**Shaikh, Heenasalim**

**3.6 years**

**LETS PREPARE FOR INTERVIEW**

**Its time to leave sears holding India**

**HADOOP AND MAPREDUCE QUESTIONS**

1. **What are the basic differences between relational database and HDFS?**

|  |  |  |
| --- | --- | --- |
|  | **RDBMS** | **Hadoop** |
| **Data Volume** | It can handle smaller data | It can handle huge amount data |
| **Data Types** | It can process only structured data | It can process structured, unstructured or semi-structured. |
| **Cost** | Licensed and payable software, | Hadoop is an open source framework. |
| **Engine** | OLTP | OLAP |
| **Processing** | Fast ACID transactions | Slow ACID transactions |
|  | Quickly access the row | Cannot quickly access the row |
| **Read/Write Speed** | Fast reading of the data | Fast writing of the data |

1. **Explain “Big Data” and what are five V’s of Big Data?**

* **Volume:**

Big data can store Petabytes/Exabytes of the data.

* **Velocity**:

And this data process with higher speed .

* **Variety:**

And data can be structured, semi-structured and un-structured data.

* **Value:-**

and providing value ie high return of investment to organization.  
**Veracity:-**

### **What is Hadoop and its components.**

Apache Hadoop is framework which is use to load, process and store the huge volume of Structured , nonstructural and semi-structured data with high speed.

Hadoop have 2 Components :-

* Storage unit– HDFS (NameNode, DataNode)
* Processing unit– YARN (ResourceManager, NodeManager)

**4. What is HDFS:-**

Hadoop distributed file system used to store variety of the data.

HDFS have two components:-

**NameNode: -**

* It is masternode which store the metadata eg replication factor and block size, block number etc
* And manages the datanodes

**DataNode:-**

Datanodes stores the actual data.

**5. What is YARN:-**

Yet another resource negotiator

YARN have two components:-

**ResourceManager:-**

* It manages the resources base on need
* allocates task to node managers.

**NodeManager:-**

It is installed program on every datanode to execute task allocated by resource manager.

### **6. List the difference between Hadoop 1 and Hadoop 2.**

### In Hadoop 1.x, “NameNode” is the single point of failure.

### In Hadoop 2.x, we have Active and Passive “NameNodes”. If the active “NameNode” fails, the passive “NameNode” come into picture.

|  |  |  |
| --- | --- | --- |
|  | Hadoop 1.x | Hadoop 2.x |
| Passive  NameNode | NameNode is a Single Point of Failure | Active & Passive NameNode | |
| Processing | MRV1 (Job Tracker & Task Tracker) | MRV2/YARN (ResourceManager & NodeManager) | |
|  | Low scalability | Higher scalability | |

### **7. What are active and passive “NameNodes”?**

Active “NameNode” is the “NameNode” which works/active in the cluster.When the active “NameNode” fails, the passive “NameNode” replaces the active “NameNode”.

### **8. How does NameNode tackle DataNode failures?**

NameNode periodically receives a Heartbeat (signal) from each of the DataNode in the cluster,

If a DataNode does not send a heartbeat message, after a specific period of time it is marked dead.

### **9. What will you do when NameNode is down?**

1. I will use the file system metadata replica (FsImage) to start a new NameNode.
2. Then, will configure the DataNodes and clients so that they will know new NameNode, t is started.

### **10. What is a checkpoint?**

1. “Checkpointing” is a process that combine FsImage, edit log produce new FsImage.
2. Checkpointing is saving the latest system state.
3. Checkpointing is performed by Secondary NameNode.
4. Hence reduces NameNode startup time when it got failed

### **11. How do you define “block” in HDFS? What is the default block size in Hadoop 1 and in Hadoop 2? Can it be changed?**, **difference between an “HDFS Block” and an “Input Split”?**

The “HDFS Block” is the physical division of the data and

“Input Split” is the logical division of the data received from the client.

**15. Name the three modes in which Hadoop can run.**

The three modes in which Hadoop can run are as follows:

1. **Standalone (local) mode:**

the components of Hadoop, such NameNode, DataNode, ResourceManager, and NodeManager, run as a **single Java process in the local filesystem**

1. **Pseudo-distributed mode:**

In this mode, master and the slave nodes services execute **on a single node.**

1. **Fully distributed mode:**

The Hadoop master and slave services **run on separate nodes**, are stated as fully distributed mode.

### **16. What is the purpose of “RecordReader” in Hadoop?**

The “RecordReader” class reads the record from source and converts it into (key, value) pairs which can be used by “Mapper” task.

**17. How do “reducers” communicate with each other?**

“Reducers” do not communicate with each other, run in isolation/runs separately.

**18. What does a “MapReduce Partitioner” do?**

A “MapReduce Partitioner” it passes common keys to the same “reducer” and

### **19. What is a “Combiner”?**

Combines the input from the “mapper” and Sends to the “reducer”. “

### **20. What are the benefits of Apache Pig over MapReduce?**

|  |  |  |
| --- | --- | --- |
| Unit | Pig | Mapreduce |
| Language | High level data processing language | Low level data processing language |
| Coding | Easy to learn | Complex to learn |
| Code length | 10 lines of code | 100 lines of code |
| Engine | Built in functions available for joins, filters, ordering, sorting | We need to write custom programs for  joins, filters, ordering, sorting |
| Read/Write Speed | Joining is faster | Joining is slow. |

### **21. What are the different data types in Pig Latin ?**

### Pig Latin can handle

### Atomic data types :- int, float, long, double,char etc.

### Complex data types:- tuple, bag and map.

### **22. What are the different relational operations in “Pig Latin” you worked with?**

Different relational operators are:

1. for each
2. filter
3. order by
4. group by
5. distinct
6. join
7. limit

### **23. What is a UDF?**

If some functionalities are unavailable in built-in operators, user create functions (UDF) in languages like java, python,ruby.

### **24. What is the default location where “Hive” stores table data?**

### /user/hive/warehouse

Data warehouse location can be set using hive.metastore.warehouse.dir  property in the hive-site.xml.

### **25. What are the core methods of a Reducer?**

### The 3 core methods of a reducer are –

### setup () – This set up parameters like data size, cache, heap size, etc.

### Function Definition- public void setup (context)

### reduce () it is heart of the reducer, which is called once per key.

### Function Definition -public void reduce (Key,Value,context)

### cleanup () - called only once for clearing all the temporary files.

### Function Definition -public void cleanup (context)

### **26. Explain about the partitioning, shuffle and sort phase**

### Shuffle Phase-

### The process of moving the mapper intermediate outputs to the reducer.

### Sort Phase- Sorts of the intermediate keys before passing to the reducer.

### Partitioning Phase-The process of assigning key value pair to respective reducers is known as partitioning.

**27. Explain about co-group in Pig.**

grouping two or more relations.

It can take 127 relations at a time.

**28. What is SerDe in Hive? How can you write your own custom SerDe?**

SerDe is a Serializer DeSerializer.

Hive uses SerDe to read and write data into tables.

**29. What are the stable versions of Hadoop?**

Release 2.7.1 (stable)

Release 2.4.1

Release 1.2.1 (stable)

**30. The number of nodes you have worked with in a cluster.**

600

### **31. What is data voume u dealt in the company recently?**

### 1TB

### **32. Name some companies that use Hadoop**

### Facebook Netflix Amazon eBay Hulu Twitter

### **33. What is commodity hardware?**

### Commodity Hardware is cheaper systems that do not have high scalability or high quality.

### Hadoop can be run on any commodity hardware and does not require super computers.

### **34. What is the port number for NameNode, Task Tracker and Job Tracker?**[**Click here to Tweet**](http://ctt.ec/wf0Uv)

### Job Tracker 50030

### Task Tracker 50060

### NameNode 50070

### **35. How can you overwrite the replication factors in HDFS?**

### hadoop fs –setrep –w 5 /user/heena

### 

### **36. What is the default replication factor?**

### 3

### **37. What are different hdfs dfs shell commands to perform copy operation?**

### $ hadoop fs -copyToLocal $ hadoop fs –copyFromLocal $ hadoop fs –put

### $hadoop fs –get

### 38. What different type of schedulers and type of scheduler did you use

### Capacity Scheduler It is designed to run Hadoop applications as a shared, multi-tenant cluster.

### It maximize throughput and utilization of the cluster.

### Fair Scheduler

### assign equal resources to application.

### **39. What is distcp?**

### **T**he program comes with Hadoop for copying bigdata to different Hadoop servers

### no reducers are used here.

### **40. Command to format the NameNode?**

### $ hdfs namenode –format

### **41. What are different daemons in YARN?**

### ResourceManager: Global resource manager.

### NodeManager: One per data node, It manages and monitors resource usage in terms of Memory, CPU.

### ApplicationMaster: One per application, Tasks are started by NodeManager

### **42. What are the site-specific configuration files in Hadoop?**

### conf/core-site.xml

### conf/hdfs-site.xml

### conf/yarn-site.xml

### conf/mapred-site.xml.

### conf/hadoop-env.sh

### conf/yarn-env.sh

### **43. What are the two main components of ResourceManager?**

### Scheduler

### It allocates the resources to various running applications such as memory, CPU, disk etc.

### ApplicationManager It accepts jobs, resource for executing the application specific ApplicationMaster and provides the service for restarting the ApplicationMaster container on failure.

### **44. What is the function of NodeManager?**

### It manages resources,

### Monitores resource usage (cpu, memory, disk, network)

### Reporting the same to the ResourceManager

### **45. What is the function of ApplicationMaster?**

### ApplicationMaster is per application

### Moves appropriate resources from the Scheduler

### Track resource status

### Monitor resource progress.

### **46. What is a partitioner and how the user can control which key will go to which reducer?**

### Partitioner do partitioning of the keys of the intermediate map-outputs.

### It uses hash function for it.

### There are two partitioner program.

### HashPartitioner.

### A custom partitioner is implemented to decide which keys go to which Reducer.

### Method use to create custom partition is :-

### public class SamplePartitioner extends Partitioner {

### @Override

### public int getPartition(Text key, Text value, int numReduceTasks) {

### }}

### **47. What are combiners and its purpose?**

### Combiners are used to

### It is used to aggregate intermediate map output locally.

### Reduce the amount of data that needs to be transferred across to the reducers.

### Hadoop may or may not execute a combiner.

### **48. How a number of partitioners and reducers are related?**

### The total numbers of partitions are the same as the number of reduce tasks for the job.

### **49. What are a Counter and its purpose?**

### The counter is facility for MapReduce applications

### to report its statistics.

### to track job progress in simple manner

### Counter can be of any Enum type.

### **50. Define different types of Counters?**

### Built in Counters:

### Map Reduce Task Counters

### Job Counters

### Custom Java Counters:

### User can perform their own counting operation by specifying their own counters.

### **51. Why Counter values are shared by all map and reduce tasks across the MapReduce framework?**

### Counters are global so shared across the MapReduce framework and aggregated at the end of the job across all the tasks.

### **53. How to set the number of reducers?**

### The number of reduces for the user sets the job:

### Job.setNumReduceTasks(int)

### -D mapreduce.job.reduces

### **52. What is the reducer and its phases?**

### Reducer reduces a set of intermediate values, which has same key .

### The framework calls reduce() for this. Syntax: reduce(WritableComparable, Iterable, Context) Reducer has three primary phases:

### Shuffle

### Sort

### Reduce

### **54. Detail description of the Reducer phases?**

### Shuffle: Providing the key value pair from mapper to reducer.

### Sort: grouping and sorting the reducer inputs on the basis of the same keys.

### The shuffle and sort phases occur simultaneously;

### Reduce: reduce(WritableComparable, Iterable, Context) method is called to

### Adding the values corresponding to same keys from grouped key value pairs. The output of the reduce task is typically written using Context.write(WritableComparable, Writable).

### **55. Can there be no Reducer?**

### Yes, the number of reducer can be zero if no reduction of values is required.

### Eg. select statement

### **-------------------------------------------------------------------------------------------------------------------------**

### **56. What can be optimum value for Reducer?**

### Value of Reducers can be: 0.95

### 1.75 multiplied by ( \* < number of maximum resources per node>)

### Increasing number of reducers

### Increases the framework overhead

### Increases load balancing

### Lowers the cost of failures

### **60. What is DistributedCache and its purpose?**

### DistributedCache is a facility provided by the MapReduce framework to cache files (text, archives, jars etc.) needed for applications. It distributes large, read-only files efficiently.

### **62. What is the default value of map and reduce max attempts?**

4

### **63. Explain InputFormat?**

### InputFormat describes the input-specification for a MapReduce job.

### The MapReduce framework then splits the input file(s) into logical InputSplits

### which is then to an individual Mapper.

### Default: TextInputFormat

### **64. What is InputSplit and RecordReader?**

### InputSplit specifies logical division of the data. InputSplit is a byte-oriented view .

### Default: FileSplit RecordReader reads pairs from an InputSplit, then processes them and presents record-oriented view.

### **65. Explain the Job OutputFormat?**

### OutputFormat describes details of the output-specification for MapReduce job.

### Default: TextOutputFormat

### **66. How is the option in Hadoop to skip the bad records?**

### SkipBadRecords class.

**69. What is Data Integrity?**

**70. What is Hadoop File System?**

### **71. What are the problems with Hadoop 1.0?**

### NameNode: No Horizontal Scalability and No High Availability

### Job Tracker: Overburdened.

### MRv1: It can only understand Map and Reduce tasks

**72. Explain Hadoop architechture in detail.**

**73. Explain Mapreduce architechture in detail .**

**What is containers**

### **HIVE**

### **1. What is Hive?**

### It is open source framework use for querying and analyzing structured big data like SQL

Hive it is not a database but querying language

The present version of hive is 0.13.1.

### **2.In which scenario Hive is good fit?**

### Data warehousing

### OLAP

### For structured query

### Fast response not needed

Not suitable for OLTP, Online Transaction Processing.

**3.What kind of data warehouse application is suitable for Hive?**

### Data warehousing

### OLAP

### For structured query

### Fast response not needed

Not suitable for OLTP, Online Transaction Processing.

**5.Features and Limitations of Hive.**

**Ans. Features of Hive**

### Data warehousing

### OLAP

### For structured query

### Fast response not needed

Not suitable for OLTP, Online Transaction Processing.

**Limitation of Hive**

Can not perform

OLTP

ACID

### High speed queries

**6. Explain about ACID transactions in Hive?**

Hive supports ACID transactions.

ACID is Atomicity, Consistency, Isolation, and Durability, Insert, Delete, and Update options is possible in Hive.

Insert

Delete

Update

### **7.what are the differences between Hive and RDBMS?**

|  |
| --- |
| HIVE: RDBMS: |
| Schema on Read Schema on write |
| Batch processing Real time processing |
| Data stored in HDFS format Data stored in tabular format |
| MapReduce Processing Non-MapReduce Processing |
| No ACID ACID |

**9.What are the different components of a Hive architecture?**

There are 4 components of Hive architechture

* **User Interface** – interface to the driver. User can send command to hive engine.
* **Compiler**- Compiler converts the query into the object form.
* **Parser** :- It parses the query and creates DAG tree
* **Metastore** – It is used to send the metadata to the compiler.
* **Execute Engine**- Mapreduce program execute for code.
* **Logical Plan of Generation.**
* **Physical Plan of Generation.**
* **Optimizer.**

### **10. What is the purpose of Hive Driver?**

### Hive Driver is used for compiling, optimizing and then executing the Hive query.

### **11. What is the purpose of storing the metadata?**

### People want to read data with a particular schema

### Eg. Manager may be interested to look attendance of the particular emplyoee

### Principle may be interested to look marks of the particular student. So to access particular field again and again and to avoid efforts, we use schema.

### **12.List the various options available with the Hive command/ the different services that can be invoked using the Hive command.**

### Syntax:

### cli

### hiveserver

### hwi

### jar

### metastore

### rcfile

### **14. Can you execute Hadoop dfs Commands from Hive CLI? How?** hive > dfs -ls / ;

### **15. How to give multiline comments in Hive Scripts?**

### Hive does not support multiline comments. We can give single line comments by using – .

### **16. What is the reason for creating a new metastore\_db whenever Hive query is run from a different directory?** In embedded mode, hive checks the metastore exists or not. If metastore not exist then it creates the local metastore.

### **17. When Hive is run in embedded mode, how to share the metastore within multiple users?** For sharing use the standalone database (like MySQL, PostGresQL) .

### **18. How can an application connect to Hive run as a server?**

### Hive can connect with three drivers:-

### Thrift Client: Hive commands can call hive command through programming languages like Java, PHP, Python, Ruby, C++ JDBC Driver: Type 4 (pure Java) JDBC Driver ODBC driver:  ODBC protocol

**19. What are the different types of tables available in HIve?**

There are two types of tabkle:- Managed table and external table.

We can store both data and metadata in managed table and if you drop the managed table, data and metadata get deleted.

In the external table if we drop table, only metadata get deleted.

**21. Explain the External Table features in Hive?**

**22. Explain the Internal Table features in Hive?**

## **23. Difference between "Internal Table" and "External Table"**

Ans. "Internal Table" also known as Managed Table, is the one that is managed by Hive. When you point data in HDFS to such table, the data is moved to Hive default location /user/hive/warehouse/. And, then if such internal table is dropped, the data is deleted along with.

"External table" on the other hand is user managed, and data is not moved to hive default directory after loading i.e, any custom location can be specified. Consecutively, when you drop such table, no data is deleted, only table schema is dropped.

**24. Mention when to choose “Internal Table” and “External Table” in Hive?**

In Hive you can choose internal table,

If data is needed for temporary basis we can use managed table ,

If data is critical then we need to use managed table

**25. Is Hive suitable to be used for OLTP systems? Why?**

It does not support ACID transactions ,So it is not suitable for OLTP system.

**26. Can a table be renamed in Hive?**

Alter Table table\_name RENAME TO new\_name

**27. Can we change the data type of a column in a hive table?**

Using REPLACE column option

ALTER TABLE table\_name REPLACE COLUMNS ……

**28. What is the need for custom Serde?**

Users need to write their own java code for complex data formatting.

**30. What are the three different modes in which hive can be run?**

* Local mode
* Distributed mode
* Pseudo distributed mode

**31. Is there a date data type in Hive?**

Yes. TIMESTAMP data stores date in java.sql.timestamp format

**32. What are collection data types in Hive?**

There are three collection data types in Hive.

* ARRAY
* MAP
* STRUCT

**33. Can we run unix shell commands from hive? Give example.**

Yes, using the ! mark before the command.

example !pwd will list current directory.

**34. Which java class handles the Input record encoding into files which store the tables in Hive?**

org.apache.hadoop.mapred.TextInputFormat

**35. Which java class handles the output record encoding into files which result from Hive queries?**

org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat

**36. What is the significance of ‘IF EXISTS” clause while dropping a table?**

If table we are dropping does not exist hive throws error table does not exist, so to avoid warning message we use if exists clause.

**37. When you point a partition of a hive table to a new directory, what happens to the data?**

The data stays in the old location only . It has to be moved manually.

**38. Will the reducer work or not if you use “Limit 1” in any Hive query or If we use the "Limit 1" in any SQL query in Hive, will Reducer work or not.**

Ans. It will work , as hive fetch random record.

**39. What are the uses of explode Hive?**

To convert complex data types into desired table formats Hive is essentially using explode.

**46. What is a Table generating Function on hive?**

A table generating function is a function which takes a single column as argument and expands it to multiple column or rows. Example explode()

**40. Write a query to insert a new column(new\_col INT) into a hive table (htab) at a position before an existing column (x\_col)**

ALTER TABLE table\_name

CHANGE COLUMN new\_col\_name new\_column\_datatype

BEFORE x\_col

**41. Does the archiving of Hive tables give any space saving in HDFS?**

No. It only reduces the number of files which become easy for namenode to manage.

**42. How can you stop a partition form being queried?**

By using the ENABLE OFFLINE clause with ALTER TABLE satatement.

**43. While loading data into a hive table using the LOAD DATA clause, how do you specify it is a hdfs file and not a local file ?**

By Omitting the LOCAL CLAUSE in the LOAD DATA statement.

**44. If you omit the OVERWRITE clause while creating a hive table,what happens to file ?**

It will append the data to existing data.

**45. What does the following query do?**

INSERT OVERWRITE TABLE employees

PARTITION (country, state)

SELECT ..., se.cnty, se.st

FROM staged\_employees se;

This will overwrite the data of the two partitions and data is seleted from temporary table.

**47. How can Hive avoid mapreduce?**

If we set the property hive.exec.mode.local.auto = true then hive will avoid mapreduce.

**48. What is the difference between LIKE and RLIKE operators in Hive?**

The LIKE operator will select data like particular word .

Example −

street\_name like ‘%Pune’

RLIKE will select data based on OR ,AND,true/false conditions

Example − street\_name RLIKE ‘.\*(Pune|Mumbai).\*’

**49. Is it possible to create Cartesian join between 2 tables, using Hive?**

No.

**50. As part of optimizing the queries in HIve, what should be the order of table size in a join query?**

In a join query the smallest table should be at first position and largest table should at last position.

**51. How will you convert the string ’51.2’ to a float value in the price column?**

select cast(price as FLOAT)

**52. What will be the result when you do cast(‘abc’ as INT)?**

Hive will return NULL

**53. Mention what are views in Hive?**

* We can save any result set data as a view in Hive
* All type of DML operations can be performed on a view

**54. Can the name of a view be same as the name of a hive table?**

No. The name of a view must be unique in one database.

**55. Can we LOAD data into a view?**

No. We cannot perform INSERT or LOAD on view.

**56. What types of costs are associated in creating index on hive tables?**

Indexes need space and processing cost

**57. Give the command to see the indexes on a table.**

SHOW INDEX ON table\_name

**58. What is the usefulness of the DISTRIBUTED BY clause in Hive?**

It decide the map output is distributed among the reducers. It is useful for streaming data.

1. **What does /\*streamtable(table\_name)\*/ do?**
2. **Can a partition be archived? What are the advantages and Disadvantages?**

Yes. A partition can be archived. Advantage is it decreases the number of files stored in namenode and it become easy for namenode to manage.

The disadvantage is less efficient query and no space saving.

1. **The following statement failed to execute. What can be the cause?**

LOAD DATA LOCAL INPATH ‘${env:HOME}/country/state/’

OVERWRITE INTO TABLE address;

1. **How do you specify the table creator name when creating a table in Hive?**

We can mention:-

TBLPROPERTIES(‘creator’= ‘Joan’)

### **64. Define the difference between Hive and HBase?**

|  |  |  |
| --- | --- | --- |
| **HBase** | | **Hive** |
| low latency | | high latency |
| Allow SQL queries | doesn’t allow SQL queries | |
| Allow ACID transactions | Allow ACID transactions | |
| It is NoSQL database | It is a data warehouse framework | |
| Not Mapreduce | MapReduce jobs | |

### **Is it possible to change the default location of a managed table?**

Yes, we can change default location of a managed table by using LOCATION ‘<hdfs\_path>’clause

### **What is a partition in Hive? What is partitioning in Hive?**

* Partitioning is grouping similar data based on a column
* A partition is sub-directory in the table directory.

### **68. How to enable dynamic partitioning in Hive?** set hive.exec.dynamic.parition.mode = nonstrict;

Example,  
insert overwrite table emp\_details\_partitioned  
partition(location)  
select \* from emp\_details;

### **What is the difference between static and dynamic partitioning in Hive?**

**Static Partition in Hive**

* In static partitioning, you must specify the partition column value with LOAD statement.
* Static Partition is timesaving than dynamic partition.
* We can alter the partition in the static partition

**Dynamic Partition in Hive**

* To load the data from a non-partitioned table, you can use a dynamic partition to insert partitions individually.
* Dynamic Partition takes more time to load data than static partition.
* Dynamic partition is suitable for large data or to partition many columns
* You can use a dynamic partition on hive external table and managed tables.

### **What is dynamic partitioning and when is it used?**

In dynamic partitioning values for partition columns are known at the runtime, i.e. It is known during loading of the data into a Hive table.

We can dynamic partition in following two cases:

1. Loading data from an existing non-partitioned table to improve the sampling and therefore, decrease the query latency.
2. When we dont know values of the partitions before
3. Finding these partition values manually for huge data sets

### **Scenario:**

**Suppose, I create a table that contains details of all the transactions done by the customers of year 2016:**

**CREATE TABLE transaction\_details (cust\_id INT, amount FLOAT, month STRING, country STRING) ROW FORMAT DELIMITED FIELDS TERMINATED BY ‘,’ ;**

**Now, after inserting 50,000 tuples in this table, I want to know the total revenue generated for each month. But, Hive is taking too much time in processing this query.** **How will you solve this problem and list the steps that I will be taking in order to do so?**

We can solve this problem of query latency by partitioning table according to each month.

As we know, we can’t partition an existing non-partitioned table directly. So, we need to perform dynamic partitioning:-

* 1. Create a partitioned table, say partitioned\_transaction:

CREATE TABLE partitioned\_transaction (cust\_id INT, amount FLOAT, country STRING) PARTITIONED BY (month STRING)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ‘,’ ;

* 1. Enable dynamic partitioning in Hive:

SET hive.exec.dynamic.partition = true;

SET hive.exec.dynamic.partition.mode = nonstrict;

* 1. Transfer the data from the non – partitioned table into the newly created partitioned table:

INSERT OVERWRITE TABLE partitioned\_transaction PARTITION (month) SELECT cust\_id, amount, country, month FROM transaction\_details;

### **How can you add a new partition for the month December in the above partitioned table?**

For adding a new partition in the above table partitioned\_transaction, we will issue the command give below:

ALTER TABLE partitioned\_transaction ADD PARTITION (month=’Dec’) LOCATION  ‘/partitioned\_transaction’;

* 1. **What is the default maximum dynamic partition that can be created by a mapper/reducer? How can you change it?**

By default the number of maximum partition is s 100. One can change it by issuing the following command:

SET hive.exec.max.dynamic.partitions.pernode = <value>

### **72.I am inserting data into a table based on partitions dynamically. But, I received an error – FAILED ERROR IN SEMANTIC ANALYSIS: Dynamic partition strict mode requires at least one static partition column.** **How will you remove this error?**

To remove this error one has to execute following commands:

SET hive.exec.dynamic.partition = true;

SET hive.exec.dynamic.partition.mode = nonstrict; by default it is strict

**73. What is bucketing ?**

 In Hive Tables or partition are subdivided into buckets based on the hash function of a column /key of the table to acces the queries more efficiently is know as bucketing.

**74. What is Buckets in Hive?**

The partitioned data is divided into different Buckets. This data is divided on the basis of Hash key of the particular columns.

### **75. Why do we need buckets?**

There are two main reasons for performing bucketing to a partition:

* A[**map side join**](https://www.edureka.co/blog/map-side-join-vs-join/)can be perform only when join key and partition key is equal.
* But when partition key differs join key we can perform a map side join by bucketing the table using the join key.
* Bucketing makes the sampling process more efficient and therefore, allows us to decrease the query time.

### **76. How Hive distributes the rows into buckets?**

Hive determines the bucket number for a data by using the formula:

hash\_function (bucketing\_column) modulo (num\_of\_buckets). \

**77. In Hive, how can you enable buckets?**

In Hive, you can enable buckets by using the following command,

set.hive.enforce.bucketing=true;

### **78. What will happen in case you have not issued the command:**‘SET hive.enforce.bucketing=true;’**before bucketing a table in Hive in Apache Hive 0.x or 1.x?**

The command:  ‘SET hive.enforce.bucketing=true;’ allows to have the correct number of reducer for bucketing a column.

The reducers generated will not equal to the number of buckets.

We can also set the number of reducer equal to the number of buckets by using set mapred.reduce.task = num\_bucket.

**79. Can We Change settings within Hive Session? If Yes, How?**

Yes we can ,

hive> SET hive.enforce.bucketing=true;

### **What kind of applications is supported by Apache Hive?**

Hive supports all those client applications that are written in:

* Java
* PHP
* Python
* C++
* Ruby
* by exposing its Thrift server.

### **Why Hive does not store metadata information in HDFS?**

Hive stores metadata information in the metastore using RDBMS instead of HDFS because

HDFS read/write operations are time consuming processes.

### **What is the difference between local and remote metastore?**

**Local Metastore:**

In local metastore , the metastore service and Hive service runs in same JVM and database runs in a separate JVM which can located in same machine or in remote machine.

**Remote Metastore:**

In the remote metastore , the metastore service and Hive service runs in separate JVM. Processes communicate with the metastore server using Thrift Network APIs.

### **What is the default database provided by Apache Hive for metastore?**

By default, Hive provides an embedded Derby database metastore.

### **Scenario:**

**Suppose I have installed Apache Hive on top of my Hadoop cluster using default metastore configuration. Then, what will happen if we have multiple clients trying to access Hive at the same time?**

The default metastore configuration allows only one Hive session to be opened at a time for accessing the metastore.

Therefore, if multiple clients try to access the metastore at the same time, they will get an error.

One has to use a standalone metastore, i.e. Local or remote metastore configuration in Apache Hive for allowing access to multiple clients concurrently.

###### **Is it possible to create multiple table in hive for same data?**

As hive creates schema and append on top of an existing data file. One can have multiple schema for one data file, schema will be saved in hive’s metastore and data will not be parsed or serialized to disk in given schema. When we will try to retrieve data, schema will be used. For example if we have 5 column (name, job, dob, id, salary) in the data file present in hive metastore then, we can have multiple schema by choosing any number of columns from the above list. (Table with 3 columns or 5 columns or 6 columns).

But while rying, if we specify any column other than above list, will result in NULL values.

###### 86. What is the maximum size of string data type supported by Hive?

Maximum size is 2 GB.

1. **Suppose, I have a CSV file – ‘sample.csv’ present in ‘/temp’ directory with the following entries:**

**id first\_name last\_name email gender ip\_address**

1 Hugh Jackman hughjackman@cam.ac.uk Male 136.90.241.52

2 David Lawrence dlawrence1@gmail.com Male 101.177.15.130

3 Andy Hall andyhall2@yahoo.com Female 114.123.153.64

4 Samuel Jackson samjackson231@sun.com Male 89.60.227.31

* + 1. Emily Rose rose.emily4@surveymonkey.com Female 119.92.21.19

### **What is indexing and why do we need it?**

We can achieve Hive query optimization using indexing. It speed up the access of a columns in a Hive database as database system does not need to read all rows everytime to find the particular row and it increases speed.

**89.How will you consume this CSV file into the Hive warehouse using built SerDe?**

SerDe stands for serializer/deserializer.

A SerDe allows us to convert the unstructured bytes into a record that we can process using Hive.

SerDes are implemented using Java. Hive comes with several built-in SerDes.

Hive provides a SerDe for working with CSV files. We can use issuing following commands:

CREATE EXTERNAL TABLE sample

(id int, first\_name string,

last\_name string, email string,

gender string, ip\_address string)

ROW FORMAT SERDE ‘org.apache.hadoop.hive.serde2.OpenCSVSerde’

STORED AS TEXTFILE LOCATION ‘/temp’;

Now, we can perform any query on the table ‘sample’:

SELECT first\_name FROM sample WHERE gender = ‘male’;

**90.Suppose, I have a lot of small CSV files present in /input directory in HDFS and I want tocreate a single Hive table corresponding to these files. The data in these files are in the format: {id, name, e-mail, country}. Now, as we know, Hadoop performance degrades when we use lots of small files.**

**So, how will you solve this problem where we want to create a single Hive table for lots of small files without degrading the performance of the system?**

We can use the SequenceFile format which will group these small files together to form a single sequence file. The steps that will be followed in doing so are as follows:

Create a temporary table:

CREATE TABLE temp\_table (id INT, name STRING, e-mail STRING, country STRING)

ROW FORMAT FIELDS DELIMITED TERMINATED BY ‘,’ STORED AS TEXTFILE;

Load the data into temp\_table:

LOAD DATA INPATH ‘/input’ INTO TABLE temp\_table;

Create a table that will store data in SequenceFile format:

CREATE TABLE sample\_seqfile (id INT, name STRING, e-mail STRING, country STRING)

ROW FORMAT FIELDS DELIMITED TERMINATED BY ‘,’ STORED AS SEQUENCEFILE;

Transfer the data from the temporary table into the sample\_seqfile table:

INSERT OVERWRITE TABLE sample SELECT \* FROM temp\_table;

Hence, a single SequenceFile is generated which contains the data present in all of the input files and therefore, the problem of having lots of small files is finally eliminated.

**91.Explain about the different types of join in Hive**.

**Ans.** There are  4 different types of **joins in Hive** –

**INNER JOIN- It returns matching rows from two tables.**

**LEFT OUTER JOIN-**  All the rows from the left table are returned even if there are no matches in the right table.

**RIGHT OUTER JOIN –** All the rows from the right table are returned even if there are no matches in the left table.

**FULL OUTER JOIN –** This join combines the records of both the left and right outer tables.

**92.How can you configure remote metastore mode in Hive?**

**Ans.**

Add the below property in hive-site.xml file   
hive.metastore.uris thrift: //IP Address:9083

**93.How does data transfer happen from HDFS to Hive?**

**Ans.**We can create the managed or external table and transfer the data between HDFS and Hive.

.

.

**Bottom of Form**

**95.Compare Pig and Hive**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Pig** | **Hive** |
| Architecture | Procedural data flow language | SQL type declarative language |
| Application | Programming purposes | Report creation |
| Operational field | Client side | Server side |
| Support for avro files | Yes | No |
| Type of Data | Apache Pig is usually used for semi structured data. | Used for Structured Data |
| Schema | Schema is optional. | Hive requires a well-defined Schema. |

**100. What are the default record and field delimiter used for hive text files?**

The default record delimiter is − \n

And the filed delimiters are − \001,\002,\003

**102. How do you list all databases whose name starts with p?**

SHOW DATABASES LIKE ‘p.\*’

**103. What does the “USE” command in hive do?**

For choosing particular database

**104. How can you delete the DBPROPERTY in Hive?**

There is no way to delete the DBPROPERTY.

**105. How do you check if a particular partition exists?**

This can be done with following query

SHOW PARTITIONS table\_name PARTITION(partitioned\_column=’partition\_value’)

**106. Explain the concatenation function in Hive with an example .**

Concatenate function will join the input strings.

We can specify the ‘N’ number of strings separated by a comma.  
Example:

CONCAT ('Heena','-','is','-','a','-','hadoop',’-’,’developer’);

If string have common delimiter the we can use

CONCAT\_WS ('delimiter',’Heena’,’is’,’a’,’hadoop’,‘developer’); too

**107. Trim and Reverse function in Hive with examples.**

Trim function will delete the white spaces in a string.  
Example:

To remove the Leading space

LTRIM(‘ INTELLIPAAT’);

To remove the trailing space

RTRIM(‘INTELLIPAAT ‘);

In Reverse function, characters are reversed in the string.

Example:

REVERSE(‘INTELLIPAAT’);

**108. Explain process to access sub directories recursively in Hive queries.**

By using below commands we can access sub directories recursively in Hive

hive> Set mapred.input.dir.recursive=true;

hive> Set hive.mapred.supports.subdirectories=true;

**109. How to skip header rows from a table in Hive?**

AFTER location in create table statement we can specify

LOCATION ‘/user/data’

TBLPROPERTIES("skip.header.line.count"="2”);

**110. What is the maximum size of string data type supported by hive?**

The maximum size of string data type supported by hive is 2 GB.

**111. What is the precedence order of HIVE configuration?**

**1.** Hive-site.XML

1. Hive-default.xml
2. Hadoop-site.xml
3. Hadoop-default.xml

**112. If you run a select \* query in Hive, Why does it not run MapReduce?**

The hive.fetch.task.conversion property of Hive lowers the latency of mapreduce and hence while executing queries like SELECT, FILTER, LIMIT, etc., it skips mapreduce function

**113. How Hive can improve performance with ORC format tables?**

We can store the hive data in highly efficient manner in the Optimized Row Columnar file format. It can simplify many Hive file format limitations. We can improve the performance by using ORC files while reading, writing and processing the data.

Set hive.compute.query.using.stats-true;

Set hive.stats.dbclass-fs;

CREATE TABLE orc\_table (

idint,

name string)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ‘\:’

LINES TERMINATED BY ‘\n’

STORES AS ORC;

**116. What is available mechanism for connecting from applications, when we run hive as a server?**

1. Thrift Client: Using thrift you can call hive commands from various programming languages. Example: C++, PHP,Java, Python and Ruby.
2. JDBC Driver: JDBC Driver supports the Type 4 (pure Java) JDBC Driver
3. ODBC Driver: ODBC Driver supports the ODBC protocol.

**118. Explain the difference between partitioning and bucketing.**

* Partitioning and Bucketing of tables is done to improve the query performance. Partitioning helps execute queries faster, only if the partitioning scheme has some common range filtering i.e. either by timestamp ranges, by location, etc. Bucketing does not work by default.
* Partitioning helps eliminate data when used in WHERE clause.

Bucketing helps organize data inside the partition into multiple files so that same set of data will always be written in the same bucket. Bucketing helps in joining various columns.

* In partitioning techni, a partition is created for every uni value of the column and there could be a situation where several tiny partitions may have to be created. However, with bucketing, one can limit it to a specific number and the data can then be decomposed in those buckets.
* Basically, a bucket is a file in Hive whereas partition is a directory.

**119. Explain about SORT BY, ORDER BY, DISTRIBUTE BY and CLUSTER BY in Hive.**

SORT BY – Data is ordered at each of ‘N’ reducers where the reducers can have overlapping range of data.

ORDER BY- This is similar to the ORDER BY in SQL where total ordering of data takes place by passing it to a single reducer.

DISTRUBUTE BY – It is used to distribute the rows among the reducers. Rows that have the same distribute by columns will go to the same reducer.

CLUSTER BY- It is a combination of DISTRIBUTE BY and SORT BY where each of the N reducers gets non overlapping range of data which is then sorted by those ranges at the respective reducers.

**120. Is it possible to compress json in Hive external table ?**

Yes, you need to gzip your files and put them as is (\*.gz) into the table location.

**121.How will you optimize Hive performance**

There are various ways to run Hive ries faster -

* Using vectorization
* Tez Execution Engine – A Hive optimization techni increases the performance of hive ries by using Tez execution engine. It works on the framework of Hadoop Yarn and executes complex-directed acyclic graphs of general data processing tasks.
* ORC File Format – Optimized Row Columnar (ORC) File Format drastically improves ry performance as it stores data in a more optimized way than other formats.
* Hive Partition – By using partitions, the entries in the different columns of the dataset are separated and stored in partitions. Hence, fetching the values requires rying only the required partitions. Thus, it reduces the time taken by a ry to return results.
* Bucketing in Hive – The data in the table are divided into more manageable portions or buckets that reduces the time taken to give results.
* Vectorization– Use of vectorized ry execution to perform scans, aggregations, filters, and joins in batches of 1024 rows at once reduces the time taken to perform operations.
* Cost-Based Optimization– It this process, Hive optimizes the logical as well as physical execution plan of the ry. The results depend on how the order joins, the type of join to performed, the degree of parallelism etc.
* Hive Index – Among the best ways of optimising, it enhances the performance of the ry. The original table is indexed to create a separate index table that acts as a reference.

**122. There is a  Python application that connects to Hive database for extracting data, creating sub tables for data processing, drops temporary tables, etc. 90% of the processing is done through hive queries which are generated from python code and are sent to hive server for execution.Assume that there are 100K rows , would it be faster to fetch 100K rows to python itself into a list of tuples and mimic the join or filter operations hive performs and avoid the executuon of 20-50 ries run against hive or you should look into hive ry optimization technis ? Which one is performance efficient ?**

**123. How Facebook Uses Hadoop,Hive and Hbase ?**

Answer: Facebook data stored on HDFS, everyday millions of photos uploaded into facebook with the help of Hadoop Facebook Messages,Likes and statues updates running on top of Hbase Hive to generate reports for third-party developers and advertisers who need to track the success of their applications or campaigns.

### **124. What is the limitation of Derby database for Hive metastore?**

With derby database, you cannot have multiple connections or multiple session’s initiated at the same time. Derby database runs in the local mode and it creates a log file so that multiple users cannot access Hive simultaneously.

### **125. Which method has to be overridden when we use custom UDF in Hive?**

Whenever you write a custom UDF in Hive, you have to extend the UDF class and you have to override the **evaluate()**function.

**126.,Why you should choose Hive instead of Hadoop MapReduce?**

### **127. How is ORC file format optimised for data storage and analysis?**

ORC stores collections of rows in one file and within the collection the row data will be stored in a columnar format. With columnar format, it is very easy to compress, thus reducing a lot of storage cost.

While rying also, it ries the particular column instead of rying the whole row as the records are stored in columnar format.

ORC has got indexing on every block based on the statistics min, max, sum, count on columns so when you ry, it will skip the blocks based on the indexing.

###### **129. What is the Hive configuration precedence order?**

There is a precedence hierarchy to setting properties. In the following list, **lower numbers take precedence over higher numbers**:

1. The Hive **SET** command
2. The command line **-hiveconf** option
3. hive-site.xml
4. hive-default.xml
5. hadoop-site.xml (or, equivalently, core-site.xml, hdfs-site.xml, and mapred-site.xml)
6. hadoop-default.xml (or, equivalently, core-default.xml, hdfs-default.xml, and mapred-default.xml)

###### **130. Describe REPEAT function in Hive with example?**

REPEAT function will repeat the input string n times specified in the command.

**Example:** REPEAT(‘Hive’,3);  
**Output:** HiveHiveHive.

###### **131. What are the Binary Storage formats supported in Hive?**

By default Hive supports text file format, however hive also supports below binary formats.

Sence Files, Avro Data files, RCFiles, ORC files, Part files

**Sence files:** General binary format. Splittable, compressible and row oriented. a typical example can be. if we have lots of small file, we may use sence file as a container, where file name can be a key and content could stored as value. it support compression which enables huge gain in performance.

**Avro datafiles:** Same as Sence file splittable, compressible and row oriented except support of schema evolution and multilingual binding support.

**RCFiles:** Record columnar file, it’s a column oriented storage file. it breaks table in row split. in each split stores that value of first row in first column and followed sub subsently.

**ORC Files:** Optimized Record Columnar files

###### 132. is HQL case sensitive?

HQL is not case sensitive.

### **135. How to create a database in Hive?**

CREATE DATABASE|SCHEMA [IF NOT EXISTS] <database name>

### **139. How to update records in Hive?**

To update records in Hive, use the following syntax-

**UPDATE** <target **table**>

**SET** <**set** clause list>

[ **WHERE** <search condition> ]

**140. What type of Read and Write operations perfor**m in Hive?

Hive provides READ Many WRITE Once.

**141. What are windowing functions in Hive?**

OVER, RANK

### **142. Explain map side join in Hive?**

Map join is a feature in Hive that speeds up queries as it works without reducers.

It enables a table to be loaded into memory to perform a join within a mapper, without using the Map/Reduce step. It loads a smaller table in memory and the joins in the map phase of the Map/Reduce operation.

**TECHNICAL INTERVIEW STIONS:-**

**156. What is Managed table?**

**157. What is external table?**

**158. Tell me difference between managed table or external table**

**159. How to load the data from the hdfs ?**

**160. How to load the data from the local?**

**161. Tell the commnd to rename the tabl column?**

**162. How to add new columns into hive table ?**

**163. How to change the data types of the hive table?**

**164. What is clustering?**

**165. What is portioning and types of the partitioning?**

**166. What is custom partitioning?**

**167. What is dynamic partitioning?**

**168. How to do dynmic patitioning?**

**169. How to add the partition to hive table?**

**alter table gold\_\_sales\_sears\_pos\_tax\_general add partition (transaction\_year\_nbr='1996') location '/gold/transaction/pos/sears/tax\_aud\_s5\_general/transaction\_year\_nbr=1996';**

**hive -e "alter table gold\_\_sales\_sears\_pos\_payment add if not exists partition (transaction\_year\_nbr=$PROCESS\_YEAR);"**

**How to drop the partition to hive table?**

**72. What is Serialization?**

Serialization is the process in which Hive converts objects in different programming languages to suitable formats as a stream of bytes, which can be stored in HDFS and used by Hive. This may be done with structured or unstructured data bytes. using SerDe. It optimises and saves the state of an object to recreate it when the need arises.

**74. What is Hlog?**

Hlog used for recovering

* 1. Send heartbeat(loadinfo) to master
  2. Write requests handle
  3. Read request handle
  4. Flush
  5. Compaction
  6. Region Splits(Manage)

### Hadoop Architechture/processing

### Explain whole mapreducer processing

**75. Explain how Hive Deserialize and serialize the data?**

### Usually, while read/write the data, the user first communicate with inputformat. Then it connects with Record reader to read/write record.  To serialize the data, the data goes to row. Here deserialized custom serde use object inspector to deserialize the data in fields.

**76. Give examples of the SerDe classes which hive uses to Serialize and Deserilize data ?**

Hive currently use these SerDe classes to serialize and deserialize data:

• MetadataTypedColumnsetSerDe: This SerDe is used to read/write delimited records like CSV, tab-separated control-A separated records (quote is not supported yet.)

• ThriftSerDe: This SerDe is used to read/write thrift serialized objects. The class file for the Thrift object must be loaded first.

• DynamicSerDe: This SerDe also read/write thrift serialized objects, but it understands thrift DDL so the schema of the object can be provided at runtime. Also it supports a lot of different protocols,including TBinaryProtocol, TJSONProtocol, TCTLSeparatedProtocol (which writes data in delimited records).

**77. How do you write your own custom SerDe ? In most cases, users want to write a Deserializer** instead of a SerDe, because users just want to read their own data format instead of writing to it. •For example, the RegexDeserializer will deserialize the data using the configuration parameter ‘regex’, and possibly a list of column names. •If your SerDe supports DDL (basically, SerDe with parameterized columns and column types), you probably want to implement a Protocol based on DynamicSerDe, instead of writing a SerDe from scratch. The reason is that the framework passes DDL to SerDe through “thrift DDL” format, and it’s non-trivial to write a “thrift DDL” parser.

### **78. How is SerDe different from File format in Hive?**

SerDe stands for Serializer and Deserializer. It determines how to encode and decode the field values or the column values from a record that is: how you serialize and deserialize the values of a column

But file format determines how records are stored in key value format or how do you retrieve the records from the table.

### **79. What is RegexSerDe?**

Regex stands for a regular expression. Whenever you want to have a kind of pattern matching, based on the pattern matching, you have to store the fields. RegexSerDe is present in **org.apache.hadoop.hive.contrib.serde2.RegexSerDe.**

In the SerDeproperties, you have to define your input pattern and output fields. For example, you have to get the column values from line **xyz/pq@def** if you want to take xyz, pq and def separately.

To extract the pattern, you can use:  
‘input.regex’ = ‘(.\*)/(.\*)@(.\*)’  
To specify how to store them, you can use  
‘output.format.string’ = ‘%1$s%2$s%3$s’;

**=============================================================================**

**PIG**

==========================================================================.

**1. What are different modes of execution in Apache Pig?**

Answer: Apache Pig runs in 2 modes-

* + 1. (Local Mode) - In Local mode , we need local host and all the files are stage on local host.
    2. Hadoop MapReduce (Java) - In MapReduce mode, we need the Hadoop cluster and all the files are stage on hadoop cluster.

2. **Does Pig support multi-line commands?**

Answer: Yes same as java /\*\*/

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

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

**DUMP** displays the results on screen.

**DESCRIBE** displays the schema of a particular relation.

**ILLUSTRATE** displays step by step execution of pig statements.

**EXPLAIN** displays the execution plan for pig latin statements.

**4. What Are The Debugging Tools Used For Apache Pig Scripts?**

describe and explain are the important debugging utilities in Apache Pig.

**5. What is the difference between store and dumps commands?**

Answer: Dump Command output data displayed on the terminal, but it’s not stored anywhere. For store command data is store in local file system or HDFS.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*All The Best\*\*\*\*\*\*\*\*\*\*\*

**6. Explain The Need For Mapreduce While Programming In Apache Pig.?**

The Pig engine converts the queries into MapReduce jobs and thus MapReduce acts as the execution engine and is needed to run the programs.

7. **Explain About The Bloommapfile.?**

BloomMapFile is a class, that extends the MapFile class. It is used in HBase table format to provide quick membership test for the keys using dynamic bloom filters.

8**. What Do You Mean By A Bag In Pig?**

Collection of tuples

**9. What Is The Usage Of Foreach Operation In Pig Scripts?**

**Answer :**

FOREACH operation in Apache Pig is used to apply transformation to each element in the data bag, it is used to select column names of relation that user wants.

**Syntax-** FOREACH data\_bagname GENERATE exp1, exp2.

**10. Explain About The Different Complex Data Types In Pig.?**

**Answer :**

**Apache Pig supports 3 complex data types:**

**Maps-**These are key, value stores joined together using #.

**Tuples-** Just similar to the row in a table, where different items are separated by a comma.

**Bags-** Unordered collection of tuples. Bag allows multiple duplicate tuples.

**11. What Does Flatten Do In Pig?**

Sometimes there is data in a tuple or a bag and if we want to remove nesting from that data, then Flatten modifier in Pig can be used. Flatten un-nests bags and tuples. Eg. For tuples, the Flatten operator will replace tuples with the fields

**12. How Do Users Interact With The Shell In Apache Pig?**

Using Grunt

**13. To start Grunt in local mode use**

“pig –x local”

**14.To start grunt in mapreduce mode use**

pig

`

**17. What Do You Know About The Case Sensitivity Of Apache Pig?**

**Answer :**

It is difficult to say whether Apache Pig is case sensitive or case insensitive. For eg. user defined functions, relations and field names in pig are case sensitive i.e. the function  COUNT is not the same as function count

or

A=load ‘foo’ is not same as a=load ‘foo’.

keywords in Apache Pig are case insensitive i.e. LOAD is same as load.

19**. Differentiate Between Piglatin And Hiveql?**

It is necessary to specify the schema in HiveQL, whereas it is optional in PigLatin.

HiveQL is a declarative language, whereas PigLatin is procedural.

HiveQL follows a flat relational data model, whereas PigLatin has nested relational data model.

* 1. **Is Piglatin A Strongly Typed Language? If Yes, Then How Did You Come To The Conclusion?**

**Answer :**

In a strongly typed language, the user has to declare the type of all variables upfront. In Apache Pig, when you describe the schema of the data, it expects the data to come in the same format you mentioned.

However, when the schema is not known, the script will adapt to actually data types at runtime. So, it can be said that PigLatin is strongly typed in most cases but in rare cases it is gently typed.

**21. What Do You Understand By An Inner Bag And Outer Bag In Pig?**

A relation inside a bag is referred to as inner bag and outer bag is just a relation in Pig.

**22. Differentiate Between Group And Cogroup Operators.?**

**Answer :**

Both GROUP and COGROUP operators are identical and can work with one or more relations.

GROUP operator is generally used to group the data in a single relation for better readability,

whereas COGROUP can be used to group the data in 2 or more relations. COGROUP is combination of GROUP and JOIN, i.e., it groups the tables based on a column and then joins them on the grouped columns.

It is possible to cogroup up to 127 relations at a time.

**23. Explain The Difference Between Count\_star And Count Functions In Apache Pig?**

COUNT function does not include the NULL value when counting the number of elements in a bag,

whereas COUNT\_STAR (0 function includes NULL values while counting.

**24. How Will You Merge The Contents Of Two Or More Relations And Divide A Single Relation Into Two Or More Relations?**

This can be accomplished using the UNION and SPLIT operators.

**25. I Have A Relation R. How Can I Get The Top 10 Tuples From The Relation R.?**

TOP (10)

**26. What Are The Commonalities Between Pig And Hive?**

HiveQL and PigLatin both convert the commands into MapReduce jobs.

They cannot be used for OLAP transactions as it is difficult to execute low latency ries.

**27. What Are The Different Types Of Udf’s In Java Supported By Apache Pig?**

Algebraic, Eval and Filter functions

**28. You Have A File Employee.txt In The Hdfs Directory With 100 Records. You Want To See Only The First 10 Records From The Employee.txt File. How Will You Do This?**

**In Pig**

The first step would be to load the file employee.txt into with the relation name as Employee.

The first 10 records of the employee data can be obtained using the limit operator -

Result= limit employee 10.

In HDFS :-

Hadoop fs -/use/hshaik0/employee.txt | head -5

* 1. **Explain About The Scalar Datatypes In Apache Pig.?**

integer, float, double, long, bytearray and char array

**30. How Do Users Interact With Hdfs In Apache Pig?**

Using the grunt shell.

* 1. **What Is The Use Of Having Filters In Apache Pig?**

Apache Pig has filters records based on a given condition or predicate.

**Example:-**

X= load ‘inputs’ as(name,address)

Y = filter X by symbol matches ‘Mr.\*’;

* 1. **What Is A Udf In Pig?**

If the in-built operators do not provide some functions then programmers can implement those functionalities by writing user defined functions using other programming languages like Java, Python, Ruby, etc. These User Defined Functions (UDF’s) can then be embedded into a Pig Latin Script.

**33. Can You Join Multiple Fields In Apache Pig Scripts?**

Yes, it is possible to join multiple fields in PIG scripts because the join operations takes records from one input and joins them with another input. This can be achieved by specifying the keys for each input and the two rows will be joined when the keys are equal.

**35. How to cobmbine / concatenate Strings?**

CONCAT(“fruits”,”apple”)

**36. How to FILTER the data?**

FILTER grouped\_data\_gen

BY (int)eligible\_store\_record\_count > 0;

**37. How to sorting in Pig?**

ORDER data by c1,c2,c3;

**38. How can we do optimization in Pig?**

1: Adding compression techniques

and adjusting the mappers\*/

set io.compression.codec.lzo.class com.hadoop.compression.lzo.LzoCodec

set pig.tmpfilecompression true

set pig.tmpfilecompression.codec lzo

2:Setting Mappers:-

set mapred.compress.map.output true

set mapred.min.split.size 524288;

SET mapred.max.split.size 134217728

SET pig.maxCombinedSplitSize 4000000

3:Setting reducers

SET mapreduce.map.java.opts: -Xmx3072m

SET mapreduce.reduce.java.opts: -Xmx6144m

/\*

**39. How to generate map in pig?**

HASH\_MAP = FOREACH FILE1 GENERATE TOMAP(key,value);

**40. How to split Strig?**

TIMESTAMP\_NOW = FOREACH FILTER\_HEADER\_FILE GENERATE STRSPLIT($20,' ',2) as timestamp;

YEAR\_MONTH\_DATE = FOREACH SPLITANTS\_OF\_DATE GENERATE STRSPLIT($0,'-',3);

SPLITANTS = FOREACH TIMESTAMP\_NOW GENERATE FLATTEN(TOBAG(timestamp));

**41. How to make string smallcase?**

VENDOR\_CHK\_STR\_GENERATE = FOREACH REPLACE\_BLANKS GENERATE LOWER($1) AS VENDOR\_CHK\_STR;

**42. Give me cogroup Example?**

FINAL\_OUTPUT = COGROUP $LOAD\_TS BY $0,YEAR\_MONTH\_DATE\_DAY.YEAR BY $0

**43. Compare Apache Pig And Sql?**

**Answer :**

Apache Pig differs from SQL in its usage for ETL, lazy evaluation, store data at any given point of time in the pipeline, support for pipeline splits and explicit declaration of execution plans. SQL is oriented around ries which produce a single result. SQL has no in-built mechanism for splitting a data processing stream and applying different operators to each sub-stream.

Apache Pig allows user code to be included at any point in the pipeline whereas if SQL where to be used data needs to be imported to the database first and then the process of cleaning and transformation beginsTECHNICAL INTERVIEW QUSTIONS

**44. Write a syntax for loading the file ?**

a = load '/home/auto/hshaik0/simple1.txt' USING PigStorage(',') as (id:chararray,name:chararray,adress:chararray,local\_id:chararray);

**45. Explain the joins in pig?**

**46. Which are different types of the storages in pig**

**47. How we access csv files in pig?**

**48. How to define variable in Pig**

%declare load\_ts `date +"%Y-%m-%d %H:%M:%S"`

url\_chk\_str matches '.\* KMART.COM.\*' ? 'Kmart mWeb' :'1'

VENDOR\_CHK\_STR matches '.\*(huawei).\*' ? 'huawei' : VENDOR\_CHK\_STR

**49. How to check data is empty or not?**

IsEmpty() OR IsNull?

**50. Why we use flatten and group ?**

51. What is bag and how it forms?

By using TOBAG(timestamp)

**52. Why we use trim?**

**53. How to do casting in Pig?**

(chararray)(int)(name);

**SQOOP**

**1.What is Sqoop?**

“**SQ**L to Had**oop** and Hadoop to SQL”

Open source to transfer data between Hadoop and relational databases , Sqoop is the best tool

Sqoop uses two main tools. Like:

* [**Sqoop import**](https://data-flair.training/blogs/sqoop-import/) (Copy data from RDBMS to HDFS)
* [**Sqoop export**](https://data-flair.training/blogs/sqoop-export/) (Copy data from HDFS to RDBMS)

**4. How can you control the number of mappers used by the sqoop command?**  
**Ans.**To control the number of mappers executed by a sqoop command we use the parameter –num-mappers.

**5. What is the default extension of the files produced from a sqoop import using the –compress parameter?**  
**Ans.** .gz,tar

**6. What are the basic commands in Apache Sqoop and its uses?**

uses of Apache Sqoop basic commands are:

1. Codegen- It helps to generate code to interact with database records.
2. Create- hive-table- It helps to Import a table definition into a hive
3. Eval- It helps to evaluate SQL statement and display the results
4. Export- It helps to export an HDFS directory into a database table
5. Help- It helps to list the available commands
6. Import- It helps to import a table from a database to HDFS
7. Import-all-tables- It helps to import tables from a database to HDFS
8. List-databases- It helps to list available databases on a server
9. List-tables- It helps to list tables in a database
10. Version- It helps to display the version information

**7. How Sqoop word came? Sqoop is which type of tool and the main use of sqoop?**  
**Ans.** Sqoop word came from **SQ**L+HAD**OOP**=SQOOP.   
Basically, it is a data transfer tool. We use Sqoop to import and export a large amount of data from RDBMS to HDFS and vice versa.

Apache Sqoop – Basically, for importing data in Apache Sqoop, HDFS is the destination  
Apache Flume – In Apache Flume, data generally flow to HDFS through channels

**e. Architecture**  
Apache Sqoop – Basically, it has connector based architecture. However, that means the connectors know a great deal in connecting with the various data sources. Also to fetch data correspondingly.  
Apache Flume – However, it has agent-based architecture. Basically, it means code written in Flume is we call agent that may responsible for fetching the data.  
Also, learn complete comparison, follow link [**Apache Sqoop vs Flume- Comparison**](https://data-flair.training/blogs/apache-sqoop-vs-flume/)

### **11. I have around 300 tables in a database. I want to import all the tables from the database except the tables named Table298, Table 123, and Table299. How can I do this without having to import the tables one by one?**

This can be accomplished using the import-all-tables import command in Sqoop and

Then specifying the exclude-tables option with it as follows-

sqoop import-all-tables

--connect –username –password

--exclude-tables Table298, Table 123, Table 299

**13. What is the default database of Apache Sqoop?**  
The default database of Apache Sqoop is MySQL.

18.**Shed light on the advantage of utilizing –password-file rather than –P option**  
The –password-file option is usually used inside the Sqoop script file. On the other hand, the –P option is able to read the standard input along with the column name parameters.

**19. Is the JDBC driver fully capable to connect Sqoop on the databases?**  
The JDBC driver is not capable to connect Sqoop on the databases. This is the reason that Sqoop requires both the connector and JDBC driver.

* 1. **Describe the procedure involved in executing an incremental data load in Sqoop**

You should be well aware of the fact that in Sqoop, the process of performing additional data load is to update the uploaded data. This data is often referred to as delta data. In Sqoop, this delta data can be altered with the use of incremental load command. Additionally, it can be said that with the help of Sqoop, the import command can also perform additional load. By loading the data into the hive without overwriting it, its efficiency can be maintained in a significant manner. This is possible only with the help of incremental data load.

It is also essential for you to illustrate the various types of incremental data load. They are as follows:  
Progressive Mode: This variety usually determines the number of new rows. Moreover, it also possesses a value that can best resemble the Append functions.

**Value:** This denotes the maximum amount that is derived from the check column from the previous import operation.

**The Check Column feature:** This function is helpful in specifying the number of columns that should be assessed to determine the number of rows to be imported.

**26. Shed light on the procedure of updating the rows that have been directly exported**  
In order to update the existing rows that have been exported, you have to use a particular parameter. This parameter is in the form of update key. You can also opt to use a list of comma-separated commands. This would help you to identify a row in a uni fashion. A majority of the columns are used in the Where clause of the update ry that has been already been generated. Moreover, all the other types of table columns should be used in the SET portion of the generated ry.

**28. Define Sqoop metastore**  
It is also known as a shared metadata repository with the help of which the local users can execute and define various types of list tables. In order to connect to the metastore, you have to make changes to the Sqoop –site.xml.

**29. Does Sqoop uses the maps reduce function? If it does then shed light on the reasons**  
Apache Sqoop also uses the Map-Reduce function of Hadoop to obtain data from the relational databases. During the process of importing data, Sqoop controls the mappers and their numbers. The mappers who access RDBMS come across denial of service attacks. Hence, it can be said that with the help of Sqoop, big data can be efficiently managed.

**34. How will you update the rows that are already exported? Write sqoop command to show all the databases in MySQL server.**

By using the parameter – update-key we can update existing rows. Comma-separated list of columns is used which unily identifies a row. All of these columns are used in the WHERE clause generated UPDATE ry. All other table columns will be used in the SET part of the ry.  
The command below is used to show all the databases in MySQL server.

$ sqoop list –databases –connect jdbc:mysql://database.test.com/

**35. The purpose of sqoop-merge is:**  
This tool combines 2 datasets where entries in one dataset overwrite entries of an older dataset preserving only the new version of the records between both the data sets.

**37. How to enter into Mysql prompt, and explain the command parameter indicates?**

The command for entering into Mysql prompt is “mysql –u root –p”  
-u indicatesthe user  
Root indicates username  
-p indicates password.

**40. How can you list all the columns of a table using Apache sqoop?**

There is no straight way to list all the columns of a table in Apache Sqoop like sqoop-list-columns, so first we should retrieve the columns of the particular table and transform to a file containing the column names of particular table.Syntax is:

Sqoop import –m1 –connect ‘jdbc:sqlserver://servername;database=databasename;

Username-DeZyre;password=mypassword’ –ry “SELECT column\_name,DATA\_TYPE FROM INFORMATION\_SCHEMA columns WHEREtable\_name=’mytableofinterest’ AND \$CONDITIONS” –target-dir ‘mytableofinterest\_column\_name’.

**43. For each sqoop copying into HDFS how many MapReduce jobs and tasks will be submitted?**

There are 4 jobs that will be submitted to each Sqoop copying into HDFS and no reduce tasks are scheduled.

**44. How can Sqoop be used in Java programs?**

In the Java code Sqoop jar is included in the classpath. The required parameters are created to Sqoop programmatically like for CLI (command line interface). Sqoop.runTool() method also invoked in Java code.

**45. I am having around 500 tables in a database. I want to import all the tables from the database except the tables named Table 498, Table 323, and Table 199. How can we do this without having to import the tables one by one?**

This can be proficient using the import-all-tables, import command in Sqoop and by specifying the exclude-tables option with it as follows-  
sqoop import-all-tables  
–connect –username –password –exclude-tables Table498, Table 323, Table 199

**46. Explain the significance of using –split-by clause in Apache Sqoop?**

split-by is a clause, it is used to specify the columns of the table which are helping to generate splits for data imports during importing the data into the Hadoop cluster. This clause specifies the columns and helps to improve the performance via greater parallelism. And also it helps to specify the column that has an even distribution of data to create splits,that data is imported.

**47. What is the process to perform an incremental data load in Sqoop?**

Answer: The process to perform incremental data load in Sqoop is to synchronize the modified or updated data (often referred as delta data) from RDBMS to Hadoop. The delta data can be facilitated through the incremental load command in Sqoop.

Incremental load can be performed by using Sqoop import command or by loading the data into hive without overwriting it. The different attributes that need to be specified during incremental load in Sqoop are-

* Mode (incremental) –The mode defines how Sqoop will determine what the new rows are. The mode can have value as Append or Last Modified.
* Col (Check-column) –This attribute specifies the column that should be examined to find out the rows to be imported.
* Value (last-value) –This denotes the maximum value of the check column from the previous import operation.

command line.

**48. What is the significance of using –compress-codec parameter?**

Answer: To get the out file of a sqoop import in formats other than .gz like .bz2 we use the –compress -code parameter.

**52. How can you check all the tables present in a single database using Sqoop?**

Answer: The command to check the list of all tables present in a single database using Sqoop is as follows-

Sqoop list-tables –connect jdbc: mysql: //localhost/user;

**53. What is the standard location or path for Hadoop Sqoop scripts?**

Answer: /usr/bin/Hadoop Sqoop.

**54. How can we import a subset of rows from a table without using the where clause?**

Answer: We can run a filtering ry on the database and save the result to a temporary table in database.

Then use the sqoop import command without using the –where clause.

**59. How can you see the list of stored jobs in sqoop metastore?**

Answer: sqoop job –list

**70. What type of databases Sqoop can support?**

Answer: MySQL, Oracle, PostgreSQL, IBM, Netezza and Teradata. Every database connects through jdbc driver.

**71. What is the purpose of sqoop-merge?**

Answer: The merge tool combines two datasets where entries in one dataset should overwrite entries of an older dataset preserving only the newest version of the records between both the data sets.

**72. Can we import the data with “Where” condition?**

Answer: Yes, Sqoop has a special option to export/import a particular data.

**73. What are the limitations of importing RDBMS tables into Hcatalog directly?**

Answer: There is an option to import RDBMS tables into Hcatalog directly by making use of –hcatalog –database option with the –hcatalog –table but the limitation to it is that there are several arguments like –as-avro file , -direct, -as-sencefile, -target-dir , -export-dir are not supported.

**74. what are the majorly used commands in sqoop?**

Answer: In Sqoop Majorly Import and export command are used. But below commands are also useful sometimes. codegen, eval, import-all-tables, job, list-database, list-tables, merge, metastore.

**75. What is the usefulness of the options file in sqoop.**

Answer: The options file is used in sqoop to specify the command line values in a file and use it in the sqoop commands.

For example the –connect parameter’s value and –user name value scan be stored in a file and used again and again with different sqoop commands.

**76. what are the common delimiters and escape character in sqoop?**

Answer: The default delimiters are a comma(,) for fields, a newline(\n) for records

Escape characters are \b,\n,\r,\t,\”, \\’,\o etc

**77. while loading table from MySQL into HDFS, if we need to copy tables with maximum possible speed, what can you do?**

Answer: We need to use -direct argument in import command to use direct import fast path and this -direct can be used only with MySQL and PostGreSQL as of now.

**78. How can you sync a exported table with HDFS data in which some rows are deleted?**

Answer: Truncate the target table and load it again.

**79. Differentiate between Sqoop and distCP.**

Answer: DistCP utility can be used to transfer data between clusters whereas Sqoop can be used to transfer data only between Hadoop and RDBMS.

**80. How can you import only a subset of rows form a table?**

Answer: By using the WHERE clause in the sqoop import statement we can import only a subset of rows.

**81. How do you clear the data in a staging table before loading it by Sqoop?**

Answer: By specifying the –clear-staging-table option we can clear the staging table before it is loaded. This can be done again and again till we get proper data in staging.

 82. **Is it possible to do an incremental import using Sqoop?**

Answer: Yes, Sqoop supports two types of incremental imports-

1. Append
2. Last Modified

To insert only rows Append should be used in import command and for inserting the rows and also updating Last-Modified should be used in the import command.

**83. How can you export only a subset of columns to a relational table using sqoop?**

Answer: By using the –column parameter in which we mention the required column names as a comma separated list of values.

**84. Which database the sqoop metastore runs on?**

Answer: Running sqoop-metastore launches a shared HSQLDB database instance on the current machine.

**85. How will you update the rows that are already exported?**

Answer: The parameter –update-key can be used to update existing rows. In it a comma-separated list of columns is used which unily identifies a row. All of these columns is used in the WHERE clause of the generated UPDATE ry. All other table columns will be used in the SET part of the ry.

**86. You have a data in HDFS system, if you want to put some more data to into the same table, will it append the data or overwrite?**

Answer: No it can’t overwrite, one way to do is copy the new file in HDFS.

**94. Is JDBC driver enough to connect sqoop to the databases?**

No. Sqoop needs both JDBC and connector to connect to a database.

When to use --target-dir and when to use --warehouse-dir while importing data?How can you import only a subset of rows form a table?

By using the WHERE clause in the sqoop import statement we can import only a subset of rows.

**95. How can we import a subset of rows from a table without using the where clause?**

We can run a filtering ry on the database and save the result to a temporary table in database.

Then use the sqoop import command without using the --where clause

**96. What is the advantage of using --password-file rather than -P option while preventing the display of password in the sqoop import statement?**

The --password-file option can be used inside a sqoop script while the -P option reads from standard input , preventing automation.

**97. How can you avoid importing tables one-by-one when importing a large number of tables from a database?**

Using the command

sqoop import-all-tables

--connect

--usrename

--password

--exclude-tables table1,table2 ..

This will import all the tables except the ones mentioned in the exclude-tables clause.

**98.When the source data keeps getting updated frently, what is the approach to keep it in sync with the data in HDFS imported by sqoop?**

sqoop can have 2 approaches.

**a** − To use the --incremental parameter with append option where value of some columns are checked and only in case of modified values the row is imported as a new row.

**b** − To use the --incremental parameter with lastmodified option where a date column in the source is checked for records which have been updated after the last import.

**99.What is the usefulness of the options file in sqoop.**

The options file is used in sqoop to specify the command line values in a file and use it in the sqoop commands.

For example the --connect parameter's value and --user name value scan be stored in a file and used again and again with different sqoop commands.

* 1. **Is it possible to add a parameter while running a saved job?**

Yes, we can add an argument to a saved job at runtime by using the --exec option

sqoop job --exec jobname -- -- newparameter

* 1. **How do you fetch data which is the result of join between two tables?**

By using the --ry parameter in place of --table parameter we can specify a sql ry. The result of the ry will be imported.

* 1. **How can we slice the data to be imported to multiple parallel tasks?**

Using the --split-by parameter we specify the column name based on which sqoop will divide the data to be imported into multiple chunks to be run in parallel.

* 1. **How can you choose a name for the mapreduce job which is created on submitting a free-form query port?**

By using the --mapreduce-job-name parameter. Below is a example of the command.

sqoop import \

--connect jdbc:mysql://mysql.example.com/sqoop \

--username sqoop \

--password sqoop \

--ry 'SELECT normcities.id, \

countries.country, \

normcities.city \

FROM normcities \

JOIN countries USING(country\_id) \

WHERE $CONDITIONS' \

--split-by id \

--target-dir cities \

--mapreduce-job-name normcities

Before starting the data transfer using mapreduce job, sqoop takes a long time to retrieve the minimum and maximum values of columns mentioned in –split-by parameter. How can we make it efficient?What is the difference between the parameters sqoop.export.records.per.statement and sqoop.export.statements.per.transactionHow will you implement all-or-nothing load using sqoop?

Using the staging-table option we first load the data into a staging table and then load it to the final target table only if the staging load is successful.

* 1. **How can you schedule a sqoop job using Oozie?**

Oozie has in-built sqoop actions inside which we can mention the sqoop commands to be executed.

Sqoop imported a table successfully to HBase but it is found that the number of rows is fewer than expected. What can be the cause?Give a sqoop command to show all the databases in a MySql server.

* 1. **What do you mean by Free Form Import in Sqoop?**

How can you force sqoop to execute a free form Sql ry only once and import the rows serially.In a sqoop import command you have mentioned to run 8 parallel Mapreduce task but sqoop runs only 4.

* 1. **What can be the reason?What is the importance of --split-by clause in running parallel import tasks in sqoop?**
  2. **What does this sqoop command achieve?**

$ sqoop import --connnect <connect-str> --table foo --target-dir /dest \

**111. What happens when a table is imported into a HDFS directory which already exists using the –apend parameter?**

**112. How can you control the mapping between SQL data types and Java types?**

**113. How to import only the updated rows form a table into HDFS using sqoop assuming the source has last update timestamp details for each row?**

**114. What does the following ry do?**

$ sqoop import --connect jdbc:mysql://host/dbname --table EMPLOYEES \

--where "start\_date > '2012-11-09'

**115. Give a Sqoop command to import all the records from employee table divided into groups of records by the values in the column department\_id.What does the following ry do?**

$ sqoop import --connect jdbc:mysql://db.foo.com/somedb --table sometable \

--where "id > 1000" --target-dir /incremental\_dataset --append

**116. Give a sqoop command to import data from all tables in the MySql DB DB1.Give a command to execute a stored procedure named proc1 which exports data to from MySQL db named DB1 into a HDFS directory named Dir1.What is a sqoop metastore?**

It is a tool using which Sqoop hosts a shared metadata repository. Multiple users and/or remote users can define and execute saved jobs (created with sqoop job) defined in this metastore.

Clients must be configured to connect to the metastore in sqoop-site.xml or with the --meta-connect argument.

**117. How can you see the list of stored jobs in sqoop metastore?**

sqoop job –list

**118.Give the sqoop command to see the content of the job named myjob?**

Sqoop job –show myjob

* Oracle
* Microsoft SQL
* IBM’s Netezza
* Teradata

###### **122. What are the destination types allowed in Sqoop Import command?**

Currently Sqoop Supports data imported into below services.

* HDFS
* Hive
* HBase
* HCatalog
* Accumulo

###### **123. Is Sqoop similar to distcp in hadoop?**

Partially yes, hadoop’s **distcp** command is similar to Sqoop Import command. Both submits parallel map-only jobs but **distcp** is used to copy any type of files from Local FS/HDFS to HDFS and Sqoop is for transferring the data records only between RDMBS and Hadoop eco system services, HDFS, Hive and HBase.

###### **124. How Many Mapreduce jobs and Tasks will be submitted for Sqoop copying into HDFS?**

For each sqoop copying into HDFS only one mapreduce job will be submitted with **4 map tasks**. There will not be any reduce tasks scheduled.

###### **125. How can we control the parallel copying of RDBMS tables into hadoop ?**

We can control/increase/decrease speed of copying by configuring the number of map tasks to be run for each sqoop copying process. We can do this by providing argument **-m 10 or  –num-mappers 10 argument**to sqoop import command. If we specify **-m 10**then it will submit 10 map tasks parallel at a time. Based on our requirement we can increase/decrease this number to control the copy speed.

###### **126. What is the criteria for specifying parallel copying in Sqoop with multiple parallel map tasks?**

To use multiple mappers in Sqoop, RDBMS table must have one **primary key column** (if present) in a table and the same will be used as split-by column in Sqoop process. If primary key is not present, we need to provide any uni key column or set of columns to form uni values and these should be provided to **-split-by** column argument.

###### **127. While loading tables from MySQL into HDFS, if we need to copy tables with maximum possible speed, what can you do ?**

We need to use **–direct** argument in import command to use direct import fast path and this –direct can be used only with MySQL and PostGreSQL as of now.

###### **128. What is the example connect string for Oracle database to import tables into HDFS?**

We need to use Oracle JDBC Thin driver while connecting to Oracle database via Sqoop. Below is the sample import command to pull table **employees** from oracle database **testdb**.

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | sqoop import \  --connect jdbc:oracle:thin:@oracle.example.com/testdb \  --username SQOOP \  --password sqoop \  --table employees |

###### **129. While connecting to MySQL through Sqoop, I am getting Connection Failure exception what might be the root cause and fix for this error scenario?**

This might be due to insufficient permissions to access your MySQL database over the network. To confirm this we can try the below command to connect to MySQL database from Sqoop’s client machine.

|  |  |
| --- | --- |
| 1  2 | $ mysql --host=MySql node&gt; --database=test --user= --password= |

If this is the case then we need grant permissions user @ sqoop client machine as per the answer to stion 6 in this post.

###### **130. While importing tables from Oracle database, Sometimes I am getting java.lang.IllegalArgumentException: Attempted to generate class with no columns! or NullPointerException what might be the root cause and fix for this error scenario?**

**While dealing with Oracle database from Sqoop, Case sensitivity of table names and user names matters highly. Most probably by specifying these two values in UPPER case will solve the issue unless actual names are mixed with Lower/Upper cases. If these are mixed, then we need to provide them within double quotes.**

**In case, the source table is created under different user namespace, then we need to provide table name as USERNAME.TABLENAME as shown below.**

MySQL

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | sqoop import \  --connect jdbc:oracle:thin:@oracle.example.com/ORACLE \  --username SQOOP \  --password sqoop \  --table SIVA.EMPLOYEES |

**131. The incoming value from HDFS for a particular column is NULL. How will you load that row into RDBMS in which the columns are defined as NOT NULL?**  
Using the –input-null-string parameter, a default value can be specified so that the row gets inserted with the default value for the column that it has a NULL value in HDFS.

**132. What is the significance of using –split-by clause for running parallel import tasks in Apache Sqoop?**  
–Split-by clause is used to specify the columns of the table that are used to generate splits for data imports. This clause specifies the columns that will be used for splitting when importing the data into the Hadoop cluster. —split-by clause helps achieve improved performance through greater parallelism. Apache Sqoop will create splits based on the values present in the columns specified in the –split-by clause of the import command. If the –split-by clause is not specified, then the primary key of the table is used to create the splits while data import. At times the primary key of the table might not have evenly distributed values between the minimum and maximum range. Under such circumstances –split-by clause can be used to specify some other column that has even distribution of data to create splits so that data import is efficient.

**133. You successfully imported a table using Apache Sqoop to HBase but when you ry the table it is found that the number of rows is less than expected. What could be the likely reason?**  
If the imported records have rows that contain null values for all the columns, then probably those records might have been dropped off during import because HBase does not allow null values in all the columns of a record.

**134. Explain the significance of using –split-by clause in Apache Sqoop?**  
split-by is a clause, it is used to specify the columns of the table which are helping to generate splits for data imports during importing the data into the Hadoop cluster. This clause specifies the columns and helps to improve the performance via greater parallelism. And also it helps to specify the column that has an even distribution of data to create splits,that data is imported.

**135. If the source data gets updated every now and then, how will you synchronise the data in HDFS that is imported by Sqoop?**  
Data can be synchronised using incremental parameter with data import –

–Incremental parameter can be used with one of the two options-

i) **append**-If the table is getting updated continuously with new rows and increasing row id values then incremental import with append option should be used where values of some of the columns are checked (columns to be checked are specified using –check-column) and if it discovers any modified value for those columns then only a new row will be inserted.

ii) **lastmodified** – In this kind of incremental import, the source has a date column which is checked for. Any records that have been updated after the last import based on the lastmodifed column in the source, the values would be updated.

**136. How can Sqoop be used in Java programs?,**  
In the Java code Sqoop jar is included in the classpath. The required parameters are created to Sqoop programmatically like for CLI (command line interface). Sqoop.runTool() method also invoked in Java code.

**137. Below command is used to specify the connect string that contains hostname to connect MySQL with local host and database name as test\_db –**  
–connect jdbc: mysql: //localhost/test\_db

**138. Is the above command the best way to specify the connect string in case I want to use Apache Sqoop with a distributed hadoop cluster?**

When using Sqoop with a distributed Hadoop cluster the URL should not be specified with localhost in the connect string because the connect string will be applied on all the DataNodes with the Hadoop cluster. So, if the literal name localhost is mentioned instead of the IP address or the complete hostname then each node will connect to a different database on their localhosts. It is always suggested to specify the hostname that can be seen by all remote nodes.

**139. I am having around 500 tables in a database. I want to import all the tables from the database except the tables named Table498, Table 323, and Table199. How can we do this without having to import the tables one by one?**  
This can be proficient using the import-all-tables, import command in Sqoop and by specifying the exclude-tables option with it as follows-  
sqoop import-all-tables  
–connect –username –password –exclude-tables Table498, Table 323, Table 199

**140. You use –split-by clause but it still does not give optimal performance then how will you improve the performance further.**  
Using the –boundary-ry clause. Generally, sqoop uses the SQL ry select min (), max () from to find out the boundary values for creating splits. However, if this ry is not optimal then using the –boundary-ry argument any random ry can be written to generate two numeric columns.

**141. During sqoop import, you use the clause –m or –numb-mappers to specify the number of mappers as 8 so that it can run eight parallel MapReduce tasks, however, sqoop runs only four parallel MapReduce tasks. Why?**  
Hadoop MapReduce cluster is configured to run a maximum of 4 parallel MapReduce tasks and the sqoop import can be configured with number of parallel tasks less than or equal to 4 but not more than 4.

**153. Where Sqoop come into picture in Hadoop eco system?**

Sqoop comes into picture. Sqoop acts like a intermediate layer between Hadoop and relational database systems

**154. Whether Sqoop will do aggregations?**

Sqoop just imports and exports the data; it does not do any aggregations.

**155. How to check sqoop version?**

The Sqoop version can be checked using the following command in the command line:

sqoop version

For example,  numb-mappers 10.

**156. What is sqoop direct mode?**

Using the direct mode in Sqoop, we can mention that Sqoop must use the direct import channel. This argument is used with the import command and can be used only with **PostGreSQL and MySql**.

**157. How to delete sqoop job?**

A job can be deleted in Sqoop by the following syntax:

--delete <job id>

This is used to delete save d jobs.

**168. How much memory does a sqoop client require to initialize the job?**

A Sqoop client may require a minimum of 1Gb of memory to initialize a job.

**169. How to pass the schema name in the sqoop?**

The schema name can be provided by using the following syntax:

$ **sqoop** import ... --table custom\_table -- --schema custom\_schema

**170. What is the syntax to see db in sqoop?**

The databases can be viewed by using the following command:

$sqoop  **list**-databases

**171. Why to use $conditions in sqoop?**

The $conditions are used in Sqoop to support parallelism while splitting your ry into parts. Inside the WHERE clause of the ry, the $CONDITIONS placeholder is placed. It is used to provide a WHERE clause that is explicit.

**172. How the increment works in sqoop, do we have to manually provide the last modified value every time?**

The incremental imports in Sqoop is of two types, lastmodified and append. The incremental import allows the users to obtain the rows that are recently imported, rather than the older rows.

When an import ends, the last value that can be used for another incremental import is provided in the console.  This value must be provided in subsent incremental imports to ensure that updated information is imported.

**173. How much memory does a sqoop client require to initialize the job?**

A Sqoop client may require a minimum of 1Gb of memory to initialize a job.

**SPARK**

**What is Spark ?**

Framework

* open-source
* Fast
* in-memory
* Distributed
* Big data processing

### **4. What are the languages supported by Apache Spark and which is the most popular one?**

Apache Spark supports the following four languages:

Scala, Java, Python and R.

### **6. What is YARN?**

### **YARN is useful for resource management and job scheduling.**

YARN :-

* yet another resource negotiater
* mapreduce 2

YARN conisit of 3 units :-

**Global resource manager:-**

Allocates resources to hadoop appliations

schedule task on datanodes uses fifo or fair scheduler(equally shares resource across multiple clusters)

**Node manager:-** its is installed on every node and manages resources available on particular node.

**Application Master:**­-works with node manager to start container -which actually launches and monitor resouces.

Advantage over mapreduce:-

Mapreduce :- Batch processing

YARN :- Real time processing

Datanode -75



Application Master

Creates container processes and use resource information reports created by node manager to help container process

Container

Actually launches and monitor resouces

Node Manager= Task Tracker

Manages resources on particular datanode

By creating and seending resource information reports(active,inactive re souces) to Resouce Manager and application master

Allocates resources to hadoop appliations

schedule task on datanodes uses fifo or fair scheduler(equally shares resource across multiple clusters)

Gloabal resorce Manager

### **7. Do you need to install Spark on all nodes of YARN cluster?**

No , spark can use yarn features directly by configuring some options eg. master, deploy-mode, driver-memory, executor-memory, executor-cores.

### **Explain the concept of Resilient Distributed Dataset (RDD).**

RDD -Resilient Distribution Datasets- Immutable distributed dataset

Immutable –unchanged- readonly

Distributed – across the cluster nodes

Dataset – collection of objects

We can create RDD by using following methods:-

Sc.Parallelize

makeRDD

toRDD to conver dataframe into RDD

#### [**104. what is the difference between dataframe and dataset?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

**Dataframe**

Immutable

Distributed

In memory

#### Collection of data

#### data is map to **named columns**

eg. table

we can create dataframe by 4 ways:-

* [Create Spark DataFrame from RDD](https://sparkbyexamples.com/spark/different-ways-to-create-a-spark-dataframe/#from-rdd)
* import spark.implicits.\_
* val columns = Seq("language","users\_count")
* val data = Seq(("Java", "20000"), ("Python", "100000"), ("Scala", "3000"))
* val rdd = spark.sparkContext.parallelize(data)
* val dfFromRDD1 = rdd.toDF()

columns :- .\_1 ,.\_2

val dfFromRDD1 = rdd.toDF("language","users\_count")

columns :- language,users\_count)

val dfFromRDD2 = spark.createDataFrame(rdd).toDF(columns:\_\*)

//From RDD (USING createDataFrame and Adding schema using StructType)

val schema = StructType(columns

.map(fieldName => StructField(fieldName, StringType, nullable = true)))

//convert RDD[T] to RDD[Row]

val rowRDD = rdd.map(attributes => Row(attributes.\_1, attributes.\_2))

val dfFromRDD3 = spark.createDataFrame(rowRDD,schema)

* [Create Spark DataFrame from List and Seq collection](https://sparkbyexamples.com/spark/different-ways-to-create-a-spark-dataframe/#from-collection)
* importing import spark.implicits.\_ to use toDF()
* val dfFromData1 = data.toDF()
* //From Data (USING createDataFrame)
* var dfFromData2 = spark.createDataFrame(data).toDF(columns:\_\*)
* [Creating Spark DataFrame from CSV file](https://sparkbyexamples.com/wp-admin/post.php?post=618&action=edit#from-csv)
* val df2 = spark.read
* .csv("/src/resources/file.csv")
* [Creating from TXT file](https://sparkbyexamples.com/spark/different-ways-to-create-a-spark-dataframe/#from-txt)
* val df2 = spark.read
* .text("/src/resources/file.txt")
* [Creating from JSON file](https://sparkbyexamples.com/spark/different-ways-to-create-a-spark-dataframe/#from-json)
* val df2 = spark.read
* .json("/src/resources/file.json")
* [Creating from an XML file](https://sparkbyexamples.com/spark/different-ways-to-create-a-spark-dataframe/#from-xml)
* val df = spark.read
* .format("com.databricks.spark.xml")
* .option("rowTag", "person")
* .xml("src/main/resources/persons.xml")
* [Creating from HIVE](https://sparkbyexamples.com/spark/different-ways-to-create-a-spark-dataframe/#from-hive)
* val hiveContext = new org.apache.spark.sql.hive.HiveContext(spark.sparkContext)
* val hiveDF = hiveContext.sql(“select \* from emp”)
* [Creating from RDBMS Database table](https://sparkbyexamples.com/spark/different-ways-to-create-a-spark-dataframe/#from-database)

#### 8.a) From **Mysql** table

Make sure you have MySQL library as a dependency in your pom.xml file or MySQL jars in your classpath.

val df\_mysql = spark.read.format(“jdbc”)

.option(“url”, “jdbc:mysql://localhost:port/db”)

.option(“driver”, “com.mysql.jdbc.Driver”)

.option(“dbtable”, “tablename”)

.option(“user”, “user”)

.option(“password”, “password”)

.load()

* [Creating from HBase table](https://sparkbyexamples.com/spark/different-ways-to-create-a-spark-dataframe/#hbase)
* val hbaseDF = sparkSession.read
* .options(Map(HBaseTableCatalog.tableCatalog -> catalog))
* .format("org.apache.spark.sql.execution.datasources.hbase")
* .load()
* [Other sources (Avro, Parquet e.t.c)](https://sparkbyexamples.com/spark/different-ways-to-create-a-spark-dataframe/#avro-parquet)

sqlContext.read.json("examples/src/main/resources/people.json")

val parqDF = spark.read.parquet("/tmp/output/people.parquet")

Ways to create dataframe

1. **import** spark.implicits.\_
2. **val** someDF = Seq(  
   (8, **"bat"**),  
   (64, **"mouse"**),  
   (-27, **"horse"**)  
   ).toDF(**"number"**, **"word"**)

**val** someDF = spark.createDF(  
List(  
(8, **"bat"**),  
(64, **"mouse"**),  
(-27, **"horse"**)  
), List(1  
(**"number"**, IntegerType, **true**),  
(**"word"**, StringType, **true**)  
)  
)

**Data Set**

Strongly typed

Immutable

Distributed

in memory

#### Collection of data

Data is map to pre-defined schema.

Provides Object oriented programming interface

Clubs RDD and dataframe features

we can check syntax and analysis at compile time ,optimized query using [Catalyst Query Optimizer](http://data-flair.training/blogs/spark-sql-optimization-catalyst-optimizer/)

**supports encoder which converts object to binary format.**

Let's see different ways of creating Dataset

## **Create a Dataset**

**val** dataset **=** Seq(1, 2, 3).toDS()

dataset.show()

### **Create a Dataset from an RDD**

**case** **class** Person(name**:** String, age**:** Int)

**val** personDS **=** Seq(Person("Max", 33), Person("Adam", 32), Person("Muller", 62)).toDS()

personDS.show()

### **Create a Dataset from an RDD**

To convert an RDD into a Dataset, call rdd.toDS().

Scala

Copy

**val** rdd **=** sc.parallelize(Seq((1, "Spark"), (2, "Databricks")))

**val** integerDS **=** rdd.toDS()

integerDS.show()

s

### **Create a Dataset from a DataFrame**

You can call df.as[SomeCaseClass] to convert the DataFrame to a Dataset.

Scala

Copy

**case** **class** Company(name**:** String, foundingYear**:** Int, numEmployees**:** Int)

**val** inputSeq **=** Seq(Company("ABC", 1998, 310), Company("XYZ", 1983, 904), Company("NOP", 2005, 83))

**val** df **=** sc.parallelize(inputSeq).toDF()

**val** companyDS **=** df.as[Company]

companyDS.show()

**val** rdd **=** sc.parallelize(Seq((1, "Spark"), (2, "Databricks"), (3, "Notebook")))

**val** df **=** rdd.toDF("Id", "Name")

**val** dataset **=** df.as[(Int, String)]

dataset.show()

## **Convert a Dataset to a DataFrame**

The above 2 examples dealt with using pure Datasets APIs. You can also easily move from Datasets to DataFrames and leverage the DataFrames APIs. The below example shows the word count example that uses both Datasets and DataFrames APIs.

Scala

Copy

**import** org.apache.spark.sql.functions.\_

**val** wordsDataset **=** sc.parallelize(Seq("Spark I am your father", "May the spark be with you", "Spark I am your father")).toDS()

**val** result **=** wordsDataset

.flatMap(**\_**.split(" ")) // Split on whitespace

.filter(**\_** != "") // Filter empty words

.map(**\_**.toLowerCase())

.toDF() // Convert to DataFrame to perform aggregation / sorting

.groupBy($"value") // Count number of occurrences of each word

.agg(count("\*") as "numOccurances")

.orderBy($"numOccurances" desc) // Show most common words first

result.show()

// create a Dataset using SparkSession.createDataset() and the toDS

val movies = Seq(Movie("DDLJ", "Awesome", 2018L), Movie("ADHM", "Nice", 2018L))

val moviesDS = spark.createDataset(localMovies)

moviesDS.show()

val moviesDS1 = localMovies.toDS()

localMoviesDS1.show()

// Encoders are created for case classes

case class Employee(name: String, age: Long)

val caseClassDS = Seq(Employee("Amy", 32)).toDS

caseClassDS.show()

// convert DataFrame to strongly typed Dataset

case class Movie(actor\_name:String, movie\_title:String, produced\_year:Long)

val movies = Seq(("Damon, Matt", "The Bourne Ultimatum", 2007L),

("Damon, Matt", "Good Will Hunting", 1997L))

*// Encoders for most common types are automatically provided by importing sqlContext.implicits.\_*

**val** ds **=** **Seq**(1, 2, 3).toDS()

ds.map(**\_** + 1).collect() *// Returns: Array(2, 3, 4)*

*// Encoders are also created for case classes.*

**case** **class** **Person**(name**:** String, age**:** Long)

**val** ds **=** **Seq**(**Person**("Andy", 32)).toDS()

*// DataFrames can be converted to a Dataset by providing a class. Mapping will be done by name.*

**val** path **=** "examples/src/main/resources/people.json"

**val** people **=** sqlContext.read.json(path).as[Person]

val moviesDF = movies.toDF.as[Movie]

**11. What is Executor Memory in a Spark application?**

### **Spark executers runs with some fixed memory size on worker nodes is known as spark executer memory.**

### **we can control it by setting spark.executor.memory**

### **13. What operations does RDD support?**

### RDDs support two types of operations:

transformations and actions.

Transformations: Transformations is transforming one RDD to new RDD

map,

parallelize

textFile

filter

reduceByKey

groupBykey

sortBykey

Actions :- Results of transformation.

Collect

Foreach.println

take

### **16. Define functions of SparkCore.**

1. Memory management and fault recovery
2. Scheduling, distributing and monitoring jobs on a cluster
3. Interacting with storage systems

### **17. What do you understand by Pair RDD?**

RDD which has key,value pairs.

It operates parrallely on each key

We can perform reduceByKey,sortByKey,groupByKey operations on Pai RDD.

### **18. Name the components of Spark Ecosystem.**

1. **Spark Core**: For large-scale parallel and distributed data processing
2. **Spark Streaming**: Used for processing real-time streaming data
3. **Spark SQL**: It supports HQL and SQL,Data Source API,DataFrame API,Interpreter & Optimizer,SQL Service
4. **GraphX**: Graphs and graph-parallel computation
5. **MLlib**: Performs machine learning in Apache Spark

### **What is a Part file?**

### Output is divided into multiple parts is known as part file.

### **25. How can Apache Spark be used alongside Hadoop?**

1. **HDFS**: Spark use hadoop distributed storage.
2. **MapReduce** and **YARN** :- can be use as processing framework for batch and real time processing.

### **27. What is Spark Driver?**

program runs on the master node

creates SparkContext

delivers the RDD lineage graphs to cluster manager

**274. What Is The Default Level Of Parallelism In Apache Spark?**

**Answer :**

number of partitions are considered as default level of parallelism in Apache Spark.

#### **281. Can we create RDD using Dataset like .txt file?**

Answer: Yes, by loading dataset we can create RDD.

We can create rdd using dataset

Eg. sc.textFile(“C://Users/Desktop/heena.txt”)

#### **284. What are the types of transformations in Spark?Explain them.**

Answer: Narrow and Wide Transformation are available in Spark.

When transformations are applied over single partition it is known as narrow transformation

When transformations ar applied over multiple transformation it is known as wide transformation

#### **285. Give some example for Narrow transformation?**

Answer: Map and Filter.

Give example for wide narrow transformations

Answer:

sortByKey,

reduceByKey

#### **291. What is the default storage level in Spark?**

Answer:  Memory\_Only

#### **296. Why Spark is faster than Hadoop?**

Answer: Spark is faster than Hadoop because it does processing in memory.

#### **301. What is spark shell?**

Answer: Spark Shell is a Spark Application which is written in Scala. It offers a command line environment with auto-completion.

which is helpful in developing our own Standalone Spark Application.

;/0l

#### [**131. What is reduce() action?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

It is action which take two parameters and return one.

It aggegrates dataset.

Eg of aggregate fnctions

Sum

Count

Min

Mean

Max

#### **26. What is RDD Lineage?** [**130. What is lineage graph?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

In spark, all the dependencies are logged in a graph which is called as lineage graph.

Its logical execution plan.

And RDD lineage is graph/logical execution plan of parent RDD’s.

**175. How Spark achieves fault tolerance? 263. What happens to RDD when one of the nodes on which it is distributed goes down?**

Spark Achieved fault tolerance using RDD lineage .

The rdd lineage is logical execution plan having information about actions and transformation.

The lost data can be recovered from lineage graph

#### **260. What is DAG – Directed Acyclic Graph?** [**125. What is DAG and Stage in spark processing?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Dire]cted Acyclic Graph –

Physical execution plan

DAG is set of vertices and edges , not having cycles.

Where vertices are RDD’s

Edges are transfirmations

Once action is called on RDD DAG is submitted to Dag scheduler.

Dag scheduler split DAG to stages.

Stages submitted to Task Scheduler

Task scheduler creates task and sen it to cluster manager.

### **30. What is Spark Executor?**

Cluster manager creates spark executors which run transformation and actions on worker nodes.

### **32. What do you understand by worker node?**

Clustemanager excuted executord over datanodes are known as worker nodes.

#### [**100. What are all the internal daemons used in spark?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

TaskScheduler, SparkContext, Driver, Node Manger , Application master,container, Worker,Executor, Tasks.

**233. Explain about the core components of a distributed Spark application.**

**275. Explain About The Common Workflow Of A Spark Program**

**25. When creating an RDD, what goes on internally?**

RDD lineage

Logical execution plan contains information about actions and transformations and their dependecies

#### **DAG :-**

#### **Direct Acyclic Graph**

Set of vertices and edeges Where vertices are RDD and edges are ransfirmation and action

Transformation Transformation

Transformation Transformation

Action Action

DAG Scheduler

Task Sceduler 1

Datanode

,

Application Master

Creates container

Node Manager

Manages node and send report to the global resource manager

#### **302. Write some function of Spark Context?**

Answer: Used to create Spark RDDs, accumulators, and broadcast variables, access all Spark services and run jobs also to get the status of spark application. Starting and cancelling of Job etc.

### **31. Name types of Cluster Managers in Spark.**

The Spark framework supports three major types of Cluster Managers:

1. **Standalone**: A basic manager.
2. **Apache Mesos**: Generalized/commonly-used cluster manager, also runs Hadoop MapReduce and other applications.
3. **YARN**: Responsible for resource management in Hadoop.

**278. On which all platform can Apache Spark run?**  
Spark can run on the following platforms:

* **YARN (Hadoop):**
* **Apache Mesos:**
* **EC2:**
* **Standalone:** and start the cluster. It starts communicating with each other and run.

**74. What is File System API?**

reading data from different storage devices like HDFS, AWS S3 or local FileSystem

### **28. What file systems does Spark support?**

The following three file systems are supported by Spark:

1. Hadoop Distributed File System (HDFS).
2. Local File system.
3. Amazon S3

### **48. What are the various data sources available in Spark SQL?**

#### [**99. what are all the file formats supported by spark ?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Avro,

part,

json,

xml,

csv,

tsv,

snappy,

orc,

rc   
Raw files

structured files

### **Illustrate some demerits of using Spark.**

**What are the downsides of Spark?**

### **Expensive:-**in memory computation is very expensive.

### **Not purely Real Time:-**

### In [Spark Streaming](http://data-flair.training/blogs/apache-spark-streaming-comprehensive-guide/),

Data is divided into RDD’s ,

RDD’s sent to streaming engine after fixed time interval not continuously

So it is not purely real time.

### **Delay**

### Apache Spark has higher Delay as compared to [Apache Flink](http://data-flair.training/blogs/apache-flink-big-data-unified-platform/).

### **No File Management System**

### **Manual Optimization**

### Partitioning and Caching required.

### **Back pressure is build up of data:-**

### When buffer overflow occurred, we need to manualyy remove the data.

### **List some use cases where Spark outperforms Hadoop in processing?**

### **Advantages of spark.**

**difference between Hadoop and Spark.**

**Spe**

**edy – 100 Times faster than Hadoop**

Achive through partitioning ,caching and Persistance ,broadcast variables,accumulators

**Deployment** – can be deploy through mesos,yarn or standalone mode

**Polyglot-** can be written in java,scala,python,R

**Streaming –** offers real time processing using sparkstreaming

**Easy** – Simple to learn

**Smaller code –** smaller code as compare to hadoop and java

**Higher functions available** – to write complex logic

**Scalable –** we can scale the cluster

**177. Explain about the popular use cases of Apache Spark?**

* Apache Spark is mainly used for
* Iterative machine learning.
* Interactive data analytics and processing.
* Stream processing
* Sensor data processing

### **35. What is a Sparse Vector?**

A sparse vector is collection having two arrays one for indexces an another for values.

### **37. Is it possible to run Apache Spark on Apache Mesos?**

### **Yes , mesos act as cluster manager here which executes actions and transformation on worker nodes**

### **38. How can Spark be connected to Apache Mesos?**

To connect Spark with Mesos:

1. Configure the spark driver program to connect to Mesos.
2. Install Apache Spark in the same location as Apache Mesos and

configure the property ‘spark.mesos.executor.home’

**39. How can you minimize data transfers when working with Spark?**

**Why is there a need for broadcast variables when working with Apache Spark?**

**Explain accumulators in Apache Spark.**

**Explain optimization techniques in Spark**

**Explain Partitining**

**What are the various levels of persistence in Apache Spark?**

**Explain Caching and persistence**

#### [**Difference among cache() and persist()?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

**Filter data as soon as possible**

### **Caching**

Persistaning is storing the RDD in cache that may be memory /disk

1. **MEMORY\_ONLY:**

Default

Data is stored in non-serialized format

1. **MEMORY\_AND\_DISK:**

RDD/data is stored in memory if memory is not available data is stored into disk.

expensive

1. **MEMORY\_ONLY\_SER:**

Data is stored in serialized format.

1. **DISK\_ONLY:**

Data is stored on Disk

1. **OFF\_HEAP:**

Data in off-heap memory.

**Partitioning- distributing the data across the multiple nodes**

**1: Hash Partitioning**

The data is distributed according to hashcode

The formulae Is ,

Partition no = HashCode(Value) % no\_of\_partitions available

So if we have 1,2,3,4,5,51 values in dataset and no\_of\_partition is 3

Then we find

Partition no = HashCode(1)% 3 = 1

Partition no= HashCode(2)%3= 2

Partition no= HashCode(3)%3=0

Partition no = Hashcode(4)%3=1

Partition no = hashcode(51)%3= 0

**And if you provide No of partitions = number of cores then it will perform better.**

**The values will go to that container**

**2: Range Partitioning**

Here the data is distributed according to range

Eg 1 to 5 data will go in first bucket

5 to 10 data will go in second bucket

**Explain Broadcast variables**

Sharing and storing variables across executers is known as broadcast variable.

Eg suppose finance department wants employee payroll data at appraisals time.

Then in this case the employee payroll data can be broadcasted over multilple nodes and

Multiple finance department members can aceess payroll data.

It will increase speed for computation.

Eg2 – Suppose HR members want to access performance rating for particular employees.

Then employee data can be broadcasted over multiple executers.

And each HR member can accees performance data at same time from different data nodes independelty.

Syntax= sc.broadcast(“data.csv”)

**Explain accumulators**

Accumulators is like flag to initialize variables/values.

It only supports associative operations eg.

Count

Sum

Max

Min

Mean

### **Syntax**

Val accum = sc.accumulator(0)

Sc.parallelize(Array(1,2,3)).foreach(x => accum +x);

### **52. What do you understand by Lazy Evaluation?**

### **Evaluation(computation) will not be start until action is called.**

Ie. They are lazy and they don’t work until we ask for results.

Eg. Transformation

Increased Speed

Reduce unnecessary computation time

### **43. How can you trigger automatic clean-ups in Spark to handle accumulated metadata?**

setting the parameter ‘spark.cleaner.ttl’

### **45. What is a DStream in Apache Spark?**

**Discretized Stream** (DStream) is stream of data received from data sources eg. Apache Kafka, HDFS, and Apache Flume or another datastream.

DStreams have two operations:

1. Transformations
2. Actions

### **53. What do you understand by SchemaRDD in Apache Spark RDD?**

SchemaRDD as name suggests

RDD that has schema

Collection of row objects and its schema.

We can create schema RDD by three methods:-

CreateSchemaRDD

By loading data from external data sources

Case classes

### **54. How is Spark SQL different from HQL and SQL?**

We can join SQL and HQL table in SparkSQL

**58. What is Immutable?**

Once we assigne value , it is not possible to change is immutable.

Eg RDD

**75. Why Partitions are immutable?**

Partitions uses HDFS so partitions are immutable

1. **When do you use apache spark? OR**

**What are the benefits of Spark over Mapreduce?**

**Is there is a point of learning MapReduce, then?**

**86. Say I have a huge list of numbers in RDD(say myrdd). And I wrote the following code to compute average:**

def myAvg(x, y):  
return (x+y)/2.0;  
avg = myrdd.reduce(myAvg);

**How to find average of more numbers**

**88. Say I have a huge list of numbers in a file in HDFS. Each line has one number.And I want to compute the square root of sum of squares of these numbers. How would you do it?**

**92. In a very huge text file, you want to just check if a particular keyword exists. How would you do this using Spark?93. Can you improve the performance of this code in previous answer using all partitioning techniques?**

#### [**1.Various ways to create contexts in spark ?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

a. Sparkconext  
b. Sqlcontext  
c. Sparksession  
d. Sqlcontext.sparkcontext

#### [**95.Difference between map and flatmap?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

a. Map – one input row to one output row  
b. Flatmap – one input row to multiple output rows

#### **[96.Repartition and coalesce difference?](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)**

repartition = suppose data is partitioned earlier

and if we partition data again into new number of partitions then it is know as repartition.

We can increase or decrese number of partitions here.

Val x = (1 to 12).toList.toDF(“list dataframe”)\

[97.How to create a stream in spark](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

a. Dstream  
b. Structured stream.  
c. DirectStream

#### [**103. how to submit a spark job?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Using spark-submit and just follow the following program?  
spark-submit –class org.apache.spark.examples.ClassJobName –master yarn –deploy-mode client –driver-memory 4g –num-executors 2 –executor-memory 2g –executor-cores 10  
in the above sample  
–master = cluster manager type eg.yarn,mesos,local  
driver-memory = memory size of the driver  
executor-memory is = memory size of the executor  
–num-executors = total number of executors which are running at the worker nodes.  
–executor-cores number of individual processes that the executor memory can take up.

#### [**107. how to join two dataframes in spark?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Df1.join(df2).where(df1.col1==df2.col1).where(df1.col1==df2.col1)

#### [**108. what is udfs and how to use it ?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

UDFs are user defined functions which are created to handle complex functionalities.

#### [**110. code sample to read a data from mysql ?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

spark.read.format(‘jdbc’).op/tions(driver=’com.mysql.jdbc.Driver’,url=”””jdbc:mysql://<host>:3306/<>db?user=<usr>&password=<pass>”””,dbtable=’besant’,numPartitions=4 ).load()

### **143. How do you specify the number of partitions while creating an RDD? What are the functions?**

We can pass partitions in two functions

1

val rdd = sc.parallelize(data,4)

val data = sc.textFile(“path”,4)

### **148. How can you connect Hive to Spark SQL?**

1. Put hive-site.xml on your classpath, and specify hive.metastore.uris to where your hive metastore hosted. Also see [How to connect to a Hive metastore programmatically in SparkSQL?](https://stackoverflow.com/questions/31980584/how-to-connect-to-a-hive-metastore-programmatically-in-sparksql)
2. Import org.apache.spark.sql.hive.HiveContext, as it can perform SQL query over Hive tables.
3. Define val sqlContext = new org.apache.spark.sql.hive.HiveContext(sc)
4. Verify sqlContext.sql("show tables
5. ") to see if it works

**150. What is Standalone mode?** **152. What are client mode and cluster mode?**

**Standalone mode**

Here spark cluster manges all monitoring and scheduling resources duties.

It has one master and many workers

When running Spark in the cluster mode, the Spark Driver runs inside the cluster

When running Spark in the client mode, the SparkContext and Driver program run external to the cluster;

#### **207. How many cluster modes are supported in Apache Spark?**

**153. How to run spark in Standalone client mode?**

* spark-submit \
* class org.apache.spark.examples.SparkPi \
* deploy-mode client \
* master spark//$SPARK\_MASTER\_IP:$SPARK\_MASTER\_PORT \
* $SPARK\_HOME/examples/lib/spark-examples\_version.jar 10

**154. How to run spark in Standalone cluster mode?**

* spark-submit \
* class org.apache.spark.examples.SparkPi \
* deploy-mode cluster \
* master spark//$SPARK\_MASTER\_IP:$SPARK\_MASTER\_PORT \
* $SPARK\_HOME/examples/lib/spark-examples\_version.jar 10

**155. How to run spark in YARN client mode?**

* spark-submit \
* class org.apache.spark.examples.SparkPi \
* deploy-mode client \
* master yarn \
* $11`SPARK\_HOME/examples/lib/spark-examples\_version.jar 10

**156. How to run spark in YARN cluster mode?**

* spark-submit \
* class org.apache.spark.examples.SparkPi \
* deploy-mode cluster \
* master yarn \
* $SPARK\_HOME/examples/lib/spark-examples\_version.jar 10

**158. What is the maximum number of total cores?**

You can set this value to unlimited by passing

total-execution cores argument to spark-submit, or

spark.cores.max in your Spark configuration file.

**179. How can you remove the elements with a key present in any other RDD?**

Use the subtractByKey () function

#### **195. What is use of apply and unapply method?**

apply method - used to map data

unapply method - used to unmap the data.

#### **196. Do private members of Companion classes can be accessed through companion objects in Scala?**

Yes, it is possible that private members of Companion classes can be accessed through companion objects in Scala.

#### **198. How can Auxiliary Constructors and primary constrauctor be defined in Scala?**

What is constructor”-

Constructor is use to initialize class variables

Constructor has same name as class

There are two types of constructor

Auxilary contructor are used for constructor overloading

It uses this function for the same

The class can have one or more auxiliary constrctor.

Class student(student\_id:Int,student\_name:String,salary:Double)

{

Def this(student\_id:Int,student\_name:String,salary: Double)

{

this(1,”heena”,70000)

println(“three argument auxiliary constructor”)

}

Def this(student\_id:Int,student\_name:String,salary:Double)

{

this(2,”salim”,56l478)

println(“three argment auxiliary constructor”)

}

Def this (student\_id:Int,student\_name:String )

{

this(1,”shaikh”)

println(“two argument auxilory constructor”)

}

Def this()

{

This()

Println(“zero argument auxiliary contructor”)

}

}

}

Primary constructor

Very first constructor in class is known as primary constructor

The class contains only one primary constructor

#### **199. How will you explain yield keyword in Scala?**

We use yield keyword with loops,

Eg.for loops,while loops

Eg. suppose we have array

Var a= Array(1,2,3,4,5)

For(a <- 1 to 9)

{

Yield a

}

For(a <- 9 to 1)

{

Yeild a\*2;

}

#### **201. What do you mean by Scala Traits and how it can be used in Scala programming language?**

Trait is collection of abstract and non abstract methods

The methods which are declared but not used is known as abstract methods

The methods which are declared as well as used are known as non-abstract methods

Eg

Trait calculation()

{

Def sum()

{

Var x=10;

Var y =20;

}

Def subtraction()  
{}

Def mean calculaton()

{

Var a = 0;

Var b =9 ;

}

}

Class vendor\_management extends calculation

{

Def sum()

{

Return x\*y;

}

}

Class inventory\_management extends calaculation

{

Println (“declaring this class as abstact because we extend trait but not implementing traits method

Class has its own method

Def division()

{

Var x=10;

Var y=20;

Var z= x\*y;

}

Object vendor\_object()

{

Def main(args:String[])

{

Var v = new vendor\_management();

v.sum();

}

What is abstract class and what is abstract variables?

Variables those are declared but not used are known as abstract variables

Class which extends trait but not uses trait metods are known as abstract classes

#### **203. How are Monads useful for Scala developers?**

Monald wrap one object into another

Monalds can be collection or container

Map and flatmap creates monalds

Eg

var l = List(1,2,34,5,6)

var rdd1= sc.parallelize(l)

var rdd2 = rdd1.flatMap(x => x%2);

Output:- rdd2(rdd1)

List(1,0,0,2,0)

**229. What are the common mistakes developers make when running Spark applications?**

Developers often make the mistake of-

* Developer runs applications in local mode instead of uding yarn/mesos.

Developers should use caching and persisting wherever needed

**244. Name some sources from where Spark streaming component can process real-time data.**

Apache Flume, Apache Kafka, Amazon Kinesis

**245. Name some companies that are already using Spark Streaming.**

Uber, Netflix, Pinterest.

**262. If there is certain data that we want to use again and again in different transformations, what should improve the performance?**

Persitance caching

**264. How to save RDD?**

There are few methods provided by Spark:

* **saveAsTextFile**
* **saveAsSenceFile**
* **saveAsObjectFile**

**265. What do we mean by Parat?**

columnar format for storing data

**320. Who designed Scala? Which is the latest version?**

Martin Odersky, a German computer scientist in 2001 in Switzerland.

Latest version is 2.12.6

**321. What are the advantages of Scala?**

Among various other benefits of the language, here are a few:

It is highly scalable

concurrent programming

object-oriented and functional

Singleton objects are a cleaner solution than static

**322. What is ofDim in Scala?**

ofDim() is method to create multidimensional arrays. Eg.matrix

scala> import Array.ofDim

scala> var a=ofDim[Int](3,3)

a: Array[Array[Int]] = Array(Array(0, 0, 0), Array(0, 0, 0), Array(0, 0, 0))

scala> var k=1

scala> for(i<-0 to 2){

    | for(j<-0 to 2){

    | a(i)(j)={i+k}

    | k+=1

    | }

    | k-=1

    | }

scala> a

res12: Array[Array[Int]] = Array(Array(1, 2, 3), Array(4, 5, 6), Array(7, 8, 9)).

**324. What is a BitSet?**

A bitset is collection of positive and unique elements , bitset size varies.

Import scala.collection.immutable.\_Bitset

Var bitset = Bitset(5,6,7,8,2)

Output – BitSet(2,5,6,7,8)

Bitset += 9

Bitset -=9;

Output – Bitset(2,5,6,7,8,9)

**325. What is a vector in Scala?**

Vector is collection of immutable, random, large number of elements

We can perform following operations on vector

Distinct. Drop(index),filter(),head(),take(),tail()

Var v = Vector(1,2,3,4,5,6,4,5,);

v.distinct

Vector(1,2,3,4,5,6)

v.drop(2)

Vector(1,3,4,5,6,4,5)

v :+ 3;

v :+4;

Vector(1,3,3,4,5,4,5,6)

**V:- 5**

**Vector(1,3,3,4,6,4)**

**Syntax of List , Map and Set**

**List:-**

**List is collection of immutable ,sequenced of elements**

**Import scala.collection.immmutable.List**

**Object collection**

**{**

**Def main (args :String[Array])**

**{**

**Var l = List(543,34,7788,999)**

**For (i 🡨 l)**

**{**

**If(i%2 == 0)**

**Println(“even elements)  
else if(I %2 != 0)**

**Println(“odd elements”)**

**}**

**Println(“Printing reverse elements”);**

**Println(l.reverse)**

**}**

**Set :-**

**Set is collection of unique elements**

**Import scala.collection.immutable.Set**

**Var S = Set(1,3,4,5,6,3)**

**Set(1,3,4,5,6)**

**There are two types of set**

**Bitset and treeset**

**Map**

**Map is collection of key and value pairs**

|  |
| --- |
| var imap = Map(1-> "Heena" , 2 -> "Ajinkya") |
| imap+=(3->"Shaikh") |  |
| imap+=(4->"collection") |  |
| println("imap"+ imap) |  |
|  |  |
| //Adding the values in empty map |  |
| //Syntax var map\_name:Map[type1,type2] = Map() |  |
| var empty\_map:Map[Any,Any] = Map(); |  |
| empty\_map+= (1 -> "Sourabh") |  |
| empty\_map+= ("Heena" -> "Salim") |  |
| empty\_map+= ( 2 -> "Shaikh") |  |

**Syntax of the tuple ?**

**var aTuple = scala.io.Source.fromFile("C:\\Users\\jabin\\Desktop\\sample\_test.txt");**

**var sTuple = Tuple(1,2,34,5,6)**

**println(sTuple.\_2);**

**336. How will you handle error in spark?**

**By using try catch error blocks**

|  |
| --- |
| package spark\_Context\_Programs |
|  | import org.apache.spark.{SparkConf,SparkContext} |
|  | import scala.util.{Try ,Success,Failure} |
|  |  |
|  | object Error\_handling { |
|  |  |
|  | var total =0 |
|  | def main(args:Array[String]):Unit = { |
|  | var configuration = new SparkConf().setAppName("Exceptional Handling").setMaster("local") |
|  | var sc = new SparkContext(configuration); |
|  | var list = sc.parallelize(List(1,2,3,44,55)) |
|  |  |
|  |  |
|  | def calculate():Int = |
|  | { |
|  | if(list.isEmpty()) |
|  | throw new Exception("Elements not fond"); |
| S |  |
|  | total = list.fold(0)((a,b) => a +b) |
|  | return total |
|  |  |
|  | } |
|  |  |
|  | Try(calculate()) match |
|  | { |
|  | case Success(\_) => println(s"The script completed Successfully : $total") |
|  | case Failure(\_) => println ("The script has failed"); |
|  |  |
|  | } |
|  |  |
|  | } |
|  | } |

**337. Can we perform RDD operations on the Collections?**

Yes , but for that we need toconvert the Collection into the sequence .

**338. What is case classes and why we use case classes**

**339. What is syntax of the fold operation and what is fold operation?**

|  |
| --- |
| var l = List(2,4,5,6) |
|  | val sum = l.fold(0)(\_+\_) |
|  | val sum1 = l.fold(0){ (a,b) => +  (a + b) }; |
|  |  |

**340. Which are print operations in Spark ?**

Dump

Println

Take

First

Top

**341. Which operation we use for combining and subtracting items?**

**Union and Intersection**

**342. Explain the ways of reading files in spark scala?**

**343. Explain syntax of the case statements:-**

case when import\_ind = '1' and location\_id != '8277' then '8277' else concat(trim(ship\_duns\_nbr),'\_S') end as source\_location\_id,"

|  |
| --- |
| "case when import\_ind = 1 then 'N' \n" + |
|  | "else\n" + |
|  | "case when vendor\_managed\_inventory\_cd is not null or trim(vendor\_managed\_inventory\_cd) != '' \n" + |
|  | "and vendor\_managed\_inventory\_cd = '5' or vendor\_managed\_inventory\_cd = '6' \n" + |
|  | "or vendor\_managed\_inventory\_cd = '7' or vendor\_managed\_inventory\_cd = '8' \n" + |
|  | "or vendor\_managed\_inventory\_cd = '5' or vendor\_managed\_inventory\_cd = '9' \n" + |
|  | "then 'N' \n" + |
|  | "else \n " + |
|  | "case when stock\_ind='N' and IsNull(dc\_handling\_cd != '') and dc\_hansdling\_cd is not null and dc\_handling\_cd ='CASE' then 'Y' else 'N' end \n" + |
|  | "end \n" + |
|  | "end as dc\_flowthru\_ind,dotcom\_orderable\_cd \n" + |

**344. What is registerTemp Table?**

**Schema to file or structure can be given in three ways**

* 1. Using case classes
  2. Using StructField
  3. Using inferSchema

If we use

case classes to provide schema todataframe you need to read the file using sc.textFile

var text = sc.textFile();

var header = text.first

var file = text.filter( x -> x!=header).map(x -> x.split(|)).map(x =>x(0),x(1),x(3)).toDF()

Then we can register df as temp table or

we can use select ,selectExpression,join , withcolumn,filter,trim,cast,map,alias,union

methods directly on dataframe

StructFiled =

inferSchema:- it automatically detects the schema