. SQRT

**What are the different Eval functions available in pig?**

Below are most commonly used **Eval pig functions**

1. AVG
2. CONCAT
3. MAX
4. MIN
5. SUM
6. SIZE
7. COUNT
8. COUNT\_STAR
9. DIFF
10. TOKENIZE
11. IsEmpty

**What are the different String functions available in pig?**

Below are most commonly used **STRING pig functions**

1. UPPER
2. LOWER
3. TRIM
4. SUBSTRING
5. INDEXOF
6. STRSPLIT
7. LAST\_INDEX\_OF

**What are the different Relational Operators available in pig language?**

Relational operators in pig can be categorized into the following list

1. Loading and Storing
2. Filtering
3. Grouping and joining
4. Sorting
5. Combining and Splitting
6. Diagnostic

**What are the relational operators available related to loading and storing in pig language?**

For Loading data and Storing it into HDFS, Pig uses following operators.

1. LOAD
2. STORE

LOADS, load the data from the file system. STORE, stores the data in the file system.

**What are the relational operators available related to filtering in pig language?**

Filter operators are used for data analysis through the pig. Below are some of the filtering related operators.

1. FILTER
2. FOREACH
3. DISTINCT
4. STREAM

**FILTER**Removes unwanted data from relation using the condition you provided.**FOREACH**generates data transformation. **DISTINCT**fetches distinct tuples from a specified relation. **STREAM**removes duplicate tuples from a Relation.

**What are the relational operators available related to Grouping and joining in pig language?**

Grouping and Joining operators are the most powerful operators in pig language. Because core MapReduce  creation for grouping and joins are very typical in low-level MapReduce language.

1. JOIN
2. GROUP
3. COGROUP
4. CROSS

**JOIN**is used to join two or more relations. **GROUP** is used for aggregation of a single relation**. COGROUP**is used for the aggregation of multiple relations. **CROSS**is used to create a cartesian product of two or more relations.

**What are the relational operators available related to sorting in pig language?**

Sorting operators in pig language are

1. Order
2. Limit

**ORDER**can sort a relation by one or more fields. **LIMIT**is used to restrict the size of relation to a number of tuples.

**What are the relational operators available related to combining and splitting in pig language?**

**UNION**and **SPLIT** used for combining and splitting relations in the pig.

**How would you diagnose or do exception handling in the pig?**

For exception handling of pig script, we can use following operators.

1. DUMP
2. DESCRIBE
3. ILLUSTRATE
4. EXPLAIN

**DUMP**displays the results on screen.**Describe**displays the schema of a particular relation. **ILLUSTRATE**displays step by step execution of a sequence of pig statements.**EXPLAIN**displays the execution plan for pig latin statements.

What is the difference between inner bag and outer bag?

Outer bag is nothing but a relation.

Inner bag is a relation inside a bag..