**HADOOP AND MAPREDUCE QUESTIONS**

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

|  |  |  |
| --- | --- | --- |
|  | **RDBMS** | **Hadoop** |
| **Data Types** | process only structured data | process structured, unstructured or semi-structured. |
| **Cost** | Licensed and payable software, | Hadoop is an open source framework. |
| **Data Volume** | It can handle smaller data | It can handle huge amount of the data |
| **Engine** | OLTP | OLAP |
| **Read/Write Speed** | Reading of the data is faster. | Writing of the data is faster |
|  | We can access any row very fast. | We cannot access the particular row quickly |
| **Processing** | ACID transactions are fast | ACID transactions are slow. |

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

**1:Volume:**

**Big data can store Petabytes/Exabytes of the data.**

**2:Velocity:**

**Big data can store this volume of the data in higher speed ie velocity**

**3:Variety:**

**Big data can store structured, semi-structured and un-structured data.**

**4:Value:-**

**Big data providing high ROI (return of investment) to organization.**

### 3. 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 2Components :-**

* ***Storage unit***– HDFS (NameNode, DataNode)
* ***Processing framework***– 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 in the HDFS which store the metadata eg replication factor and block size, block numberetc

And manages the datanodes

**DataNode:-**

Datanodes stores the actual data.

-----------------------------------------------------------------------------------------------------------------------------

**5:What is YARN:-**

Yet another resource negotiator

YARN have two components:-

**ResourceManager:-**

It allocates resources to application based on the need and allocates task to node managers.

**NodeManager:-**

It is installed on every datanode and it carry out task execution on every datanode.

### ****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 in the cluster.
* When the active “NameNode” fails, the passive “NameNode” replaces the active “NameNode” in the cluster.

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

NameNode periodically receives a Heartbeat (signal) from each of the DataNode in the cluster, which implies DataNode is functioning properly.

A block report contains a list of all the blocks on a DataNode. If a DataNode fails to send a heartbeat message, after a specific period of time it is marked dead.

The NameNode replicates the blocks of dead node to another DataNode using the replicas created earlier.

### ****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, I 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 takes an FsImage, edit log and combine them into a new FsImage.
2. Checkpointing is performed by Secondary NameNode.
3. Hence reduces NameNode startup time.
4. Checkpointing is same as saving the latest saystem state.

### 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?

### Blocks are smallest unit on your hard drive where data is stored.

### We can set block size by setting dfs.block.size parameter in hdfs-site.xml  .

* Hadoop 1 default block size: 64 MB
* Hadoop 2 default block size:  128 MB

### ****12. What does ‘jps’ command do?****

The ‘jps’ command helps us to check if the Hadoop daemons are running or not. It shows all the Hadoop daemons i.e namenode, datanode, resourcemanager, nodemanager etc. that are running on the machine.

### ****13. How can I restart “NameNode” or all the daemons in Hadoop?****

To stop all the daemons, use**.** ***/sbin/stop-all.sh***command.

Then to use start all the daemons.***/sbin/start-all.sh*** command.

### ****14. What is the 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 file 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***:

This is the default mode if we don’t configure anything. In this mode, all 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, all the Hadoop services, including both the master and the slave nodes services execute on a single compute node.

1. ***Fully distributed mode***:

A Hadoop deployments in which 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 “InputSplit” defines a slice of work, but does not describe how to access it. The “RecordReader” class loads the data from its source and converts it into (key, value) pairs suitable for reading by the “Mapper” task. The “RecordReader” instance is defined by the “Input Format”.

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

This is a tricky question. The “MapReduce” programming model does not allow “reducers” to communicate with each other. “Reducers” run in isolation.

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

A “MapReduce Partitioner” makes sure that all the values of a single key go to the same “reducer”, thus allowing even distribution of the map output over the “reducers”. It redirects the “mapper” output to the “reducer” by determining which “reducer” is responsible for the particular key.

### ****19. What is a “Combiner”?****

A “Combiner” is a mini “reducer” that performs the local “reduce” task. It receives the input from the “mapper” on a particular “node” and sends the output to the “reducer”. “Combiners” help in enhancing the efficiency of “MapReduce” by reducing the quantum of data that is required to be sent to the “reducers”.

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

|  |  |  |
| --- | --- | --- |
|  | **Pig** | **Mapreduce** |
| **Language** | High level data processing language | Low level data processing language |
| **Coding** | Easy coding | Complex coding. |
| **Code length** | 10 lines of code | 100 lines of code |
| **Engine** | Built in functions for joins, filters, ordering, sorting | We need to write custom programs for  joins, filters, ordering, sorting |
| **Read/Write Speed** | Joining is simple and faster | Joining is complex and 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. order by
3. filters
4. group
5. distinct
6. join
7. limit

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

If some functions are unavailable in built-in operators, we can programmatically create User Defined Functions (UDF) in languages like java, python,ruby.

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

### The default location where Hive stores table data is inside HDFS in /**user/hive/warehouse.**

### ****25. What are the core methods of a Reducer?****[**Click here to Tweet**](http://ctt.ec/e2_Z3)

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

### 1)setup () – This method of the reducer is used for configuring various parameters like the input data size, distributed cache, heap size, etc.

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

### 2)reduce () it is heart of the reducer which is called once per key with the associated reduce task.

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

### 3)cleanup () - This method is called only once at the end of reduce task for clearing all the temporary files.

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

### 26. Explain about the partitioning, shuffle and sort phase [Click here to Tweet](http://ctt.ec/cNUfx)

### Shuffle Phase-Once the first map tasks are completed, the nodes continue to perform several other map tasks and also exchange the intermediate outputs with the reducers as required. This process of moving the intermediate outputs of map tasks to the reducer is referred to as Shuffling.

### Sort Phase- Hadoop MapReduce automatically sorts the set of intermediate keys on a single node before they are given as input to the reducer.

### Partitioning Phase-The process that determines which intermediate keys and value will be received by each reducer instance is referred to as partitioning. The destination partition is same for any key irrespective of the mapper instance that generated it.

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

### COGROUP operator in Pig is used to work with multiple tuples. COGROUP operator is applied on statements that contain or involve two or more relations. The COGROUP operator can be applied on up to 127 relations at a time. When using the COGROUP operator on two tables at once-Pig first groups both the tables and after that joins the two tables on the grouped columns.

### We have further categorized Hadoop Pig Interview Questions for Freshers and Experienc

### 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 from tables. Generally, users prefer to write a Deserializer instead of a SerDe as they want to read their own data format rather than writing to it. If the SerDe supports DDL i.e. basically SerDe with parameterized columns and different column types, the users can implement a Protocol based DynamicSerDe rather than writing the SerDe from scratch.

### 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.The data volume you have worked with in your most recent project.

### 1TB

### 32. Name some companies that use Hadoop

### Facebook Netflix Amazon Adobe eBay Hulu Spotify Rubikloud Twitter

### 33. What is commodity hardware? [Click here to Tweet](http://ctt.ec/15x4e)

### Commodity Hardware refers to inexpensive systems that do not have high availability or high quality. Commodity Hardware consists of RAM because there are specific services that need to be executed on RAM. Hadoop can be run on any commodity hardware and does not require any super computer s or high end hardware configuration to execute jobs.

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

### NameNode 50070

### Job Tracker 50030

### Task Tracker 50060

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

### hadoop fs –setrep –w 5 /my/test\_dir

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

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

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

### 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

### 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 while maximizing the throughput and the utilization of the cluster.

### Fair Scheduler Fair scheduling is a method of assigning resources to applications such that all apps get, on average, an equal share of resources over time.

### 39. What is distcp?

### istcp is the program comes with Hadoop for copying large amount of data to and from Hadoop file systems in parallel.

### It is implemented as MapReduce job where copying is done through maps that run in parallel across the cluster.

### There are no reducers.

### 40. Command to format the NameNode?

### $ hdfs namenode -format

### Q34) 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

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

### ResourceManager: Global resource manager.

### NodeManager: One per data node, It manages and monitors usage of the container (resources in terms of Memory, CPU).

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

### 42. What are different daemons in YARN?

### ResourceManager: Global resource manager.

### NodeManager: One per data node, It manages and monitors usage of the container (resources in terms of Memory, CPU).

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

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

### Scheduler

### It allocates the resources (containers) to various running applications: Container elements such as memory, CPU, disk etc.

### ApplicationManager It accepts job-submissions, negotiating for container for executing the application specific ApplicationMaster and provides the service for restarting the ApplicationMaster container on failure.

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

### The NodeManager is the resource manager for the node (Per machine) and is responsible for containers, monitoring their resource usage (cpu, memory, disk, network) and reporting the same to the ResourceManager

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

### ApplicationMaster is per application and it has the responsibility of negotiating appropriate resource containers from the Scheduler, tracking their status and monitoring for progress.

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

### Partitioner controls the partitioning of the keys of the intermediate map-outputs by the default. The key to decide the partition uses hash function. Default partitioner is HashPartitioner. A custom partitioner is implemented to control, which keys go to which Reducer.

### 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 increase the efficiency of a MapReduce program. It can be used to aggregate intermediate map output locally on individual mapper outputs.

### Combiners can help reduce the amount of data that needs to be transferred across to the reducers.

### Reducer code as a combiner if the operation performed is commutative and associative.

### 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. 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.

### 50. What is IdentityMapper?

### IdentityMapper implements the mapping inputs directly to output. IdentityMapper.class is used as a default value when JobConf.setMapperClass is not set.

### 51. What is IdentityReducer?

### In IdentityReducer no reduction is performed, writing all input values directly to the output. IdentityReducer.class is used as a default value when JobConf.setReducerClass is not set

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

### Reducer reduces a set of intermediate values, which has same key to a smaller set of values. The framework then calls reduce(). Syntax: reduce(WritableComparable, Iterable, Context) method for each pair in the grouped inputs. Reducer has three primary phases:

### Shuffle

### Sort

### Reduce

### 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

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

### Shuffle: Sorted output (Mapper) à Input (Reducer). Framework then fetches the relevant partition of the output of all the mappers.

### Sort: The framework groups Reducer inputs by keys. The shuffle and sort phases occur simultaneously; while map-outputs are being fetched they are merged.

### Secondary Sort: Grouping the intermediate keys are required to be different from those for grouping keys before reduction, then Job.setSortComparatorClass(Class).

### Reduce: reduce(WritableComparable, Iterable, Context) method is called for each pair in the grouped inputs. 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.

### Q65) What can be optimum value for Reducer?

### Value of Reducers can be: 0.95

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

### Increasing number of reducers

### Increases the framework overhead

### Increases load balancing

### Lowers the cost of failures

### 56. What are a Counter and its purpose?

### The counter is a facility for MapReduce applications to report its statistics. They can be used to track job progress in a very easy and flexible manner. It is defined by MapReduce framework or by applications. Each Counter can be of any Enum type. Applications can define counters of type Enum and update them via counters.incrCounter in the map and/or reduce methods.

### 57.Define different types of Counters?

### Built in Counters:

### Map Reduce Task Counters

### Job Counters

### Custom Java Counters:

### MapReduce allows users to specify their own counters (using Java enums) for performing their own counting operation.

### 58.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.

### 59. Explain speculative execution.

### Speculative execution is a way of dealing with individual machine’s performance. As there are lots of machines in the cluster, some machines can have low performance, which affects the performance of the whole job.

### Speculative execution in Hadoop can run multiple copies of the same map or reduce task on different task tracker nodes and the results from first node to finish are used.

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

### DistributedCache is a facility provided by the MapReduce framework to cache files (text, archives, jars etc.) needed by applications. It distributes application-specific, large, read-only files efficiently. The user needs to use DistributedCache to distribute and symlink the script file.

### 61. What is the Job interface in MapReduce framework?

### Job is the primary interface for a user to describe a MapReduce job to the Hadoop framework for execution. Some basic parameters are configured for example:

### Job.setNumReduceTasks(int)

### Configuration.set(JobContext.NUM\_MAPS, int)

### Mapper

### Combiner (if any)

### Partitioner

### Reducer

### InputFormat

### OutputFormat implementations

### setMapSpeculativeExecution(boolean))/ setReduceSpeculativeExecution(boolean))

### Maximum number of attempts per task (setMaxMapAttempts(int)/ setMaxReduceAttempts(int)) etc.

### DistributedCache for large amounts of (read-only) data.

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

### The framework will try to execute a map task or reduce task by default 4 times before giving up on it.

### 63. Explain InputFormat?

### InputFormat describes the input-specification for a MapReduce job. The MapReduce framework depends on the InputFormat of the job to:

### Checks the input-specification of the job. It then splits the input file(s) into logical InputSplit instances, each of which is then assigned to an individual Mapper.

### To extract input records from the logical InputSplit for processing by the Mapper it provides the RecordReader implementation. Default: TextInputFormat

### 64. What is InputSplit and RecordReader?

### InputSplit specifies the data to be processed by an individual Mapper. In general, InputSplit presents a byte-oriented view of the input.

### 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 for a MapReduce job. The MapReduce framework depends on the OutputFormat of the job to: It checks the job output-specification

### To write the output files of the job in the pairs, it provides the RecordWriter implementation. Default: TextOutputFormat

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

### Hadoop provides an option where a certain set of bad input records can be skipped when processing map inputs. This feature can be controlled by the SkipBadRecords class.

67. **HDFS and Mapreduce Features in Hadoop Versions:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **1.x** | **0.22** | **2.x** |
| Secure Authentication | Yes | No | Yes |
| Old Configuration Names | Yes | Deprecate | Deprecate |
| New Configuration Names | No | Yes | Yes |
| Old Mapreduce API | Yes | Yes | Yes |
| New Mapreduce API | Yes | Yes | Yes |
| Mapreduce 1 Runtime | No | No | Yes |
| Mapreduce 2 Runtime | No | No | Yes |
| HDFS Federation | No | No | Yes |
| HDFS High Availability | No | No | Yes |

**68. What is Cluster Rebalancing?**

* The architecture of HDFS is in flow with data rebalancing schemes.
* A scheme automatically move data from one data node into another data node.

If in case, there is a sudden rise in particular file, a scheme dynamically creates additional replies and rebalance the data within cluster. This type of data rebalancing schemes are yet to be positioned.

**69. What is Data Integrity?**

1. Data Integrity is a method where a block of data is fetched from a datanode, but it comes corrupted. This corruption is due to faults in storage device, network and buggy software.
2. HDFS software employs checksum checking on the HDFS file contents.
3. When a client handles file contents, it verifies that data retrieved matches the checksum of the relevant checksum file.
4. If not, then the client may opt to retrieve the block from variant datanode that replicates the other block.

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

1. Hadoop File System indicating the compiler to interact with Linux local environment to HDFS environment.
2. Hadoop File System is not support the -vi command. Because HDFS is write once.
3. Hadoop File System is support for -touchz command.
4. Hadoop File System looks the only HDFS directory but not local directory.
5. We can not create a file on top of HDFS.
6. We can not create the file on local.
7. We can not update a file on top of HDFS. We can updations in local, after that file is put into the HDFS.
8. Hadoop file system does not support hard links (or) soft links.
9. Hadoop File System does not implement user quotas.
10. Error implementation is sent to stderr & output is sent to stdout.

Display detailed help for a command:  
Hadoop fs - help

**71 User Command Archive?**

* Hadoop stores the small files inefficiently such as each file get stored in a  block & namenode has to keep the metadata information in memory so with this reason most of the namenode memory will get eat up this small files only which results in a wastage of memory.
* To avoid the same problem we use hadoop archives (or) har files (.har a the extension for all the archive files).
* When creating archive directory the input is converted to mapreduce jobs, so we can call hadoop archives as a input for our mapreduce programming.

1. Hadoop archives are special format archives.  
2. Hadoop archive maps to a file system directory.  
3. Hadoop archive always has a .har extension  
4. Hadoop archive directory contains metadata.

**72. What is Serialization?**

1. Serialization transforms objects into bytes
2. Hadoop utilizes PR6 for transmitting across the network.
3. Hadoop employs a very own serialization format which is writable
4. Comparison types are crucial
5. Hadoop enables a Raw comparator, that abolishes deserialization
6. External frameworks are enables via : **enter Avro**

**73. Datanode block scanner?**

All the datanodes runs the block scanner, which periodically verifies all the blocks stored on the datanode. This allows bad blocks to be detected and fixed before they are read by clients.  
It maintains

1. A list of blocks to verify
2. It scans them one by one for checksum errors.
3. Block scanner report can be verify by visiting http://datanode:50075/blockScannerReport.

**74 What is HBASE Data Storage?**

HBASE is column oriented data storage

**Column-Oriented:**

1. The reason to store values on a per column basis instead is based on the assumption
2. That for specific queries, not all of the values are needed
3. Reduced I/O
4. The data of column-oriented databases is saved in the way grouped by columns and the following column values are stored on the contiguous disk locations. This is quite different from the conventional approach followed by the traditional databases which stores all the rows contiguously.
   1. **Storefile**: Store File for each state for each region for the table.
   2. **Block**: Blocks within a store file within a store for each region for the table
5. 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)

**75.What is Hadoop Streaming?**

A utility to enable Mapreduce code in any language: C, Perl, Python, C++, Bash etc. The examples include a python mapper and an AWK reducer.

**76. What is the difference between Base & NOSQL?**

Favours consistencies are availability (but availability is good in practice)  
Great hadoop integration (very efficient bulk loads, Mapreduce Analysis)  
Ordered range partitions(not hash)  
Automatically shards/scales (just run on more servers)  
Sparse column stronge(not key value)

### HIVE

### 1. What is Hive?

### Hive is a Hadoop based system for querying and analyzing large volumes of Structured data which is stored on HDFS or in other words Hive is an query engine built to work on top of Hadoop that can compile queries into Map Reduce jobs and run them on the cluster.

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

### Data warehousing applications where more static data is analyzed.

### Fast response time is not the criteria.

### Data is not changing rapidly.

### An abstract to underlying MapReduce programs

### Like SQL

### 3. What are the limitations of Hive?

### Hive does not provide:

### Record-level operations like INSERT, DELETE or UPDATE.

### Cannot be used for low latency jobs.

### Transaction.

### 4. What are the differences between Hive and RDBMS?

### HIVE:

### Schema on Read

### Batch processing jobs

### Data stored on HDFS

### Processed using MapReduce

### RDBMS:

### Schema on write

### Real time jobs

### Data stored on internal structure

### Processed using database

### 5. What are the components of Hive architecture?

### Hive Driver

### Metastore

### Hive CLI/HUE/HWI

### 6. What is the purpose of Hive Driver?

### Hive Driver is responsible for compiling, optimizing and then executing the HiveQL.

### 7. What is a Metastore and what it stores?

### It is a database by default Derby SQL server

### Holds metadata about table definition, column, and data types partitioning information,

### It can be stored in MySQL, derby, oracle etc.

### 8. What is the purpose of storing the metadata?

### People want to read the dataset with a particular schema in mind. For e.g.: BA and CFO of a company look at the data with a particular schema. BA may be interested in say IP addresses and timings of the clicks in a weblog while the CFO may be interested in say the clicks that were direct clicks on the website or from paid Google adds. Underneath it’s the same dataset that is accessed. This schema is used again and again. So it makes sense to store this schema in a RDBMS.

### 9. List the various options available with the Hive command.

### Syntax:

### $ ./hive –service serviceName where serviceName options are: cli help hiveserver hwi jar lineage metastore rcfile

### 10 Explain the different services that can be invoked using the Hive command.

### cli

### default service

### used to define tables, run queries, etc.

### hiveserver

### aemon that listens for Thrift connections from other processes

### hwi

### Simple web interface for running queries

### jar

### Extension of the hadoop jar command

### metastore

### External Hive metastore service to support multiple clients

### rcfile

### Tool for printing the contents of an RFFile

### 11 Can you execute Hadoop dfs Commands from Hive CLI? How?

### Hadoop dfs commands can be run from within the hive CLI by dropping the hadoop work from the command and adding a semicolon in the end.

### For Example:

### Hadoop dfs command: hadoop dfs -ls / From within hive hive > dfs -ls / ;

### 12. How to give multiline comments in Hive Scripts?

### Hive does not support multiline comments. All lines of comments should start with the string — For e.g. — This is first line of comment — This is second line of comment !!

### 13. What is the reason for creating a new metastore\_db whenever Hive query is run from a different directory?

### Embedded mode: Whenever Hive runs in embedded mode, it checks whether the metastore exists. If the metastore does not exist then it creates the local metastore.

### Property: Default value

### javax.jdo.option.ConnectionURL = “jdbc:derby:;databaseName=metastore\_db;create=true”

### 14. When Hive is run in embedded mode, how to share the metastore within multiple users?

### No. For sharing use the standalone database (like MySQL, PostGresQL) for metastore

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

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

What are the different types of tables available in HIve?

There are two types. Managed table and external table. In managed table both the data an schema in under control of hive but in external table only the schema is under control of Hive.

Is Hive suitable to be used for OLTP systems? Why?

No Hive does not provide insert and update at row level. So it is not suitable for OLTP system.

Can a table be renamed in Hive?

Alter Table table\_name RENAME TO new\_name

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

Using REPLACE column option

ALTER TABLE table\_name REPLACE COLUMNS ……

What is a metastore in Hive?

It is a relational database storing the metadata of hive tables, partitions, Hive databases etc

What is the need for custom Serde?

Depending on the nature of data the user has, the inbuilt SerDe may not satisfy the format of the data. SO users need to write their own java code to satisfy their data format requirements.

Why do we need Hive?

Hive is a tool in Hadoop ecosystem which provides an interface to organize and query data in a databse like fashion and write SQL like queries. It is suitable for accessing and analyzing data in Hadoop using SQL syntax.

What is the default location where hive stores table data?

hdfs://namenode\_server/user/hive/warehouse

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

* Local mode
* Distributed mode
* Pseudodistributed mode

Is there a date data type in Hive?

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

What are collection data types in Hive?

There are three collection data types in Hive.

* ARRAY
* MAP
* STRUCT

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

Yes, using the ! mark just before the command.

For example !pwd at hive prompt will list the current directory.

What is a Hive variable? What for we use it?

The hive variable is variable created in the Hive environment that can be referenced by Hive scripts. It is used to pass some values to the hive queries when the query starts executing.

Can hive queries be executed from script files? How?

Using the source command.

**Example −**

Hive> source /path/to/file/file\_with\_query.hql

What is the importance of .hiverc file?

It is a file containing list of commands needs to run when the hive CLI starts. For example setting the strict mode to be true etc.

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

What do you mean by schema on read?

The schema is validated with the data when reading the data and not enforced when writing data.

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

SHOW DATABASES LIKE ‘p.\*’

What does the “USE” command in hive do?

With the use command you fix the database on which all the subsequent hive queries will run.

How can you delete the DBPROPERTY in Hive?

There is no way you can delete the DBPROPERTY.

What is the significance of the line

set hive.mapred.mode = strict;

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’)

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

org.apache.hadoop.mapred.TextInputFormat

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

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

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

When we issue the command DROP TABLE IF EXISTS table\_name

Hive throws an error if the table being dropped does not exist in the first place.

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. It has to be moved manually.

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

ALTER TABLE table\_name

CHANGE COLUMN new\_col INT

BEFORE x\_col

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

No. It only reduces the number of files which becomes easier for namenode to manage.

How can you stop a partition form being queried?

By using the ENABLE OFFLINE clause with ALTER TABLE atatement.

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.

If you omit the OVERWRITE clause while creating a hive table,what happens to file which are new and files which already exist?

The new incoming files are just added to the target directory and the existing files are simply overwritten. Other files whose name does not match any of the incoming files will continue to exist.

If you add the OVERWRITE clause then all the existing data in the directory will be deleted before new data is written.

What does the following query do?

INSERT OVERWRITE TABLE employees

PARTITION (country, state)

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

FROM staged\_employees se;

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 exploe()

How can Hive avoid mapreduce?

If we set the property hive.exec.mode.local.auto to true then hive will avoid mapreduce to fetch query results.

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

The LIKE operator behaves the same way as the regular SQL operators used in select queries. Example −

street\_name like ‘%Chi’

But the RLIKE operator uses more advance regular expressions which are available in java

Example − street\_name RLIKE ‘.\*(Chi|Oho).\*’ which will select any word which has either chi or oho in it.

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

No. As this kind of Join can not be implemented in mapreduce

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 to be taken in the first position and largest table should be taken in the last position.

What is the usefulness of the DISTRIBUTED BY clause in Hive?

It controls ho wthe map output is reduced among the reducers. It is useful in case of streaming data

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

Select cast(price as FLOAT)

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

Hive will return NULL

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

No. The name of a view must be unique whne compared to all other tables and views present in the same database.

Can we LOAD data into a view?

No. A view can not be the target of a INSERT or LOAD statement.

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

Indexes occupies space and there is a processing cost in arranging the values of the column on which index is cerated.

Give the command to see the indexes on a table.

SHOW INDEX ON table\_name

This will list all the indexes created on any of the columns in the table table\_name.

What is bucketing ?

The values in a column are hashed into a number of buckets which is defined by user. It is a way to avoid too many partitions or nested partitions while ensuring optimizes query output.

What does /\*streamtable(table\_name)\*/ do?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 the archived file can be queried using hive. The disadvantage is it will cause less efficient query and does not offer any space savings.

What is a generic UDF in hive?

It is a UDF which is created using a java program to server some specific need not covered under the existing functions in Hive. It can detect the type of input argument programmatically and provide appropriate response.

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

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

OVERWRITE INTO TABLE address;

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

The TBLPROPERTIES clause is used to add the creator name while creating a table.

The TBLPROPERTIES is added like −

TBLPROPERTIES(‘creator’= ‘Joan’)

## **What is Next ?**

Further you can go through your past assignments you have done with the subject and make sure you are able to speak confidently on them. If you are fresher then interviewer does not expect you will answer very complex questions, rather you have to make your basics concepts very strong.

Second it really doesn't matter much if you could not answer few questions but it matters that whatever you answered, you must have answered with confidence. So just feel confident during your interview. We at tutorialspoint wish you best luck to have a good interviewer and all the very best for your future endeavor. Cheers :-)

## **Apache Hive Interview Questions**

Here is the comprehensive list of the most frequently asked Apache Hive Interview Questions that have been framed after deep research and discussion with the industry experts.

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

|  |  |
| --- | --- |
| **Hive vs HBase** | |
| **HBase** | **Hive** |
| 1. HBase is built on the top of HDFS | 1. It is a data warehousing infrastructure |
| 2. HBase operations run in a real-time on its database rather | 2. Hive queries are executed as MapReduce jobs internally |
| 3. Provides low latency to single rows from huge datasets | 3. Provides high latency for huge datasets |
| 4. Provides random access to data | 4. Provides random access to data |

### ****2. 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.

### ****3. Where does the data of a Hive table gets stored?****

By default, the Hive table is stored in an HDFS directory – /user/hive/warehouse. One can change it by specifying the desired directory in hive.metastore.warehouse.dir configuration parameter present in the hive-site.xml.

### ****4. What is a metastore in Hive?****

[***Metastore***](https://www.edureka.co/blog/hive-tutorial/) in Hive stores the meta data information using RDBMS and an open source ORM (Object Relational Model) layer called Data Nucleus which converts the object representation into relational schema and vice versa.

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

Hive stores metadata information in the metastore using RDBMS instead of HDFS. The reason for choosing RDBMS is to achieve low latency as HDFS read/write operations are time consuming processes.

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

Local Metastore:

In local metastore configuration, the metastore service runs in the same JVM in which the Hive service is running and connects to a database running in a separate JVM, either on the same machine or on a remote machine.

Remote Metastore:

In the remote metastore configuration, the metastore service runs on its own separate JVM and not in the Hive service JVM. Other processes communicate with the metastore server using Thrift Network APIs. You can have one or more metastore servers in this case to provide more availability.

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

By default, Hive provides an embedded Derby database instance backed by the local disk for the metastore. This is called the embedded metastore configuration.

### ****8. 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.

Following are the steps to configure MySQL database as the local metastore in Apache Hive:

* One should make the following changes in hive-site.xml:
  + javax.jdo.option.ConnectionURL property should be set to jdbc:mysql://host/dbname?createDataba  
    seIfNotExist=true.
  + javax.jdo.option.ConnectionDriverName property should be set to com.mysql.jdbc.Driver.
  + One should also set the username and password as:
    - javax.jdo.option.ConnectionUserName is set to desired username.
    - javax.jdo.option.ConnectionPassword is set to the desired password.
* The JDBC driver JAR file for MySQL must be on the Hive’s classpath, i.e. The jar file should be copied into the Hive’s lib directory.
* Now, after restarting the Hive shell, it will automatically connect to the MySQL database which is running as a standalone metastore.

### ****9. What is the difference between external table and managed table?****

Here is the key difference between an external table and managed table:

* In case of managed table, If one drops a managed table, the metadata information along with the table data is deleted from the Hive warehouse directory.
* On the contrary, in case of an external table, Hive just deletes the metadata information regarding the table and leaves the table data present in HDFS untouched.

***Note:*** I would suggest you to go through the blog on [***Hive Tutorial***](https://www.edureka.co/blog/hive-tutorial/?#data_model)to learn more about Managed Table and External Table in Hive.

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

Yes, it is possible to change the default location of a managed table. It can be achieved by using the clause – LOCATION ‘<hdfs\_path>’.

### ****11. When should we use SORT BY instead of ORDER BY?****

We should use SORT BY instead of ORDER BY when we have to sort huge datasets because SORT BY clause sorts the data using multiple reducers whereas ORDER BY sorts all of the data together using a single reducer. Therefore, using ORDER BY against a large number of inputs will take a lot of time to execute.

[[](https://www.edureka.co/big-data-and-hadoop)](https://www.edureka.co/big-data-and-hadoop" \t "_blank)

### [Big Data Hadoop Certification Training](https://www.edureka.co/big-data-and-hadoop" \t "_blank)

* *[Instructor-led Sessions](https://www.edureka.co/big-data-and-hadoop" \t "_blank)*
* *[Real-life Case Studies](https://www.edureka.co/big-data-and-hadoop" \t "_blank)*
* *[Assignments](https://www.edureka.co/big-data-and-hadoop" \t "_blank)*
* *[Lifetime Access](https://www.edureka.co/big-data-and-hadoop" \t "_blank)*

[Explore Curriculum](https://www.edureka.co/big-data-and-hadoop" \t "_blank)

### ****12. What is a partition in Hive?****

Hive organizes tables into partitions for grouping similar type of data together based on a column or partition key. Each Table can have one or more partition keys to identify a particular partition. Physically, a partition is nothing but a sub-directory in the table directory.

### ****13. Why do we perform partitioning in Hive?****

Partitioning provides granularity in a Hive table and therefore, reduces the query latency by scanning only **relevant** partitioned data instead of the whole data set.

For example, we can partition a transaction log of an e – commerce website based on month like Jan, February, etc. So, any analytics regarding a particular month, say Jan, will have to scan the Jan partition (sub – directory) only instead of the whole table data.

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

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

One may use dynamic partition in following two cases:

* Loading data from an existing non-partitioned table to improve the sampling and therefore, decrease the query latency.
* When one does not know all the values of the partitions before hand and therefore, finding these partition values manually from a huge data sets is a tedious task.

### ****15. 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 the table according to each month. So, for each month we will be scanning only the partitioned data instead of whole data sets.

As we know, we can’t partition an existing non-partitioned table directly. So, we will be taking following steps to solve the very problem:

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 ‘,’ ;

2. Enable dynamic partitioning in Hive:

SET hive.exec.dynamic.partition = true;

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

3. 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;

Now, we can perform the query using each partition and therefore, decrease the query time.

### ****16. 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’;

***Note:*** I suggest you to go through the dedicated blog on [***Hive Commands***](https://www.edureka.co/blog/hive-commands-with-examples) where all the commands present in Apache Hive have been explained with an example.

### ****17. 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 that can be created by a mapper or reducer is set to 100. One can change it by issuing the following command:

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

***Note:***You can set the total number of dynamic partitions that can be created by one statement by using: SET hive.exec.max.dynamic.partitions = <value>

### ****18. Scenario:****

***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;

***Things to Remember:***

[[](https://www.edureka.co/big-data-and-hadoop)](https://www.edureka.co/big-data-and-hadoop" \t "_blank)

### [Big Data Hadoop Certification Training](https://www.edureka.co/big-data-and-hadoop" \t "_blank)

[Watch The Course Preview](https://www.edureka.co/big-data-and-hadoop" \t "_blank)

* By default, hive.exec.dynamic.partition configuration property is set to False in case you are using Hive whose version is prior to 0.9.0.
* hive.exec.dynamic.partition.mode is set to strict by default. Only in non – strict mode Hive allows all partitions to be dynamic.

### ****19. 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/)requires the data belonging to a unique join key to be present in the same partition. But what about those cases where your partition key differs from that of join key? Therefore, in these cases you 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.

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

Hive determines the bucket number for a row by using the formula: hash\_function (bucketing\_column) modulo (num\_of\_buckets). Here, hash\_function depends on the column data type. For integer data type, the hash\_function will be:

hash\_function (int\_type\_column)= value of int\_type\_column

### ****21. 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 one to have the correct number of reducer while using ‘CLUSTER BY’ clause for bucketing a column. In case it’s not done, one may find the number of files that will be generated in the table directory to be not equal to the number of buckets. As an alternative, one may also set the number of reducer equal to the number of buckets by using set mapred.reduce.task = num\_bucket.

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

One of the Hive query optimization methods is Hive index. Hive index is used to speed up the access of a column or set of columns in a Hive database because with the use of index the database system does not need to read all rows in the table to find the data that one has selected.

### ****23. Scenario:****

***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

5 Emily Rose rose.emily4@surveymonkey.com Female 119.92.21.19

***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 and many other third-party SerDes are also available.

Hive provides a specific SerDe for working with CSV files. We can use this SerDe for the sample.csv by 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’;

### ****24. Scenario:****

**Suppose, I have a lot of small CSV files present in /input directory in HDFS and I want to create 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?**

One 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)

[[](https://www.edureka.co/big-data-and-hadoop)](https://www.edureka.co/big-data-and-hadoop" \t "_blank)

### [Big Data Hadoop Certification Training](https://www.edureka.co/big-data-and-hadoop" \t "_blank)

[Weekday / Weekend BatchesSee Batch Details](https://www.edureka.co/big-data-and-hadoop" \t "_blank)

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.

### ****Conclusion:****

I hope you find this blog on Apache Hive Interview Questions to be informative and helpful. You are welcome to check out our other interview question blogs as well that covers all the components present in Hadoop framework. Kindly, refer to the links given below and enjoy the reading:

* [***Top 50 Hadoop Interview Questions***](https://www.edureka.co/blog/interview-questions/top-50-hadoop-interview-questions-2016/)
* [***Hadoop Cluster Interview Questions***](https://www.edureka.co/blog/interview-questions/hadoop-interview-questions-hadoop-cluster/)
* [**Hadoop HDFS Interview Questions**](https://www.edureka.co/blog/interview-questions/hadoop-interview-questions-hdfs-2/)
* [***Hadoop MapReduce Interview Questions***](https://www.edureka.co/blog/interview-questions/hadoop-interview-questions-mapreduce/)
* [***Pig Interview Questions***](https://www.edureka.co/blog/interview-questions/hadoop-interview-questions-pig/)
* [***HBase Interview Questions***](https://www.edureka.co/blog/interview-questions/hbase-interview-questions/)

Got a question for us? Please mention it in the comments section of this Apache Hive Interview Questions and we will get back to you.

**Que 1. What is Apache Hive?**

**Ans.** Basically, a tool which we call a data warehousing tool is Hive. However, Hive gives [**SQL queries**](https://data-flair.training/blogs/sql-subquery/) to perform an analysis and also an abstraction. Although, Hive it is not a database it gives you logical abstraction over the databases and the tables.

**Que 2. What kind of applications is supported by Apache Hive?**

**Ans.** All those client applications which are written in [**Java**](https://data-flair.training/blogs/java-tutorial/), PHP, [**Python**](https://data-flair.training/blogs/python-tutorial-for-beginners/), C++ or Ruby by exposing its thrift server, Hive supports them.

**Que 3. Is Hive suitable to be used for OLTP systems? Why?**

**Ans.** No, it is not suitable for OLTP system since it does not offer insert and update at the row level.

**Que 4. Where does the data of a Hive table gets stored?**

**Ans.** In an **HDFS directory** – /user/hive/warehouse, the Hive table is stored, by default only. Moreover, by specifying the desired directory in hive.metastore.warehouse.dir configuration parameter present in the hive-site.xml, one can change it.   
  
**Que 5. What is a metastore in Hive?**

**Ans**. Basically, to store the metadata information in the Hive we use [**Metastore**](https://data-flair.training/blogs/apache-hive-metastore/). Though, it is possible by using **RDBMS** and an open source ORM (Object Relational Model) layer called Data Nucleus. That converts the object representation into the relational schema and vice versa.

**Que 6. Why does Hive not store metadata information in HDFS?**

**Ans.** Using RDBMS instead of HDFS, Hive stores metadata information in the metastore. Basically, to achieve low latency we use RDBMS. Because **HDFS read/write operations** are time-consuming processes.  
  
**Que 7. What is the difference between local and remote metastore?**

**Ans. Local Metastore:**

It is the metastore service runs in the same [**JVM**](https://data-flair.training/blogs/java-virtual-machine-jvm/) in which the Hive service is running and connects to a database running in a separate JVM. Either on the same machine or on a remote machine.  
  
**Remote Metastore:**  
  
In this configuration, the metastore service runs on its own separate JVM and not in the Hive service JVM.  
  
**Que 8. What is the default database provided by Apache Hive for metastore?**

**Ans.** It offers an embedded Derby database instance backed by the local disk for the metastore, by default. It is what we call embedded metastore configuration.

**Que 9. What is the difference between the external table and managed table?**

**Ans. Managed table**

The metadata information along with the table data is deleted from the Hive warehouse directory if one drops a managed table.\

**External table**  
Hive just deletes the metadata information regarding the table. Further, it leaves the table data present in HDFS untouched.

[**Read more about Hive internal tables vs External tables**](https://data-flair.training/blogs/hive-internal-tables-vs-external-tables-comparison/)

**Que 10. Is it possible to change the default location of a managed table?**

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

**Hive Interview Questions for Freshers- Q. 1,2,3,4,5,7,8,9,10**

**Hive Interview Questions for Experience- Q. 6**  
  
**Que 11. When should we use SORT BY instead of ORDER BY?**

**Ans**. Despite [**ORDER BY**](https://data-flair.training/blogs/hiveql-order-by-query/) we should use SORT BY. Especially while we have to sort huge datasets. The reason is SORT BY clause sorts the data using multiple reducers. ORDER BY sorts all of the data together using a single reducer. Hence, using ORDER BY will take a lot of time to execute a large number of inputs.

**Que 12. What is a partition in Hive?**

**Ans.** Basically, for the purpose of grouping similar type of data together on the basis of column or partition key, Hive organizes tables into partitions. Moreover, to identify a particular partition each table can have one or more partition keys. On defining [**Hive Partition**](https://data-flair.training/blogs/apache-hive-partitioning/), in other words, it is a sub-directory in the table directory.

**Que 13. Why do we perform partitioning in Hive?**

**Ans.** In a Hive table, Partitioning provides granularity. Hence, by scanning only relevant partitioned data instead of the whole dataset it reduces the query latency.  
  
**Que 14. What is dynamic partitioning and when is it used?**

**Ans.** Dynamic partitioning values for partition columns are known in the runtime. In other words, it is known during loading of the data into a Hive table.

* **Usage:**

1. While we Load data from an existing non-partitioned table, in order to improve the sampling. Thus it decreases the query latency.
2. Also, while we do not know all the values of the partitions beforehand. Thus, finding these partition values manually from a huge dataset is a tedious task.

**Que 15. Why do we need buckets?**

**Ans.** Basically, for [**performing bucketing**](https://data-flair.training/blogs/bucketing-in-hive/) to a partition there are two main reasons:

* A map side join requires the data belonging to a unique join key to be present in the same partition.
* It allows us to decrease the query time. Also, makes the sampling process more efficient.

**Que 16. How Hive distributes the rows into buckets?**

**Ans.** By using the formula: hash\_function (bucketing\_column) modulo (num\_of\_buckets) Hive determines the bucket number for a row. Basically, hash\_function depends on the column data type. Although, hash\_function for integer data type will be:  
hash\_function (int\_type\_column)= value of int\_type\_column

**Que 17.  What is indexing and why do we need it?**

**Ans.** [**Hive index**](https://data-flair.training/blogs/hive-view-hive-index/) is a Hive query optimization techniques. Basically, we use it to speed up the access of a column or set of columns in a Hive database. Since, the database system does not need to read all rows in the table to find the data with the use of the index, especially that one has selected.

**Que 18. What is the use of Hcatalog?**

**Ans.** Basically, to share data structures with external systems we use Hcatalog. It offers access to hive metastore to users of other tools on Hadoop. Hence, they can read and write data to hive’s data warehouse.

**Que 19. Where is table data stored in Apache Hive by default?**

**Ans.** hdfs: //namenode\_server/user/hive/warehouse

**Que 20. Are multi-line comments supported in Hive?**

**Ans.**No

**Hive Interview Questions for Freshers- Q. 12,13,14,15,17,18,19,20**

**Hive Interview Questions for Experience- Q. 11,16**

**Que 21. What is ObjectInspector functionality?**

**Ans.** To analyze the structure of individual columns and the internal structure of the row objects we use ObjectInspector. Basically, it provides access to complex objects which can be stored in multiple formats in Hive.

**Que 22. Explain about the different types of join in Hive**.

**Ans.** There are  4 different types of [**joins in HiveQL**](https://data-flair.training/blogs/hive-join/) –

* **JOIN-** It is very similar to Outer Join in SQL
* **FULL OUTER JOIN –** This join Combines the records of both the left and right outer tables. Basically, that fulfill the join condition.
* **LEFT OUTER JOIN-** Through this Join, All the rows from the left table are returned even if there are no matches in the right table.
* **RIGHT OUTER JOIN –** Here also, all the rows from the right table are returned even if there are no matches in the left table.

**Que 23. How can you configure remote metastore mode in Hive?**  
**Ans.** Basically, hive-site.xml file has to be configured with the below property, to configure metastore in Hive –  
hive.metastore.uris  
 thrift: //node1 (or IP Address):9083  
 IP address and port of the metastore host

**Que 24. Is it possible to change the default location of Managed Tables in Hive, if so how?**

**Ans.** Yes, by using the LOCATION keyword while creating the managed table, we can change the default location of Managed tables. But the one condition is, the user has to specify the storage path of the managed table as the value of the LOCATION keyword.

**Que 25.  How does data transfer happen from HDFS to Hive?**

**Ans.**Basically, the user need not LOAD DATA that moves the files to the /user/hive/warehouse/. But only if data is already present in HDFS. Hence, using the keyword external that creates the table definition in the hive metastore  the user just has to define the table.  
Create external table table\_name (  
 id int,  
 myfields string  
)  
location ‘/my/location/in/hdfs’;

**Que 26. What are the different components of a Hive architecture?**

**Ans.** There are several components of [**Hive Architecture**](https://data-flair.training/blogs/apache-hive-architecture/). Such as –

1. User Interface – Basically, it calls the execute interface to the driver. Further, driver creates a session handle to the query. Then sends the query to the compiler to generate an execution plan for it.
2. Metastore – It is used to Send the metadata to the compiler. Basically, for the execution of the query on receiving the send MetaData request.
3. Compiler- It generates the execution plan. Especially, that is a DAG of stages where each stage is either a metadata operation, a map or reduce job or an operation on HDFS.
4. Execute Engine- Basically,  by managing the dependencies for submitting each of these stages to the relevant components we use Execute engine.

**Que 27. Wherever (Different Directory) I run the hive query, it creates new metastore\_db, please explain the reason for it?**

**Ans.** Basically, it creates the local metastore, while we run the hive in embedded mode. Also, it looks whether metastore already exist or not before creating the metastore. Hence, in configuration file hive-site.xml. Property is “javax.jdo.option.ConnectionURL” with default value “jdbc:derby:;databaseName=metastore\_db;create=true” this property is defined. Hence, to change the behavior change the location to the absolute path, thus metastore will be used from that location.

**Que 28. Is it possible to use the same metastore by multiple users, in case of the embedded hive?**

**Ans.** No, we cannot use metastore in sharing mode. It is possible to use it in standalone “real” database. Such as MySQL or PostGresSQL.

**Que 29. Usage of Hive.**

**Ans.** Here, we will look at following [**Hive**](https://hive.apache.org/) usages.  
– We use Hive for Schema flexibility as well as evolution.  
– Moreover, it is possible to portion and bucket, tables in Apache Hive.  
– Also, we can use JDBC/ODBC drivers, since they are available in Hive.

**Que 30. Features and Limitations of Hive.**

**Ans. Features of Hive**

1. The best feature is it offers data summarization, query, and analysis in much easier manner.
2. To process data without actually storing in HDFS, Hive supports external tables.
3. Moreover, it fits the low-level interface requirement of Hadoop perfectly.

* **Limitation of Hive**

1. We can not perform real-time queries with Hive. Also, it does not offer row-level updates.
2. Moreover,  for interactive data browsing Hive offers acceptable latency.
3. Also, we can say Hive is not the right choice for online transaction processing.

**Hive Interview Questions for Freshers- Q. 22,24,25,26,28,29,30**

**Hive Interview Questions for Experience- Q. 21,23,27**

Top of Form

**Get the most demanding skills of IT Industry - Learn Hadoop  
**

Bottom of Form

## **Summary**

As a result, we have we have seen top 30 Hive Interview Questions and Answers. Thus, once you go through it, you will get an in-depth knowledge of questions which may frequently ask in Hive interview. We hope all these questions will help you in preparing well for your Hive interviews ahead.  
  
Moreover, we encourage you to add the Interview Questions in Hive which you came across if you have attended any Hadoop interviews previously. Also, if you want to

**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 |

**2. What is the definition of Hive? What is the present version of Hive and explain about ACID transactions in Hive?**

Hive is an open source data warehouse system. We can use Hive for analyzing and querying in large data sets of Hadoop files. It’s similar to SQL. The present version of hive is 0.13.1. Hive supports ACID transactions: The full form of ACID is Atomicity, Consistency, Isolation, and Durability. ACID transactions are provided at the row levels, there are Insert, Delete, and Update options so that Hive supports ACID transaction.

* Insert
* Delete
* Update

Want to learn more about Hive? Go through this insightful blog “[What is Hive](https://intellipaat.com/blog/what-is-apache-hive/)?”

**3. Explain what is a Hive variable. What do we use it for?**

Hive variable is basically created in the Hive environment that is referenced by Hive scripting languages. It provides to pass some values to the hive queries when the query starts executing. It uses the source command.

Become Hadoop Certified in 85 hrs.

**GET CERTIFIED**

**4. What kind of data warehouse application is suitable for Hive? What are the types of tables in Hive?**

Hive is not considered as a full database. The design rules and regulations of Hadoop and HDFS put restrictions on what Hive can do.Hive is most suitable for data warehouse applications.  
Where :

* Analyzing the relatively static data.
* Less Responsive time.
* No rapid changes in data.Hive doesn’t provide fundamental features required for OLTP, Online Transaction Processing.Hive is suitable for data warehouse applications in large data sets.**Two types of tables in Hive**

1. Managed table.
2. External table.

Get a better understanding of Hive by going through this [Hive Tutorial](https://intellipaat.com/tutorial/hadoop-tutorial/apache-hive/) now.

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

Yes we can change the settings within Hive session, using the SET command. It helps to change Hive job settings for an exact query.  
Example: The following commands shows buckets are occupied according to the table definition.

hive> SET hive.enforce.bucketing=true;

We can see the current value of any property by using SET with the property name. SET will list all the properties with their values set by Hive.

hive> SET hive.enforce.bucketing;

hive.enforce.bucketing=true

And this list will not include defaults of Hadoop. So we should use the below like

SET -v

It will list all the properties including the Hadoop defaults in the system.

**Interested in learning Hive? Well, we have the comprehensive**[*Hive Training Course*](https://intellipaat.com/big-data-hadoop-training/#curriculum)**to give you a head start in your career.**

**6. Is it possible to add 100 nodes when we have 100 nodes already in Hive? How?**

**Yes, we can add the nodes by following the below steps.**

1. Take a new system create a new username and password.
2. Install the SSH and with master node setup ssh connections.
3. Add ssh public\_rsa id key to the authorized keys file.
4. Add the new data node host name, IP address and other details in /etc/hosts slaves file  
   192.168.1.102 slave3.in slave3.
5. Start the Data Node on New Node.
6. Login to the new node like suhadoop or ssh -X hadoop@192.168.1.103.
7. Start HDFS of a newly added slave node by using the following command  
   ./bin/hadoop-daemon.sh start data node.
8. Check the output of jps command on a new node

**7. 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 ('Intellipaat','-','is','-','a','-','eLearning',’-’,’provider’);

Output:

Intellipaat-is-a-eLearning-provider

So, every time we set the limits of the strings by ‘-‘. If it is common for every strings, then Hive provides another command

CONCAT\_WS. In this case,we have to specify the set limits of operator first.

CONCAT\_WS ('-',’Intellipaat’,’is’,’a’,’eLearning’,‘provider’);

Output: Intellipaat-is-a-eLearning-provider.

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

Trim function will delete the spaces associated with a string.  
Example:

TRIM(‘ INTELLIPAAT ‘);

Output:

INTELLIPAAT

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’);

Output:

TAAPILLETNI

**9. How to change the column data type in Hive? Explain RLIKE in Hive.**

We can change the column data type by using ALTER and CHANGE.  
The syntax is :

ALTER TABLE table\_name CHANGE column\_namecolumn\_namenew\_datatype;

Example: If we want to change the data type of the salary column from integer to bigint in the employee table.  
ALTER TABLE employee CHANGE salary salary BIGINT;RLIKE: Its full form is Right-Like and it is a special function in the Hive. It helps to examine the two substrings. i.e, if the substring of A matches with B then it evaluates to true.  
Example:

‘Intellipaat’ RLIKE ‘tell’ ◊ True

‘Intellipaat’ RLIKE ‘^I.\*’ ◊ True (this is a regular expression)

[Learn what is Hadoop Hive](https://intellipaat.com/blog/what-is-hive-hadoop/) in this detailed blog post now.

[**Learn Hive from Experts! Enrol Today**](https://intellipaat.com/big-data-hadoop-training/#course-content)

**10. What are the components used in Hive query processor?**

The components of a Hive query processor include

* Logical Plan of Generation.
* Physical Plan of Generation.
* Execution Engine.
* Operators.
* UDF’s and UDAF’s.
* Optimizer.
* Parser.
* Semantic Analyzer.
* Type Checking

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

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

**12. 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;

Hive tables can be pointed to the higher level directory and this is suitable for the directory structure which is like /data/country/state/city/

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

Header records in log files  
System=….  
Version=…  
Sub-version=….  
In the above three lines of headers that we do not want to include in our Hive query. To skip header lines from our tables in the Hive,set a table property that will allow us to skip the header lines.

CREATE EXTERNAL TABLE employee (

name STRING,

job STRING,

dob STRING,

id INT,

salary INT)

ROW FORMAT DELIMITED FIELDS TERMINATED BY ‘ ‘ STORED AS TEXTFILE

LOCATION ‘/user/data’

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

**14. What is the maximum size of string data type supported by hive? Mention the Hive support binary formats.**

The maximum size of string data type supported by hive is 2 GB.  
Hive supports the text file format by default and it supports the binary format Sequence files, ORC files, Avro Data files, Parquet files.  
Sequence files: Splittable, compressible and row oriented are the general binary format.  
ORC files: Full form of ORC is optimized row columnar format files. It is a Record columnar file and column oriented storage file. It divides the table in row split. In each split stores that value of the first row in the first column and followed sub subsequently.  
AVRO data files: It is same as a sequence file splittable, compressible and row oriented, but except the support of schema evolution and multilingual binding support.

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

We are using a precedence hierarchy for setting the properties

1. SET Command in HIVE
2. The command line –hiveconf option
3. Hive-site.XML
4. Hive-default.xml
5. Hadoop-site.xml
6. Hadoop-default.xml

**16. 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 overhead and in effect when executing queries like SELECT, FILTER, LIMIT, etc., it skips mapreduce function

**17. 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;

Need a reason to learn Apache Hadoop and Hive? Well, go through this blog post to find out why [Hadoop is the new black](https://intellipaat.com/blog/hadoop-is-the-new-black/).

**18. Explain the functionality of Object-Inspector.**

It helps to analyze the internal structure of row object and individual structure of columns in HIVE. It also provides a uniform way to access complex objects that can be stored in multiple formats in the memory.  
Instance of Java class  
A standard Java object  
A lazily initialized object  
The Object-Inspector tells structure of the object and also ways to access the internal fields inside the object.

**19. Whenever we run hive query, new metastore\_db is created. Why?**

Local metastore is created when we run Hive in embedded mode. And before creating it checks whether the metastore exists or not and this metastore property is defined in the configuration file hive-site.xml. Property is“javax.jdo.option.ConnectionURL” with default value “jdbc:derby:;databaseName=metastore\_db;create=true”.So to change the behavior of the location to an absolute path, so that from that location meta-store will be used.

**20. Differentiate between Hive and HBase**

|  |  |
| --- | --- |
| **Hive** | **HBase** |
| Enables most of the SQL queries | This doesn’t allow SQL queries |
| Doesn’t support record level insert, update, and delete operations on table | It supports |
| It is a data warehouse framework | It is NoSQL database |
| Hive run on the top of MapReduce | HBase runs on the top of HDFS |

**21. How can we access the sub directories recursively?**

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;

Hive tables can be pointed to the higher level directory and this is suitable for the directory structure which is like /data/country/state/city/

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

Hadoop developers consider the array as their inputs and convert them into a separate table row. To convert complicate data types into desired table formats Hive is essentially using explode.

**23. 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.

**24. How do we write our own custom SerDe?**

End users want to read their own data format instead of writing, so the user wants to write a Deserializer than SerDe.  
Example: The RegexDeserializer will deserialize the data using the configuration parameter ‘regex’, and a list of column names.  
If our SerDe supports DDL, we probably want to implement a protocol based on DynamicSerDe. It’s non-trivial to write a “thrift DDL” parser.

**25. Mention the date data type in Hive. Name the Hive data type collection.**

The TIMESTAMP data type stores date in java.sql.timestamp format.

**Three collection data types in Hive**

1. ARRAY
2. MAP
3. STRUCT

**26. Can we run UNIX shell commands from Hive? Can Hive queries be executed from script files? How? Give an example.**

Yes, we can run UNIX shell commands from Hive using the! Mark before the command .For example: !pwd at hive prompt will list the current directory.  
We can execute Hive queries from the script files by using the source command.  
Example −

Hive> source /path/to/file/file\_with\_query.hql

[**« Previous**](https://intellipaat.com/interview-question/pig-interview-questions/)

[**Next »**](https://intellipaat.com/interview-question/sqoop-interview-questions/)

Top of Form



Bottom of Form

#### RECOMMENDED COURSES

[](https://intellipaat.com/big-data-hadoop-training/)

**Self Paced**

[**LEARN MORE**](https://intellipaat.com/big-data-hadoop-training/)

Lifetime Access and 24/7 Support

**IBM Certification**

[**LEARN MORE**](https://intellipaat.com/big-data-hadoop-training/)

What IBM Certified Course contains? [**CLICK HERE**](https://intellipaat.com/interview-question/hive-interview-questions/)

[**Browse all courses**](https://intellipaat.com/all-courses/big-data/)

* 

**Paschal Ositadima**

Career-oriented Training.Thanks for making this platform available. I hope to bring more people from Nigeria to embrace intellipaat as the bridge to filling the gaps in their career needs.

* 

**John Chioles**

Good Experience.I will definitely recommend this training to others as well as plan to attend other courses from intellipaat.. gained good experience from intellipaat.

* 

**Rich Baker**

AMAZING Course!!This training was very informative and helped a lot in understanding Hadoop. You all do a fantastic job! Pls stay in touch.

#### INTERVIEW QUESTIONS

* [Top Apache Spark Interview Questions And Answers](https://intellipaat.com/interview-question/apache-spark-interview-questions/)
* [Top Mapreduce Interview Questions And Answers](https://intellipaat.com/interview-question/map-reduce-interview-questions/)
* [Top HDFS Interview Questions And Answers](https://intellipaat.com/interview-question/hdfs-interview-questions/)
* [Top PIG Interview Questions – Most Asked](https://intellipaat.com/interview-question/pig-interview-questions/)
* [Top Sqoop Interview Questions – Most Asked](https://intellipaat.com/interview-question/sqoop-interview-questions/)
* [Top Kafka Interview Questions – Most Asked](https://intellipaat.com/interview-question/kafka-interview-questions/)
* [Top Couchbase Interview Questions – Most Asked](https://intellipaat.com/interview-question/couchbase-interview-questions/)
* [Top DevOps Interview Questions – Most Asked](https://intellipaat.com/interview-question/devops-interview-questions/)

### "0 Responses on Top Hive Interview Questions – Most Asked"

### Leave a Message

You must be [logged in](https://intellipaat.com/login/?redirect_to=https%3A%2F%2Fintellipaat.com%2Finterview-question%2Fhive-interview-questions%2F) to post a comment.

**) Explain what is Hive?**

Hive is an ETL and Data warehousing tool developed on top of Hadoop Distributed File System (HDFS). It is a data warehouse framework for querying and analysis of data that is stored in HDFS. Hive is an open-source-software that lets programmers analyze large data sets on Hadoop.

**2) When to use Hive?**

* Hive is useful when making data warehouse applications
* When you are dealing with static data instead of dynamic data
* When application is on high latency (high response time)
* When a large data set is maintained
* When we are using queries instead of scripting

**3) Mention what are the different modes of Hive?**

Depending on the size of data nodes in Hadoop, Hive can operate in two modes.

These modes are,

* Local mode
* Map reduce mode

**4) Mention when to use Map reduce mode?**

Map reduce mode is used when,

* It will perform on large amount of data sets and query going to execute in a parallel way
* Hadoop has multiple data nodes, and data is distributed across different node we use Hive in this mode
* Processing large data sets with better performance needs to be achieved

**5) Mention key components of Hive Architecture?**

Key components of Hive Architecture includes,

* User Interface
* Compiler
* Metastore
* Driver
* Execute Engine

[](https://career.guru99.com/wp-content/uploads/2016/09/Apache_Hive_logo.svg_.png)

**6) Mention what are the different types of tables available in Hive?**

There are two types of tables available in Hive.

* **Managed table**: In managed table, both the data and schema are under control of Hive
* **External table**: In the external table, only the schema is under the control of Hive.

**7) Explain what is Metastore in Hive?**

Metastore is a central repository in Hive.  It is used for storing schema information or metadata in the external database.

**8) Mention what Hive is composed of ?**

Hive consists of 3 main parts,

1. Hive Clients
2. Hive Services
3. Hive Storage and Computing

**9) Mention what are the type of database does Hive support ?**

For single user metadata storage, Hive uses derby database and for multiple user Metadata or shared Metadata case Hive uses MYSQL.

**10) Mention Hive default read and write classes?**

Hive default read and write classes are

1. TextInputFormat/HiveIgnoreKeyTextOutputFormat
2. SequenceFileInputFormat/SequenceFileOutputFormat

**11) Mention what are the different modes of Hive?**

Different modes of Hive depends on the size of data nodes in Hadoop.

These modes are,

* Local mode
* Map reduce mode

**12) Why is Hive not suitable for OLTP systems?**

Hive is not suitable for OLTP systems because it does not provide insert and update function at the row level.

**13) Mention what is the difference between Hbase and Hive?**

Difference between Hbase and Hive is,

* Hive enables most of the SQL queries, but HBase does not allow SQL queries
* Hive does not support record level insert, update, and delete operations on table
* Hive is a data warehouse framework whereas HBase is NoSQL database
* Hive run on the top of MapReduce, HBase runs on the top of HDFS

**14) Explain what is a Hive variable? What for we use it?**

Hive variable is created in the Hive environment that can be referenced by Hive scripts. It is used to pass some values to the hive queries when the query starts executing.

**15) Mention what is ObjectInspector functionality in Hive?**

ObjectInspector functionality in Hive is used to analyze the internal structure of the columns, rows, and complex objects.  It allows to access the internal fields inside the objects.

**16) Mention what is (HS2) HiveServer2?**

It is a server interface that performs following functions.

* It allows remote clients to execute queries against Hive
* Retrieve the results of mentioned queries

Some advanced features Based on Thrift RPC in its latest version include

* Multi-client concurrency
* Authentication

**17) Mention what Hive query processor does?**

Hive query processor convert graph of MapReduce jobs with the execution time framework.  So that the jobs can be executed in the order of dependencies.

**18) Mention what are the components of a Hive query processor?**

The components of a Hive query processor include,

* Logical Plan Generation
* Physical Plan Generation
* Execution Engine
* Operators
* UDF’s and UDAF’s
* Optimizer
* Parser
* Semantic Analyzer
* Type Checking

**19) Mention what is Partitions in Hive?**

Hive organizes tables into partitions.

* It is one of the ways of dividing tables into different parts based on partition keys.
* Partition is helpful when the table has one or more Partition keys.
* Partition keys are basic elements for determining how the data is stored in the table.

**20) Mention when to choose “Internal Table” and “External Table” in Hive?**

In Hive you can choose internal table,

* If the processing data available in local file system
* If we want Hive to manage the complete lifecycle of data including the deletion

You can choose External table,

* If processing data available in HDFS
* Useful when the files are being used outside of Hive

**21) Mention if we can name view same as the name of a Hive table?**

No. The name of a view must be unique compared to all other tables and as views present in the same database.

**22) Mention what are views in Hive?**

In Hive, Views are Similar to tables. They are generated based on the requirements.

* We can save any result set data as a view in Hive
* Usage is similar to as views used in SQL
* All type of DML operations can be performed on a view

**23) 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.

**24) What is Buckets in Hive?**

* The data present in the partitions can be divided further into Buckets
* The division is performed based on Hash of particular columns that is selected in the table.

**25) In Hive, how can you enable buckets?**

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

set.hive.enforce.bucketing=true;

**26) In Hive, can you overwrite Hadoop MapReduce configuration in Hive?**

Yes, you can overwrite Hadoop MapReduce configuration in Hive.

**27) Explain how can you change a column data type in Hive?**

You can change a column data type in Hive by using command,

ALTER TABLE table\_name CHANGE column\_name column\_name new\_datatype;

**28) Mention what is the difference between order by and sort by in Hive?**

* SORT BY will sort the data within each reducer. You can use any number of reducers for SORT BY operation.
* ORDER BY will sort all of the data together, which has to pass through one reducer. Thus, ORDER BY in hive uses a single

**29) Explain when to use explode in Hive?**

Hadoop developers sometimes take an array as input and convert into a separate table row. To convert complex data types into desired table formats, Hive use explode.

**30) Mention how can you stop a partition form being queried?**

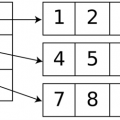
You can stop a partition form being queried by using the ENABLE OFFLINE clause with ALTER TABLE statement.

**Refer our**[**Hive Tutorials**](http://www.guru99.com/hive-tutorials.html?utm_source=crosslinking&utm_medium=referral&utm_campaign=click)**for an extra edge in your interview.**

[[Download PDF](https://career.guru99.com/top-30-hive-interview-questions/?format=pdf) Download PDF](https://career.guru99.com/top-30-hive-interview-questions/?format=pdf)

[Facebook](https://career.guru99.com/#facebook)[Twitter](https://career.guru99.com/#twitter)

### You Might Like:

[**[](https://career.guru99.com/technical-interview-questions/)Technical Interview Questions & Answers**](https://career.guru99.com/technical-interview-questions/) [**[](https://career.guru99.com/top-50-array-interview-questions-answers/)Top 50 Array Interview Questions & Answers**](https://career.guru99.com/top-50-array-interview-questions-answers/) [**[](https://career.guru99.com/top-15-social-media-manager-interview-questions/)Top 15 Social Media Interview Questions & Answers**](https://career.guru99.com/top-15-social-media-manager-interview-questions/) [**[](https://career.guru99.com/top-18-data-analyst-interview-questions/)Top 30 Data Analyst Interview Questions & Answers**](https://career.guru99.com/top-18-data-analyst-interview-questions/)

0REPLIES

### Leave a Comment

Top of Form

Name \*  Email \* 





Bottom of Form

### RECENT

* [https://career.guru99.com/wp-content/uploads/2018/06/process-36x36.png**Top 25 Microservices Interview Questions and Answers**](https://career.guru99.com/microservices-interview-questions-and-answer/)
* [https://career.guru99.com/wp-content/uploads/2018/05/Ansible_logo.svg_-36x36.png**Top 19 Ansible Interview Questions and Answers**](https://career.guru99.com/ansible-interview-questions-answers/)
* [https://career.guru99.com/wp-content/uploads/2018/05/SDLC_-_Software_Development_Life_Cycle-36x36.jpg**Top 27 SDLC Interview Questions and Answers**](https://career.guru99.com/sdlc-interview-questions-answers/)
* [https://career.guru99.com/wp-content/uploads/2018/05/Logo-talend-36x36.jpg**Top 30 Talend interview Questions & Answers**](https://career.guru99.com/talend-interview-questions-answers/)

© Copyright - Career Guru99 2019[google-play-store](https://play.google.com/store/apps/details?id=com.vector.guru99&hl=en)

* [Home](https://career.guru99.com/)
* [About](https://career.guru99.com/about-me/)
* [Privacy Policy](https://career.guru99.com/privacy-policy/)
* [Quiz](https://career.guru99.com/quiz/)
* [Jobs](https://career.guru99.com/jobs/)
* [Contact](https://career.guru99.com/contact/)

# **vv**Hive Interview Questions and Answers for 2018

26 Apr 2016

Last Update made on March 20, 2018

|  |  |
| --- | --- |
| Hive Interview Questions and Answers 2017 | Preparing for a Hadoop job interview then this list of most commonly asked Hive Interview questions and answers will help you ace your hadoop job interview.These Hive Interview questions and answers are formulated just to make candidates familiar with the nature of questions that are likely to be asked in a Hadoop job interview on the subject of Hive. |

## **Hadoop Hive Interview Questions and Answers**

**1) What is the difference between Pig and Hive ?**

|  |  |  |
| --- | --- | --- |
| ****Criteria**** | ****Pig**** | ****Hive**** |
| 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. |
| Language | It is a procedural data flow language. | Follows SQL Dialect and is a declarative language. |
| Purpose | Mainly used for programming. | It is mainly used for reporting. |
| General Usage | Usually used on the client side of the hadoop cluster. | Usually used on the server side of the hadoop cluster. |
| Coding Style | Verbose | More like SQL |
| **Pig vs Hive** | | |

For a detailed answer on the difference between Pig and Hive, refer this link -

https://www.dezyre.com/article/difference-between-pig-and-hive-the-two-key-components-of-hadoop-ecosystem/79

**2) What is the difference between HBase and Hive ?**

|  |  |
| --- | --- |
| ****HBase**** | ****Hive**** |
| HBase does not allow execution of SQL queries. | Hive allows execution of most SQL queries. |
| HBase runs on top of HDFS. | Hive runs on top of Hadoop MapReduce. |
| HBase is a NoSQL database. | Hive is a datawarehouse framework. |
| Supports record level insert, updated and delete operations. | Does not support record level insert, update and delete. |
| Hive vs HBase | |

[](https://goo.gl/iHkLeJ)

**2) I do not need the index created in the first question anymore. How can I delete the above index named index\_bonuspay?**

DROP INDEX index\_bonuspay ON employee;

### ****Test Your Practical Hadoop Knowledge****

**[](https://www.dezyre.com/big-data-hadoop-projects/finding-unique-urls-using-hadoop-hive?utm_source=DeZyre_Blog&utm_medium=QABanner_1&utm_campaign=blog_hiveinterviewquestions_246)**

**3) Can you list few commonly used Hive services?**

* Command Line Interface (cli)
* Hive Web Interface (hwi)
* HiveServer (hiveserver)
* Printing the contents of an RC file using the tool rcfilecat.
* Jar
* Metastore

For the complete list of big data companies and their salaries- [**CLICK HERE**](https://docs.google.com/forms/d/1LFuWEKQKCLR231qR9WE5PZakJj77fTDIW6ox5328HFM/viewform)

**4) Suppose that I want to monitor all the open and aborted transactions in the system along with the transaction id and the transaction state. Can this be achieved using Apache Hive?**

Hive 0.13.0 and above version support SHOW TRANSACTIONS command that helps administrators monitor various hive transactions.

### ****FREE eBook on 250 Hadoop Interview Questions and Answers****

**[](http://eepurl.com/crvfzH)**

**5) What is the use of Hcatalog?**

Hcatalog can be used to share data structures with external systems. Hcatalog provides access to hive metastore to users of other tools on Hadoop so that they can read and write data to hive’s data warehouse.

**6) Write a query to rename a table Student to Student\_New.**

Alter Table Student RENAME to Student\_New

Which companies use Hive extensively? This could be one of the possible Hive Interview Questions asked at your next Hadoop Job interview.



**7) Where is table data stored in Apache Hive by default?**

hdfs: //namenode\_server/user/hive/warehouse

[](https://www.dezyre.com/big-data-hadoop-projects/using-apache-hive-for-real-time-queries-and-analytics?utm_source=DeZyre_Blog&utm_medium=QABanner_2&utm_campaign=blog_hiveinterviewquestions_246)

**8) 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 technique, a partition is created for every unique 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.

[](https://www.dezyre.com/big-data-hadoop-projects/design-a-hadoop-architecture?utm_source=DeZyre_Blog&utm_medium=QABanner_3&utm_campaign=blog_hiveinterviewquestions_246)

**9) Explain about the different types of partitioning in Hive?**

Partitioning in Hive helps prune the data when executing the queries to speed up processing. Partitions are created when data is inserted into the table. In static partitions, the name of the partition is hardcoded into the insert statement whereas in a dynamic partition, Hive automatically identifies the partition based on the value of the partition field.

Based on how data is loaded into the table, requirements for data and the format in which data is produced at source- static or dynamic partition can be chosen. In dynamic partitions the complete data in the file is read and is partitioned through a MapReduce job based into the tables based on a particular field in the file. Dynamic partitions are usually helpful during ETL flows in the data pipeline.

When loading data from huge files, static partitions are preferred over dynamic partitions as they save time in loading data. The partition is added to the table and then the file is moved into the static partition. The partition column value can be obtained from the file name without having to read the complete file.

**10) When executing Hive queries in different directories, why is metastore\_db created in all places from where Hive is launched?**

When running Hive in embedded mode, it creates a local metastore. When you run the query, it first checks whether a metastore already exists or not. The property javax.jdo.option.ConnectionURL defined in the hive-site.xml has a default value jdbc: derby: databaseName=metastore\_db; create=true.

The value implies that embedded derby will be used as the Hive metastore and the location of the metastore is metastore\_db which will be created only if it does not exist already. The location metastore\_db is a relative location so when you run queries from different directories it gets created at all places from wherever you launch hive. This property can be altered in the hive-site.xml file to an absolute path so that it can be used from that particular location instead of creating multiple metastore\_db subdirectory multiple times.

[](https://goo.gl/UyLCXN)

**11) How will you read and write HDFS files in Hive?**

i) TextInputFormat- This class is used to read data in plain text file format.

ii) HiveIgnoreKeyTextOutputFormat- This class is used to write data in plain text file format.

iii) SequenceFileInputFormat- This class is used to read data in hadoop SequenceFile format.

iv) SequenceFileOutputFormat- This class is used to write data in hadoop SequenceFile format.

**12) What are the components of a Hive query processor?**

Query processor in Apache Hive converts the SQL to a graph of MapReduce jobs with the execution time framework so that the jobs can be executed in the order of dependencies. The various components of a query processor are-

* Parser
* Semantic Analyser
* Type Checking
* Logical Plan Generation
* Optimizer
* Physical Plan Generation
* Execution Engine
* Operators
* UDF’s and UDAF’s.

**13) Differentiate between describe and describe extended.**

Describe database/schema- This query displays the name of the database, the root location on the file system and comments if any.

Describe extended database/schema- Gives the details of the database or schema in a detailed manner.

**14) Is it possible to overwrite Hadoop MapReduce configuration in Hive?**

Yes, hadoop MapReduce configuration can be overwritten by changing the hive conf settings file.

**15) I want to see the present working directory in UNIX from hive. Is it possible to run this command from hive?**

Hive allows execution of UNIX commands with the use of exclamatory (!) symbol. Just use the ! Symbol before the command to be executed at the hive prompt. To see the present working directory in UNIX from hive run !pwd at the hive prompt.

**16)  What is the use of explode in Hive?**

Explode in Hive is used to convert complex data types into desired table formats. explode UDTF basically emits all the elements in an array into multiple rows.

**17) 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.

**18) Difference between HBase and Hive.**

* HBase is a NoSQL database whereas Hive is a data warehouse framework to process Hadoop jobs.
* HBase runs on top of HDFS whereas Hive runs on top of Hadoop MapReduce.

**19) Write a hive query to view all the databases whose name begins with “db”**

SHOW DATABASES LIKE ‘db.\*’

**20) How can you prevent a large job from running for a long time?**

This can be achieved by setting the MapReduce jobs to execute in strict mode set hive.mapred.mode=strict;

The strict mode ensures that the queries on partitioned tables cannot execute without defining a WHERE clause.

What do u think is more popular among the developers – Pig or Hive?



**21) What is a Hive Metastore?**

Hive Metastore is a central repository that stores metadata in external database.

**22) Are multiline comments supported in Hive?**

No

**23) What is ObjectInspector functionality?**

ObjectInspector is used to analyse the structure of individual columns and the internal structure of the row objects. ObjectInspector in Hive provides access to complex objects which can be stored in multiple formats.

**24) Explain about the different types of join in Hive.**

HiveQL has 4 different types of joins –

JOIN- Similar to Outer Join in SQL

FULL OUTER JOIN – Combines the records of both the left and right outer tables that fulfil the join condition.

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.

**25) How can you configure remote metastore mode in Hive?**

To configure metastore in Hive, hive-site.xml file has to be configured with the below property –

 hive.metastore.uris

   thrift: //node1 (or IP Address):9083

   IP address and port of the metastore host

**26) Is it possible to change the default location of Managed Tables in Hive, if so how?**

Yes, we can change the default location of Managed tables using the LOCATION keyword while creating the managed table. The user has to specify the storage path of the managed table as the value to the LOCATION keyword.

**27) How data transfer happens from HDFS to Hive?**

If data is already present in HDFS then the user need not LOAD DATA that moves the files to the /user/hive/warehouse/. So the user just has to define the table using the keyword external that creates the table definition in the hive metastore.

Create external table table\_name (

  id int,

  myfields string

)

location '/my/location/in/hdfs';

**28) In case of embedded Hive, can the same metastore be used by multiple users?**

We cannot use metastore in sharing mode. It is suggested to use standalone real database like PostGreSQL and MySQL.

**29)  The partition of hive table has been modified to point to a new directory location. Do I have to move the data to the new location or the data will be moved automatically to the new location?**

Changing the point of partition will not move the data to the new location. It has to be moved manually to the new location from the old one.

**30)  What will be the output of cast (‘XYZ’ as INT)?**

It will return a NULL value.

[**Master Hadoop by working on interesting Hadoop Hive Real-Time Projects**](https://www.dezyre.com/projects/big-data-projects/apache-hive-projects%20www.dezyre.com/projects/big-data-projects/apache-hadoop-projects?utm_source=DeZyreBlog&utm_medium=TextLink&utm_campaign=Blog_HiveInte246)

**31) What are the different components of a Hive architecture?**

Hive Architecture consists of a –

* User Interface – UI component of the Hive architecture calls the execute interface to the driver.
* Driver create a session handle to the query and sends the query to the compiler to generate an execution plan for it.
* Metastore - Sends the metadata to the compiler for the execution of the query on receiving the sendMetaData request.
* Compiler- Compiler generates the execution plan which is a DAG of stages where each stage is either a metadata operation, a map or reduce job or an operation on HDFS.
* Execute Engine- Execution engine is responsible for submitting each of these stages to the relevant components by managing the dependencies between the various stages in the execution plan generated by the compiler.

**32) What happens on executing the below query? After executing the below query, if you modify   the column –how will the changes be tracked?**

Hive> CREATE INDEX index\_bonuspay ON TABLE employee (bonus)

AS 'org.apache.hadoop.hive.ql.index.compact.CompactIndexHandler';

The query creates an index named index\_bonuspay which points to the bonus column in the employee table. Whenever the value of bonus is modified it will be stored using an index value.

**33) What is the default database provided by Hive for Metastore ?**

Derby is the default database.

**34) 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.

### ****Scenario based or Real-Time Interview Questions on Hadoop Hive****

1. How will you optimize Hive performance?

There are various ways to run Hive queries faster -

* Using Apache Tez execution engine
* Using vectorization
* Using ORCFILE
* Do cost based query optimization.

1. Will the reducer work or not if you use “Limit 1” in any HiveQL query?
2. Why you should choose Hive instead of Hadoop MapReduce?
3. I create a table which contains transaction details of customers for the year 2018.   
   CREATE TABLE transaction\_details (cust\_id INT, amount FLOAT, month STRING, country STRING) ROW FORMAT DELIMITED FIELDS TERMINATED BY ‘,’ ;  
   I have inserted 60K tuples in this table and now want to know the total revenue that has been generated for each month. However, Hive takes too much time to process this query. List all the steps that you would follow to solve this problem.
4. 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 queries run against hive or you should look into hive query optimization techniques ? Which one is performance efficient ?

### ****Other Interview Questions on Hadoop Hive****

1. Explain the difference between SQL and Apache Hive.
2. Why mapreduce will not run if you run select \* from table in hive?

We hope that these Hive Interview questions and answers have pre-charged you for your next Hadoop interview on the subject of Hive. Let us know about your experience on Hive interview questions in Hadoop interviews in the comments below.

#### **Posts Related to Hadoop Interview Questions-**

1 What is Hive ?

Answer: Hive is a data warehouse software which is used for facilitates querying and managing large data sets residing in distributed storage.Hive language almost look like SQL language called HiveQL.Hive also allows traditional map reduce programs to customize mappers and reducers when it is inconvenient or inefficient to execute the logic in HiveQL (User Defined Functions UDFS).

Q2 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.

Q3 What is the difference between HBase and Hive?

Answer: Both hive and hbase can be used in different technologies that are based on Hadoop. Hive happens to be a infrastructure warehouse of data which is used on Hadoop whereas HBase is NoSQL. The key value stores which run on Hadoop themselves. Hive will also help those who know about SQL run a few jobs in MapReduce when Hbase will also support 4 of the operations such as put, get, scan and delete. The Hbase happens to be good for querying for data but Hive on the other hand is good for querying data is analytical and is collected over a while.

Q5 What is Hive Metastore ?

Answer: Hive Meta store is a database that stores metadata of your hive tables like table name,column name,data types,table location,number of buckets in the table etc.

Q6 Hive new version supported Hadoop Versions ?

Answer: This release works with Hadoop 0.20.x, 0.23.x.y, 1.x.y, 2.x.y

Q7 Which companies are mostly using Hive ?

Answer: Facebook,Netflix

Q8 Wherever (Different Directory) I run hive query, it creates new metastore\_db, please explain the reason for it?

Answer: Whenever you run the hive in embedded mode, it creates the local metastore. And before creating the metastore it looks whether metastore already exist or not. This property is defined in configuration file hive – site.xml. Property is “javax.jdo.option.ConnectionURL” with default value “jdbc:derby:;databaseName=metastore\_db;create=true”. So to change the behavior change the location to absolute path, so metastore will be used from that location.

Q9 Is it possible to use same metastore by multiple users, in case of embedded hive?

Answer: No, it is not possible to use metastore in sharing mode. It is recommended to use standalone “real” database like MySQL or PostGresSQL.

Q10 What is the functionality of Query Processor in Apached Hive ?

Answer: This component implements the processing framework for converting SQL to a graph of map/reduce jobs and the execution time framework to run those jobs in the order of dependencies.

Q11 Is multi line comment supported in HIVE Script?

Answer: NO

Q12 What is the functionality of Query Processor in Apache Hive?

Answer: This components implements the processing framework for converting SQL to graph of map/reduce jobs and the execution time framework to run those jobs in the order od dependencies.

Q13 what is a Hive Metastore?

Answer: Hive Metastore is a central repository that stores metadata in external database.

Q14 Explain about the SMB Join in Hive.

Answer: In SMB join in Hive, each mapper reads a bucket from the first table and the corresponding bucket from the second table and then a merge sort join is performed. Sort Merge Bucket (SMB) join in hive is mainly used as there is no limit on file or partition or table join. SMB join can best be used when the tables are large. In SMB join the columns are bucketed and sorted using the join columns. All tables should have the same number of buckets in SMB join.

Q15 What is ObjectInspector functionality?

Answer: ObjectInspector is used to analyze the structure of individual columns and the internal structure of the row objects. ObjectInspector in Hive provides access to complex objects which can be stored in multiple formats.

Q16 Is it possible to use same metastore by multiple users, in case of embedded hive?

Answer: No, it is not possible to use metastore in sharing mode. It is recomended to use standalone “real” database like MySQL or PostGreSQL.

Q17 Explain about the different types of join in Hive.

Answer: HiveQL has 4 different types of joins – JOIN- Similar to Outer Join in SQL

1. FULL OUTER JOIN – Combines the records of both the left and right outer tables that fulfil the join condition.
2. LEFT OUTER JOIN- All the rows from the left table are returned even if there are no matches in the right table.
3. RIGHT OUTER JOIN-All the rows from the right table are returned even if there are no matches in the left table.

Q18 Is it possible to change the default location of Managed Tables in Hive, if so how?

Answer: Yes, we can change the default location of Managed tables using the LOCATION keyword while creating the managed table. The user has to specify the storage path of the managed table as the value to the LOCATION keyword.

Q19 How can you connect an application, if you run Hive as a server?

Answer: When running Hive as a server, the application can be connected in one of the 3 ways-

1. ODBC Driver-This supports the ODBC protocol
2. JDBC Driver- This supports the JDBC protocol
3. Thrift Client- This client can be used to make calls to all hive commands using different programming language like PHP, Python, Java, C++ and Ruby.

Q20 Which classes are used by the Hive to Read and Write HDFS Files

Answer: Following classes are used by Hive to read and write HDFS files:

* TextInputFormat/HiveIgnoreKeyTextOutputFormat: These 2 classes read/write data in plain text file format.
* SequenceFileInputFormat/SequenceFileOutputFormat: These 2 classes read/write data in hadoop SequenceFile format.

Q21 What are the types of tables in Hive?

Answer: There are two types of tables.

1. Managed tables.
2. External tables.

Only the drop table command differentiates managed and external tables. Otherwise, both type of tables are very similar.

Q22 Is it possible to create multiple table in hive for same data?

Answer: 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).

Q23 What kind of datawarehouse application is suitable for Hive?

Answer: Hive is not a full database. The design constraints and limitations of Hadoop and HDFS impose limits on what Hive can do. Hive is most suited for data warehouse applications, where

1) Relatively static data is analyzed,

2) Fast response times are not required, and

3) When the data is not changing rapidly.

Hive doesn’t provide crucial features required for OLTP, Online Transaction Processing. It’s closer to being an OLAP tool, Online Analytic Processing.So, Hive is best suited for data warehouse applications, where a large data set is maintained and mined for insights, reports, etc.

Q24 What is the maximum size of string data type supported by Hive?

Answer: Maximum size is 2 GB.

Q25 What are the Binary Storage formats supported in Hive?

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

Sequence Files, Avro Data files, RCFiles, ORC files, Parquet files

* Sequence files: General binary format. splittable, compressible and row oriented. a typical example can be. if we have lots of small file, we may use sequence 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 Sequence 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 subsequently.
* ORC Files: Optimized Record Columnar files.

Q26 CONCAT function in Hive with Example?

Answer: CONCAT function will concat the input strings. You can specify any number of strings separated by comma.

Q27 is HQL case sensitive?

Answer: HQL is not case sensitive.

Q28 REPEAT function in Hive with example?

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

Q29 Describe REVERSE function in Hive with example?

Answer: REVERSE function will reverse the characters in a string.

Q30 LOWER or LCASE function in Hive with example?

Answer: LOWER or LCASE function will convert the input string to lower case characters.

Q31 UPPER or UCASE function in Hive with example?

Answer: UPPER or UCASE function will convert the input string to upper case characters.

Q33 Rename a table in Hive – How to do it?

Answer: Using ALTER command, we can rename a table in Hive.

ALTER TABLE hive\_table\_name RENAME  TO new\_name;

Q34 Difference between order by and sort by in hive?

Answer: SORT BY will sort the data within each reducer. You can use any number of reducers for SORT BY operation.

ORDER BY will sort all of the data together, which has to pass through one reducer. Thus, ORDER BY in hive uses single reducer.

ORDER BY guarantees total order in the output while SORT BY only guarantees ordering of the rows within a reducer. If there is more than one reducer, SORT BY may give partially ordered final results.

Q35 RLIKE in Hive?

Answer: RLIKE (Right-Like) is a special function in Hive where if any substring of A matches with B then it evaluates to true. It also obeys Java regular expression pattern. Users don’t need to put % symbol for a simple match in RLIKE.

Q36 Difference between external table and internal table in HIVE ?

Answer: Hive has a relational database on the master node it uses to keep track of state. For instance, when you CREATE TABLE FOO(foo string) LOCATION ‘hdfs://tmp/’;, this table schema is stored in the database. If you have a partitioned table, the partitions are stored in the database(this allows hive to use lists of partitions without going to the filesystem and finding them, etc). These sorts of things are the ‘metadata’.

When you drop an internal table, it drops the data, and it also drops the metadata. When you drop an external table, it only drops the meta data. That means hive is ignorant of that data now. It does not touch the data itself.

Q37 Does Hive support record level Insert, delete or update?

Answer: Hive does not provide record-level update, insert, or delete. Henceforth, Hive does not provide transactions too. However, users can go with CASE statements and built in functions of Hive to satisfy the above DML operations. Thus, a complex update query in a RDBMS may need many lines of code in Hive.

Q38 Is Hive suitable to be used for OLTP systems? Why?

Answer: No Hive does not provide insert and update at row level. So it is not suitable for OLTP system.

Q39 What kind of datawarehouse application is suitable for Hive?

Answer: Hive is not a full database. The design constraints and limitations of Hadoop and HDFS impose limits on what Hive can do.

Hive is most suited for data warehouse applications, where

* Relatively static data is analyzed,
* Fast response times are not required, and
* When the data is not changing rapidly.

Hive doesn’t provide crucial features required for OLTP, Online Transaction Processing. It’s closer to being an OLAP tool, Online Analytic Processing.So, Hive is best suited for data warehouse applications, where a large data set is maintained and mined for insights, reports, etc.

Q40 Can we change the data type of a column in a hive table?

Answer: Using REPLACE column option

ALTER TABLE table\_name REPLACE COLUMNS ……

Q41 TRIM function in Hive with example?

Answer: TRIM function will remove the spaces associated with a string. xample:

TRIM(‘  Hadoop  ‘);

Output: Hadoop.

Q42 Why do we need Hive?

Answer: Hive is a tool in Hadoop ecosystem which provides an interface to organize and query data in a databse like fashion and write SQL like queries. It is suitable for accessing and analyzing data in Hadoop using SQL syntax.

Q43 Is there a date data type in Hive?

Answer: Yes. The TIMESTAMP data types stores date in java.sql.timestamp format

Q44 What is a Hive variable? What for we use it?

Answer: The hive variable is variable created in the Hive environment that can be referenced by Hive scripts. It is used to pass some values to the hive queries when the query starts executing.

Q45 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 ?

Answer: By Omitting the LOCAL CLAUSE in the LOAD DATA statement.

Q46 What does the “USE” command in hive do?

Answer: With the use command you fix the database on which all the subsequent hive queries will run.

Q47 How can you delete the DBPROPERTY in Hive?

Answer: There is no way you can delete the DBPROPERTY.

Q48 Does the archiving of Hive tables give any space saving in HDFS?

Answer: No. It only reduces the number of files which becomes easier for namenode to manage.

Q49 What is the usefulness of the DISTRIBUTED BY clause in Hive?

Answer: It controls ho wthe map output is reduced among the reducers. It is useful in case of streaming data.

Q50 Can a partition be archived? What are the advantages and Disadvantages?

Answer: Yes. A partition can be archived. Advantage is it decreases the number of files stored in namenode and the archived file can be queried using hive. The disadvantage is it will cause less efficient query and does not offer any space savings.

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

### Part 1 – Hive Interview Questions (Basic)

This first part covers basic Hive Interview Questions and Answers.

#### 1. List out the different components of Hive architecture?

**Answer:**  
There are five core components in Hive architecture are listed below:  
**•User Interface (UI):** It acts as a communicator between users and drivers when the user writes the queries the UI accepts it and runs it on the driver, there are two types of interface available they are Command line and GUI interface.  
**•Driver:** It maintains the life cycle of the HiveQL query. It receives the queries from the user interface and creates the session to process the query.  
**•Compiler:** It receives the query plans from the driver and gets the required information from Metastore in order to execute the plan.  
**•Metastore:** It stores the information about the data like a table; it can be of an internal or external table. It sends the metadata information to the compiler to execute the query.  
•Execute Engine: Hive service will execute the result in execution engine; it executes the query in MapReduce to process the data. It is responsible for controlling each stage for all these components.

#### 2. Which are the different types of modes that Hive can operate?

**Answer:**  
This is the common Hive Interview Questions asked in an interview. Hive can operate on two modes based on the size of data,  
These modes are:  
•Map reduce Mode  
•Local Mode

 Popular Course in this category

**Data Scientist Course**43 Online Courses | 170+ Hours | Verifiable Certificate of Completion | Lifetime Access   
4.8 (1,145 ratings)

Course Price   
₹16499 ~~₹39999~~   
[**View Course**](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner1)

Related Courses

[SAS Training Course](https://www.educba.com/data-science/courses/sas-training-course/?btnz=edu-blg-inline-banner1)[Machine Learning Course](https://www.educba.com/data-science/courses/machine-learning-course/?btnz=edu-blg-inline-banner1)[AWS Training Course](https://www.educba.com/data-science/courses/aws-training-course/?btnz=edu-blg-inline-banner1)

#### 3. Which are the scenarios where Hive can be used and cannot be used?

**Answer**:  
When you’re creating Data warehouse applications when your data is [Static](https://www.educba.com/course/static-and-dynamic-routing/) when your application does not need high response time, when the data volume is huge, when the data is not changing rapidly and when you are using queries instead of scripting. [Hive supports only OLAP transaction](https://www.educba.com/install-hive/) it is not suitable for [OLTP](https://www.educba.com/what-is-oltp/)transactions.

Let us move to the next Hive Interview Questions.

#### 4. What are the file formats that Hive supports? List the type of applications that are supported by HIVE?

**Answer:**  
By default,[Hive supports Text File format](https://www.educba.com/what-is-a-hive/) and it also supports the binary file format such as Sequence file, ORC files, Parquet files, Avro Data files.  
**•Sequence file:** It is generally a binary format file, which can be compressed and is splittable.  
**•ORC file:** Optimized Row Columnar file is recorded column-based file and column-oriented storage file.  
**•Parquet file:** It is a column-oriented binary file it is highly efficient for large-scale queries.  
**•Avro Data file:** It is same as sequence file format which is a splittable, compressible and row-oriented file.  
The maximum size of string data type allowed in Hive is 2 GB.

Hive is a data warehouse framework that is suitable for those applications that are written in [Java](https://www.educba.com/course/java-8-tutorials/), [C++](https://www.educba.com/c-programming-language-basics/), [PHP](https://www.educba.com/course/php-mysql-for-beginners/), [Python](https://www.educba.com/course/advanced-python-iot-iot-based-data-analysis/) or [Ruby](https://www.educba.com/course/ruby-programming-training/).

#### 5. What are the different types of table that are available in Hive?

**Answer:**  
There are two types of a table in Hive application, they are:  
**•Managed Tables:** The data and [schema](https://www.educba.com/course/xml-4/) are in control of Hive.  
**•External Tables:** Only the schema is in control of the Hive.

### Part 2 – Hive Interview Questions (Advanced)

Let us now have a look at the advanced Hive Interview Questions.

#### 6. What is a Metastore in Hive? List and explain the different types of Hive Metastores configuration?

**Answer:**  
Metastore in [Hive is used to store the metadata information](https://www.educba.com/install-hive/), it is a central repository in Hive. It allows storing the metadata information in an external database. By default, Hive stores Metadata information in Derby database but it can also be stored in other databases such as [Oracle](https://www.educba.com/course/oracle-application-express-online-training/), [MySql](https://www.educba.com/course/mysql-training/) etc.  
There are three types of Metastore configuration, they are:  
**•Embedded metastore:** It is a default mode; it can locally access the Hive library, all the command line operations are done in embedded mode. The Hive service, the metastore service, and the database run in same JVM.  
**•Local metastore:** It stores data in an external database such as MySql or Oracle. The Hive service and metastore service runs in same JVM, it connects to the database that is running in separate JVM.  
**•Remote metastore:** It uses the remote mode to run queries, here the metastore service and hive service runs in a separate JVM. You can have multiple metastore servers to increase the availability.

#### 7. What is a Hive Query Processor? What are the different components of the Hive Query Processor?

**Answer:**  
This is the frequently asked Hive Interview Questions in an interview. Hive Query Processor is used to convert SQL to MapReduce jobs. Based on the order of dependencies the jobs are executed.  
The components of Hive Query Processor are listed below:  
•Semantic Analyser  
•UDF’s and UDAF’s  
•Optimizer  
•Operator  
•Parser  
•Execution Engine  
•Type Checking  
•Logical Plan Generation  
•Physical Plan Generation

#### 8. What is the functionality of Object-Inspector in Hive?

**Answer:**  
It is composed of Hive that is used to identify the structure of the individual columns and internal structure of row objects. The complex objects that are stored in multiple formats can be accessed using Object-Inspector in Hive.  
Object-Inspector will identify the structure of an object and ways to access the internal fields inside the object.

Let us move to the next Hive Interview Questions.

#### 9. What are the different ways to connect the applications to Hive Server?

**Answer:**  
There are three ways to connect the applications to the Hive server, they are:  
**•Thrift Client:** This is used to run all the hive commands using a different [programming language](https://www.educba.com/course/using-programming-languages-ubuntu/) such as [Java](https://www.educba.com/course/java/), [C++](https://www.educba.com/c-programming-language-basics/), [PHP](https://www.educba.com/course/php-database-basic-tutorials/), [Python](https://www.educba.com/course/web-application-development-pyramid-micro-framework-python/) or [Ruby](https://www.educba.com/course/ruby-rails-training/).  
**•ODBC Driver:** This will support the ODBC protocol  
**•JDBC Driver:** This will support the JDBC protocol

#### 10. What is the default read and write classes in Hive?

**Answer:**  
Below is the read and write classes available in Hive:  
•TextInputFormat – This class is used to read data in plain text format.  
•HiveIgnoreKeyTextOutputFormat – This class is used to write data in plain text format.  
•SequenceFileInputFormat – This class is used to read data in Hadoop Sequence file format.  
•SequenceFileOutputFormat – This class is used to write data in Hadoop Sequence file format.

## Top 15 Hive Interview Questions

### 1. What is Hive?

Hive is a data warehousing tool. It is an abstraction and it gives SQL queries to perform an analysis. It gives you logical abstraction over the databases and the tables but it is not a database.

### 2. What is Hive a metastore?

Hive contains two things: data and the metadata. The metadata contains the (column names, partitions information, bucketing information, SerDe etc.) i.e., the data about the actual table this is by default stored in the Derby database, we can also configure it to Oracle or MySQL database.

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

With derby database, you cannot have multiple connections or multiple sessions instantiated 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.

### 4. What are managed and external tables?

We have got two things, one of which is data present in the HDFS and the other is the metadata, present in some database.

There are two categories of Hive tables i.e., Managed and External Tables.

In the Managed tables, both the data and the metadata are managed by Hive and if you drop the managed table, both data and metadata are deleted.

There are some situations where your data will be controlled by some other application and you want to read that data but you must allow Hive to delete that data.

In such case, you can create an external table in Hive. In the external table, metadata is controlled by Hive but the actual data will be controlled by some other application. So, when you delete a table accidentally, only the metadata will be lost and the actual data will reside wherever it is.

### 5. What are the complex data types in Hive?

* Map – The**Map**contains a key-value pair where you can search for a value using the key.
* Struct – A**Struct**is a collection of elements of different data types. For example, if you take the address, it can have different data types. For example, pin code will be in Integer format.
* Array – An**Array**will have a collection of homogeneous elements. For example, if you take your skillset, you can have N number of skills
* Uniontype – It represents a column which can have a value that can belong to any of the data types of your choice.

### 6. How does partitioning help in the faster execution of queries?

With the help of partitioning, a subdirectory will be created with the name of the partitioned column and when you perform a query using the WHERE clause, only the particular sub-directory will be scanned instead of scanning the whole table. This gives you faster execution of queries.

### 7. How to enable dynamic partitioning in Hive?

Related to partitioning there are two types of partitioning Static and Dynamic. In the static partitioning, you will specify the partition column while loading the data.

Whereas in dynamic partitioning, you push the data into Hive and then Hive decides which value should go into which partition. To enable dynamic partitioning, you have set the below property  
set hive.exec.dynamic.parition.mode = nonstrict;

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

### 8. How does bucketing help in the faster execution of queries?

If you have to join two large tables, you can go for reduce side join. But if both the tables have the same number of buckets or same multiples of buckets and also sorted on the same column there is a possibility of SMBMJ in which all the joins take place in the map phase itself by matching the corresponding buckets.

Buckets are basically files that are created inside the HDFS directory.

There are different properties which you need to set for bucket map joins and they are as follows:

* set hive.enforce.sortmergebucketmapjoin = false;
* set hive.auto.convert.sortmerge.join =  false;
* set hive.optimize.bucketmapjoin =  true;
* set hive.optimize.bucketmapjoin.sortedmerge = true;

### 9. How to enable bucketing in Hive?

By default bucketing is disabled in Hive, you can enforce to enable it by setting the below property  
set hive.enforce.bucketing  = true;

### 10. 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.

### 11. What are the different file formats in Hive?

There are different file formats supported by Hive

* Text File format
* Sequence File format
* RC file format
* Parquet
* Avro
* ORC

Every file format has its own characteristics and Hive allows you to choose easily the file format which you wanted to use.

### 12. 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.

### 13. 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’;

### 14. 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 querying also, it queries the particular column instead of querying 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 query, it will skip the blocks based on the indexing.

### 15. How to access HBase tables from Hive?

Using Hive-HBase storage handler, you can access the HBase tables from Hive and once you are connected, you can query HBase using the SQL queries from Hive. You can also join multiple tables in HBase from Hive and retrieve the result.

Along with the above Hive interview questions, also check our [Hbase tutorial](https://acadgild.com/blog/hbase-tutorial-beginners-guide) . These 2 articles would be helpful to attend a Hadoop interview. Enroll for [Big Data and Hadoop Training](https://acadgild.com/big-data/big-data-development-training-certification) with ACADGILD and become a successful Hadoop developer.

###### 1. 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)

###### 2. How do change settings within Hive Session?

We can change settings from within a session, too, using the SET command. This is useful for changing Hive or MapReduce job settings for a particular query. For example, the following command ensures buckets are populated according to the table definition.



|  |  |
| --- | --- |
| 1  2 | hive> SET hive.enforce.bucketing=true; |

To see the current value of any property, use SET with just the property name:



|  |  |
| --- | --- |
| 1  2  3 | hive> SET hive.enforce.bucketing;  hive.enforce.bucketing=true |

By itself, **SET** will list all the properties and their values set by Hive. This list will not include Hadoop defaults, unless they have been explicitly overridden in one of the ways covered in the above answer. Use **SET -v** to list all the properties in the system, including Hadoop defaults.

###### 3. How to print header on Hive query results?

We need to use following set command before our query to show column headers in STDOUT.



|  |  |
| --- | --- |
| 1  2 | hive> set hive.cli.print.header=true; |

###### 4. How to get detailed description of a table in Hive?

Use below hive command to get a detailed description of a hive table.



|  |  |
| --- | --- |
| 1  2 | hive> describe extended <tablename>; |

###### 5. How to access sub directories recursively in Hive queries?

To process directories recursively in Hive, we need to set below two commands in hive session. These two parameters work in conjunction.



|  |  |
| --- | --- |
| 1  2  3 | hive> Set mapred.input.dir.recursive=true;  hive> Set hive.mapred.supports.subdirectories=true; |

Now hive tables can be pointed to the higher level directory. This is suitable for a scenario where the directory structure is as following: */data/country/state/city*

###### 6. How to skip header rows from a table in Hive?

Suppose while processing some log files, we may find header records.

System=….  
Version=…  
Sub-version=….

Like above, It may have 3 lines of headers that we do not want to include in our Hive query. To skip header lines from our tables in Hive we can set a table property that will allow us to skip the header lines.



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | CREATE EXTERNAL TABLE userdata (  name STRING,  job STRING,  dob STRING,  id INT,  salary INT)  ROW FORMAT DELIMITED FIELDS TERMINATED BY ‘ ‘ STORED AS TEXTFILE  LOCATION ‘/user/data’  TBLPROPERTIES("skip.header.line.count"="3”); |

###### 7. 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 querying, if we specify any column other than above list, will result in NULL values.

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

Maximum size is 2 GB.

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

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

Sequence Files, Avro Data files, RCFiles, ORC files, Parquet files

**Sequence files:** General binary format. splittable, compressible and row oriented. a typical example can be. if we have lots of small file, we may use sequence 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 Sequence 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 subsequently.

**ORC Files:** Optimized Record Columnar files

###### 10. is HQL case sensitive?

HQL is not case sensitive.

###### 11. Describe CONCAT function in Hive with Example?

CONCAT function will concatenate the input strings. We can specify any number of strings separated by comma.

**Example:** CONCAT (‘Hive’,’-‘,’is’,’-‘,’a’,’-‘,’data warehouse’,’-‘,’in Hadoop’);  
**Output:** Hive-is-a-data warehouse-in Hadoop

So, every time we delimit the strings by ‘-‘. If it is common for all the strings, then Hive provides another command **CONCAT\_WS**. Here you have to specify the delimit operator first.

**Syntax:** CONCAT\_WS (‘-‘,’Hive’,’is’,’a’,’data warehouse’,’in Hadoop’);  
**Output:** Hive-is-a-data warehouse-in Hadoop

###### 12. 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.

###### 13. Describe REVERSE function in Hive with example?

REVERSE function will reverse the characters in a string.

**Example:** REVERSE(‘Hive’);  
**Output:** eviH

###### 14. Describe TRIM function in Hive with example?

TRIM function will remove the spaces associated with a string.

**Example:** TRIM(‘ Hadoop ‘);  
**Output:** Hadoop.

If we want to remove only leading or trailing spaces then we can specify the below commands respectively.

LTRIM(‘ Hadoop’);  
RTRIM(‘Hadoop ‘);

###### 15. Describe RLIKE in Hive with an example?

RLIKE (Right-Like) is a special function in Hive where if any substring of A matches with B then it evaluates to true. It also obeys Java regular expression pattern. Users don’t need to put % symbol for a simple match in RLIKE.

Examples: **‘Express’ RLIKE ‘Exp’ –> True**  
**‘Express’ RLIKE ‘^E.\*’ –> True** (Regular expression)

Moreover, RLIKE will come handy when the string has some spaces. Without using TRIM function, RLIKE satisfies the required scenario. Suppose if A has value ‘Express  ‘ (2 spaces additionally) and B has value ‘Express’. In these situations, RLIKE will work better without using TRIM.

**‘Express  ‘ RLIKE ‘Express’ –> True**

**Note:**RLIKE evaluates to NULL if A or B is NULL.

### Share this:

**

### *About Siva*

*Senior Hadoop developer with 4 years of experience in designing and architecture solutions for the Big Data domain and has been involved with several complex engagements. Technical strengths include Hadoop, YARN, Mapreduce, Hive, Sqoop, Flume, Pig, HBase, Phoenix, Oozie, Falcon, Kafka, Storm, Spark, MySQL and Java.*

[*View all posts by Siva →*](http://hadooptutorial.info/author/siv535gmail-com/)

### Leave a comment

Top of Form

Your email address will not be published. Required fields are marked \*

VisualText

File

Edit

View

Insert

Format

Tools

Table

Name \*

Email \*

Website



Bottom of Form

## 5 thoughts on “Hive Interview Questions and Answers for experienced Part – 4”

* 

[Reply ↓](http://hadooptutorial.info/hive-interview-questions-and-answers-part-4/#comment-1906)

manjula[*August 26, 2015 at 6:58 pm*](http://hadooptutorial.info/hive-interview-questions-and-answers-part-4/#comment-1906)

hi please share me hive questions and ansewrs

* 

[Reply ↓](http://hadooptutorial.info/hive-interview-questions-and-answers-part-4/#comment-3707)

sriniwaas[*December 7, 2015 at 4:33 am*](http://hadooptutorial.info/hive-interview-questions-and-answers-part-4/#comment-3707)

is it possible to create multiple table in hive for same data.

this question not understand. can you please explain first yes or no.

please let me know first wither we  create multiple table in hive for same data or not?

thanks in advance.

* + 

[Reply ↓](http://hadooptutorial.info/hive-interview-questions-and-answers-part-4/#comment-3727)

[Siva](http://hadooptutorial.info/members/siv535gmail-com/) Post author[*December 8, 2015 at 1:33 pm*](http://hadooptutorial.info/hive-interview-questions-and-answers-part-4/#comment-3727)

yes

* 

[Reply ↓](http://hadooptutorial.info/hive-interview-questions-and-answers-part-4/#comment-17236)

Arun[*July 14, 2016 at 4:00 pm*](http://hadooptutorial.info/hive-interview-questions-and-answers-part-4/#comment-17236)

Siva. Thanks for Publishing this. Its very exhaustive and provides a clear good view of Hive.

### Post navigation

## **Hive Interview Questions**

### [1. List down the segments of a Hive question processor?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled1)

The segments of a Hive question processor are as follows:

* Logical Plan Generation
* Physical Plan Generation
* UDF’s and UDAF’s
* Execution Engine
* Operators
* Semantic Analyzer
* Optimizer
* Type Checking
* Parser

### [2. What does meta-store means in Hive?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled2)

Meta-store in Hive stores the meta information utilizing RDBMS and an open source ORM (Object Relational Model) layer called Data Nucleus which changes over the object portrayal into a relational schema.

Hive meta-store comprises of two major units:

* A service that gives meta-store access to other Apache Hive administrations.
* Disk storage for the Hive metadata, which is separate from HDFS stockpiling.

### [3. Specify the different types of tables accessible in Hive?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled3)

Managed and External tables are the two different kinds of tables in hive used to enhance how information is loaded, managed and controlled

Two types of tables, which are used are:

* **Managed Table** -: Managed table is also known as an internal table. This is the default table in Hive. When we make a table in Hive without specifying it as external, naturally we will get a Managed table. If we make a table as a managed table, the table will be made in a specific area in HDFS.
* **External table:** External table is made up for external use as when the information is utilized outside Hive. At whatever point we need to erase the table’s meta information and we need to keep the table’s information as it seems to be, we utilize External table. The external table just erases the pattern of the table.

### [4. What do you understand by Hive?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled4)

Hive is an information warehouse programming which is utilized for encourages questioning and overseeing vast data sets residing in dispersed storage. Hive language nearly looks like SQL language called HiveQL. Hive also permits conventional map to reduce projects to customize mappers and reducers when it is awkward or wasteful to execute the logic in HiveQL (User Defined Functions UDFS)

### [5. What is the major difference between local and remote meta-store?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled5)

* **Local Meta-store:**In local meta-store design, the meta-store service keeps running in the same JVM in which the Hive service is running and associates with a database running in a different JVM, either on a similar machine or a remote machine.
* **Remote Meta-store:**In the remote meta-store design, the meta-store service keeps running alone separating JVM and not in the Hive benefit JVM. Different procedures communicate with the meta-store server utilizing Thrift Network APIs. You can have at least one meta-store servers for this situation to give greater accessibility.

### [6. What do Hive variable means? How can we utilize it?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled6)

Hive variable is made in the Hive condition that can be referenced by Hive contents. It is utilized to pass a few values to the hive inquiries when the queries begin executing.

### [7. Clarify about the SMB Join in Hive?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled7)

In SMB join in Hive, every mapper peruses a bucket from the first table and the relating bucket from the second table, and after that, a merge sort join is performed. Sort Merge Bucket (SMB) joins in the hive is for the most utilized as there are no restrictions on file or segment or table join. SMB join can best be utilized when the tables are huge. In SMB join the sections are bucketed and arranged to utilize the join segments. All tables ought to have a similar number of buckets in SMB join.

### [8. How can a developer utilize Hive?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled8)

Hive is helpful when influencing information to warehouse applications when you are dealing with static information rather than dynamic information.

* When the application is on high latency (high reaction time)
* When a big data set collection is kept up
* When we are utilizing queries instead of scripting

### [9. What is the man difference between HBase and Hive?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled9)

Both hive and HBase can be utilized in different technologies that depend on Hadoop. Hive happens to be an infrastructure warehouse of information, which is utilized on Hadoop while HBase is NoSQL. The key esteem stores which keep running on Hadoop themselves. The hive will also enable the individuals who know about SQL run a few of jobs in MapReduce when Hbase will also bolster 4 of the activities, for example, put, get, scan and erase. The HBase happens to be useful for questioning for information yet Hive then again is useful for questioning information is analytical and is gathered over a while.

### [10. What is (HS2) Hive Server2?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled10)

Hive Server2 is a server interface. Various functions, which are followed by Hive Server2 are as follows:

* Works against Hive by enabling remote customers to execute questions.
* The outcomes of inquiries specified are retrieved

**Propelled highlights:**

* Multi-customer concurrency
* Authentication

### [11. What is the utilization of Hcatalog?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled11)

Hcatalog can be utilized to share information structures with external systems. Hcatalog gives access to hive meta-store to clients of other devices on Hadoop with the goal that they can read and compose information to hive’s data warehouse.

### [12. Say when to pick “Inward Table” and “Outside Table” in Hive?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled12)

In Hive, you can pick an internal table

* If the preparing data accessible in the local file system.
* If we need Hive to deal with the entire lifecycle of data including the cancellation

You can pick an External table

* If processing information accessible in HDFS
* Useful when the documents are being utilized outside of Hive

### [13. What are the various uses of explode Hive?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled13)

Hadoop developers consider the exhibit as their inputs and convert them into a different table row. To change over data types into wanted table formats Hive is basically utilizing detonate.

### [14. Say what the views are in Hive?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled14)

Views are Similar to tables In Hive; They are produced based on various requirements:

* Any results can be spared asset data as a view in Hive
* Similar to views utilized as a part of SQL in use.
* All kind of DML tasks can be performed on a view.

### [15. What in Hive made out of?](https://www.onlineinterviewquestions.com/hive-interview-questions/" \l "collapseUnfiled15)

**Ans7.** Hive is made out of

* Clients
* Services
* Storage and Computing
* ‹
* 1
* [2](https://www.onlineinterviewquestions.com/hive-interview-questions/page/2/)
* [›](https://www.onlineinterviewquestions.com/hive-interview-questions/page/2/)
  1. What is Hive Metastore? Ans : Hive metastore is a database that stores metadata about your Hive tables (eg. table name, column names and types, table location, storage handler being used, number of buckets in the table, sorting columns if any, partition columns if any, etc.). When you create a table, this metastore gets updated with the information related to the new table which gets queried when you issue queries on that table. 2. Wherever (Different Directory) I run hive query, it creates new metastore\_db, please explain the reason for it? Ans: Whenever you run the hive in embedded mode, it creates the local metastore. And before creating the metastore it looks whether metastore already exist or not. This property is defined in configuration file hive-site.xml. Property is “javax.jdo.option.ConnectionURL” with default value “jdbc:derby:;databaseName=metastore\_db;create=true”. So to change the behavior change the location to absolute path, so metastore will be used from that location. 3. Is it possible to use same metastore by multiple users, in case of embedded hive? Ans: No, it is not possible to use metastore in sharing mode. It is recommended to use standalone “real” database like MySQL or PostGresSQL. 4. Is multiline comment supported in Hive Script ? Ans: No. 5. If you run hive as a server, what are the available mechanism for connecting it from application? Ans: There are following ways by which you can connect with the Hive Server: 1. Thrift Client: Using thrift you can call hive commands from a various programming languages e.g. C++, Java, PHP, Python and Ruby. 2. JDBC Driver : It supports the Type 4 (pure Java) JDBC Driver 3. ODBC Driver: It supports ODBC protocol. 6. What is SerDe in Apache Hive ? Ans : A SerDe is a short name for a Serializer Deserializer. Hive uses SerDe (and FileFormat) to read and write data from tables. An important concept behind Hive is that it DOES NOT own the Hadoop File System (HDFS) format that data is stored in. Users are able to write files to HDFS with whatever tools/mechanism takes their fancy("CREATE EXTERNAL TABLE" or "LOAD DATA INPATH," ) and use Hive to correctly "parse" that file format in a way that can be used by Hive. A SerDe is a powerful (and customizable) mechanism that Hive uses to "parse" data stored in HDFS to be used by Hive. 7. Which classes are used by the Hive to Read and Write HDFS Files Ans : Following classes are used by Hive to read and write HDFS files •TextInputFormat/HiveIgnoreKeyTextOutputFormat: These 2 classes read/write data in plain text file format. •SequenceFileInputFormat/SequenceFileOutputFormat: These 2 classes read/write data in hadoop SequenceFile format. 8. Give examples of the SerDe classes whihc hive uses to Serializa and Deserilize data ? Ans : 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). 9. How do you write your own custom SerDe ? Ans : •In most cases, users want to write "thrift DDL" format, and it's non-trivial to write a "thrift DDL" parser. 10. What is ObjectInspector functionality ? Ans : Hive uses ObjectInspector to analyze the internal structure of the row object and also the structure of the individual columns. ObjectInspector provides a uniform way to access complex objects that can be stored in multiple formats in the memory, including: •Instance of a Java class (Thrift or native Java) •A standard Java object (we use java.util.List to represent Struct and Array, and use java.util.Map to represent Map) •A lazily-initialized object (For example, a Struct of string fields stored in a single Java string object with starting offset for each field) A complex object can be represented by a pair of ObjectInspector and Java Object. The ObjectInspector not only tells us the structure of the Object, but also gives us ways to access the internal fields inside the Object. 11. What is the functionality of Query Processor in Apached Hive ? Ans: This component implements the processing framework for converting SQL to a graph of map/reduce jobs and the execution

### ****What is Hive?****

Hive is a tool to perform ETL (extract, transform and load) and Data warehousing functions. It was developed as a sub-project of Hadoop Distributed File System (HDFS). Hive makes operations like data encapsulation, ad-hoc queries, analysing large sets of data simpler.

### ****How to check Hive version?****

To check Hive version, you can use Putty. Open Putty and type **leo-ingesting.vip.name.com** in the hostname. Click on open and then enter username and password. Follow the command set below to find the version of Hive.

$ **bash**

**bash**-3.00$ Hive

**Hive** history file=/tmp/rkost/Hive\_job\_log\_rkost\_201207010451\_1212680168.txt

**Hive**> set mapred.job.queue.name=hdmi-technology;

**Hive**> select \* from table LIMIT 1;

### ****What is SerDe in Hive?****

SerDe is the short form for Serializer/Deserializer. Hive uses SerDe to take care of serialization and deserialization. It also interprets the outcomes of serialization considering individual fields for processing. SerDe enables Hive to read the information from a table, and then write it back as outputs of HDFS in any format you want. Anyone can write his or her own SerDe for his or her choice of data formats.

### ****What is Hive metastore?****

Hive Metastore is the chief storehouse of Hive’s metadata. It stores metadata i.e. the schema and locations of Hive tables and partitions in the form of a relational database. It also stores data such as table name, table location, column names and types, sorting columns if any, partition columns, number of buckets in the table, storage handler, etc. The metastore updates itself whenever you create a table. You can directly access this information from the metastore.

### ****When running Hive as a server what connection options are available?****

You can connect to the Hive Server in three ways.

* Thrift Client: You can use different programming languages such as C++, Java, PHP, Python etc. to call any Hive command with thrive client.
* JDBC Driver: This type of connection supports Type 4 JDBC Driver i.e. pure Java.
* ODBC Driver: This connection works on ODBC protocol.

### ****How to create an external table in Hive?****

The following syntax will help you create an external table in Hive-

**CREATE** **EXTERNAL** **TABLE** [**IF** **NOT** **EXISTS**] [db\_name.] table\_name

[(col\_name data\_type [COMMENT col\_comment], ...)]

[COMMENT table\_comment]

[**ROW** FORMAT row\_format]

[FIELDS TERMINATED **BY** **char**]

[STORED **AS** file\_format]

[LOCATION hdfs\_path];

### ****How to run Hive script?****

You have to input the following command to run Hive script.

Hive–f /home/cloudera/sample.sql

The complete path of the location of Script file must be present to successfully run the script.

### ****What is lateral view in Hive?****

A lateral view is used to apply the User Defined Tabular Function (UDTF) to every row of a table. It then joins the output and input rows to form a virtual table. Lateral view works in conjunction with user-defined table generating functions like explode().

### ****What is hcatalog in Hive?****

HCatalog in Hive is a metadata and table management system that enables storing data in any format irrespective of the structure. Both structured and unstructured data can be processed, stored and shared in HCatalog. This capability in combination with the ‘**schema on read**’ nature of Hadoop reduces cycle time and encourages exploration.

### ****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.

### ****What is bucketing in Hive? How bucketing works in Hive?****

Bucketing in Hive allows the user to divide table data sets into parts that are more manageable. It gives an acceptable structure to Hive tables when performing queries on very large datasets. Bucketing works by subdividing partitions in a table containing huge datasets into parts that are more manageable or work directly on the table without partitions.

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

In hive, a database refers to a namespace or a group of tables. To create a database, you have to use the Create Database command. The syntax to create a database is:

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

### ****How to load JSON file in Hive?****

To load a JSON file in Hive, you must first create a JSON table in hive. Then load and display the actual contents of the schema. After that, fetch the data values in the existing JSON hierarchy by using get\_json\_object().

### ****Explain serialization in Hive?****

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.

### ****How will you optimize Hive performance?****

**You can optimise the performance of Hive in one of the following ways-**

* Tez Execution Engine – A Hive optimization technique increases the performance of hive queries 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 query 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 querying only the required partitions. Thus, it reduces the time taken by a query 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 query 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 query. 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 query. The original table is indexed to create a separate index table that acts as a reference.

### ****What is the primary purpose of Hive in the Hadoop architecture?****

Hadoop is used to crunch large amounts of data with the help of an array of tools. Hive is one such tool that is used to query and analyse huge datasets. It is an open source data warehousing system, which exclusively stores data in Hadoop storage.

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

Partition in Hive is used to organise tables into partitions by dividing a table into associated parts. The division depends on the values of the columns like date, city, and department. Every table in the Hive can have more than one partition keys to identify the partition.

### ****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 in with every LOAD statement.
* Static Partition is timesaving while loading data compared to dynamic partition.
* You have the ability to alter the partition in the static partition
* The partition column value can be obtained from the filename, day of date etc. without reading the entire file.

**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 vertex and vectorization in Hive?****

Vectorization allows Hive to process a group of rows together instead of processing a single row at a given time. Every batch is an array of a primitive type. In vectorization, operations of the entire column vector at one go, which improves the instruction pipelines as well as cache usage.

### ****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> ]

**1) Explain Hive in brief?**

To query and analyze large data sets stored in HDFS, an open source data warehouse framework is developed on top of Hadoop known as Hive. Hive helps programmers perform the following operations easy and fast

* Data encapsulation
* Ad hoc queries
* Analysis of large datasets

**2) Explain the External Table features in Hive?**

Unlike RDBMS, where data and table are tightly coupled, External Table in Hive and its data in HDFS is loosely coupled. External Tables are created on top of data residing in HDFS. Even if we drop the external table in the Hive, the data mapped to it continues to reside inside HDFS.

**3) Explain the Internal Table features in Hive?**

Internal Table in Hive is similar to the tables we have in RDBMS. The data and table schema are tightly coupled. If we drop the internal table in the Hive, the data stored inside it will get deleted.

**4) What type of Read and Write operations perform in Hive?**

Hive provides READ Many WRITE Once.

**5) Explain HCatalog Functionality in Hive?**

HCatalog is a table and storage management layer of Hadoop, which supports reading and writing files in any file format for which a Hive SerDe (serializer-deserializer) can be written. By default, HCatalog supports following File formats. They are RCFile, CSV, JSON and Sequence. In case, we want to use a custom file format, we have to provide the InputFormat, OutputFormat and SerDe for that custom file format.

**6) What SerDe means?**

SerDe is a framework to serialize and deserialize IO.

**7) What are windowing functions in Hive?**

OVER, RANK

**8) What are the different types of metastores that Hive provides?**

Three Modes Embedded mode, Local mode, remote mode.

**9) How client can interact with Hive?**

We can interact with Hive using Web GUI and Java Database Connectivity(JDBC) interface. Generally, clients use command line interface (CLI) to interact with the Hive.

**10) What are the file formats that Hive supports?**

Hive supports four file formats. These file formats are TEXTFILE, SEQUENCEFILE, ORC and RCFILE (Record Columnar File).

**11) When will Hive use MySql database for metadata storage?**

Hive use MYSQL to store Hive tables and partitions metadata, when it had to handle multiple concurrent Hive sessions.

**12) What is the storage location for internal table in Hive?**

/user/Hive/warehouse

**13) What is the name of advanced Hive version used in cloudera distribution and list its advantages?**

The advanced Hive version used in Cloudera is impala. If can produce results to a query much faster compared to the Hive. It’s a pure Cloudera enterprise edition.

**14) Explain partitions in Hive?**

Based on the Partition keys present in a table which is the basis for determining how the data is stored, partitions are created inside the Hive.

**15) Explain the use Buckets Hive warehouse?**

Buckets are used for efficient querying of the Data i.e. present in that partition. The data inside the partitions can be further divided into buckets. This division is performed based on hash of a particular column, we select in the table.

**16) Explain Hive working modes?**

Hive works in two modes. They are Interactive Mode and Non Interactive Mode. In interactive mode, when you type Hive, it directly goes to Hive Mode(Hive Shell). Non Interactive Mode is about executing the code directly in console file Mode.

**17) Explain MetaStore in Hive?**

MetaStore is used to store metadata of Hive Tables like their columns, column types and it’s partition structure in an RDBMS table. MetaStore service runs in the same JVM as the services of the Hive are running.

**18) What is HiveSERVER in Hive?**

HiveSERVER is an API that allows the clients(JDBC) to execute the queries on Hive Data warehouse and get the desired results. To process and execute a query, compiler and Execution engine in HiveSERVER interact with each other.

**19) Some key Differences between Hive and relational databases?**

In relational databases, tables are created first and then the data get inserted into that table. We can execute DML commands like Insert, Update and Delete on those tables. In hive, data is stored first and tables are created on top of it. We can't execute, delete and update commands on Hive, because date get replicated to multiple nodes.

**20) Explain the function of Execution Engine in Hive Architecture?**

Execution Engine (EE) is a key component of Hive. It is used to execute the query by directly communicating with Job Tracker, Name Node and Data Nodes. When we execute a Hive query, it will generate series of MR Jobs in the backend. In this scenario, the execution engine acts as a bridge between Hive and Hadoop to process the query. For DFS operations, Execution Engine communicate with the Name Node.

## **Difference between "Sort By" and "Group by" in Hive. How they work?**

Ans. SORT BY sorts the data per reducer, it provides ordering of the rows within a reducer. If there are more than one reducer, "sort by" may give partially ordered final results. Whereas GROUP BY aggregate records by the specified columns which allows you to perform aggregation functions on non-grouped columns (such as SUM, COUNT, AVG, etc).

## **2. If we use the "Limit 1" in any SQL query in Hive, will Reducer work or not.**

Ans. I think Reducer will work, because as per Hive documentation -- Limit indicates the number of rows to be returned. The rows returned are chosen at random. The following query returns 5 rows from t1 at random.

SELECT \* FROM t1 LIMIT 5

Having to randomly pick, it has to have complete result output from Reducer.

## **- How to optimize Hive Performance?**

Ans. These links should answer this

* [5 WAYS TO MAKE YOUR HIVE QUERIES RUN FASTER](http://hortonworks.com/blog/5-ways-make-hive-queries-run-faster/)
* [5 Tips for efficient Hive queries with Hive Query Language](https://www.qubole.com/blog/big-data/5-tips-for-efficient-hive-queries/)

## **- 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 /ust/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.

## **- What is the main difference between Hive and SQL**

Ans. Hive is a Datawarehousing layer on top of hadoop that provides SQL like row table interface to users for analyzing underlying data. It employs HiveQL (HQL) language for this which is loosely based on SQL-92 standards.

SQL is a standard RDBMS language for accessing and manipulating databases.

[share](https://stackoverflow.com/a/38455580)[improve this answer](https://stackoverflow.com/posts/38455580/edit)

answered Jul 19 '16 at 10:12

[[https://www.gravatar.com/avatar/31aaca0248cebfdf24cf8706b9af4e78?s=32&d=identicon&r=PG&f=1](https://stackoverflow.com/users/5630544/shakti-tokas)](https://stackoverflow.com/users/5630544/shakti-tokas)

[Shakti Tokas](https://stackoverflow.com/users/5630544/shakti-tokas)

**41**4

* select \* from table limit 1, will not start MR after fetch operation changes in hive. – [Sanket\_patil](https://stackoverflow.com/users/2559446/sanket-patil) [Feb 18 at 12:39](https://stackoverflow.com/questions/35049403/few-hive-interview-questions#comment96276900_38455580)
* check "hive.fetch.task.conversion" parameter here: [cwiki.apache.org/confluence/display/Hive/…](https://cwiki.apache.org/confluence/display/Hive/Configuration+Properties) – [Sanket\_patil](https://stackoverflow.com/users/2559446/sanket-patil)[Feb 18 at 12:52](https://stackoverflow.com/questions/35049403/few-hive-interview-questions#comment96277370_38455580)

add a comment

0

I am new to Hadoop and Hive as well so I can't give you a complete answer.

From what I've read in the book "Hadoop The Definitive Guide" the key difference between Hive and SQL is that Hive (HiveQL) was created with MapReduce in mind. Hive's SQL dialect is supposed to make it easier for people to interact with Hadoop without needing to know a lot about Java (and SQL is well known by data professionals anyway).

As time has went on, Hive has become more compliant to the SQL standard. It blends a mix of MySQL and Oracle's SQL dialects with SQL-92.

**The Main Difference**

From what I've read, the biggest difference is that RDBMS have schema's that are typically schema on write. This means that data needs to conform to the schema when you load it in the database. In Hive, it uses schema on read because it doesn't verify the data when it is loaded.

**Information obtained from Hadoop The Definitive Guide**

Really good book and gives a good overview of all the technologies involved.

**EDIT:**

**For external and internal tables, check out this response:**

[Difference between Hive internal tables and external tables?](https://stackoverflow.com/questions/17038414/difference-between-hive-internal-tables-and-external-tables)

**Information regarding Sort By and Group By**

**Sort By:**

Hive uses the columns in SORT BY to sort the rows before feeding the rows to a reducer. The sort order will be dependent on the column types. If the column is of numeric type, then the sort order is also in numeric order. If the column is of string type, then the sort order will be lexicographical order.

**Difference between Sort By and Order By**

(Taken from the link provided maybe this will help with the difference between Group By and Sort By)

Hive supports SORT BY which sorts the data per reducer. The difference between "order by" and "sort by" is that the former guarantees total order in the output while the latter only guarantees ordering of the rows within a reducer. If there are more than one reducer, "sort by" may give partially ordered final results.

Note: It may be confusing as to the difference between SORT BY alone of a single column and CLUSTER BY. The difference is that CLUSTER BY partitions by the field and SORT BY if there are multiple reducers partitions randomly in order to distribute data (and load) uniformly across the reducers.

Basically, the data in each reducer will be sorted according to the order that the user specified.

**Group By:**

Group By is done using aggregation. It is pretty much done the same as you would normally in any other SQL dialect.

INSERT OVERWRITE TABLE pv\_gender\_sum

SELECT pv\_users.gender, count (DISTINCT pv\_users.userid)

FROM pv\_users

GROUP BY pv\_users.gender;

This query selects pv\_users.gender and counts the distinct user\_ids from the users table. In order to do count the users in a gender, you would first have to group all the users who are a certain gender together. (Query taken from the group by link below)

<https://cwiki.apache.org/confluence/display/Hive/LanguageManual+SortBy>

<https://cwiki.apache.org/confluence/display/Hive/LanguageManual+GroupBy>

**Information on Optimizing Hive Performance**

<http://hortonworks.com/blog/5-ways-make-hive-queries-run-faster/>

**Optimizing Joins**

<https://www.facebook.com/notes/facebook-engineering/join-optimization-in-apache-hive/470667928919/>

**General Hive Performance Tips**

<https://streever.atlassian.net/wiki/display/HADOOP/Hive+Performance+Tips>

**Some extra resources**

**SQL to Hive Cheat Sheet**

<http://hortonworks.com/wp-content/uploads/downloads/2013/08/Hortonworks.CheatSheet.SQLtoHive.pdf>

**Hive LIMIT Documentation**

<https://cwiki.apache.org/confluence/display/Hive/LanguageManual+Select#LanguageManualSelect-LIMITClause>

**Best of luck in your interview!**

Hive Interview Questions and Answers USA: +1 469 522 9879 | INDIA : +91 988 599 1924 | EMAIL ID: info@onlineitguru.com 1 What is Hive? Hive is a data warehouse software which is used for facilitates querying and managing large data sets residing in distributed storage. Hive language almost look like SQL language called HiveQL. Hive also allows traditional map reduce programs to customize mappers and reducers when it is inconvenient or inefficient to execute the logic in HiveQL (User Defined Functions UDFS) 2 What is Hive Metastore? Hive metastore is a database that stores metadata about your Hive tables (eg. Table name, column names and types,table location, storage handler being used, number of buckets in the table, sorting columns if any, partition columns if any, etc.). When you create a table,this metastore gets updated with the information related to the new table which gets queried when you issue queries on that table. Hive is a central repository of hive metadata. it has 2 parts services and data. by default it uses derby db in local disk. it is referred as embedded metastore configuration. It tends to the limitation that only one session can be served at any given point of time. 3 Which classes are used by the Hive to Read and Write HDFS Files? Following classes are used by Hive to read and write HDFS files •TextInputFormat/HiveIgnoreKeyTextOutputFormat: These 2 classes read/write data in plain text file format. •SequenceFileInputFormat/SequenceFileOutputFormat: These 2 classes read/write data in hadoop SequenceFile format. 4 What is Object Inspector functionality? Hive uses Object Inspector to analyze the internal structure of the row object and also the structure of the individual columns. Object Inspector provides a uniform way to access complex objects that can be stored in multiple formats in the memory, including: •Instance of a Java class (Thrift or native Java) •A standard Java object (we use java.util.List to represent Struct and Array, and use java.util.Map to represent Map) •A lazily-initialized object (For example, a Struct of string fields stored in a single Java string object with starting offset for each field). Hive Interview Questions and Answers USA: +1 469 522 9879 | INDIA : +91 988 599 1924 | EMAIL ID: info@onlineitguru.com A complex object can be represented by a pair of ObjectInspector and Java Object. The ObjectInspector not only tells us the structure of the Object, but also gives us ways to access the internal fields inside the Object. 5 What is the functionality of Query Processor in Apached Hive? This component implements the processing framework for converting SQL to a graph of map/reduce jobs and the execution time framework to run those jobs in the order of dependencies. 6 If you run hive as a server, what are the available mechanism for connecting it from application? There are following ways by which you can connect with the Hive Server: 1. Thrift Client: Using thrift you can call hive commands from a various programming languages e.g. C++,Java, PHP, Python and Ruby. 2. JDBC Driver : It supports the Type 4 (pure Java) JDBC Driver 3. ODBC Driver: It supports ODBC protocol. 7 What kind of data warehouse application is suitable for Hive? Hive is not a full database. The design constraints and limitations of Hadoop and HDFS impose limits on what Hive can do. Hive is most suited for data warehouse applications, where 1) Relatively static data is analyzed, 2) Fast response times are not required, and 3) When the data is not changing rapidly. Hive doesn’t provide crucial features required for OLTP, Online Transaction Processing. It’s closer to being an OLAP tool, Online Analytic Processing. So, Hive is best suited for data warehouse applications, where a large data set is maintained and mined for insights, reports, etc. 8 Which database hive used for Metadata store? What are the metastore configuration hive supports? Hive can use derby by default and can have three type metastore configuration. It supports • Embedded Metastore Hive Interview Questions and Answers USA: +1 469 522 9879 | INDIA : +91 988 599 1924 | EMAIL ID: info@onlineitguru.com • Local Metastore • Remote Metastore Embedded uses derby db to store data backed by file stored in disk. It can’t support multi session at same time and services of metastore runs in same JVM as hive. Local Metastore: In this case we need to have stand alone db like MySql, which would be communicated by metastore services.Benefit of this approach is, it can support multiple hive session at a time. and service still runs in same process as Hive. Remote Metastore: Metastore and Hive service would run in different process. with stand alone Mysql kind db. 9 what are Binary storage formats hive supports? Hive natively supports text file format, however hive also has support for other binary formats. Hive supports Sequence, Avro, RCFiles. 1. Sequence files :-General binary format. splittable, compressible and row oriented. a typical example can be.if we have lots of small file, we may use sequence file as a container, where file name can be a key andcontent could stored as value. it support compression which enables huge gain in performance. 2. Avro datafiles:-Same as Sequence file splittable, compressible and row oriented except support of schema evolution and multilingual binding support. 3. 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 subsequently.. 10 Is it possible to use same metastore by multiple users, in case of embedded hive? No, it is not possible to use metastore in sharing mode. It is recommended to use standalone “real” database like MySQL or PostGresSQL. 11 What is Apache Hcatalog ? HCatalog is built on top of the Hive metastore and incorporates Hive’s DDL. Apache Hcatalog is a table and data management layer for hadoop,we can process the data on Hcatalog by using APache pig,Apache Mapreduce and Apache Hive. There is no need to worry in Hcatalog where data is stored and which format of data generated. HCatalog displays data from RCFile format, text files, or sequence files in a tabular view. It also provides REST. APIs so that external systems can access these tables’ metadata. Hive Interview Questions and Answers USA: +1 469 522 9879 | INDIA : +91 988 599 1924 | EMAIL ID: info@onlineitguru.com 12 What is the work of Hive/Hcatalog ? Hive/HCatalog also enables sharing of data structure with external systems including traditional data management tools. 13 What is WebHCatServer ? The WebHcatServer provides a REST – like web API for Hcatalog. Applications make HTTP requests to run Pig,Hive, and HCatalog DDL from within applications. 14 What is Hive Present Version ? Hive-0.13.1 15 What is the stable version of Hive ? Hive-0.12.0 16 Is it possible to create multiple table in hive for same data? Hive creates schema and append on top of an existing data file. One can have multiple schema for one data file,schema would be saved in hive’s metastore and data will not be parsed read or serialized to disk in given schema.When s/he will try to retrieve data schema will be used. Lets say if my file have 5 column(Id,Name,Class,Section,Course) we can have multiple schema by choosing any number of column. 17 Wherever (Different Directory) I run hive query, it creates new metastore\_db, please explain the reason for it? Whenever you run the hive in embedded mode, it creates the local metastore. And before creating the metastore it looks whether metastore already exist or not. This property is defined in configuration file hive-site.xml. Property is “javax.jdo.option.ConnectionURL”withdefaultvalue“jdbc:derby:;databaseName=metastore\_db; create=true”. So to change the behavior change the location to absolute path, so metastore will be used from that location. 18 What is SerDe in Apache Hive ? A SerDe is a short name for a Serializer Deserializer. Hive uses SerDe (and FileFormat) to read and write data from tables. An important concept behind Hive is that it DOES NOT own the Hadoop File System (HDFS) format that data is stored in. Users are able to write files to HDFS with whatever tools/mechanism takes their fancy(“CREATE EXTERNAL TABLE” or “LOAD DATA INPATH,” ) and use Hive to correctly “parse” that file format in a way that can be used by Hive. A SerDe is a powerful (and customizable) mechanism that Hive uses to “parse” data stored in HDFS to be used by Hive. Hive Interview Questions and Answers USA: +1 469 522 9879 | INDIA : +91 988 599 1924 | EMAIL ID: info@onlineitguru.com 19 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). 20 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. 21 What are the types of tables in Hive? There are two types of tables. 1. Managed tables. 2. External tables. Only the drop table command differentiates managed and external tables. Otherwise, both type of tables are very similar. I When you drop an internal table, it drops the data, and it also drops the metadata. When you drop an external table, it only drops the meta data. That means hive is ignorant of that data now. It does not touch the data itself. 22 Does Hive support record level Insert, delete or update? Hive does not provide record-level update, insert, or delete. Henceforth, Hive does not provide transactions too. However, users can go with CASE statements and built in functions of Hive to satisfy the above DML operations. Thus, a complex update query in a RDBMS may need many lines of code in Hive. Hive Interview Questions and Answers USA: +1 469 522 9879 | INDIA : +91 988 599 1924 | EMAIL ID: info@onlineitguru.com 23 Difference between SQL and HiveQL ? 24 what is Partition? To increase performance Hive has the capability to partition data 1 The values of partitioned column divide a table into segments 2 Entire partitions can be ignored at query time 3 Similar to relational databases’ indexes but not as granular 25 what is bucketing? Mechanism to query and examine random samples of data • Break data into a set of buckets based on a hash function of a "bucket column" .

#### [.Which File System Support Hive?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-400)

TextFile, SequenceFile, RCFile, AVRO, ORC, Parquet.

#### [What is Hive Metastore?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-401)

It is the central repository of Apache Hive metadata. It stores metadata for Hive tables (like their schema and location) and [partitions](https://data-flair.training/blogs/wp-content/uploads/sites/2/2017/09/apache-hive-partitioning.jpg) in a relational database. It also provides client access to this information with the help of metastore service API

#### [What is the default database provided by Apache Hive for metastore?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-402)

Hive offers an embedded Derby database instance backed by the local disk for the metastore, by default. To this concept what we call embedded metastore configuration.

#### [What is Hive Operators? What are the Types of Hive Operators?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-404)

Apache Hive provides Different Built-in operators for data operations to be implemented on the tables present inside Apache Hive warehouse.

Hive operators are used for mathematical operations on operands. It returns specific value as per the logic applied.

Types Of Hive Operators Are:

* Relational Operators
* Arithmetic Operators
* Logical Operators
* String Operators
* Operators on Complex Types

#### [What is Bucketing in Hive?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-406)

For decomposing table data sets into more manageable parts, Apache Hive offers another technique. That technique is called as a Bucketing in Hive.

 In Hive Tables or partition are subdivided into buckets based on the hash function of a column in the table to give extra structure to the data that may be used for more efficient queries.

#### [What is Internal Table and external Table in Hive?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-407)

**Internal Table (Managed table):**Managed table is also Known as Internal table. This is the default table in Hive. When user create a table in Hive without specifying it as external, by default we will get a Managed table.

If we create a table as a managed table, the table will be created in a specific location in HDFS.

By default, the table data will be created in /usr/hive/warehouse directory of HDFS.

**External Tables:**External table is mostly created for external use as when the data is used outside Hive. Whenever we want to delete the table’s metadata and we want to keep the table’s data as it is, we use External table. External table only deletes the schema of the table.a

#### [How can Client Interact with Hive?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-409)

 There are three ways of Interacting with Hive:

**I Hive Thrift Client:**

Basically,  with any programming language that supports thrift, we can interact with HIVE.

**IIIJDBC Driver:**

However, to connect to the HIVE Server the BeeLine CLI uses JDBC Driver.

**III**. **ODBC Driver:**

Also, we can use an ODBC Driver application. Since that support ODBC to connect to the HIVE server.

#### [Name Key components of Hive Architecture?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-410)

* User Interface
* Compiler
* Metastore
* Driver
* Execute Engine

#### [.What is SerDe in Apache Hive ?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-411)

A SerDe is a short name for a Serializer Deserializer. Hive uses SerDe to read and write data from tables. An important concept behind Hive is that it DOES NOT own the Hadoop File System format that data is stored in. Users are able to write files to HDFS with whatever tools/mechanism takes their fancy("CREATE EXTERNAL TABLE" or "LOAD DATA INPATH," ) and use Hive to correctly "parse" that file format in a way that can be used by Hive. A SerDe is a powerful (and customizable) mechanism that Hive uses to "parse" data stored in HDFS to be used by Hive.

#### [What is ObjectInspector functionality?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-413)

Hive uses ObjectInspector to analyze the internal structure of the row object and also the structure of the individual columns. ObjectInspector provides a uniform way to access complex objects that can be stored in multiple formats in the memory, including:

Instance of a Java class (Thrift or native Java)

* A standard Java object (we use java.util.List to represent Struct and Array, and use java.util.Map to represent Map)
* A lazily-initialized object (For example, a Struct of string fields stored in a single Java string object with starting offset for each field) A complex object can be represented by a pair of ObjectInspector and Java Object. The ObjectInspector not only tells us the structure of the Object, but also gives us ways to access the internal fields inside the Object.

#### [Can we change the default location of Managed tables?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-415)

Yes, with the help of LOCATION keyword, we can change the default location of Managed tables while creating the managed table in Hive. However, to do so, the user needs to specify the storage path of the managed table as the value to the LOCATION keyword, that will help to change the default location of a managed table.

#### [Which classes are used in Hive to Read and Write HDFS Files?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-417)

 Following classes are used by Hive to read and write HDFS files

* TextInputFormat/HiveIgnoreKeyTextOutputFormat: These 2 classes read/write data in plain text file format
* SequenceFileInputFormat/SequenceFileOutputFormat: These 2 classes read/write data in hadoop SequenceFile format.

#### [Mention some Hive DDL Commands?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-418)

 However, there are several types of Hive DDL commands, we commonly use. such as:

1. Create Database Statement
2. Hive Show Database
3. Drop database
4. Creating Hive Tables
5. Browse the table
6. Altering and Dropping Tables
7. Hive Select Data from Table
8. Hive Load Data

#### [What is Hcatalog?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-419)

HCatalog is a table and storage management layer for Hadoop that enables users with different data processing tools — Pig, MapReduce — to more easily read and write data on the grid.

Hcatalog can be used to share data structures with external systems. Hcatalog provides access to hive metastore to users of other tools on Hadoop so that they can read and write data to hive data warehouse.

#### [What is Hive Thrift Server?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-420)

The main purpose of Hive Thrift server is it allows access to Hive over a single port.

Thrift server is also known as Thrift Server.However, for scalable cross-language services development Thrift is a software framework. Also, it allows clients using languages including Java, C++, Ruby, and many others, to programmatically access Hive remotely.

#### [.What is Hiveserver2 (HS2)?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-421)

The HiveServer2 is a server interface and part of Hive Services that enables remote clients to execute queries against Hive and retrieve the results. The current implementation(HS2), based on Thrift RPC which has improved version of Hive Server 1 and supports multi-client concurrency and authentication. It is designed to provide better support for open API clients like JDBC and ODBC Drivers.

#### [How data transfer happens from HDFS to Hive?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-422)

If data is already present in HDFS then the user need not LOAD DATA that moves the files to the /user/hive/warehouse/. So the user simply has to define the table using the keyword external that creates the table definition in the hive metastore.

Create external table table\_name (

  id int,

  myfields string

)

location '/my/location/in/hdfs';

#### [9.Why Hive does not store metadata information in HDFS?](https://www.zeolearn.com/interview-questions/hive#collapse-beginner-423)

Hive stores metadata information in the metastore using RDBMS instead of HDFS. The main reason for choosing RDBMS is to achieve low latency because HDFS read/write operations are time consuming processes.

 was asked to do a design task a week in advance that involved designing a feature of the Hive app using as a starting point their current app, and present the concept to the Head of UX in an interview.

#### **Q1). What is Hive?**

Hive is a data warehouse system that is used to facilitate queries and managing large datasets residing in distributed storage. Hive programming language looks very much similar to the SQL language and named as the HiveQL. The platform allows traditional MapReduce programs to customize mapper and reducers when it is not convenient or efficient to execute logic in HiveQL.

#### **Q2). How Hive and HBase are different?**

Hive and HBase are generally used for different technologies based on the Hadoop. Hive is given as an infrastructure warehouse of data that is based on Hadoop and HBase in NoSQL. Queries in Hive are executed as MapReduce jobs internally and HBase operations run in the real-time. Hive provides high latency for huge datasets and HBase offers the low latency. Both platforms give random access to the data.

#### **Q3). What are the programming languages supported in Hive?**

Java, PHP, Python, C++, Ruby etc.

#### **Q4). How will you define a Metastore in Hive?**

A Metastore in Hive will store the metadata information using RDBMS and an open source layer called the Data Nucleus that converts objects relationship into the relational schema and vice versa.

#### **Q5). Why Hive stores the metadata information in Metastore, not in HDFS?**

Hive stores metadata information in Metastore using RDBMS, not the HDFS. The reason for choosing RDBMS is low latency. At the same time, read/write operations in HDFS is quite a time consuming when compared to RDBMS.

#### **Q6). How does the data of a Hive table get stored?**

By default, data for a Hive table is stored in the HDFS directory. Users had the flexibility to change the directory path by specifying any other desired directory.

### ****Hive Interview Questions Answers for Experienced****

Here you can check top hive interview questions with answers for the experienced user.

#### **Q7). How will you differentiate the local Metastore and remote Metastore from each other?**

**Local Metastore:**

In the case of local Metastore configuration, the Metastore service runs in the same JVM in which Hive services are running and connects to a different database running on to a different JVM. This JVM would either lie on the same machine or the remote machine.

**Remote Metastore:**

In the case of remote Metastore configuration, the Metastore service runs on its own separate JVM, not in the Hive service JVM. You could have multiple Metastore servers here to provide maximum availability.

#### **Q8). What is default database supported by the Hive Metastore?**

An Embedded Debry database instance is supported by the Hive Metastore by default. It can also be named as the embedded Metastore configuration.

#### **Q9). How will you differentiate the external table and managed table in Hive?**

In the case of a managed table, if one puts the command “drop a managed table” then it will delete the complete table along with metadata information from the Hive warehouse dictionary. In contrary, when we put the same command for an external table then it will delete only metadata information but table data is still the same.

#### **Q10). What is the method to change the default location of a managed table?**

You should simply use the command – LOCATION ‘<hdfs\_path>’.

#### **Q11). Define a partition in Hive.**

In Hive, data is organized in partitions where a similar type of data is grouped together based on the partition key. Each table could have one or more partitions keys to identify a specific partition. In simple words, a partition is nothing but a sub-directory in the table directory.

#### **Q12). When should use SORT BY command instead of Order By command?**

You should use Sort By when you are interested in sorting huge datasets because the command will use multiple reducers and makes the things little simpler. At the same time, Order By command is suitable for small datasets as it is using a single reducer only.

#### **Q13). What is the need for partitioning in Hive?**

Partitions in Hive are used to provide greater granularity and reduces the query latency by scanning relevant data partitions instead of the whole dataset.

#### **Q14). How will you define the dynamic partitioning in Hive and when is it required?**

In dynamic partitioning, values for partition columns are known in the real-time or we can say values are identified when data is loaded into a Hive table. Here are two possible situations when dynamic partitioning can be used on priority.

* When loading data from an existing non-partitioned table to improve the sampling and decrease the overall latency.
* When someone doesn’t know the value of partitions, calculating values for these partitions could be a tedious task. So, this would be a great idea opting for dynamic partitions here.

#### **Q15). How many dynamic partitions can be created by a mapper or reducer in Hive? How can you change the default length?**

By default, a maximum number of partitions that can be created by a mapper or reducer include 100. Once can change this default count by writing the following command –

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

You can set a limit yourself in maximum values of dynamic partitions with the command given below.

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

**Summary**

I hope you find this blog for Apache Hive Interview Questions and Answers relevant for you. You may check our other related blogs too like Hadoop Interview Questions, HDFS interview questions and more. We wish you luck with your next interview!

#### **Related Interview Questions**

[Content Writers Interview Questions](https://www.andlearning.org/content-writers-interview-questions/)

[Front-End Developer Interview Questions](https://www.andlearning.org/front-end-developer-interview-questions/)

[Splunk Interview Questions](https://www.andlearning.org/splunk-interview-questions/)

[Ansible Interview Questions](https://www.andlearning.org/ansible-interview-questions/)

[Magento Interview Questions](https://www.andlearning.org/magento-interview-questions/)

[Hive Interview Questions](https://www.andlearning.org/hive-interview-questions/)

[Boostrap Interview Questions](https://www.andlearning.org/bootstrap-interview-questions/)

[Social Media Interview Questions](https://www.andlearning.org/social-media-marketing-interview-questions/)

[Agile Interview Questions](https://www.andlearning.org/agile-testing-interview-questions/)

[Devops Interview Questions](https://www.andlearning.org/devops-interview-questions/)

[Docker Interview Questions](https://www.andlearning.org/docker-interview-questions/)

[Common Job Interview Questions](https://www.andlearning.org/common-job-interview-questions/)

[Salesforce Interview Questions](https://www.andlearning.org/salesforce-interview-questions/)

[WordPress Interview Questions](https://www.andlearning.org/wordpress-interview-questions/)

[Software Testing Interview Questions](https://www.andlearning.org/software-testing-interview-questions/)

[Java Interview Questions](https://www.andlearning.org/java-interview-questions/)

[Angularjs Interview Questions](https://www.andlearning.org/angularjs-interview-questions/)

[Graphic Designer Interview Questions](https://www.andlearning.org/graphic-designer-interview-questions/)

[Selenium Interview Questions](https://www.andlearning.org/selenium-tester-interview-questions/)

[Web Designer Interview Questions](https://www.andlearning.org/web-designers-interview-questions/)

[SEO Interview Questions](https://www.andlearning.org/seo-interview-questions-answers/)

[Jquery Interview Questions](https://www.andlearning.org/jquery-interview-questions/)

[Networking Interview Questions](https://www.andlearning.org/hardware-and-networking-interview-questions/)

[HR Interview Questions](https://www.andlearning.org/hr-interview-questions/)

[Spring Interview Questions](https://www.andlearning.org/spring-interview-questions/)

[C# Interview Questions](https://www.andlearning.org/c-sharp-interview-questions/)

[PHP Interview Questions](https://www.andlearning.org/php-interview-questions/)

[JavaScript Interview Questions](https://www.andlearning.org/javascript-interview-questions/)

[SQL Interview Questions](https://www.andlearning.org/sql-interview-questions/)

[AWS Interview Questions](https://www.andlearning.org/aws-interview-questions/)

**TECHNICAL INTERVIEW QUESTIONS:-**

**PIG**

**1. List the Primitive Data Types?**

**DataTypes:**  
TINYINT,BIGINT,

**2. What problem does Apache Pig solve?**

**Scenario**  
1. MapReduce paradigm presented by Hadoop is low level and rigid so developing can be challenging.  
2. Jobs are (mainly) in Java where developer needs to think in terms of map and reduce

**Problem**  
1. Many common operations like filters, projections, joins requires a custom code  
2. Not everyone is a Java expert!!!  
3. MapReduce has a long development cycle

**3. What is Apache Pig?**

Apache Pig is a platform for analyzing large data sets that consists high-level language for expressing data analysis programs, with infrastructure for evaluating these programs.  
**Goals:** Ease of programming, Improved Code readability, Flexible, Extensible

**Pig Features:**  
Ease of programming:

* Generates MapReduce programs automatically
* Fewer lines of code

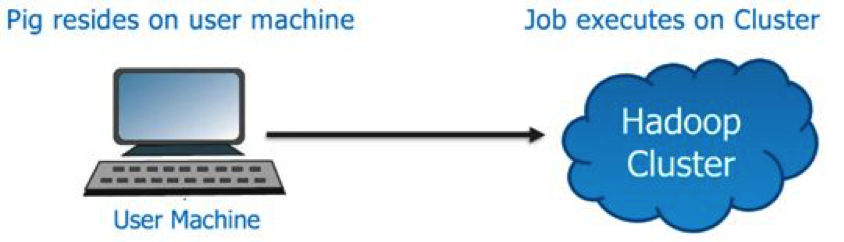
Flexible:

* Metadata is optional

Extensible:

* Easy extensible by UDFs

Resides on the client machine



**4. In which scenario MapReduce is a better fit than Pig?**

Some problems are harder to express in Pig. For example:

1. Complex grouping or joins
2. Combining lot of datasets
3. Replicated join
4. Complex cross products

In such cases, Pig’s MAPREDUCE relational operator can be used which allows plugging in Java MapReduce job.

**5. In which scenario Pig is better fit than MapReduce?**

Pig provides common data operations (joins, filters, group by, order by, union) and nested data types (tuple, bag and maps), which are missing from MapReduce.

**6. Where not to use Pig?**

1. Completely unstructured data. For example: images, audio, video
2. When more power to optimize the code is required
3. Retrieving a single record in a very large dataset

**7. What can be feed to Pig?**

We can input structured, semi-structured or unstructured data to Pig.  
For example, CSV’s, TSV’s, Delimited Data, Logs

**8. What are the components of Apache Pig platform?**

**Pig Engine**  
Parser, Optimizer and produces sequences of MapReduce programs

**Grunt**  
Pig’s interactive shell  
It allows users to enter Pig Latin interactively and interact with HDFS

**Pig Latin**  
High level and easy to understand dataflow language  
Provides ease of programming, extensibility and optimization.

**9. What are the execution modes in Pig?**

Pig has **two** execution modes:

**Local mode**  
No Hadoop / HDFS installation is required  
All processing takes place in only one local JVM  
Used only for quick prototyping and debugging Pig Latin script  
pig -x local

**MapReduce mode (Default)**  
Parses, checks and optimizes locally

1. Plans execution as one MapReduce job
2. Submits job to Hadoop
3. Monitors job progress

pig or pig -x mapreduce

**10. Different running modes for running Pig?**

Pig has **two** running modes:

**Interactive mode**  
Pig commands runs one at a time in the grunt shell

**Batch mode**  
Commands are in pig script file.

**11. What are the different ways to develop PigLatin scripts?**

Plugins are available which features such as syntax/error highlighting, auto completion etc.  
Eclipse plugins

1. PigEditor
2. PigPen
3. Pig-Eclipse

Vim, Emacs, TextMate plugins also available

**12. What are the Data types in Pig?**

**Scalar Types**  
Int, long, float, double, chararray, bytearray, boolean (since Release 0.10.0)

**Complex Types**  
Map, Tuple, Bag

**13. Which type in Pig is not required to fit in Memory?**

1. Bag is the type not required to fit in memory, as it can be quite large.
2. It can store bags to disk when necessary.

**14. What is a Map in Pig?**

Map is a chararray to data element mapping, where data element be of any Pig data type.  
It can also be called as a set of key-value pairs where

 Keys → chararray and Values → any pig data type  
For example [‘student’#’Mahi’, ’Rank’#1]

**15. What is a Tuple in Pig? (~ RDBMS row in a table)**

A tuple is an ordered set of fields; fields can be of any data type.  
It can also be called as a sequence of fields of any type.

**16. What are the components of Pig Execution Environment?**

The components of Apache Pig Execution Environment are:

* **Pig Scripts**: Pig scripts are submitted to the Apache Pig execution environment which can be written in Pig Latin using built-in operators and UDFs can be embedded in it.
* **Parser**: The Parser does the type checking and checks the syntax of the script. The parser outputs a DAG (directed acyclic graph). DAG represents the Pig Latin statements and logical operators.
* **Optimizer:** The Optimizer performs the optimization activities like split, merge, transform, reorder operators, etc. The optimizer provides the automatic optimization feature to Apache Pig. The optimizer basically aims to reduce the amount of data in the pipeline.
* **Compiler**: The Apache Pig compiler converts the optimized code into MapReduce jobs automatically.
* **Execution Engine**: Finally, the MapReduce jobs are submitted to the execution engine. Then, the MapReduce jobs are executed and the required result is produced.

## **17. What are the different ways of executing Pig script?**

There are three ways to execute the Pig script:

* **Grunt Shell**: This is Pig’s interactive shell provided to execute all Pig Scripts.
* **Script File**: Write all the Pig commands in a script file and execute the Pig script file. This is executed by the Pig Server.
* **Embedded Script**: If some functions are unavailable in built-in operators, we can programmatically create User Defined Functions (UDF) to bring that functionality using other languages like Java, Python, Ruby, etc. and embed it in the Pig Latin Script file. Then, execute that script file.

## **18. What is a bag in Pig Latin?**

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

## **19. What do you understand by an inner bag and outer bag in Pig?**

Outer bag or relation is nothing but a bag of tuples. Here relations are similar as relations in relational databases. For example:

{(Linkin Park, California), (Metallica, Los Angeles), (Mega Death, Los Angeles)}

An inner bag contains a bag inside a tuple. For Example:

(Los Angeles, {(Metallica, Los Angeles), (Mega Death, Los Angeles)})

(California, {(Linkin Park, California)})

## **20. List the diagnostic operators in Pig.**

Pig supports a number of diagnostic operators that you can use to debug Pig scripts.

* **DUMP:**Displays the contents of a relation to the screen.
* **DESCRIBE:** Return the schema of a relation.
* **EXPLAIN:** Display the logical, physical, and MapReduce execution plans.
* **ILLUSTRATE:** Gives the step-by-step execution of a sequence of statements.

## **21. Does ‘ILLUSTRATE’ run a MapReduce job?**

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

ILLUSTRATE operator is used to review how data is transformed through a sequence of Pig Latin statements. ILLUSTRATE command is your best friend when it comes to debugging a script. This command alone might be a good reason for choosing Pig over something else.

Syntax: illustrate relation\_name;

## **22. List the relational operators in Pig.**

All Pig Latin statements operate on relations (and operators are called relational operators). Different relational operators in Pig Latin are:

* **COGROUP**: Joins two or more tables and then perform GROUP operation on the joined table result.
* **CROSS**: CROSS operator is used to compute the cross product (Cartesian product) of two or more relations.
* **DISTINCT**: Removes duplicate tuples in a relation.
* **FILTER**: Select a set of tuples from a relation based on a condition.
* **FOREACH**: Iterate the tuples of a relation, generating a data transformation.
* **GROUP**: Group the data in one or more relations.
* **JOIN**: Join two or more relations (inner or outer join).
* **LIMIT**: Limit the number of output tuples.
* **LOAD**: Load data from the file system.
* **ORDER**: Sort a relation based on one or more fields.
* **SPLIT**: Partition a relation into two or more relations.
* **STORE**: Store data in the file system.
* **UNION**: Merge the content of two relations. To perform a UNION operation on two relations, their columns and domains must be identical.

## **23. Is the keyword ‘DEFINE’ like a function name?**

Yes, the keyword ‘DEFINE’ is like a function name.

DEFINE statement is used to assign a name (alias) to a UDF function or to a streaming command.

* The function has a long package name that you don’t want to include in a script, especially if you call the function several times in that script. The constructor for the function takes string parameters. If you need to use different constructor parameters for different calls to the function you will need to create multiple defines – one for each parameter set.
* The streaming command specification is complex. The streaming command specification requires additional parameters (input, output, and so on). So, assigning an alias makes it easier to access.

**24. What is the function of co-group in Pig?**

COGROUP takes members of different relations, binds them by similar fields, and creates a bag that contains a single instance of both relations where those relations have common fields. Co-group operation joins the data set by grouping one particular data set only.

It groups the elements by their common field and then returns a set of records containing two separate bags. The first bag consists of the first data set record with the common data set and the second bag consists of the second data set records with the common data set.

## **25. Can we say co-group is a group of more than 1 data set?**

Co-group is a group of data sets. More than one data set, co-group will group all the data sets and join them based on the common field. Hence, we can say that co-group is a group of more than one data set and join of that data set as well.

## **26. The difference between GROUP and COGROUP operators in Pig?**

Group and Cogroup operators are identical. For readability, GROUP is used in statements involving one relation and COGROUP is used in statements involving two or more relations. Group operator collects all records with the same key. Cogroup is a combination of group and join, it is a generalization of a group instead of collecting records of one input depends on a key, it collects records of n inputs based on a key. At a time, we can Cogroup up to 127 relations.

## **27. You have a file personal\_data.txt in the HDFS directory with 100 records. You want to see only the first 5 records from the employee.txt file. How will you do this?**

For getting only 5 records from 100 records we use limit operator.

First load the data in Pig:

personal\_data = LOAD “/personal\_data.txt” USING PigStorage(‘,’) as (parameter1, Parameter2, …);

Then Limit the data to 5 records:

limit\_data = LIMIT personal\_data 5;

## **28. What are the different execution modes available in Pig?**

The execution modes in Apache Pig are:

* **MapReduce Mode**: This is the default mode, which requires access to a Hadoop cluster and HDFS installation. Since, this is a default mode, it is not necessary to specify -x flag (you can execute pig OR pig -x mapreduce). The input and output in this mode are present on HDFS.
* **Local Mode:** With access to a single machine, all files are installed and run using a local host and file system. Here the local mode is specified using ‘-x flag’ (pig -x local). The input and output in this mode are present on local file system.

**29 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.

**30 What do you mean by a bag in Pig?**

Collection of tuples is referred as a bag in Apache Pig

**31 What is the usage of foreach operation in Pig scripts?**

FOREACH operation in Apache Pig is used to apply transformation to each element in the data bag, so that respective action is performed to generate new data items.

Syntax- FOREACH data\_bagname GENERATE exp1, exp2

**32 Explain about the different complex data types in Pig.**

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. Tuples can have multiple attributes.
* Bags- Unordered collection of tuples. Bag allows multiple duplicate tuples.

**33 What does Flatten do in Pig?**

Sometimes there is data in a tuple or a bag and if we want to remove the level of nesting from that data, then Flatten modifier in Pig can be used. Flatten un-nests bags and tuples. For tuples, the Flatten operator will substitute the fields of a tuple in place of a tuple, whereas un-nesting bags is a little complex because it requires creating new tuples.

**34 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.

**35 What are the various diagnostic operators available in Apache Pig?**

1. **Dump Operator-**It is used to display the output of pig Latin statements on the screen, so that developers can debug the code.
2. **Describe Operator-**Explained in apache pig interview question no- 10
3. **Explain Operator-**Explained in apache pig interview question no -10
4. **Illustrate Operator-** Explained in apache pig interview question no -11

**36 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.

**37 I have a relation R. How can I get the top 10 tuples from the relation R.?**

TOP () function returns the top N tuples from a bag of tuples or a relation. N is passed as a parameter to the function top () along with the column whose values are to be compared and the relation R.

**38 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 queries.

**39  What are the different types of UDF’s in Java supported by Apache Pig?**

Algebraic, Eval and Filter functions are the various types of UDF’s supported in Pig.

**40 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?**

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.

**41 Explain about the scalar datatypes in Apache Pig.**

integer, float, double, long, bytearray and char array are the available scalar datatypes in Apache Pig.

**42 How do users interact with HDFS in Apache Pig ?**

Using the grunt shell.

**43 What is the use of having Filters in Apache Pig ?**

**Just like the where clause in SQL, Apache Pig has filters to extract records based on a given condition or predicate. The record is passed down the pipeline if the predicate or the condition turn to true. Predicate contains various operators like ==, <=,!=, >=.**

**Example -**

X= load ‘inputs’ as(name,address)

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

**44 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.

**45 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.

**46 Does Pig support multi-line commands?**

Yes

What are the common hadoop PIG interview questions, that you have been asked in a Hadoop Job Interview? Let us know in comments below, to help the big data community.

## **47 Difference between**[Pig and Hive](https://www.dezyre.com/Big-Data-and-Hadoop/19)

The below tabular data will give you an overview on the basic difference between Pig and Hive:

|  |  |  |
| --- | --- | --- |
| **Characteristic** | **Pig** | **Hive** |
| Language Name | Pig Latin | HiveQL |
| Type of Language | Dataflow | Declarative (SQL Dialect) |
| Developed By | Yahoo | Facebook |
| Data Structures Supported | Nested and Complex |  |
| Relational Complete | YES | YES |
| Schema Optional | YES | NO |
| Turing Complete | YES,when you extend it with Java User Defined Functions. | YES, when you extend it with Java User Defined Functions. |
| Pig vs Hive | | |

**48 What is the difference between logical and physical plans?**

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

**49 Can we say cogroup is a group of more than 1 data set?**

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

**50 why should we use ‘orderby’ keyword in pig scripts?**

Answer: The order statement sorts your data for you, producing a total order of your output data.The syntax of order is similar to group. You indicate a key or set of keys by which you wish to order your data

input2 = load ‘daily’ as (exchanges, stocks);

grpds = order input2 by exchanges;

**51 Pig Features ?**

Answer: i) Data Flow Language

User Specifies a Sequence of Steps where each step specifies only a single high-level data transformation.

1. ii) User Defined Functions (UDF)

iii)Debugging Environment

1. iv) Nested data Model

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

Answer: The pig is easy to learn: Pig is easy to learn, it overcomes the need for writing complex MapReduce programs to some extent. Pig works in a step by step manner. So it is easy to write, and even better, it is easy to read.

It can handle heterogeneous data: Pig can handle all types of data – structured, semi-structured, or unstructured.

* Pig is Faster: Pig’s multi-query approach combines certain types of operations together in a single pipeline, reducing the number of times data is scanned.
* Pig does more with less: Pig provides the common data operations (filters, joins, ordering, etc.) And nested data types (e.g. Tuples, bags, and maps) which can be used in processing data.
* Pig is Extensible: Pig is easily extensible by UDFs – including Python, Java, JavaScript, and Ruby so you can use them to load, aggregate and analysis. Pig insulates your code from changes to the Hadoop Java API.

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

Answer: The physical form of execution of pig script happens at this stage. Physical plan is responsible for converting operators to Physical Plan.

**54 What Is Difference Between Mapreduce and Pig ?**

Answer:

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

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

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

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

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

**56 Why do we need Pig?**

Answer: Pig is a high level scripting language that is used with Apache Hadoop. Pig excels at describing data analysis problems as data flows. Pig is complete in that you can do all the required data manipulations in Apache Hadoop with Pig. In addition through the User Defined Functions(UDF) facility in Pig you can have Pig invoke code in many languages like JRuby, Jython and Java. Conversely you can execute Pig scripts in other languages. The result is that you can use Pig as a component to build larger and more complex applications that tackle real business problems.

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

Answer: Below are most commonly used STRING pig functions

* UPPER
* LOWER
* TRIM
* SUBSTRING
* INDEXOF
* STRSPLIT
* LAST\_INDEX\_OF

**58 What is a relation in Pig?**

Answer: A Pig relation is a bag of tuples. A Pig relation is similar to a table in a relational database, where the tuples in the bag correspond to the rows in a table. Unlike a relational table, however, Pig relations don t require that every tuple contain the same number of fields or that the fields in the same position (column) have the same type.

**59 What is a tuple?**

Answer: A tuple is an ordered set of fields and A field is a piece of data.

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

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

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

Answer: In the Logical plan stage of Pig, statements are parsed for syntax error. Validation of input files and the data structure of the file is also analysed. A DAG (Directed Acyclic Graph) of operators as nodes and data flow as edges are then created. Optimization of pig scripts also materialized to the logical plan.

**62 What is UDF in Pig?**

Answer: The pig has wide-ranging inbuilt functions, but occasionally we need to write complex business logic, which may not be implemented using primitive functions. Thus, Pig provides support to allow writing User Defined Functions (UDFs) as a way to stipulate custom processing.

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

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

Answer: Following are the primitive data types in pig

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

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

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

**63 why should we use ‘distinct’ keyword in pig scripts?**

Answer: The distinct statement is very simple. It removes duplicate records. It works only on entire records, not on individual fields:

input2 = load ‘daily’ as (exchanges, stocks);

grpds = distinct exchanges;

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

Answer: Below are most commonly used math pig functions

* ABS
* ACOS
* EXP
* LOG
* ROUND
* CBRT
* RANDOM
* SQRT

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

Answer: Below are most commonly used Eval pig functions

* AVG
* CONCAT
* MAX
* MIN
* SUM
* SIZE
* COUNT
* COUNT\_STAR
* DIFF
* TOKENIZE
* IsEmpty

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

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

1. LOAD
2. STORE

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

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

Answer: COGROUP operator in Pig is used to work with multiple tuples. COGROUP operator is applied on statements that contain or involve two or more relations. The COGROUP operator can be applied on up to 127 relations at a time. When using the COGROUP operator on two tables at once-Pig first groups both the tables and after that joins the two tables on the grouped columns.

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

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

**69 What are different modes of execution in Apache Pig?**

Answer: Apache Pig runs in 2 modes- one is the “Pig (Local Mode) Command Mode” and the other is the “Hadoop MapReduce (Java) Command Mode”. Local Mode requires access to only a single machine where all files are installed and executed on a local host whereas MapReduce requires accessing the Hadoop cluster.

**70 Does Pig support multi-line commands?**

Answer: Yes

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

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

* DUMP
* DESCRIBE
* ILLUSTRATE
* EXPLAIN

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

**72What is the difference between store and dumps commands?**

Answer: Dump Command after process the data displayed on the terminal, but it’s not stored anywhere. Where as store store in local file system or HDFS and output execute in a folder. In the protection environment most opften hadoop developer used ‘store’ command to store data in the HDFS.

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

* 1. **uestion 1. 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 queries 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 begins.

* 1. **Question 2. Explain The Need For Mapreduce While Programming In Apache Pig.?**

**Answer :**

Apache Pig programs are written in a query language known as Pig Latin that is similar to the SQL query language. To execute the query, there is a need for an execution engine. The Pig engine converts the queries into MapReduce jobs and thus MapReduce acts as the execution engine and is needed to run the programs.

* 1. **Question 3. Explain About The Bloommapfile.?**

**Answer :**

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.

* 1. **Question 4. What Do You Mean By A Bag In Pig?**

**Answer :**

Collection of tuples is referred as a bag in Apache Pig.

* 1. **Question 5. 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, so that respective action is performed to generate new data items.

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

* 1. **Question 6. 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. Tuples can have multiple attributes.

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

* 1. **Question 7. What Does Flatten Do In Pig?**

**Answer :**

Sometimes there is data in a tuple or a bag and if we want to remove the level of nesting from that data, then Flatten modifier in Pig can be used. Flatten un-nests bags and tuples. For tuples, the Flatten operator will substitute the fields of a tuple in place of a tuple, whereas un-nesting bags is a little complex because it requires creating new tuples.

* 1. **Question 8. How Do Users Interact With The Shell In Apache Pig?**

**Answer :**

Using Grunt i.e. Apache Pig’s interactive shell, users can interact with HDFS or the local file system.

**To start Grunt, users should invoke Apache Pig with no command:**

Executing the command “pig –x local” will result in the prompt -

grunt >

This is where PigLatin scripts can be run either in local mode or in cluster mode by setting the configuration in PIG\_CLASSPATH.

To exit from grunt shell, press CTRL+D or just type exit.

* 1. **Question 9. What Are The Debugging Tools Used For Apache Pig Scripts?**

**Answer :**

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

explain utility is helpful for Hadoop developers, when trying to debug error or optimize PigLatin scripts. explain can be applied on a particular alias in the script or it can be applied to the entire script in the grunt interactive shell. explain utility produces several graphs in text format which can be printed to a file.

describe debugging utility is helpful to developers when writing Pig scripts as it shows the schema of a relation in the script. For beginners who are trying to learn Apache Pig can use the describe utility to understand how each operator makes alterations to data. A pig script can have multiple describes.

* 1. **Question 10. What Is Illustrate Used For In Apache Pig?**

**Answer :**

Executing pig scripts on large data sets, usually takes a long time. To tackle this, developers run pig scripts on sample data but there is possibility that the sample data selected, might not execute your pig script properly.

For instance, if the script has a join operator there should be at least a few records in the sample data that have the same key, otherwise the join operation will not return any results. To tackle these kind of issues, illustrate is used. illustrate takes a sample from the data and whenever it comes across operators like join or filter that remove data, it ensures that only some records pass through and some do not, by making modifications to the records such that they meet the condition. illustrate just shows the output of each stage but does not run any MapReduce task.

* 1. **Question 11. Explain About The Execution Plans Of A Pig Script?<br> Or<br> Differentiate Between The Logical And Physical Plan Of An Apache Pig Script?**

**Answer :**

Logical and Physical plans are created during the execution of a pig script. Pig scripts are based on interpreter checking. Logical plan is produced after semantic checking and basic parsing and no data processing takes place during the creation of a logical plan. For each line in the Pig script, syntax check is performed for operators and a logical plan is created. Whenever an error is encountered within the script, an exception is thrown and the program execution ends, else for each statement in the script has its own logical plan.

A logical plan contains collection of operators in the script but does not contain the edges between the operators.

After the logical plan is generated, the script execution moves to the physical plan where there is a description about the physical operators, Apache Pig will use, to execute the Pig script. A physical plan is more or less like a series of MapReduce jobs but then the plan does not have any reference on how it will be executed in MapReduce. During the creation of physical plan, cogroup logical operator is converted into 3 physical operators namely –Local Rearrange, Global Rearrange and Package. Load and store functions usually get resolved in the physical plan.

* 1. **Question 12. 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 instance, 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 X=load ‘foo’ is not same as x=load ‘foo’. On the other hand, keywords in Apache Pig are case insensitive i.e. LOAD is same as load.

* 1. **Question 13. What Are Some Of The Apache Pig Use Cases You Can Think Of?**

**Answer :**

Apache Pig big data tools, is used in particular for iterative processing, research on raw data and for traditional ETL data pipelines. As Pig can operate in circumstances where the schema is not known, inconsistent or incomplete- it is widely used by researchers who want to make use of the data before it is cleaned and loaded into the data warehouse.

To build behavior prediction models, for instance, it can be used by a website to track the response of the visitors to various types of ads, images, articles, etc.

* 1. **Question 14. Differentiate Between Piglatin And Hiveql?**

**Answer :**

* + - 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. **Question 15. 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, i.e. it continues to work with data that does not live up to its expectations.

* 1. **Question 16. What Do You Understand By An Inner Bag And Outer Bag In Pig?**

**Answer :**

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

* 1. **Question 17. 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 more like a 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.

* 1. **Question 18. Explain The Difference Between Count\_star And Count Functions In Apache Pig?**

**Answer :**

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.

* 1. **Question 19. What Are The Various Diagnostic Operators Available In Apache Pig?**

**Answer :**

* + - **Dump Operator-** It is used to display the output of pig Latin statements on the screen, so that developers can debug the code.
    - **Describe Operator-**Explained in apache pig interview question no- 10
    - **Explain Operator-**Explained in apache pig interview question no -10
    - **Illustrate Operator-** Explained in apache pig interview question no -11
  1. **Question 20. How Will You Merge The Contents Of Two Or More Relations And Divide A Single Relation Into Two Or More Relations?**

**Answer :**

This can be accomplished using the UNION and SPLIT operators.

* 1. **Question 21. I Have A Relation R. How Can I Get The Top 10 Tuples From The Relation R.?**

**Answer :**

TOP () function returns the top N tuples from a bag of tuples or a relation. N is passed as a parameter to the function top () along with the column whose values are to be compared and the relation R.

* 1. **Question 22. What Are The Commonalities Between Pig And Hive?**

**Answer :**

* + - 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 queries.
  1. **Question 23. What Are The Different Types Of Udf’s In Java Supported By Apache Pig?**

**Answer :**

Algebraic, Eval and Filter functions are the various types of UDF’s supported in Pig.

* 1. **Question 24. 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?**

**Answer :**

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.

* 1. **Question 25. Explain About The Scalar Datatypes In Apache Pig.?**

**Answer :**

integer, float, double, long, bytearray and char array are the available scalar datatypes in Apache Pig.

* 1. **Question 26. How Do Users Interact With Hdfs In Apache Pig?**

**Answer :**

Using the grunt shell.

* 1. **Question 27. What Is The Use Of Having Filters In Apache Pig?**

**Answer :**

Just like the where clause in SQL, Apache Pig has filters to extract records based on a given condition or predicate. The record is passed down the pipeline if the predicate or the condition turn to true. Predicate contains various operators like ==, <=,!=, >=.

**Example:-**

X= load ‘inputs’ as(name,address)

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

* 1. **Question 28. What Is A Udf In Pig?**

**Answer :**

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.

* 1. **Question 29. Can You Join Multiple Fields In Apache Pig Scripts?**

**Answer :**

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.

* 1. **Question 30. Does Pig Support Multi-line Commands?**

**Answer :**

Yes.

| **Apache Pig Related Tutorials** | |
| --- | --- |
| [Apache Tapestry Tutorial](https://www.wisdomjobs.com/e-university/apache-tapestry-tutorial-1223.html) | [Apache Cassandra Tutorial](https://www.wisdomjobs.com/e-university/apache-cassandra-tutorial-1241.html) |
| [Apache Solr Tutorial](https://www.wisdomjobs.com/e-university/apache-solr-tutorial-1297.html) | [Apache Storm Tutorial](https://www.wisdomjobs.com/e-university/apache-storm-tutorial-1298.html) |
| [Apache Hive Tutorial](https://www.wisdomjobs.com/e-university/apache-hive-tutorial-1308.html) | [Apache Pig Tutorial](https://www.wisdomjobs.com/e-university/apache-pig-tutorial-1327.html) |
| [Apache Flume Tutorial](https://www.wisdomjobs.com/e-university/apache-flume-tutorial-1341.html) | [Apache Kafka Tutorial](https://www.wisdomjobs.com/e-university/apache-kafka-tutorial-1342.html) |
| [Apache Ant Tutorial](https://www.wisdomjobs.com/e-university/apache-ant-tutorial-1398.html) | [Apache Tajo Tutorial](https://www.wisdomjobs.com/e-university/apache-tajo-tutorial-1693.html) |
| [Apache Impala Tutorial](https://www.wisdomjobs.com/e-university/apache-impala-tutorial-1698.html) | [Apache Presto Tutorial](https://www.wisdomjobs.com/e-university/apache-presto-tutorial-1791.html) |
| [Apache POI PPT Tutorial](https://www.wisdomjobs.com/e-university/apache-poi-ppt-tutorial-1947.html) | [Apache Struts 2 Tutorial](https://www.wisdomjobs.com/e-university/apache-struts-2-tutorial-1973.html) |

TECHNICAL QUESTIONS

**SQOOP**

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

transfer data between Hadoop and relational database servers, Sqoop is the best tool

Moreover, Sqoop uses two main tools. Like:

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

**Que 1. Mention the best features of Apache Sqoop.**

**Ans.** Apache Sqoop is a tool in Hadoop ecosystem have several advantages. Like

1. Parallel import/export
2. Connectors for all major RDBMS Databases
3. Import results of SQL query
4. Incremental Load
5. Full Load
6. Kerberos Security Integration
7. Load data directly into [**Hive**](https://data-flair.training/blogs/apache-hive-tutorial/) / [**HBase**](https://data-flair.training/blogs/hadoop-hbase-tutorial/)
8. Compression
9. Support for Accumulo

**Que 38. 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.

**Que 39. What is the default extension of the files produced from a sqoop import using the –compress parameter?**

**Ans.** .gz,tar

**Que 43. What are the basic commands in Apache Sqoop and its uses?**

**Ans.** The basic commands of Apache Sqoop are:  
[**Codegen**](https://data-flair.training/blogs/sqoop-codegen/), Create-hive-table, [**Eval**](https://data-flair.training/blogs/sqoop-eval/), [**Export**](https://data-flair.training/blogs/sqoop-export/), Help,[**Import**](https://data-flair.training/blogs/sqoop-import/), [**Import-all-tables**](https://data-flair.training/blogs/sqoop-import-all-tables/),[**List-databases**](https://data-flair.training/blogs/sqoop-list-databases/), [**List-tables**](https://data-flair.training/blogs/sqoop-list-tables/), Versions.  
Moreover, 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

**Que 44. 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.  
Follow this link to know more about [**Sqoop**](https://data-flair.training/blogs/sqoop-introduction/)

**Que 50. Difference Between Apache Sqoop vs Flume.**

**Ans.** So, let’s discuss all the differences on the basis of features.

**a. Data Flow**  
Apache Sqoop – Basically, [**Sqoop**](http://sqoop.apache.org/) works with any type of relational database system (RDBMS) that has the basic JDBC connectivity. Also, Sqoop can import data from NoSQL databases like MongoDB, Cassandra and along with it. Moreover, it allows data transfer to Apache Hive or HDFS.  
[Apache Flume](https://data-flair.training/blogs/apache-flume-tutorial/)**–**Likewise, Flume works with streaming data sources those are generated continuously in Hadoop environments. Like log files.

**b. Type of Loading**  
Apache Sqoop – Basically,  Sqoop load is not driven by events.  
Apache Flume – Here, data loading is completely event-driven.

**c. When to use**  
Apache Sqoop – However, if the data is being available in Teradata, Oracle, MySQL, PostreSQL or any other JDBC compatible database it is considered an ideal fit.  
Apache Flume – While we move bulk of streaming data from sources likes JMS or spooling directories, it is the best choice.

**d. Link to HDFS**  
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/)

|  |  |
| --- | --- |
| **Sqoop** | **Flume** |
| Used for importing data from structured data sources like RDBMS. | Used for moving bulk streaming data into HDFS. |
| It has a connector based architecture. | It has a agent based architecture. |
| Data import in sqoop is not event driven. | Data load in flume is event driven |
| HDFS is the destination for importing data. | Data flows into HDFS through one or more channels. |
| **Sqoop vs Flume** | |

### ****2) What is the default file format to import data using Apache Sqoop?****

Sqoop allows data to be imported using two file formats

#### **i) Delimited Text File Format**

This is the default file format to import data using Sqoop. This file format can be explicitly specified using the –as-textfile argument to the import command in Sqoop. Passing this as an argument to the command will produce the string based representation of all the records to the output files with the delimited characters between rows and columns.

#### **ii) Sequence File Format**

It is a binary file format where records are stored in custom record-specific data types which are shown as Java classes. Sqoop automatically creates these data types and manifests them as java classes.

### ****3) 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 by specifying the exclude-tables option with it as follows-

sqoop import-all-tables

--connect –username –password --exclude-tables Table298, Table 123, Table 299

**Q2) How can you import large objects like BLOB and CLOB in Sqoop?**  
The direct import function is not supported by Sqoop in case of CLOB and BLOB objects. Hence, if you have to import large purposes, you can use JDBC based imports. This can be done without introducing the direct argument of the import utility.

**Q3) What is the default database of Apache Sqoop?**  
The default database of Apache Sqoop is MySQL.

**Q4) Describe the process of executing a free-form SQL query to import rows**  
To achieve a free-form SQL query, you have to use the –m1 option. This would create only one Mapreduce task. This would then import the rows directly.

**Related Article:** [Evaluating Performance of Distributed Systems with MapReduce](https://mindmajix.com/mapreduce/evaluating-performance-of-distributed-systems-with-mapreduce)

**Q5) Describe the importance of using –compress-codec parameter**  
The –compress-codec parameter can be used to get the export file of the Sqoop import in the mentioned formats.

**Q6) What is the significance of Eval tool?**  
Sqoop Eval would help you to make use of the sample SQL queries. This can be against the database as it can preview the results that are displayed on the console. Interestingly, with the help of the Eval tool, you would be well aware of the fact that the desired data can be imported correctly or not.

**Q7) What is the meaning of Free form import in Sqoop?**  
With the use of Sqoop, one can import the relational database query. This can be done using column and table name parameters.

**Q8) 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.

**Q9) 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.

**Q10) What is the meaning of Input Split in Hadoop?**  
Input Split is that kind of a function which is associated with splitting the input files into various chunks. These chunks can also assign each split to a mapper in the ongoing process of data correction.

**11) Illustrate the utility of the Help Command in Sqoop**  
The help command in Sqoop can be utilized to list the various available commands.

**12) Shed light on the service of Codegen command in Sqoop**  
The Codegen command is associated with the generation of code so that it can appropriately interact with the database records.

**Q13) 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.

**Related Article:** [An Overview Of Hadoop Hive](https://mindmajix.com/hadoop/an-overview-of-hadoop-hive)

**Q14) Illustrate on the process of listing all the columns of a table with the help of Apache Sqoop**  
To contain all the columns, you do not have any direct command like the Sqoop indexed columns. However, you can also indirectly achieve this. You can do that by retrieving the columns of the desired tables. After that, you can redirect them to a set of files that can be viewed in a standard manner. This also contains the columns of a particular table.

**Q15) What is the default file format in order to import data with the utilization of Apache Sqoop?**  
At the time of answering this question, you should know that there are two file formats that can be used in the case of importing data. These are as follows:

**Sequencing the file format**

It is a commonly observed fact that a sequence file format is also known by the name of binary file format. The records of these binary file formats are usually stored in the custom record data types. Moreover, Sqoop can automatically create a varied data types and also manifests them in the form of Java classes.

**Delimiting the text file format**

This is the usual file format in importing data. Additionally, it can be said that in order to avail the import command in Sqoop, this file format can be specified. You can specify the file format with the use of text file argument command. On the other hand, when you pass this argument, you would produce a string-based representation of varied types of records. You can also create the output files with the use of delimited characters between columns and rows.

[Check Out Hadoop Tutorials](https://mindmajix.com/hadoop)

**Q16) List all the basic commands in Apache Sqoop along with their applications**  
The basic controls in Apache Sqoop along with their uses are:

**1. Export:** This function helps to export the HDFS directory into a database table  
**2. List Tables:**This function would help the user to list all tables in a particular database.  
**3. Codegen:** This function would help you to generate code so that you can interact with varied types of database records.   
**4. Create:** This function allows a user to import the table definition within the hive of databases.  
**5. Eval:**This function would always help you to assess the SQL statement and display the results.  
**6. Version:** This function would help you to depict the information related to the text of the database.   
**7. Import all tables:** This function would help a user to import all the tables from a database to HDFS.  
**8. List all the databases:**This function would assist a user to create a list of the available databases on a particular server.

**Related Article:** [What is Apache Hadoop Sqoop](https://mindmajix.com/hadoop/what-is-apache-hadoop-sqoop)

**Q17) What is the meaning of Sqoop Validation?**  
It refers to the manner in which data validation happens when it is copied. It can also be executed by either exporting or importing the data. It can also be done with the help of a basic comparison between the row counts from the source. You can also opt to use the option to make sure that you are comparing the row counts between the target as well as the source. During the time of the imports, all the rows can be deleted and added. In this context, it is important to note that during the whole process, Sqoop keeps a tab on the changes that have been affected.

**Q18) Give a basic introduction to Sqoop**  
When it comes to transfer data between relational database servers and Hadoop, you should know that Sqoop is one of the best tools. In order to be more specific, you should use it in importing data from various types of relational databases. It is important for you to note that you can import data from varied types of databases such as MySQL, HDFS, and Hadoop. It is also interesting to note that you have the option to export data from the Hadoop file with the help of Sqoop. This functionality is being provided by the Apache Software Foundation.

It is also important to mention that Sqoop utilizes two main tools. They are in the form of Sqoop export and Sqoop import. With the help of these two tools, you can now extract data information form varied types of databases.

**Related Article:**[Difference between HBase and RDBMS – Hadoop](https://mindmajix.com/hadoop/difference-between-hbase-rdbms)

**Q19) What are the limitations of importing the RDBMS tables into the Hcatalog directly?**  
In order to import the tables into the Hcatalog in a direct manner, you have to make sure that you are using the –Hcatalog database option. However, in this process, you would face a limitation of importing the tables. It is in the form of the fact that this option do not supports a plethora of arguments like –direct, –as-Avro file and -export-dir.

**Q20) 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 unique fashion. A majority of the columns are used in the Where clause of the update query that has been already been generated. Moreover, all the other types of table columns should be used in the SET portion of the generated query.

**Q21) What is the significance of the Sqoop Import Mainframe tool? Shed light on its purpose too**  
The Sqoop Import Mainframe tool can also be used to import all the important datasets which lies in a partitioned dataset. The partitioned dataset is also known as PDS. The PDS is also known to a directory on varied types of open systems. It is important for you to note that in a dataset, the various types of records would be stored as a single text field with the help of the entire record. This tool would always help you to make sure that you are importing the right types of data tools and that too in a proper manner.

**Q22) 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.

**Q23) 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.

**Q24) Describe the practicality of opting for Sqoop nowadays**  
Apache Sqoop is regarded as an excellent help for those individuals who face challenges in transferring data out of the data warehouse. It is also used for importing data from RDBMS to HDFS. With the help of Sqoop, the users can also import more than one table. Interestingly, with the use of Apache Sqoop, the data selected columns can be easily exported. Furthermore, Sqoop is also compatible with a majority of JDBC databases. Here is the list of questions which would help you to crack the Sqoop interview.

**1. Compare Sqoop and Flume**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Sqoop** | **Flume** |
| Application | Importing data from RDBMS | Moving bulk streaming data into HDFS |
| Architecture | Connector  – connecting to respective data | Agent – fetching of the right data |
| Loading of data | Event driven | Not event driven |

**2. Name a few import control commands. How can Sqoop handle large objects?**

Import control commands are used to import RDBMS data

**Append:** Append data to an existing dataset in HDFS. –append

**Columns:** columns to import from the table. –columns  
<col,col……> • Where: where clause to use during import. —

where The common large objects are Blog and Clob.Suppose the object is less than 16 MB, it is stored inline with the rest of the data. If there are big objects, they are temporarily stored in a subdirectory with the name \_lob. Those data are then materialized in memory for processing. If we set lob limit as ZERO (0) then it is stored in external memory.

**3. How can we import data from particular row or column? What is the destination types allowed in Sqoop import command?**

Sqoop allows to Export and Import the data from the data table based on the where clause. The syntax is

--columns

<col1,col2……> --where

--query

Example:

sqoop import –connect jdbc:mysql://db.one.com/corp --table INTELLIPAAT\_EMP --where “start\_date> ’2016-07-20’ ”

sqoopeval --connect jdbc:mysql://db.test.com/corp --query “SELECT \* FROM intellipaat\_emp LIMIT 20”

sqoop import –connect jdbc:mysql://localhost/database --username root --password aaaaa –columns “name,emp\_id,jobtitle”

Sqoop supports data imported into following services:

* HDFS
* Hive
* Hbase
* Hcatalog
* Accumulo

Learn about the complete Hadoop ecosystem in this [blog post](https://intellipaat.com/blog/hadoop-ecosystem-of-india/).

Learn Hadoop in 85 hrs from experts

Top of Form



Bottom of Form

**4. Role of JDBC driver in sqoop setup? Is the JDBC driver enough to connect the sqoop to the database?**

Sqoop needs a connector to connect the different relational databases. Almost all Database vendors make a JDBC connector available specific to that Database, Sqoop needs a JDBC driver of the database for interaction.  
No, Sqoop needs JDBC and a connector to connect a database.

**Interested in learning Sqoop? Well, we have the comprehensive**[*Training Course*](https://intellipaat.com/big-data-hadoop-training/#curriculum)**to give you a head start in your career.**

**5. Using Sqoop command how can we control the number of mappers?.**

We can control the number of mappers by executing the parameter –num-mapers in sqoop command. The –num-mappers arguments control the number of map tasks, which is the degree of parallelism used. Start with a small number of map tasks, then choose a high number of mappers starting the performance may down on the database side.

**Syntax:** -m, –num-mappers

**6.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 uniquely identifies a row. All of these columns are used in the WHERE clause generated UPDATE query. All other table columns will be used in the SET part of the query.  
The command below is used to show all the databases in MySQL server.

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

[**Wish to Learn Sqoop? Click Here**](https://intellipaat.com/big-data-hadoop-training/#course-content)

**7. Define Sqoop metastore? What is the purpose of Sqoop-merge?**

Sqoop meta store is a tool for using hosts in a shared metadata repository. Multiple users and remote users can define and execute saved jobs defined in metastore. End users configured to connect the metastore in sqoop-site.xml or with the

–meta-connect argument.

**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.

**8. Explain the saved job process in Sqoop.**

Sqoop allows us to define saved jobs which make this process simple. A saved job records the configuration information required to execute a Sqoop command at a later time. sqoop-job tool describes how to create and work with saved jobs. Job descriptions are saved to a private repository stored in $HOME/.sqoop/.

We can configure Sqoop to instead use a shared metastore, which makes saved jobs offered to multiple users across a shared cluster. Starting the metastore is covered by the section on the sqoop-metastore tool.

**9. How Sqoop word came ? Sqoop is which type of tool and the main use of sqoop?**

Sqoop word came from SQL+HADOOP=SQOOP. And Sqoop is a data transfer tool.  
The main use of Sqoop is to import and export the large amount of data from RDBMS to HDFS and vice versa.

Become Hadoop Certified in 85 hrs.

**GET CERTIFIED**

**10. 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.

**11. I am getting connection failure exception during connecting to Mysql through Sqoop, what is the root cause and fix for this error scenario?**

This will happen when there is lack of permissions to access our Mysql database over the network. We can try the below command to confirm the connect to Mysql database from aSqoop client machine.  
$ mysql –host=MySqlnode> –database=test –user= –password=  
We can grant the permissions with below commands.

mysql> GRANT ALL PRIVILEGES ON \*.\* TO ‘%’@’localhost’;

mysql> GRANT ALL PRIVILEGES ON \*.\* TO ‘ ’@’localhost’;

Give your career a big boost by going through our Apache [Sqoop Training Course](https://intellipaat.com/big-data-hadoop-training/#sample-video) now!

**12. I am getting java.lang.IllegalArgumentException: during importing tables from oracle database.what might be the root cause and fix for this error scenario?**

Sqoop commands are case- sensitive of table names and user names.  
By specifying the above two values in UPPER case, it will resolve the issue.  
In case, the source table is created under different user namespace,then table name should be like USERNAME.TABLENAME as shown below  
sqoop import  
–connect jdbc:oracle:thin:@intellipaat.testing.com/INTELLIPAAT  
–username SQOOP  
–password sqoop  
–table COMPANY.EMPLOYEES

**13. 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’ –query “SELECT column\_name,DATA\_TYPE FROM INFORMATION\_SCHEMA columns WHEREtable\_name=’mytableofinterest’ AND \$CONDITIONS” –target-dir ‘mytableofinterest\_column\_name’.

**14. How to create a table in Mysql and how to insert the values into the table ?**

To create a table in mysql using the below command

mysql> create table tablename( col1 datatype, col2 datatype,…………);

Example –

mysql> create table INTELLIPAAT(emp\_idint,emp\_namevarchar(30),emp\_salint);

Insert the values into the table

mysql> insert into table name(value1,value2,value3,………);

Example-

mysql> insert into INTELLIPAAT(1234,’aaa’,20000);

mysql> insert into INTELLIPAAT(1235,’bbb’,10000);

mysql> insert into INTELLIPAAT(1236,’ccc’,15000);

**15. What are the basic commands in Hadoop Sqoop and its uses?**

The basic commands of HadoopSqoop are

* Codegen, Create-hive-table, Eval, Export, Help, Import, Import-all-tables, List-databases, List-tables,Versions.
* Useof HadoopSqoop basic commands
* Codegen- It helps to generate code to interact with database records.
* Create-hive-table- It helps to Import a table definition into a hive
* Eval- It helps to evaluateSQL statement and display the results
* Export-It helps to export an HDFS directory into a database table
* Help- It helps to list the available commands
* Import- It helps to import a table from a database to HDFS
* Import-all-tables- It helps to import tables from a database to HDFS
* List-databases- It helps to list available databases on a server
* List-tables-It helps to list tables in a database
* Version-It helps to display the version information

**16. Is sqoop same as to distcp in hadoop?**

No. Because the only distcp import command is same as Sqoop import command and both the commands submit parallel map-only jobs but both command functions are different. Distcp is used to copy any type of files from Local filesystem to HDFS and Sqoop is used for transferring the data records between RDBMS and Hadoop eco- system service.

**17. 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.

**18. 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.

**19. 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

**20. 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.

**Take charge of your career by going through this professionally designed Apache**[*Hadoop Developer Course*](https://intellipaat.com/big-data-hadoop-training/#certification)**.**

**Q1 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.

**Q2 How Sqoop can be used in a Java program?**

Answer: The Sqoop jar in classpath should be included in the java code. After this the method Sqoop.runTool () method must be invoked. The necessary parameters should be created to Sqoop programmatically just like for command line.

**Q3 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.

**Q4 How are large objects handled in Sqoop?**

Answer: Sqoop provides the capability to store large sized data into a single field based on the type of data. Sqoop supports the ability to store-

* CLOB ‘s – Character Large Objects
* BLOB’s –Binary Large Objects

Large objects in Sqoop are handled by importing the large objects into a file referred as “LobFile” i.e. Large Object File. The LobFile has the ability to store records of huge size, thus each record in the LobFile is a large object.

**Q5 What is a disadvantage of using –direct parameter for faster data load by sqoop?**

Answer: The native utilities used by databases to support faster load do not work for binary data formats like Sequence File.

**Q6 Is it possible to do an incremental import using Sqoop?**

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

* Append
* 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.

**Q7 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;

**Q8 How can you control the number of mappers used by the sqoop command?**

Answer: The Parameter –num-mappers is used to control the number of mappers executed by a sqoop command. We should start with choosing a small number of map tasks and then gradually scale up as choosing high number of mappers initially may slow down the performance on the database side.

**Q9 What is the standard location or path for Hadoop Sqoop scripts?**

Answer: /usr/bin/Hadoop Sqoop.

**Q10 How can we import a subset of rows from a table without using the where clause?**

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

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

**Q11 When the source data keeps getting updated frequently, what is the approach to keep it in sync with the data in HDFS imported by sqoop?**

Answer: qoop 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.

**Q12 What is a sqoop metastore?**

Answer: 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.

**Q13 Can free form SQL queries be used with Sqoop import command? If yes, then how can they be used?**

Answer: Sqoop allows us to use free form SQL queries with the import command. The import command should be used with the –e and – query options to execute free form SQL queries. When using the –e and –query options with the import command the –target dir value must be specified.

**Q14 Tell few import control commands:**

Answer: –Append

–Columns

–Where

These command are most frequently used to import RDBMS Data.

**Q15 Can free form SQL queries be used with Sqoop import command? If yes, then how can they be used?**

Answer: Sqoop allows us to use free form SQL queries with the import command. The import command should be used with the –e and – query options to execute free form SQL queries. When using the –e and –query options with the import command the –target dir value must be specified.

**Q16 How can you see the list of stored jobs in sqoop metastore?**

Answer: sqoop job –list

**Q17 What type of databases Sqoop can support?**

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

**Q18 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.

**Q19 HOw sqoop can handle large objects?**

Answer: Blog and Clob columns are common large objects. If the object is less than 16MB, it stored inline with the rest of the data. If large objects, temporary stored in\_lob subdirectory. Those lobs processes in a streaming fashion. Those data materialized in memory for processing. IT you set LOB to 0, those lobs objects placed in external storage.

**Q20 What is the importance of eval tool?**

Answer: It allows user to run sample SQL queries against Database and preview the results on the console. It can help to know what data can import? The desired data imported or not?

**Q21 What is the default extension of the files produced from a sqoop import using the –compress parameter?**

Answer: .gz

**Q22 Can we import the data with “Where” condition?**

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

**Q23 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-sequencefile, -target-dir , -export-dir are not supported.

**Q24 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.

**Q25 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.

**Q26 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

**Q27 What are the two file formats supported by sqoop for import?**

Answer: Delimited text and Sequence Files.

**Q28 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.

**Q29 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.

**Q30 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.

**Q31 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.

**Q32 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.

**Q33 What is Sqoop?**

Answer: Sqoop is an open source project that enables data transfer from non-hadoop source to hadoop source. It can be remembered as SQL to Hadoop -> SQOOP. It allows user to specify the source and target location inside the Hadoop.

**Q34 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.

**Q35 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.

**Q36 Which database the sqoop metastore runs on?**

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

**Q37 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 uniquely identifies a row. All of these columns is used in the WHERE clause of the generated UPDATE query. All other table columns will be used in the SET part of the query.

**Q38 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.

**Q39 Where can the metastore database be hosted?**

Answer: The metastore database can be hosted anywhere within or outside of the Hadoop cluster.

**Q40 Which is used to import data in Sqoop ?**

Answer: In SQOOP import command is used to import RDBMS data into HDFS. Using import command we can import a particular table into HDFS.

**Q41 What is the role of JDBC driver in a Sqoop set up?**

Answer: To connect to different relational databases sqoop needs a connector. Almost every DB vendor makes this connecter available as a JDBC driver which is specific to that DB. So Sqoop needs the JDBC driver of each of the database it needs to interact with.

**Q42 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?**

Answer: By using the lastmodified mode. Rows where the check column holds a timestamp more recent than the timestamp specified with –last-value are imported.

**Q43 What is InputSplit in Hadoop?**

Answer: When a hadoop job is run, it splits input files into chunks and assign each split to a mapper to process. This is called Input Split.

**Q44 Hadoop sqoop word came from ?**

Answer: Sql + Hadoop = sqoop

**Q45 What is the work of Export In Hadoop sqoop ?**

Answer: Export the data from HDFS to RDBMS

**Q46 Use of Codegen command in Hadoop sqoop ?**

Answer: Generate code to interact with database records

**Q47 Use of Help command in Hadoop sqoop ?**

Answer: List available commands

**Q48 How can you schedule a sqoop job using Oozie?**

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

**Q49 What are the two file formats supported by sqoop for import?**

Answer: Delimited text and Sequence Files.

**Q50 What is a sqoop metastore?**

Answer: 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.

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

What is the role of JDBC driver in a Sqoop set up?

To connect to different relational databases sqoop needs a connector. Almost every DB vendor makes this connecter available as a JDBC driver which is specific to that DB. So Sqoop needs the JDBC driver of each of the database it needs to inetract with.

What is the role of JDBC driver in a Sqoop set up?

To connect to different relational databases sqoop needs a connector. Almost every DB vendor makes this connecter available as a JDBC driver which is specific to that DB. So Sqoop needs the JDBC driver of each of the database it needs to inetract with.

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.

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

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

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

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.

What is the default extension of the files produced from a sqoop import using the --compress parameter?

.gz

What is the significance of using --compress-codec parameter?

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

What is a disadvantage of using --direct parameter for faster data load by sqoop?

The native utilities used by databases to support faster laod do not work for binary data formats like SequenceFile.

How can you control the number of mappers used by the sqoop command?

The Parameter --num-mapers is used to control the number of mappers executed by a sqoop command. We should start with choosing a small number of map tasks and then gradually scale up as choosing high number of mappers initially may slow down the performance on the database side.

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.

When the source data keeps getting updated frequently, 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.

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.

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

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

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

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.

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

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 \

--query '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.

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

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.

How will you update the rows that are already exported?How can you sync a exported table with HDFS data in which some rows are deleted?How can you export only a subset of columns to a relational table using sqoop?

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

How can we load to a column in a relational table which is not null but the incoming value from HDFS has a null value?

By using the –input-null-string parameter we can specify a default value and that will allow the row to be inserted into the target table.

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.What do you mean by Free Form Import in Sqoop?How can you force sqoop to execute a free form Sql query 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. What can be the reason?What is the importance of --split-by clause in running parallel import tasks in sqoop?What does this sqoop command achieve?

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

What happens when a table is imported into a HDFS directory which already exists using the –apend parameter?How can you control the mapping between SQL data types and Java types?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?What are the two file formats supported by sqoop for import?Give a sqoop command to import the columns employee\_id,first\_name,last\_name from the MySql table EmployeeGive a sqoop command to run only 8 mapreduce tasks in parallel

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

-m 8

What does the following query do?

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

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

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 query do?

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

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

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.

What is the purpose of sqoop-merge?

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.

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

sqoop job –list

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

Sqoop job –show myjob

Which database the sqoop metastore runs on?

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

Where can the metastore database be hosted?

The metastore database can be hosted anywhere within or outside of the Hadoop cluster..

### ****Sqoop Interview Questions and Answers for Experienced****

###### **1. What is Sqoop?**

Sqoop is an open source tool that enables users to transfer bulk data between Hadoop eco system and relational databases.

###### **2. What are the relational databases supported in Sqoop?**

Below are the list of RDBMSs that are supported by Sqoop Currently.

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

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

Currently Sqoop Supports data imported into below services.

* HDFS
* Hive
* HBase
* HCatalog
* Accumulo

###### **4. 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.

###### **5. What are the majorly used commands in Sqoop?**

In Sqoop Majorly**Import** and **export** commands are used. But below commands are also useful some times.

* codegen
* eval
* import-all-tables
* job
* list-databases
* list-tables
* merge
* metastore

###### **6. When Importing tables from MySQL to what are the precautions that needs to be taken care w.r.t to access?**

In MySQL, we need to make sure that we have granted all privileges on the databases, that needs to be accessed, should be given to all users at destination hostname. If Sqoop is being run under localhost and MySQL is also present on the same then we can grant the permissions with below two commands from MySQL shell logged in with ROOT user.



|  |  |
| --- | --- |
| 1  2  3  4 | $ mysql -u root -p  mysql> GRANT ALL PRIVILEGES ON \*.\* TO '%'@'localhost';  mysql> GRANT ALL PRIVILEGES ON \*.\* TO ''@'localhost'; |

###### **7. What if my MySQL server is running on MachineA and Sqoop is running on MachineB for the above question?**

From MachineA login to MySQL shell and perform the below command as root user. If using hostname of second machine, then that should be added to /etc/hosts file of first machine.



|  |  |
| --- | --- |
| 1  2  3  4 | $ mysql -u root -p  mysql> GRANT ALL PRIVILEGES ON \*.\* TO '%'@'MachineB hostname or Ip address';  mysql> GRANT ALL PRIVILEGES ON \*.\* TO ''@'MachineB hostname or Ip address'; |

###### **8. 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.

###### **9. 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.

###### **10. 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 unique key column or set of columns to form unique values and these should be provided to **-split-by** column argument.

###### **11. 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.

###### **12. 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 |

###### **13. 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 Question 6 in this post.

###### **14. 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 |

**Q1. What is Sqoop ?**  
Sqoop is a tool designed to transfer data between Hadoop and relational database servers. It is used to import data from relational databases such as MySQL, Oracle to Hadoop HDFS, and export from Hadoop file system to relational databases.

**Q2. What is Sqoop metastore?**  
Sqoop metastore is a shared metadata repository for remote users to define and execute saved jobs created using sqoop job defined in the metastore. The sqoop –site.xml should be configured to connect to the metastore.

**Q3. What are the two file formats supported by sqoop for import?**  
Delimited text and Sequence Files.

**Q4. What is the difference between Sqoop and DistCP command in Hadoop?**  
Both distCP (Distributed Copy in Hadoop) and Sqoop transfer data in parallel but the only difference is that distCP command can transfer any kind of data from one Hadoop cluster to another whereas Sqoop transfers data between RDBMS and other components in the Hadoop ecosystem like HBase, Hive, HDFS, etc.

**Q5. Compare Sqoop and Flume**  
**Sqoop vs Flume**

|  |  |
| --- | --- |
| ****Sqoop**** | ****Flume**** |
| Used for importing data from structured data sources like RDBMS. | Used for moving bulk streaming data into HDFS. |
| It has a connector based architecture. | It has a agent based architecture. |
| Data import in sqoop is not evetn driven. | Data load in flume is event driven |
| HDFS is the destination for importing data. | Data flows into HDFS through one or more channels. |

**Q6. What do you mean by Free Form Import in Sqoop?**  
Sqoop can import data form a relational database using any SQL query rather than only using table and column name parameters.

**Q7. Does Apache Sqoop have a default database?**  
Yes, MySQL is the default database

**Q8. How can you execute a free form SQL query in Sqoop to import the rows in a sequential manner?**  
This can be accomplished using the –m 1 option in the Sqoop import command. It will create only one MapReduce task which will then import rows serially.

**Q9.  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 by specifying the exclude-tables option with it as follows-

sqoop import-all-tables

–connect –username –password –exclude-tables Table298, Table 123, Table 299

**Q10. How can I import large objects (BLOB and CLOB objects) in Apache Sqoop?**  
Apache Sqoop import command does not support direct import of BLOB and CLOB large objects. To import large objects, I Sqoop, JDBC based imports have to be used without the direct argument to the import utility.

**Q11. How will you list all the columns of a table using Apache Sqoop?**  
Unlike sqoop-list-tables and sqoop-list-databases, there is no direct command like sqoop-list-columns to list all the columns. The indirect way of achieving this is to retrieve the columns of the desired tables and redirect them to a file which can be viewed manually containing the column names of a particular table.

Sqoop import –m 1 –connect ‘jdbc: sqlserver: //nameofmyserver; database=nameofmydatabase; username=DeZyre; password=mypassword’ –query “SELECT column\_name, DATA\_TYPE FROM INFORMATION\_SCHEMA.Columns WHERE table\_name=’mytableofinterest’ AND \$CONDITIONS” –target-dir ‘mytableofinterest\_column\_name’

**Q12. 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.

**Q13. 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.

**Q14. What is the default file format to import data using Apache Sqoop?**  
Sqoop allows data to be imported using two file formats

**i) Delimited Text File Format**

This is the default file format to import data using Sqoop. This file format can be explicitly specified using the –as-textfile argument to the import command in Sqoop. Passing this as an argument to the command will produce the string based representation of all the records to the output files with the delimited characters between rows and columns.

**ii) Sequence File Format**

It is a binary file format where records are stored in custom record-specific data types which are shown as Java classes. Sqoop automatically creates these data types and manifests them as java classes.

**Q15. What are the basic commands in HadoopSqoop and its uses?**  
The basic commands of HadoopSqoop are

* + Codegen, Create-hive-table, Eval, Export, Help, Import, Import-all-tables, List-databases, List-tables,Versions.
  + Useof HadoopSqoop basic commands
  + Codegen- It helps to generate code to interact with database records.
  + Create-hive-table- It helps to Import a table definition into a hive
  + Eval- It helps to evaluateSQL statement and display the results
  + Export-It helps to export an HDFS directory into a database table
  + Help- It helps to list the available commands
  + Import- It helps to import a table from a database to HDFS
  + Import-all-tables- It helps to import tables from a database to HDFS
  + List-databases- It helps to list available databases on a server
  + List-tables-It helps to list tables in a database
  + Version-It helps to display the version information

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.

**Q16. You successfully imported a table using Apache Sqoop to HBase but when you query 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.

**Q17. 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.

**Q18. 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.

**Q19. 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.

**Q20.  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

**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.

**Q21. 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

**Q22.  You use –split-by clause but it still does not give optimal performance then how will you improve the performance further.**  
Using the –boundary-query clause. Generally, sqoop uses the SQL query select min (), max () from to find out the boundary values for creating splits. However, if this query is not optimal then using the –boundary-query argument any random query can be written to generate two numeric columns.

**Q23. 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.

**Q24. How can you see the list of stored jobs in sqoop metastore?**  
sqoop job –list

**Q25. Give a sqoop command to import data from all tables in the MySql DB DB1.**  
sqoop –tables –connect jdbc:mysql://host/DB1

**Q26. Where can the metastore database be hosted?**  
The metastore database can be hosted anywhere within or outside of the Hadoop cluster..

1. **The first and best function of Sqoop?**
   1. Sqoop can import individual tables or entire databases into HDFS. The data is stored in the native directories and files in the HDFS file system.
2. **What is export in Sqoop?**
   1. Sqoop can export data directly from HDFS into a relational database using a target table definition based on the specifics of the target database.
3. **Why Sqoop uses mapreduce in import/export operations?**
   1. Sqoop uses MapReduce to import and export the data, which provides parallel operation as well as fault tolerance.
4. **What does it mean non-Hadoop data?**
   1. It is one of the most popular Big Data tools that leverages the competency to haul out data from a non-Hadoop data store by transforming information into a form that can be easily accessed and used by Big Data Hadoop
5. **What are the underlying technologies in Sqoop?**
   1. It has a high degree of dependence on underlying technologies like HDFS and MapReduce.
6. **What is the data loading or import in Sqoop?**
   1. Load directly into Hive tables, creating HDFS files in the background and the Hive metadata automatically
7. **Sqoop imports data into three kinds of data storage what are those?**
   1. Hive Tables
   2. HDFS files
   3. Hbase (HBase is an open-source, distributed, versioned, non-relational database modeled after Google’s Bigtable)
8. **The nine functions of Sqoop?**
   1. Full Load
   2. Incremental Load
   3. Parallel import/export
   4. Import results of SQL query
   5. Compression
   6. Connectors for all major RDBMS Databases
   7. Kerberos Security Integration
   8. Load data directly into Hive/Hbase
   9. Support for Accumulo
9. **Where Sqoop come into picture in Hadoop eco system?**
   1. Sqoop comes into picture. Sqoop acts like a intermediate layer between Hadoop and relational database systems
10. **Whether Sqoop will do aggregations?**
    1. Sqoop just imports and exports the data; it does not do any aggregations.

**What is sqoop and What does sqoop stand for?**

Sqoop stands for SQL to Hadoop and Hadoop to SQL. The Apache Sqoop is a component within the Hadoop ecosystem. It helps in the movement of data between the different relational database servers in the Hadoop file structure.  It can pull data from other relational databases such as Oracle to the Hadoop file system.

**What is sqoop used for?**

The primary task of Sqoop is to transfer bulk information between relational database servers. Sqoop helps the developers to handle these huge bulks of data which in turn helps them to gather relevant insights from the data. It eases up the process and allows better export and import of information.

**What is the default database of Sqoop?**

The default database of Sqoop is MySql.

**How to check sqoop version?**

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

sqoop version

**How to set a number of mappers in Apache sqoop?**

Sqoop controls the number of mappers that are currently accessing the RDBMS to avoid the DOS attacks on the entire system. This number of mappers can be set using the following command:

clause-m **or** numb-mappers.

For example,  numb-mappers 10.

**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**.

**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.

**What is sqoop eval tool?**

The primary task of the Sqoop Eval is to let the users execute simple database queries. Both DDL and DML statements can be executed easily using Sqoop Eval.  

**Why sqoop is used in Hadoop?**

Sqoop is used to import datasets into the HDFS. It can also be used for export datastores from Hadoop to external databases. Sqoop is easier to use while transferring data for the developers, as it automates the process. It makes use of the MapReduce framework to transfer data, that provides fault tolerance. It is easy to use Sqoop as it has a command based interface.

**What is boundary query in sqoop?**

The valid query required to obtain the minimum and maximum values for the column split using split by command can be mentioned using the boundary query parameter.

**How to import multiple tables in sqoop?**

The import all tables tool can be used to import multiple tables in Sqoop. The following syntax has to be used:

$ sqoop **import**-all-tables (generic-args) (**import**-args)

**However, to use the tool successfully, the following conditions have to be met:**

* The tables should have a single column primary key.
* All the columns of each table must be imported.
* Using conditions having a WHERE clause and non-default splitting columns must be avoided.

**How to change sqoop date format?**

The date format is Sqoop can be altered by the SET DATEFORMAT 'ymd' just after starting the connection.

**What is the difference between Sqoop and Flume?**

**The difference between Sqoop and Flume is depicted below:**

| **Sqoop** | **Flume** |
| --- | --- |
| It is used for importing and exporting datastores from relational databases. | It is used for pulling logs and crash reports into the Hadoop file system. |
| It does not have event-driven loading. | It has event-driven loading. |
| It has an architecture based on connectors. | It has an architecture based on agents, where the agent code is used to pull the data. |
| Used to move data faster to produce analytic results. | It is used when the analysis of patterns and root causes is necessary for corporations. |

**What happens when Sqoop fails in between the large data transfer job?**

During data transfer between two databases, if the job fails, some portion of the data might get saved in the database. This can be solved by using the staging table option.

**What is the use of split by in sqoop?**

The split by command is used to mention the column that will be used to split the data that is being imported while data transfer. Based on certain column values, Sqoop uses split by to split the data.

**How to use split by in sqoop?**

The split by command can be used in Sqoop using the following syntax:

--split-by student.id

This command will split the data while importing based on the student.id values.

**What is accumulo in sqoop?**

The Apache Acccumulo is a key value store that is developed over the Hadoop File system. The handling of larger sets of data is easier with Accumulo. The data in Accumulo is stored in the Haddop File system. In Sqoop, there is a functionality that allows users to directly import a table from the Accumulo data store.

**How to grant access on password file in Sqoop?**

The password file option can be used to access the protected file. The file can be present in the HDFS directory or it can be kept in the system where the command is being executed. The following code will be needed:

$**echo** -n "password" > /etc/sqoop/conf/passwords/mysql-password.txt

$**chmod** 400 /etc/sqoop/conf/passwords/mysql-password.txt

$**sqoop** import --connect jdbc:netezza://localhost/MYDB

--**username** testuser

--**table** ORDERS

--**password**-file /etc/sqoop/conf/passwords/mysql-password.txt

**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 subsequent incremental imports to ensure that updated information is imported.

**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.

**Define free form import in sqoop.**

Free-form query import allows the users to mention the query they want to use while importing data. The query parameter will be used.

**What is boundary query in sqoop?**

The valid query required to obtain the minimum and maximum values for the column split using split by command can be mentioned using the boundary query parameter.

**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

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

The databases can be viewed by using the following command:

$sqoop  **list**-databases

**Why to use $conditions in sqoop?**

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

**What is sqoop and What does sqoop stand for?**

Sqoop stands for SQL to Hadoop and Hadoop to SQL. The Apache Sqoop is a component within the Hadoop ecosystem. It helps in the movement of data between the different relational database servers in the Hadoop file structure.  It can pull data from other relational databases such as Oracle to the Hadoop file system.

**What is sqoop used for?**

The primary task of Sqoop is to transfer bulk information between relational database servers. Sqoop helps the developers to handle these huge bulks of data which in turn helps them to gather relevant insights from the data. It eases up the process and allows better export and import of information.

**What is the default database of Sqoop?**

The default database of Sqoop is MySql.

**How to check sqoop version?**

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

sqoop version

**How to set a number of mappers in Apache sqoop?**

Sqoop controls the number of mappers that are currently accessing the RDBMS to avoid the DOS attacks on the entire system. This number of mappers can be set using the following command:

clause-m **or** numb-mappers.

For example,  numb-mappers 10.

**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**.

**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.

**What is sqoop eval tool?**

The primary task of the Sqoop Eval is to let the users execute simple database queries. Both DDL and DML statements can be executed easily using Sqoop Eval.  

**Why sqoop is used in Hadoop?**

Sqoop is used to import datasets into the HDFS. It can also be used for export datastores from Hadoop to external databases. Sqoop is easier to use while transferring data for the developers, as it automates the process. It makes use of the MapReduce framework to transfer data, that provides fault tolerance. It is easy to use Sqoop as it has a command based interface.

**What is boundary query in sqoop?**

The valid query required to obtain the minimum and maximum values for the column split using split by command can be mentioned using the boundary query parameter.

**How to import multiple tables in sqoop?**

The import all tables tool can be used to import multiple tables in Sqoop. The following syntax has to be used:

$ sqoop **import**-all-tables (generic-args) (**import**-args)

**However, to use the tool successfully, the following conditions have to be met:**

* The tables should have a single column primary key.
* All the columns of each table must be imported.
* Using conditions having a WHERE clause and non-default splitting columns must be avoided.

**How to change sqoop date format?**

The date format is Sqoop can be altered by the SET DATEFORMAT 'ymd' just after starting the connection.

**What is the difference between Sqoop and Flume?**

**The difference between Sqoop and Flume is depicted below:**

| **Sqoop** | **Flume** |
| --- | --- |
| It is used for importing and exporting datastores from relational databases. | It is used for pulling logs and crash reports into the Hadoop file system. |
| It does not have event-driven loading. | It has event-driven loading. |
| It has an architecture based on connectors. | It has an architecture based on agents, where the agent code is used to pull the data. |
| Used to move data faster to produce analytic results. | It is used when the analysis of patterns and root causes is necessary for corporations. |

**What happens when Sqoop fails in between the large data transfer job?**

During data transfer between two databases, if the job fails, some portion of the data might get saved in the database. This can be solved by using the staging table option.

**What is the use of split by in sqoop?**

The split by command is used to mention the column that will be used to split the data that is being imported while data transfer. Based on certain column values, Sqoop uses split by to split the data.

**How to use split by in sqoop?**

The split by command can be used in Sqoop using the following syntax:

--split-by student.id

This command will split the data while importing based on the student.id values.

**What is accumulo in sqoop?**

The Apache Acccumulo is a key value store that is developed over the Hadoop File system. The handling of larger sets of data is easier with Accumulo. The data in Accumulo is stored in the Haddop File system. In Sqoop, there is a functionality that allows users to directly import a table from the Accumulo data store.

**How to grant access on password file in Sqoop?**

The password file option can be used to access the protected file. The file can be present in the HDFS directory or it can be kept in the system where the command is being executed. The following code will be needed:

$**echo** -n "password" > /etc/sqoop/conf/passwords/mysql-password.txt

$**chmod** 400 /etc/sqoop/conf/passwords/mysql-password.txt

$**sqoop** import --connect jdbc:netezza://localhost/MYDB

--**username** testuser

--**table** ORDERS

--**password**-file /etc/sqoop/conf/passwords/mysql-password.txt

**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 subsequent incremental imports to ensure that updated information is imported.

**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.

**Define free form import in sqoop.**

Free-form query import allows the users to mention the query they want to use while importing data. The query parameter will be used.

**What is boundary query in sqoop?**

The valid query required to obtain the minimum and maximum values for the column split using split by command can be mentioned using the boundary query parameter.

**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

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

The databases can be viewed by using the following command:

$sqoop  **list**-databases

**Why to use $conditions in sqoop?**

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

SPARK

## **Apache Spark Interview Questions And Answers**

### ****1. Compare Hadoop and Spark.****

We will compare Hadoop MapReduce and Spark based on the following aspects:

|  |  |  |
| --- | --- | --- |
| **Apache Spark vs. Hadoop** | | |
| **Feature Criteria** | **Apache Spark** | **Hadoop** |
| **Speed** | 100 times faster than Hadoop | Decent speed |
| **Processing** | Real-time & Batch processing | Batch processing only |
| **Difficulty** | Easy because of high level modules | Tough to learn |
| **Recovery** | Allows recovery of partitions | Fault-tolerant |
| **Interactivity** | Has interactive modes | No interactive mode except Pig & Hive |

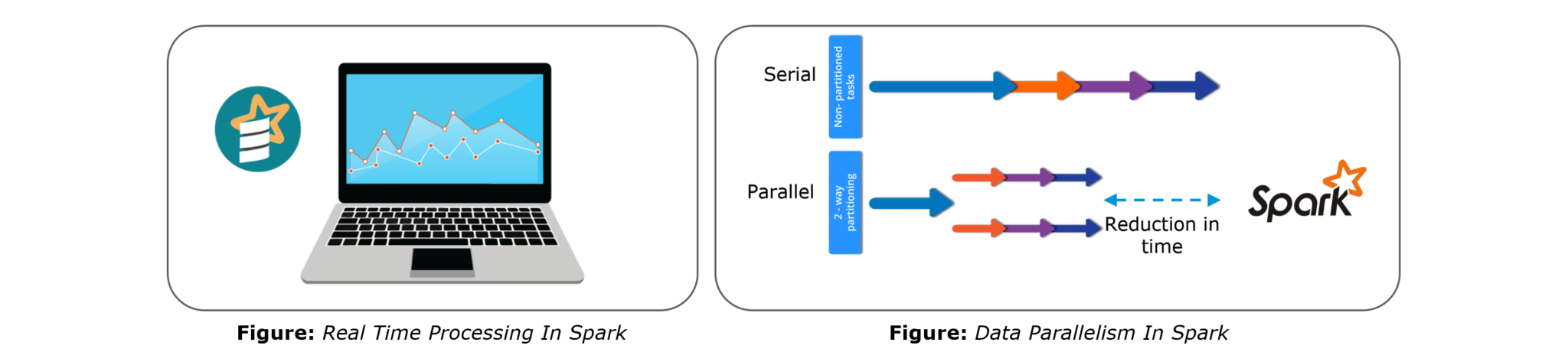
**Table:** Apache Spark versus Hadoop

Let us understand the same using an interesting analogy.

“Single cook cooking an entree is regular computing. Hadoop is multiple cooks cooking an entree into pieces and letting each cook her piece.Each cook has a separate stove and a food shelf. The first cook cooks the meat, the second cook cooks the sauce. This phase is called “Map”. A the end the main cook assembles the complete entree. This is called “Reduce”. For Hadoop, the cooks are not allowed to keep things on the stove between operations. Each time you make a particular operation, the cook puts results on the shelf. This slows things down.For Spark, the cooks are allowed to keep things on the stove between operations. This speeds things up. Finally, for Hadoop the recipes are written in a language which is illogical and hard to understand. For Spark, the recipes are nicely written.” – Stan Kladko*, Galactic Exchange.io*

### ****2. What is Apache Spark?****

* [***Apache Spark***](https://www.edureka.co/blog/spark-tutorial/) is an open-source cluster computing framework for real-time processing.
* It has a thriving open-source community and is the most active Apache project at the moment.
* Spark provides an interface for programming entire clusters with implicit data parallelism and fault-tolerance.

Spark is of the most successful projects in the Apache Software Foundation. Spark has clearly evolved as the market leader for Big Data processing. Many organizations run Spark on clusters with thousands of nodes. Today, Spark is being adopted by major players like Amazon, eBay, and Yahoo!

[**GET STARTED WITH SPARK**](https://www.edureka.co/apache-spark-scala-training)

### ****3. Explain the key features of Apache Spark.****

The following are the key features of Apache Spark:

1. **Polyglot**
2. **Speed**
3. **Multiple Format Support**
4. **Lazy Evaluation**
5. **Real Time Computation**
6. **Hadoop Integration**
7. **Machine Learning**

Let us look at these features in detail:

1. **Polyglot**: Spark provides high-level APIs in Java, Scala, Python and R. Spark code can be written in any of these four languages. It provides a shell in Scala and Python. The Scala shell can be accessed through **./bin/spark-shell** and Python shell through **./bin/pyspark** from the installed directory.
2. **Speed**: Spark runs upto 100 times faster than Hadoop MapReduce for large-scale data processing. Spark is able to achieve this speed through controlled partitioning. It manages data using partitions that help parallelize distributed data processing with minimal network traffic.
3. **Multiple Formats**: Spark supports multiple data sources such as Parquet, JSON, Hive and Cassandra. The Data Sources API provides a pluggable mechanism for accessing structured data though Spark SQL. Data sources can be more than just simple pipes that convert data and pull it into Spark.
4. **Lazy Evaluation**: Apache Spark delays its evaluation till it is absolutely necessary. This is one of the key factors contributing to its speed. For transformations, Spark adds them to a DAG of computation and only when the driver requests some data, does this DAG actually gets executed.
5. **Real Time Computation**: Spark’s computation is real-time and has less latency because of its in-memory computation. Spark is designed for massive scalability and the Spark team has documented users of the system running production clusters with thousands of nodes and supports several computational models.
6. **Hadoop Integration**: Apache Spark provides smooth compatibility with Hadoop. This is a great boon for all the Big Data engineers who started their careers with Hadoop. Spark is a potential replacement for the MapReduce functions of Hadoop, while Spark has the ability to run on top of an existing Hadoop cluster using YARN for resource scheduling.
7. **Machine Learning**: Spark’s MLlib is the machine learning component which is handy when it comes to big data processing. It eradicates the need to use multiple tools, one for processing and one for machine learning. Spark provides data engineers and data scientists with a powerful, unified engine that is both fast and easy to use.

### ****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. Among these languages, Scala and Python have interactive shells for Spark. The Scala shell can be accessed through**./bin/spark-shell**and the Python shell through **./bin/pyspark**. Scala is the most used among them because Spark is written in Scala and it is the most popularly used for Spark.

### ****5. What are benefits of Spark over MapReduce?****

Spark has the following benefits over MapReduce:

1. Due to the availability of in-memory processing, Spark implements the processing around 10 to 100 times faster than Hadoop MapReduce whereas MapReduce makes use of persistence storage for any of the data processing tasks.
2. Unlike Hadoop, Spark provides inbuilt libraries to perform multiple tasks from the same core like batch processing, Steaming, Machine learning, Interactive SQL queries. However, Hadoop only supports batch processing.
3. Hadoop is highly disk-dependent whereas Spark promotes caching and in-memory data storage.
4. Spark is capable of performing computations multiple times on the same dataset. This is called iterative computation while there is no iterative computing implemented by Hadoop.

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

Similar to Hadoop, YARN is one of the key features in Spark, providing a central and resource management platform to deliver scalable operations across the cluster. YARN is a distributed container manager, like Mesos for example, whereas Spark is a data processing tool. Spark can run on YARN, the same way Hadoop Map Reduce can run on YARN. Running Spark on YARN necessitates a binary distribution of Spark as built on YARN support.

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

No, because Spark runs on top of YARN. Spark runs independently from its installation. Spark has some options to use YARN when dispatching jobs to the cluster, rather than its own built-in manager, or Mesos. Further, there are some configurations to run YARN. They include master, deploy-mode, driver-memory, executor-memory, executor-cores, and queue.

### ****8. Is there any benefit of learning MapReduce if Spark is better than MapReduce?****

Yes, MapReduce is a paradigm used by many big data tools including Spark as well. It is extremely relevant to use MapReduce when the data grows bigger and bigger. Most tools like Pig and Hive convert their queries into MapReduce phases to optimize them better.

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

RDD stands for Resilient Distribution Datasets. An RDD is a fault-tolerant collection of operational elements that run in parallel. The partitioned data in RDD is immutable and distributed in nature. There are primarily two types of RDD:

1. Parallelized Collections: Here, the existing RDDs running parallel with one another.
2. Hadoop Datasets: They perform functions on each file record in HDFS or other storage systems.

RDDs are basically parts of data that are stored in the memory distributed across many nodes. RDDs are lazily evaluated in Spark. This lazy evaluation is what contributes to Spark’s speed.

### ****10. How do we create RDDs in Spark?****

Spark provides two methods to create RDD:

1. By parallelizing a collection in your Driver program.

2. This makes use of SparkContext’s ‘parallelize’

|  |  |
| --- | --- |
| 1  2  3 | method val DataArray = Array(2,4,6,8,10)    val DataRDD = sc.parallelize(DataArray) |

3. By loading an external dataset from external storage like HDFS, HBase, shared file system.

### ****11. What is Executor Memory in a Spark application?****

Every spark application has same fixed heap size and fixed number of cores for a spark executor. The heap size is what referred to as the Spark executor memory which is controlled with the spark.executor.memory property of the **–executor-memory** flag. Every spark application will have one executor on each worker node. The executor memory is basically a measure on how much memory of the worker node will the application utilize.

### ****12. Define Partitions in Apache Spark.****

As the name suggests, partition is a smaller and logical division of data similar to ‘split’ in MapReduce. It is a logical chunk of a large distributed data set. Partitioning is the process to derive logical units of data to speed up the processing process. Spark manages data using partitions that help parallelize distributed data processing with minimal network traffic for sending data between executors. By default, Spark tries to read data into an RDD from the nodes that are close to it. Since Spark usually accesses distributed partitioned data, to optimize transformation operations it creates partitions to hold the data chunks. Everything in Spark is a partitioned RDD.

### ****13. What operations does RDD support?****

RDD (Resilient Distributed Dataset) is main logical data unit in Spark. An RDD has distributed a collection of objects. Distributed means, each RDD is divided into multiple partitions. Each of these partitions can reside in memory or stored on the disk of different machines in a cluster. RDDs are immutable (Read Only) data structure. You can’t change original RDD, but you can always transform it into different RDD with all changes you want.

RDDs support two types of operations: transformations and actions.

Transformations: Transformations create new RDD from existing RDD like map, reduceByKey and filter we just saw. Transformations are executed on demand. That means they are computed lazily.

Actions: Actions return final results of RDD computations. Actions triggers execution using lineage graph to load the data into original RDD, carry out all intermediate transformations and return final results to Driver program or write it out to file system.

### ****14. What do you understand by Transformations in Spark?****

Transformations are functions applied on RDD, resulting into another RDD. It does not execute until an action occurs. map() and filter() are examples of transformations, where the former applies the function passed to it on each element of RDD and results into another RDD. The filter() creates a new RDD by selecting elements from current RDD that pass function argument.

|  |  |
| --- | --- |
| 1  2  3 | val rawData=sc.textFile("path to/movies.txt")    val moviesData=rawData.map(x=&gt;x.split("\t")) |

As we can see here, rawData RDD is transformed into moviesData RDD. Transformations are lazily evaluated.

### ****15. Define Actions in Spark.****

An action helps in bringing back the data from RDD to the local machine. An action’s execution is the result of all previously created transformations. Actions triggers execution using lineage graph to load the data into original RDD, carry out all intermediate transformations and return final results to Driver program or write it out to file system.

reduce() is an action that implements the function passed again and again until one value if left. take()action takes all the values from RDD to a local node.

|  |  |
| --- | --- |
| 1 | moviesData.saveAsTextFile(“MoviesData.txt”) |

As we can see here, moviesData RDD is saved into a text file called MoviesData.txt.

[[](https://www.edureka.co/apache-spark-scala-training)](https://www.edureka.co/apache-spark-scala-training" \t "_blank)

### [Apache Spark and Scala Certification Training](https://www.edureka.co/apache-spark-scala-training" \t "_blank)

* *[Instructor-led Sessions](https://www.edureka.co/apache-spark-scala-training" \t "_blank)*
* *[Real-life Project](https://www.edureka.co/apache-spark-scala-training" \t "_blank)*
* *[Assignments](https://www.edureka.co/apache-spark-scala-training" \t "_blank)*
* *[Lifetime Access](https://www.edureka.co/apache-spark-scala-training" \t "_blank)*

[Explore Curriculum](https://www.edureka.co/apache-spark-scala-training" \t "_blank)

### ****16. Define functions of SparkCore.****

Spark Core is the base engine for large-scale parallel and distributed data processing. The core is the distributed execution engine and the Java, Scala, and Python APIs offer a platform for distributed ETL application development. SparkCore performs various important functions like memory management, monitoring jobs, fault-tolerance, job scheduling and interaction with storage systems. Further, additional libraries, built atop the core allow diverse workloads for streaming, SQL, and machine learning. It is responsible for:

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?****

Apache defines PairRDD functions class as

|  |  |
| --- | --- |
| 1 | class PairRDDFunctions[K, V] extends Logging with HadoopMapReduceUtil with Serializable |

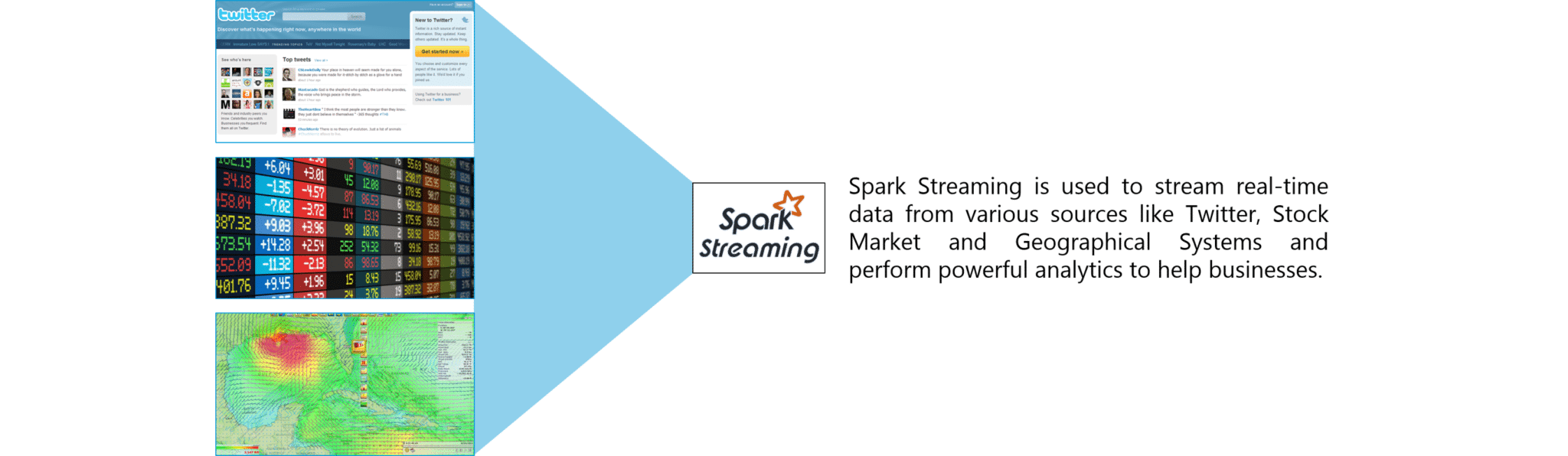
Special operations can be performed on RDDs in Spark using key/value pairs and such RDDs are referred to as Pair RDDs. Pair RDDs allow users to access each key in parallel. They have areduceByKey() method that collects data based on each key and a join() method that combines different RDDs together, based on the elements having the same key.

### ****18. Name the components of Spark Ecosystem.****

1. **Spark Core**: Base engine for large-scale parallel and distributed data processing
2. **Spark Streaming**: Used for processing real-time streaming data
3. **Spark SQL**: Integrates relational processing with Spark’s functional programming API
4. **GraphX**: Graphs and graph-parallel computation
5. **MLlib**: Performs machine learning in Apache Spark

### ****19. How is Streaming implemented in Spark? Explain with examples.****

Spark Streaming is used for processing real-time streaming data. Thus it is a useful addition to the core Spark API. It enables high-throughput and fault-tolerant stream processing of live data streams. The fundamental stream unit is DStream which is basically a series of RDDs (Resilient Distributed Datasets) to process the real-time data. The data from different sources like Flume, HDFS is streamed and finally processed to file systems, live dashboards and databases. It is similar to batch processing as the input data is divided into streams like batches.

****

**Figure:** Spark Interview Questions – Spark Streaming

### ****20. Is there an API for implementing graphs in Spark?****

GraphX is the Spark API for graphs and graph-parallel computation. Thus, it extends the Spark RDD with a Resilient Distributed Property Graph.

The property graph is a directed multi-graph which can have multiple edges in parallel. Every edge and vertex have user defined properties associated with it. Here, the parallel edges allow multiple relationships between the same vertices. At a high-level, GraphX extends the Spark RDD abstraction by introducing the Resilient Distributed Property Graph: a directed multigraph with properties attached to each vertex and edge.

To support graph computation, GraphX exposes a set of fundamental operators (e.g., subgraph, joinVertices, and mapReduceTriplets) as well as an optimized variant of the Pregel API. In addition, GraphX includes a growing collection of graph algorithms and builders to simplify graph analytics tasks.

### ****21. What is PageRank in GraphX?****

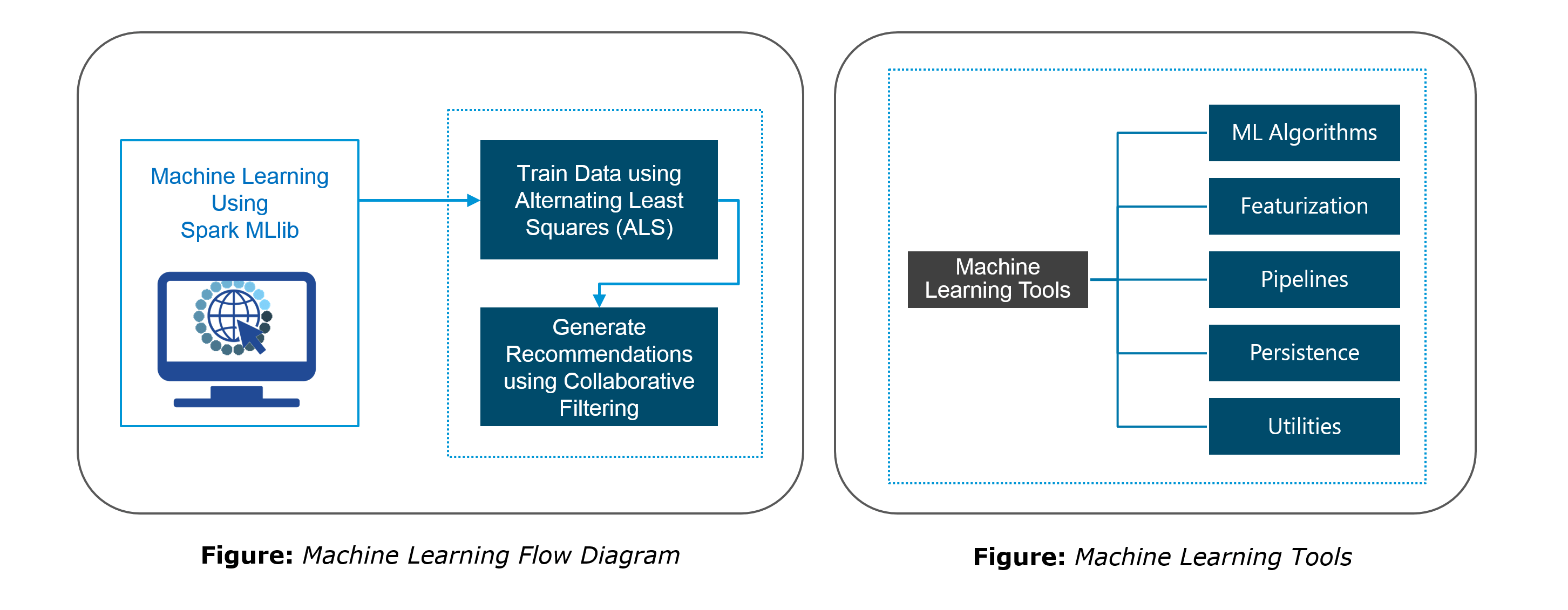
PageRank measures the importance of each vertex in a graph, assuming an edge from u to v represents an endorsement of v’s importance by u. For example, if a Twitter user is followed by many others, the user will be ranked highly.

GraphX comes with static and dynamic implementations of PageRank as methods on the PageRank Object. Static PageRank runs for a fixed number of iterations, while dynamic PageRank runs until the ranks converge (i.e., stop changing by more than a specified tolerance). GraphOps allows calling these algorithms directly as methods on Graph.

[**LEARN SPARK FROM EXPERTS**](https://www.edureka.co/apache-spark-scala-training)

### ****22. How is machine learning implemented in Spark?****

MLlib is scalable machine learning library provided by Spark. It aims at making machine learning easy and scalable with common learning algorithms and use cases like clustering, regression filtering, dimensional reduction, and alike.

****

### ****23. Is there a module to implement SQL in Spark? How does it work?****

Spark SQL is a new module in Spark which integrates relational processing with Spark’s functional programming API. It supports querying data either via SQL or via the Hive Query Language. For those of you familiar with RDBMS, Spark SQL will be an easy transition from your earlier tools where you can extend the boundaries of traditional relational data processing.

Spark SQL integrates relational processing with Spark’s functional programming. Further, it provides support for various data sources and makes it possible to weave SQL queries with code transformations thus resulting in a very powerful tool.

The following are the four libraries of Spark SQL.

1. Data Source API
2. DataFrame API
3. Interpreter & Optimizer
4. SQL Service

### Spark SQL - Spark Interview Questions - Edureka****24. What is a Parquet file?****

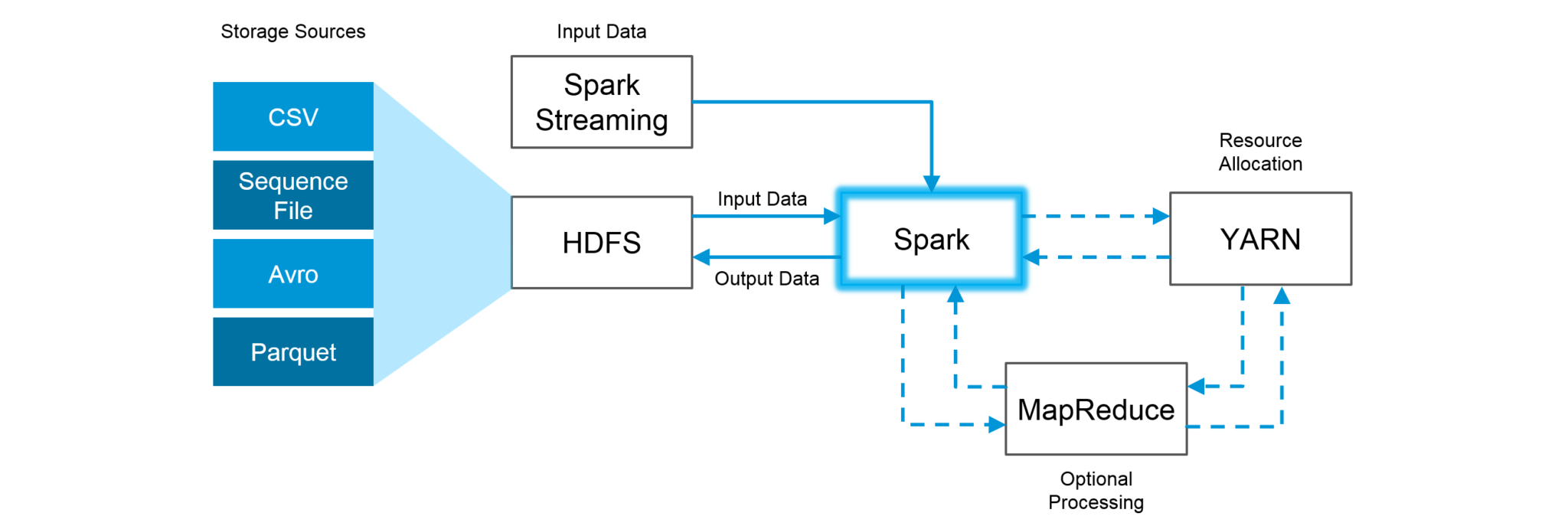
Parquet is a columnar format file supported by many other data processing systems. Spark SQL performs both read and write operations with Parquet file and consider it be one of the best big data analytics formats so far.

Parquet is a columnar format, supported by many data processing systems. The advantages of having a columnar storage are as follows:

1. Columnar storage limits IO operations.
2. It can fetch specific columns that you need to access.
3. Columnar storage consumes less space.
4. It gives better-summarized data and follows type-specific encoding.

### ****25. How can Apache Spark be used alongside Hadoop?****

The best part of Apache Spark is its compatibility with Hadoop. As a result, this makes for a very powerful combination of technologies. Here, we will be looking at how Spark can benefit from the best of Hadoop. Using Spark and Hadoop together helps us to leverage Spark’s processing to utilize the best of Hadoop’s HDFS and YARN.

**Figure:**Using Spark and Hadoop

Hadoop components can be used alongside Spark in the following ways:

1. **HDFS**: Spark can run on top of HDFS to leverage the distributed replicated storage.
2. **MapReduce**: Spark can be used along with MapReduce in the same Hadoop cluster or separately as a processing framework.
3. **YARN**: Spark applications can also be run on YARN (Hadoop NextGen).
4. **Batch & Real Time Processing**: MapReduce and Spark are used together where MapReduce is used for batch processing and Spark for real-time processing.

### ****26. What is RDD Lineage?****

Spark does not support data replication in the memory and thus, if any data is lost, it is rebuild using RDD lineage. RDD lineage is a process that reconstructs lost data partitions. The best is that RDD always remembers how to build from other datasets.

### ****27. What is Spark Driver?****

Spark Driver is the program that runs on the master node of the machine and declares transformations and actions on data RDDs. In simple terms, a driver in Spark creates SparkContext, connected to a given Spark Master.  
The driver also delivers the RDD graphs to Master, where the standalone cluster manager runs.

### ****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

### ****29. List the functions of Spark SQL.****

Spark SQL is capable of:

1. Loading data from a variety of structured sources.
2. Querying data using SQL statements, both inside a Spark program and from external tools that connect to Spark SQL through standard database connectors (JDBC/ODBC). For instance, using business intelligence tools like Tableau.
3. Providing rich integration between SQL and regular Python/Java/Scala code, including the ability to join RDDs and SQL tables, expose custom functions in SQL, and more.

### ****30. What is Spark Executor?****

When SparkContext connects to a cluster manager, it acquires an Executor on nodes in the cluster. Executors are Spark processes that run computations and store the data on the worker node. The final tasks by SparkContext are transferred to executors for their execution.

### ****31. Name types of Cluster Managers in Spark.****

The Spark framework supports three major types of Cluster Managers:

1. **Standalone**: A basic manager to set up a cluster.
2. **Apache Mesos**: Generalized/commonly-used cluster manager, also runs Hadoop MapReduce and other applications.
3. **YARN**: Responsible for resource management in Hadoop.

### ****32. What do you understand by worker node?****

Worker node refers to any node that can run the application code in a cluster. The driver program must listen for and accept incoming connections from its executors and must be network addressable from the worker nodes.

Worker node is basically the slave node. Master node assigns work and worker node actually performs the assigned tasks. Worker nodes process the data stored on the node and report the resources to the master. Based on the resource availability, the master schedule tasks.

### ****33. Illustrate some demerits of using Spark.****

The following are some of the demerits of using Apache Spark:

1. Since Spark utilizes more storage space compared to Hadoop and MapReduce, there may arise certain problems.
2. Developers need to be careful while running their applications in Spark.
3. Instead of running everything on a single node, the work must be distributed over multiple clusters.
4. Spark’s “in-memory” capability can become a bottleneck when it comes to cost-efficient processing of big data.
5. Spark consumes a huge amount of data when compared to Hadoop.

### ****34. List some use cases where Spark outperforms Hadoop in processing.****

1. **Sensor Data Processing**: Apache Spark’s “In-memory” computing works best here, as data is retrieved and combined from different sources.
2. **Real Time Processing**: Spark is preferred over Hadoop for real-time querying of data. e.g. Stock Market Analysis, Banking, Healthcare, Telecommunications, etc.
3. **Stream Processing**: For processing logs and detecting frauds in live streams for alerts, Apache Spark is the best solution.
4. **Big Data Processing**:Spark runs upto 100 times faster than Hadoop when it comes to processing medium and large-sized datasets.

### ****35. What is a Sparse Vector?****

A sparse vector has two parallel arrays; one for indices and the other for values. These vectors are used for storing non-zero entries to save space.

[[](https://www.edureka.co/apache-spark-scala-training)](https://www.edureka.co/apache-spark-scala-training" \t "_blank)

### [Apache Spark and Scala Certification Training](https://www.edureka.co/apache-spark-scala-training" \t "_blank)

[Watch The Course Preview](https://www.edureka.co/apache-spark-scala-training" \t "_blank)

|  |  |
| --- | --- |
| 1 | Vectors.sparse(7,Array(0,1,2,3,4,5,6),Array(1650d,50000d,800d,3.0,3.0,2009,95054)) |

The above sparse vector can be used instead of dense vectors.

|  |  |
| --- | --- |
| 1 | val myHouse = Vectors.dense(4450d,2600000d,4000d,4.0,4.0,1978.0,95070d,1.0,1.0,1.0,0.0) |

### ****36. Can you use Spark to access and analyze data stored in Cassandra databases?****

Yes, it is possible if you use Spark Cassandra Connector.To connect Spark to a Cassandra cluster, a Cassandra Connector will need to be added to the Spark project. In the setup, a Spark executor will talk to a local Cassandra node and will only query for local data. It makes queries faster by reducing the usage of the network to send data between Spark executors (to process data) and Cassandra nodes (where data lives).

### ****37. Is it possible to run Apache Spark on Apache Mesos?****

Yes, Apache Spark can be run on the hardware clusters managed by Mesos. In a standalone cluster deployment, the cluster manager in the below diagram is a Spark master instance. When using Mesos, the Mesos master replaces the Spark master as the cluster manager. Mesos determines what machines handle what tasks. Because it takes into account other frameworks when scheduling these many short-lived tasks, multiple frameworks can coexist on the same cluster without resorting to a static partitioning of resources.

### ****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. Spark binary package should be in a location accessible by Mesos.
3. Install Apache Spark in the same location as that of Apache Mesos and configure the property ‘spark.mesos.executor.home’ to point to the location where it is installed.

### ****39. How can you minimize data transfers when working with Spark?****

Minimizing data transfers and avoiding shuffling helps write spark programs that run in a fast and reliable manner. The various ways in which data transfers can be minimized when working with Apache Spark are:

1. Using Broadcast Variable- Broadcast variable enhances the efficiency of joins between small and large RDDs.
2. Using Accumulators – Accumulators help update the values of variables in parallel while executing.

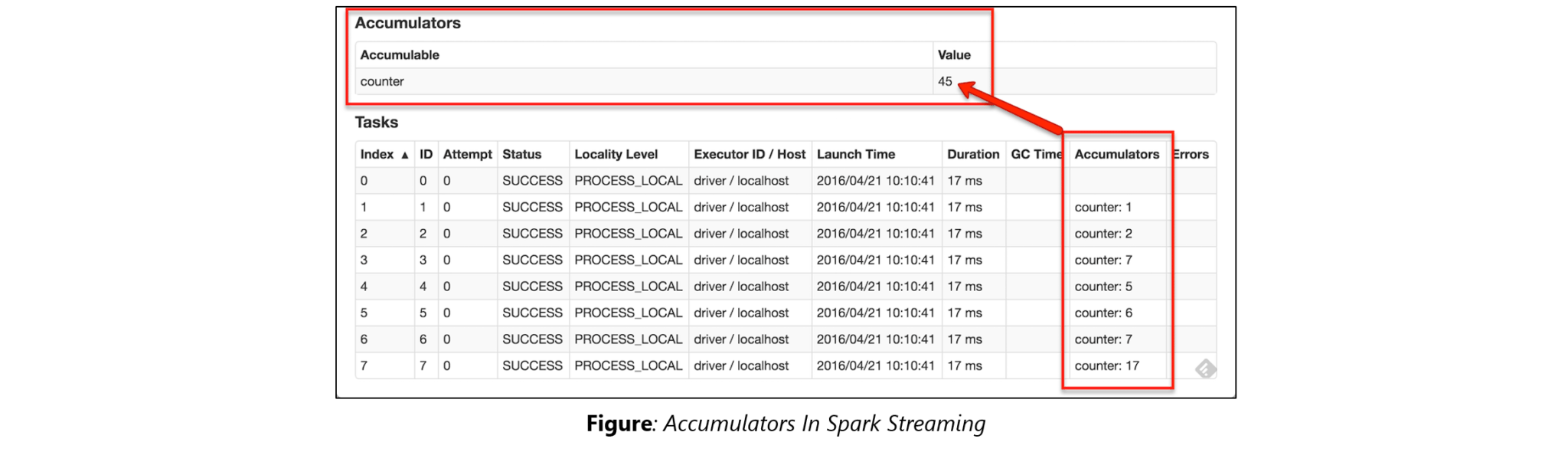
The most common way is to avoid operations ByKey, repartition or any other operations which trigger shuffles.

### ****40. What are broadcast variables?****

Broadcast variables allow the programmer to keep a read-only variable cached on each machine rather than shipping a copy of it with tasks. They can be used to give every node a copy of a large input dataset in an efficient manner. Spark also attempts to distribute broadcast variables using efficient broadcast algorithms to reduce communication cost.

### Broadcast Variables - Spark Interview Questions - Edureka****41. Explain accumulators in Apache Spark.****

Accumulators are variables that are only added through an associative and commutative operation. They are used to implement counters or sums. Tracking accumulators in the UI can be useful for understanding the progress of running stages. Spark natively supports numeric accumulators. We can create named or unnamed accumulators.

****

### ****42. Why is there a need for broadcast variables when working with Apache Spark?****

Broadcast variables are read only variables, present in-memory cache on every machine. When working with Spark, usage of broadcast variables eliminates the necessity to ship copies of a variable for every task, so data can be processed faster. Broadcast variables help in storing a lookup table inside the memory which enhances the retrieval efficiency when compared to an RDD lookup().

[**LEARN SPARK FROM EXPERTS**](https://www.edureka.co/apache-spark-scala-training)

### ****43. How can you trigger automatic clean-ups in Spark to handle accumulated metadata?****

You can trigger the clean-ups by setting the parameter ‘spark.cleaner.ttl’ or by dividing the long running jobs into different batches and writing the intermediary results to the disk.

### ****44. What is the significance of Sliding Window operation?****

Sliding Window controls transmission of data packets between various computer networks. Spark Streaming library provides windowed computations where the transformations on RDDs are applied over a sliding window of data. Whenever the window slides, the RDDs that fall within the particular window are combined and operated upon to produce new RDDs of the windowed DStream.

### DStream Sliding Window - Spark Interview Questions - Edureka****45. What is a DStream in Apache Spark?****

***Discretized Stream***(DStream) is the basic abstraction provided by Spark Streaming. It is a continuous stream of data. It is received from a data source or from a processed data stream generated by transforming the input stream. Internally, a DStream is represented by a continuous series of RDDs and each RDD contains data from a certain interval. Any operation applied on a DStream translates to operations on the underlying RDDs.

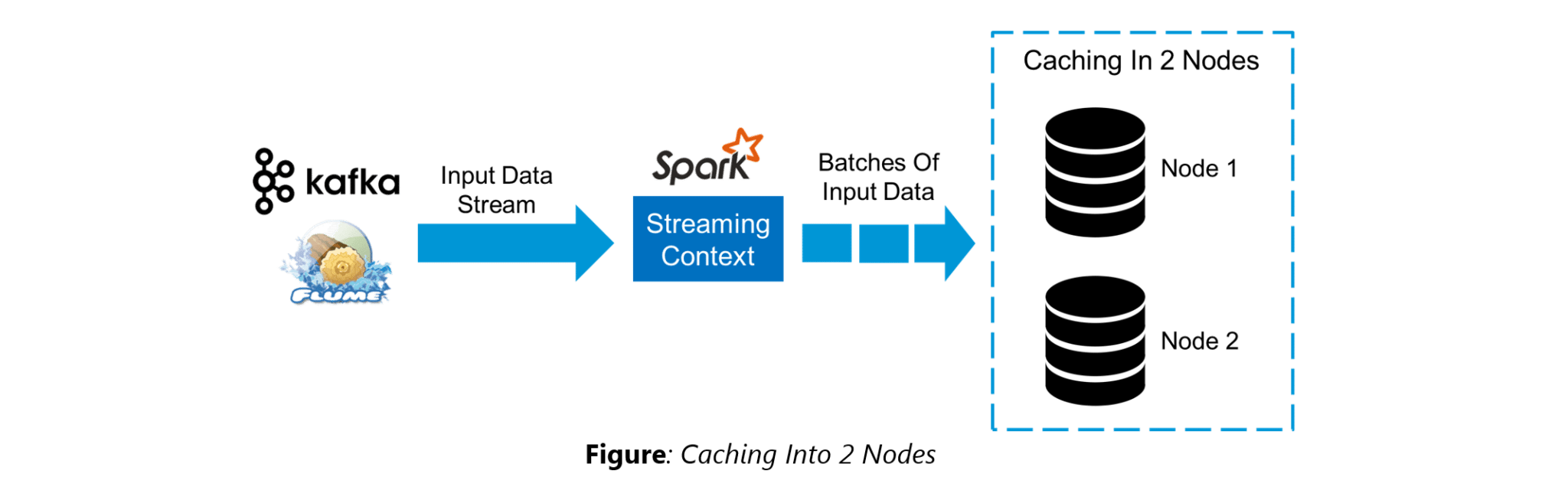
DStreams can be created from various sources like Apache Kafka, HDFS, and Apache Flume. DStreams have two operations:

1. Transformations that produce a new DStream.
2. Output operations that write data to an external system.

There are many DStream transformations possible in Spark Streaming. Let us look at**filter(*func*)**. filter(*func*) returns a new DStream by selecting only the records of the source DStream on which func returns true.

### DStream Filter - Spark Interview Questions - Edureka****46. Explain Caching in Spark Streaming.****

DStreams allow developers to cache/ persist the stream’s data in memory. This is useful if the data in the DStream will be computed multiple times. This can be done using the persist() method on a DStream. For input streams that receive data over the network (such as Kafka, Flume, Sockets, etc.), the default persistence level is set to replicate the data to two nodes for fault-tolerance.

****

### ****47. When running Spark applications, is it necessary to install Spark on all the nodes of YARN cluster?****

Spark need not be installed when running a job under YARN or Mesos because Spark can execute on top of YARN or Mesos clusters without affecting any change to the cluster.

### ****48. What are the various data sources available in Spark SQL?****

Parquet file, JSON datasets and Hive tables are the data sources available in Spark SQL.

### ****49. What are the various levels of persistence in Apache Spark?****

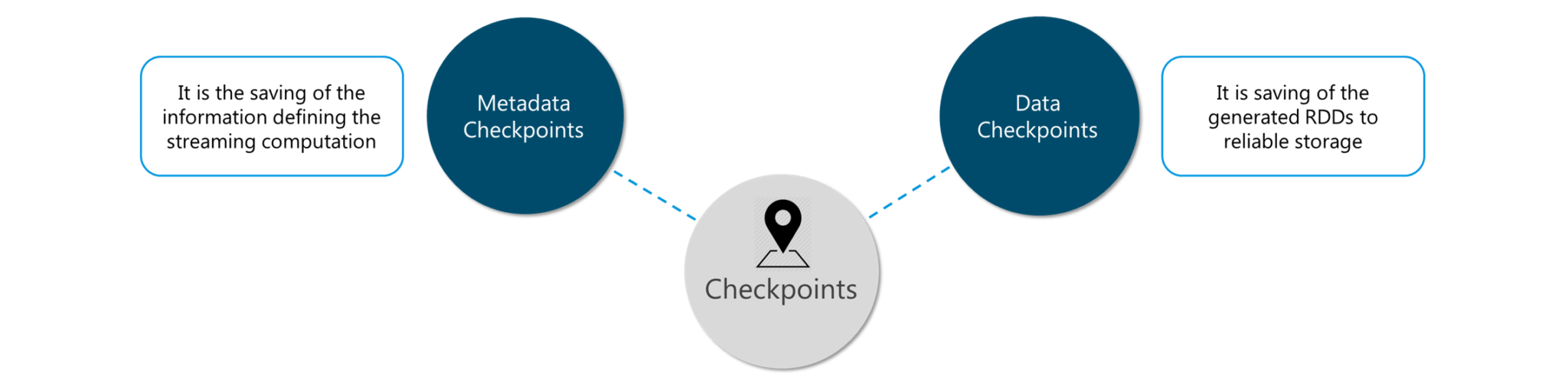
Apache Spark automatically persists the intermediary data from various shuffle operations, however, it is often suggested that users call persist () method on the RDD in case they plan to reuse it. Spark has various persistence levels to store the RDDs on disk or in memory or as a combination of both with different replication levels.

The various storage/persistence levels in Spark are:

1. MEMORY\_ONLY: Store RDD as deserialized Java objects in the JVM. If the RDD does not fit in memory, some partitions will not be cached and will be recomputed on the fly each time they’re needed. This is the default level.
2. MEMORY\_AND\_DISK: Store RDD as deserialized Java objects in the JVM. If the RDD does not fit in memory, store the partitions that don’t fit on disk, and read them from there when they’re needed.
3. MEMORY\_ONLY\_SER: Store RDD as *serialized* Java objects (one byte array per partition).
4. MEMORY\_AND\_DISK\_SER: Similar to MEMORY\_ONLY\_SER, but spill partitions that don’t fit in memory to disk instead of recomputing them on the fly each time they’re needed.
5. DISK\_ONLY: Store the RDD partitions only on disk.
6. OFF\_HEAP: Similar to MEMORY\_ONLY\_SER, but store the data in off-heap memory.

### ****50. Does Apache Spark provide checkpoints?****

Checkpoints are similar to checkpoints in gaming. They make it run 24/7 and make it resilient to failures unrelated to the application logic.

**Figure:** Spark Interview Questions – Checkpoints

Lineage graphs are always useful to recover RDDs from a failure but this is generally time-consuming if the RDDs have long lineage chains. Spark has an API for checkpointing i.e. a REPLICATE flag to persist. However, the decision on which data to checkpoint – is decided by the user. Checkpoints are useful when the lineage graphs are long and have wide dependencies.

### ****51. How Spark uses Akka?****

Spark uses Akka basically for scheduling. All the workers request for a task to master after registering. The master just assigns the task. Here Spark uses Akka for messaging between the workers and masters.

### ****52. What do you understand by Lazy Evaluation?****

Spark is intellectual in the manner in which it operates on data. When you tell Spark to operate on a given dataset, it heeds the instructions and makes a note of it, so that it does not forget – but it does nothing, unless asked for the final result. When a transformation like map() is called on an RDD, the operation is not performed immediately. Transformations in Spark are not evaluated till you perform an action. This helps optimize the overall data processing workflow.

### Lazy Evaluation - Spark Interview Questions - Edureka****53. What do you understand by SchemaRDD in Apache Spark RDD?****

SchemaRDD is an RDD that consists of row objects (wrappers around the basic string or integer arrays) with schema information about the type of data in each column.

SchemaRDD was designed as an attempt to make life easier for developers in their daily routines of code debugging and unit testing on SparkSQL core module. The idea can boil down to describing the data structures inside RDD using a formal description similar to the relational database schema. On top of all basic functions provided by common RDD APIs, SchemaRDD also provides some straightforward relational query interface functions that are realized through SparkSQL.

Now, it is officially renamed to DataFrame API on Spark’s latest trunk.

### ****54. How is Spark SQL different from HQL and SQL?****

Spark SQL is a special component on the Spark Core engine that supports SQL and Hive Query Language without changing any syntax. It is possible to join SQL table and HQL table to Spark SQL.

### ****55. Explain a scenario where you will be using Spark Streaming.****

When it comes to Spark Streaming, the data is streamed in real-time onto our Spark program.

Twitter Sentiment Analysis is a real-life use case of Spark Streaming. Trending Topics can be used to create campaigns and attract a larger audience. It helps in crisis management, service adjusting and target marketing.

**Q) Apache Spark Vs Hadoop**

|  |  |  |  |
| --- | --- | --- | --- |
| **Spark Vs Hadoop** | | | |
| **Features** | **Spark** | **Hadoop** |  |
| Data processing | Part of hadoop, hence batch processing | Batch Processing even for high volumes |  |
| Streaming Engine | Apache spark straming - micro batches | Map-Reduce |  |
| Data Flow | Direct Acyclic Graph-DAG | Map-Reduce |  |
| Computation Model | Collect and process | Map-Reduce batch oriented model |  |
| Performance | Slow due to batch processing | Slow due to batch processing |  |
| Memory Management | Automatic memory management in latest release | Dynamic and static - Configurable |  |
| Fault Tolerance | Recovery available without extra code | Highly fault tolerant due to Map-Reduce |  |
| Scalability | Highly scalable - sSpark Cluster(8000 Nodes) | Highly scalable - Produces large number of nodes |  |

**Q1) What is Spark?**

Spark is a parallel data processing framework. It allows to develop fast, unified big data application combine batch, streaming and interactive analytics.

**Q2) Why Spark?**

Spark is the third generation distributed data processing platform. It’s unified bigdata solution for all bigdata processing problems such as batch , interacting, streaming processing.So it can ease many bigdata problems.

**Q3) What is RDD?**

Spark’s primary core abstraction is called Resilient Distributed Datasets. RDD is a collection of partitioned data that satisfies these properties. Immutable, distributed, lazily evaluated, catchable are common RDD properties.

**Q4) What is Immutable?**

Once created and assign a value, it’s not possible to change, this property is called Immutability. Spark is by default immutable, it does not allow updates and modifications. Please note data collection is not immutable, but data value is immutable.

**Q5) What is Distributed?**

RDD can automatically the data is distributed across different parallel computing nodes.

**Q6) What is Lazy evaluated?**

If you execute a bunch of programs, it’s not mandatory to evaluate immediately. Especially in Transformations, this Laziness is a trigger.

**Q6) What is Catchable?**

Keep all the data in-memory for computation, rather than going to the disk. So Spark can catch the data 100 times faster than Hadoop.

**Q7) What is Spark engine responsibility?**

Spark responsible for scheduling, distributing, and monitoring the application across the cluster.

**Q8) What are common Spark Ecosystems?**

* Spark SQL(Shark) for SQL developers,
* Spark Streaming for streaming data,
* MLLib for machine learning algorithms,
* GraphX for Graph computation,
* SparkR to run R on Spark engine,
* BlinkDB enabling interactive queries over massive data are common Spark ecosystems.  GraphX, SparkR, and BlinkDB are in the incubation stage.

**Q9) What is Partitions?**

Partition is a logical division of the data, this idea derived from Map-reduce (split). Logical data specifically derived to process the data. Small chunks of data also it can support scalability and speed up the process. Input data, intermediate data, and output data everything is Partitioned RDD.

Learn [Spark vs Hadoop](https://mindmajix.com/spark/whats-better-to-learn-first-spark-vs-hadoop) What's Better to Learn First.

**Q10) How spark partition the data?**

Spark use map-reduce API to do the partition the data. In Input format we can create number of partitions. By default HDFS block size is partition size (for best performance), but its’ possible to change partition size like Split.

**Q11) How Spark store the data?**

Spark is a processing engine, there is no storage engine. It can retrieve data from any storage engine like HDFS, S3 and other data resources.

**Q12) Is it mandatory to start Hadoop to run spark application?**

No not mandatory, but there is no separate storage in Spark, so it use local file system to store the data. You can load data from local system and process it, Hadoop or HDFS is not mandatory to run spark application.

**Q13) What is SparkContext?**

When a programmer creates a RDDs, SparkContext connect to the Spark cluster to create a new SparkContext object. SparkContext tell spark how to access the cluster. SparkConf is key factor to create programmer application.

**Q14) What is SparkCore functionalities?**

SparkCore is a base engine of apache spark framework. Memory management, fault tolarance, scheduling and monitoring jobs, interacting with store systems are primary functionalities of Spark.

**Q15) How SparkSQL is different from HQL and SQL?**

SparkSQL is a special component on the sparkCore engine that support SQL and HiveQueryLanguage without changing any syntax. It’s possible to join SQL table and HQL table.

**Q16) When did we use Spark Streaming?**

Spark Streaming is a real time processing of [streaming data](https://mindmajix.com/spark/streaming-big-data-with-apache-spark) API. Spark streaming gather streaming data from different resources like web server log files, social media data, stock market data or Hadoop ecosystems like Flume, and Kafka.

**Q17) How Spark Streaming API works?**

Programmer set a specific time in the configuration, with in this time how much data gets into the Spark, that data separates as a batch. The input stream (DStream) goes into spark streaming. Framework breaks up into small chunks called batches, then feeds into the spark engine for processing. Spark Streaming API passes that batches to the core engine. Core engine can generate the final results in the form of streaming batches. The output also in the form of batches. It can allows streaming data and batch data for processing.

**Q18) What is Spark MLlib?**

Mahout is a machine learning library for [Hadoop](https://mindmajix.com/hadoop-training), similarly MLlib is a Spark library. MetLib provides different algorithms, that algorithms scale out on the cluster for data processing. Most of the data scientists use this MLlib library.

**Q19) What is GraphX?**

GraphX is a Spark API for manipulating Graphs and collections. It unifies ETL, other analysis, and iterative graph computation. It’s fastest graph system, provides fault tolerance and ease of use without special skills.

**Q20) What is File System API?**

FS API can read data from different storage devices like HDFS, S3 or local FileSystem. Spark uses FS API to read data from different storage engines.

**Q21) Why Partitions are immutable?**

Every transformation generates new partition.  Partitions use HDFS API so that partition is immutable, distributed and fault tolerance. Partition also aware of data locality.

**Q22) What is Transformation in spark?**

Spark provides two special operations on RDDs called transformations and Actions. Transformation follows lazy operation and temporary hold the data until unless called the Action. Each transformation generates/return new RDD. Example of transformations: Map, flatMap, groupByKey, reduceByKey, filter, co-group, join, sortByKey, Union, distinct, sample are common spark transformations.

**Q23) What is Action in Spark?**

Actions are RDD’s operation, that value returns back to the spar driver programs, which kick off a job to execute on a cluster. Transformation’s output is an input of Actions. reduce, collect, takeSample, take, first, saveAsTextfile, saveAsSequenceFile, countByKey, foreach are common actions in Apache spark.

**Q24) What is RDD Lineage?**

Lineage is an RDD process to reconstruct lost partitions. Spark not replicate the data in memory, if data lost, Rdd use linege to rebuild lost data.Each RDD remembers how the RDD build from other datasets.

**Q25) What is Map and flatMap in Spark?**

The map is a specific line or row to process that data. In FlatMap each input item can be mapped to multiple output items (so the function should return a Seq rather than a single item). So most frequently used to return Array elements.

**Q26) What are broadcast variables?**

Broadcast variables let programmer keep a read-only variable cached on each machine, rather than shipping a copy of it with tasks. Spark supports 2 types of shared variables called broadcast variables (like Hadoop distributed cache) and accumulators (like Hadoop counters). Broadcast variables stored as Array Buffers, which sends read-only values to work nodes.

**Q27) What are Accumulators in Spark?**

Spark of-line debuggers called accumulators. Spark accumulators are similar to Hadoop counters, to count the number of events and what’s happening during job you can use accumulators. Only the driver program can read an accumulator value, not the tasks.

**Q28) How RDD persist the data?**

There are two methods to persist the data, such as persist() to persist permanently and cache() to persist temporarily in the memory. Different storage level options there such as MEMORY\_ONLY, MEMORY\_AND\_DISK, DISK\_ONLY and many more. Both persist() and cache() uses different options depends on the task.

**Q29) When do you use apache spark? OR  What are the benefits of Spark over Mapreduce?**

* Spark is really fast. As per their claims, it runs programs up to 100x faster than Hadoop MapReduce in memory, or 10x faster on disk. It aptly utilizes RAM to produce the faster results.
* In map reduce paradigm, you write many Map-reduce tasks and then tie these tasks together using Oozie/shell script. This mechanism is very time consuming and the map-reduce task has heavy latency.
* And quite often, translating the output out of one MR job into the input of another MR job might require writing another code because Oozie may not suffice.
* In Spark, you can basically do everything using single application/console (pyspark or scala console) and get the results immediately. Switching between ‘Running something on cluster’ and ‘doing something locally’ is fairly easy and straightforward. This also leads to less context switch of the developer and more productivity.
* Spark kind of equals to MapReduce and Oozie put together.

**Q30) Is there is a point of learning MapReduce, then?**

Yes. For the following reason:

* Mapreduce is a paradigm used by many big data tools including Spark. So, understanding the MapReduce paradigm and how to convert a problem into series of MR tasks is very important.
* When the data grows beyond what can fit into the memory on your cluster, the Hadoop Map-Reduce paradigm is still very relevant.
* Almost, every other tool such as Hive or Pig converts its query into MapReduce phases. If you understand the Mapreduce then you will be able to optimize your queries better.

**Q31) When running Spark on Yarn, do I need to install Spark on all nodes of Yarn Cluster?**

Since spark runs on top of Yarn, it utilizes yarn for the execution of its commands over the cluster’s nodes.  
So, you just have to install Spark on one node.

[Check Out Apache Spark Tutorials](https://mindmajix.com/apache-spark-tutorial)

**Q32) What are the downsides of Spark?**

Spark utilizes the memory. The developer has to be careful. A casual developer might make following mistakes:

* She may end up running everything on the local node instead of distributing work over to the cluster.
* She might hit some webservice too many times by the way of using multiple clusters.

The first problem is well tackled by Hadoop Map reduce paradigm as it ensures that the data your code is churning is fairly small a point of time thus you can make a mistake of trying to handle whole data on a single node.  
The second mistake is possible in Map-Reduce too. While writing Map-Reduce, user may hit a service from inside of map() or reduce() too many times. This overloading of service is also possible while using Spark.

**Q33) What is an RDD?**

The full form of RDD is resilience distributed dataset. It is a representation of data located on a network which is

* Immutable – You can operate on the rdd to produce another rdd but you can’t alter it.
* Partitioned / Parallel – The data located on RDD is operated in parallel. Any operation on RDD is done using multiple nodes.
* Resilience – If one of the node hosting the partition fails, another nodes takes its data.

RDD provides two kinds of operations: Transformations and Actions.

**Q34) What is Transformations?**

The transformations are the functions that are applied on an RDD (resilient distributed data set). The transformation results in another RDD. A transformation is not executed until an action follows.  
The example of transformations are:

1. map() – applies the function passed to it on each element of RDD resulting in a new RDD.
2. filter() – creates a new RDD by picking the elements from the current RDD which pass the function argument.

**Q35) What are Actions?**

An action brings back the data from the RDD to the local machine. Execution of an action results in all the previously created transformation. The example of actions are:

* reduce() – executes the function passed again and again until only one value is left. The function should take two argument and return one value.
* take() – take all the values back to the local node form RDD.

**Q36) 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);

**Q37) What is wrong with it? And How would you correct it?**

The average function is not commutative and associative;  
I would simply sum it and then divide by count.  
def sum(x, y):  
return x+y;  
total = myrdd.reduce(sum);  
avg = total / myrdd.count();  
The only problem with the above code is that the total might become very big thus over flow. So, I would rather divide each number by count and then sum in the following way.  
cnt = myrdd.count();  
def devideByCnd(x):  
return x/cnt;  
myrdd1 = myrdd.map(devideByCnd);  
avg = myrdd.reduce(sum);

**Q38) 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?**

# We would first load the file as RDD from HDFS on spark  
numsAsText = sc.textFile(“hdfs://hadoop1.knowbigdata.com/user/student/sgiri/mynumbersfile.txt”);  
# Define the function to compute the squares  
def toSqInt(str):  
v = int(str);  
return v\*v;  
#Run the function on spark rdd as transformation  
nums = numsAsText.map(toSqInt);  
#Run the summation as reduce action  
total = nums.reduce(sum)  
#finally compute the square root. For which we need to import math.  
import math;  
print math.sqrt(total);

**Q39) Is the following approach correct? Is the sqrtOfSumOfSq a valid reducer?**

numsAsText =sc.textFile(“hdfs://hadoop1.knowbigdata.com/user/student/sgiri/mynumbersfile.txt”);  
def toInt(str):  
return int(str);  
nums = numsAsText.map(toInt);  
def sqrtOfSumOfSq(x, y):  
return math.sqrt(x\*x+y\*y);  
total = nums.reduce(sum)  
import math;  
print math.sqrt(total);  
A: Yes. The approach is correct and **sqrtOfSumOfSq** is a valid reducer.

**Q40) Could you compare the pros and cons of the your approach (in Question 2 above) and my approach (in Question 3 above)?**

You are doing the square and square root as part of reduce action while I am squaring in map() and summing in reduce in my approach.  
My approach will be faster because in your case the reducer code is heavy as it is calling math.sqrt() and reducer code is generally executed approximately n-1 times the spark RDD.  
The only downside of my approach is that there is a huge chance of integer overflow because I am computing the sum of squares as part of map.

**Q41) If you have to compute the total counts of each of the unique words on spark, how would you go about it?**

#This will load the bigtextfile.txt as RDD in the spark lines = sc.textFile(“hdfs://hadoop1.knowbigdata.com/user/student/sgiri/bigtextfile.txt”);  
#define a function that can break each line into words  
def toWords(line):  
return line.split();  
# Run the toWords function on each element of RDD on spark as flatMap transformation.  
# We are going to flatMap instead of map because our function is returning multiple values.  
words = lines.flatMap(toWords);  
# Convert each word into (key, value) pair. Her key will be the word itself and value will be 1.  
def toTuple(word):  
return (word, 1);  
wordsTuple = words.map(toTuple);  
# Now we can easily do the reduceByKey() action.  
def sum(x, y):  
return x+y;  
counts = wordsTuple.reduceByKey(sum)  
# Now, print  
counts.collect()

**Q41) In a very huge text file, you want to just check if a particular keyword exists. How would you do this using Spark?**

lines = sc.textFile(“hdfs://hadoop1.knowbigdata.com/user/student/sgiri/bigtextfile.txt”);  
def isFound(line):  
if line.find(“mykeyword”) > -1:  
return 1;  
return 0;  
foundBits = lines.map(isFound);  
sum = foundBits.reduce(sum);  
if sum > 0:  
print “FOUND”;  
else:  
print “NOT FOUND”;

**Q42) Can you improve the performance of this code in previous answer?**

Yes. The search is not stopping even after the word we are looking for has been found. Our map code would keep executing on all the nodes which is very inefficient.  
We could utilize accumulators to report whether the word has been found or not and then stop the job. Something on these line:  
import thread, threading  
from time import sleep  
result = “Not Set”  
lock = threading.Lock()  
accum = sc.accumulator(0)  
def map\_func(line):  
#introduce delay to emulate the slowness  
sleep(1);  
if line.find(“Adventures”) > -1:  
accum.add(1);  
return 1;  
return 0;  
def start\_job():  
global result  
try:  
sc.setJobGroup(“job\_to\_cancel”, “some description”)  
lines = sc.textFile(“hdfs://hadoop1.knowbigdata.com/user/student/sgiri/wordcount/input/big.txt”);  
result = lines.map(map\_func);  
result.take(1);  
except Exception as e:  
result = “Cancelled”  
lock.release()  
def stop\_job():  
while accum.value < 3 :  
sleep(1);  
sc.cancelJobGroup(“job\_to\_cancel”)  
supress = lock.acquire()  
supress = thread.start\_new\_thread(start\_job, tuple())  
supress = thread.start\_new\_thread(stop\_job, tuple())  
supress = lock.acquire()  
[/tab]  
Facing technical problem in your current IT job, let us help you. MindMajix has highly technical people who can assist you in solving technical problems in your project.  
We have come across many developers in USA, Australia and other countries who have recently got the job but they are struggling to survive in the job because of less technical knowledge, exposure and the kind of work given to them.  
We are here to help you.

Let us know your profile and kind of help you are looking for and we shall do our best to help you out. The job support is provided by Mindmajix Technical experts who have more than 10 years of work experience on IT technologies landscape.  
How does the job support work?  
\* We see your project and technologies used, if we are 100% confident then we agree to support you.  
\* We work on the monthly basis  
\* No of hours of Support:  Based on customer need and the pricing also varies  
\* We support you to solve your technical problem and guide you in the right direction.

Explore Apache Spark Sample Resumes! Download & Edit, Get Noticed by Top Employers![**Download Now!**](https://mindmajix.com/apache-spark-sample-resumes)

#### **Subscribe For Free Demo**

Top of Form

Phone \*

E-mail Address \*

SUBSCRIBE

Bottom of Form

Free Demo for Corporate & Online Trainings.



##### **About The Author**

Ravindra Savaram is a Content Lead at [Mindmajix.com](https://mindmajix.com/). His passion lies in writing articles on the most popular IT platforms including Machine learning, DevOps, Data Science, Artificial Intelligence, RPA, Deep Learning, and so on. You can stay up to date on all these technologies by following him on [LinkedIn](https://www.linkedin.com/in/savaram-ravindra-48064641/)and [Twitter](https://twitter.com/s11ravindra).

[DMCA.com Protection Status](https://www.dmca.com/Protection/Status.aspx?ID=5b159fca-d195-4289-83e3-b69a0818b38d&refurl=https://mindmajix.com/apache-spark-interview-questions)

#### **Social Share**

[PREVIOUS](https://mindmajix.com/scrum-interview-questions)

[NEXT](https://mindmajix.com/mobile-application-testing-interview-questions)

#### **DROP US A QUERY**

Top of Form

Full Name

E-mail Address

Course Name



Phone Number \*

Message

CONTACT US

Bottom of Form

#### **Categories**

* [PHP](https://mindmajix.com/php)
* [SAP WM](https://mindmajix.com/sap-wm)
* [Data Modeling](https://mindmajix.com/data-modeling)
* [Cyber Security](https://mindmajix.com/cyber-security)
* [WORDPRESS](https://mindmajix.com/wordpress)
* [Git](https://mindmajix.com/git)
* [IBM Security QRadar SIEM](https://mindmajix.com/ibm-security-qradar-siem)
* [SAP FICA](https://mindmajix.com/sap-fica)
* [Tally](https://mindmajix.com/tally)
* [IBM Websphere Portal Developer](https://mindmajix.com/ibm-websphere-portal-developer)

[VIEW MORE](https://mindmajix.com/blog-category)

#### **Popular Courses in 2019**

#### [JENKINS Training](https://mindmajix.com/jenkins-training)

 921 Learners

#### [VMware Training](https://mindmajix.com/vmware-training)

 3402 Learners

#### [Selenium Training](https://mindmajix.com/selenium-training)

 4807 Learners

#### **Related Articles**

[How to Manipulate Structured Data Using Apache Spark SQL](https://mindmajix.com/spark/how-to-manipulate-structured-data-using-apache-spark-sql)May 10, 2017

[Apache Spark Tutorial](https://mindmajix.com/apache-spark-tutorial)May 16, 2017

[Streaming Big Data with Apache Spark](https://mindmajix.com/spark/streaming-big-data-with-apache-spark)May 24, 2017

[Exploring Apache Spark StandAlone Cluster](https://mindmajix.com/spark/exploring-apache-spark-standalone-cluster)May 17, 2017

Mindmajix  
  
Mindmajix - Online global training platform connecting individuals with the best trainers around the globe. With the diverse range of courses, Training Materials, Resume formats and On Job Support, we have it all covered to get into IT Career. Instructor Led Training - Made easy.

### Company

* [Home](https://mindmajix.com/)
* [About Us](https://mindmajix.com/about)
* [Courses](https://mindmajix.com/all-courses)
* [Sample Resumes](https://mindmajix.com/sample-resumes)
* [Blog](https://mindmajix.com/blog)
* [Contact Us](https://mindmajix.com/contact)
* [Reviews](https://mindmajix.com/reviews)
* [Write for us](https://mindmajix.com/write-for-us)

### Trending Courses

* [Tableau Training](https://mindmajix.com/tableau-training)
* [Oracle DBA training](https://mindmajix.com/oracle-dba-training)
* [Qlikview Training](https://mindmajix.com/qlikview-training)
* [Docker Training](https://mindmajix.com/docker-training)
* [JBoss Training](https://mindmajix.com/jboss-training)
* [Informatica Training](https://mindmajix.com/informatica-training)
* [Cassandra Training](https://mindmajix.com/apache-cassandra-training)
* [Blockchain Training](https://mindmajix.com/blockchain-training)

### Contact info

* 244 Fifth Avenue, Suite 1222 New York(NY) United States (US) - 10001
* USA : +1 917 456 8403
* [info@mindmajix.com](mailto:info@mindmajix.com)
* #811, 10th A Main, Suite No.506 1st Floor, Indira Nagar Bangalore, India - 560038
* India : +91 905 240 3388
* [info@mindmajix.com](mailto:info@mindmajix.com)

Copyright © 2019 Mindmajix Technologies Inc. All Rights Reserved

* [Privacy Policy](https://mindmajix.com/terms-of-service-and-privacy-policy)

* [Terms of use](https://mindmajix.com/terms-of-service-and-privacy-policy)

* [Refund Policy](https://mindmajix.com/terms-of-service-and-privacy-policy)
* [Call us on : USA - +1 201 378 0518 | IND - +91 9246 333 245](javascript:void(0);)
* [Drop us a Query](javascript:void(0);)
* [Call us on : USA - +1 201 378 0518 | IND - +91 9246 333 245](javascript:void(0);)
* [Drop us a Query](javascript:void(0);)

**Q) Apache Spark Vs Hadoop**

|  |  |  |  |
| --- | --- | --- | --- |
| **Spark Vs Hadoop** | | | |
| **Features** | **Spark** | **Hadoop** |  |
| Data processing | Part of hadoop, hence batch processing | Batch Processing even for high volumes |  |
| Streaming Engine | Apache spark straming - micro batches | Map-Reduce |  |
| Data Flow | Direct Acyclic Graph-DAG | Map-Reduce |  |
| Computation Model | Collect and process | Map-Reduce batch oriented model |  |
| Performance | Slow due to batch processing | Slow due to batch processing |  |
| Memory Management | Automatic memory management in latest release | Dynamic and static - Configurable |  |
| Fault Tolerance | Recovery available without extra code | Highly fault tolerant due to Map-Reduce |  |
| Scalability | Highly scalable - sSpark Cluster(8000 Nodes) | Highly scalable - Produces large number of nodes |  |

**Q1) What is Spark?**

Spark is a parallel data processing framework. It allows to develop fast, unified big data application combine batch, streaming and interactive analytics.

**Q2) Why Spark?**

Spark is the third generation distributed data processing platform. It’s unified bigdata solution for all bigdata processing problems such as batch , interacting, streaming processing.So it can ease many bigdata problems.

**Q3) What is RDD?**

Spark’s primary core abstraction is called Resilient Distributed Datasets. RDD is a collection of partitioned data that satisfies these properties. Immutable, distributed, lazily evaluated, catchable are common RDD properties.

**Q4) What is Immutable?**

Once created and assign a value, it’s not possible to change, this property is called Immutability. Spark is by default immutable, it does not allow updates and modifications. Please note data collection is not immutable, but data value is immutable.

**Q5) What is Distributed?**

RDD can automatically the data is distributed across different parallel computing nodes.

**Q6) What is Lazy evaluated?**

If you execute a bunch of programs, it’s not mandatory to evaluate immediately. Especially in Transformations, this Laziness is a trigger.

**Q6) What is Catchable?**

Keep all the data in-memory for computation, rather than going to the disk. So Spark can catch the data 100 times faster than Hadoop.

**Q7) What is Spark engine responsibility?**

Spark responsible for scheduling, distributing, and monitoring the application across the cluster.

**Q8) What are common Spark Ecosystems?**

* Spark SQL(Shark) for SQL developers,
* Spark Streaming for streaming data,
* MLLib for machine learning algorithms,
* GraphX for Graph computation,
* SparkR to run R on Spark engine,
* BlinkDB enabling interactive queries over massive data are common Spark ecosystems.  GraphX, SparkR, and BlinkDB are in the incubation stage.

**Q9) What is Partitions?**

Partition is a logical division of the data, this idea derived from Map-reduce (split). Logical data specifically derived to process the data. Small chunks of data also it can support scalability and speed up the process. Input data, intermediate data, and output data everything is Partitioned RDD.

Learn [Spark vs Hadoop](https://mindmajix.com/spark/whats-better-to-learn-first-spark-vs-hadoop) What's Better to Learn First.

**Q10) How spark partition the data?**

Spark use map-reduce API to do the partition the data. In Input format we can create number of partitions. By default HDFS block size is partition size (for best performance), but its’ possible to change partition size like Split.

**Q11) How Spark store the data?**

Spark is a processing engine, there is no storage engine. It can retrieve data from any storage engine like HDFS, S3 and other data resources.

**Q12) Is it mandatory to start Hadoop to run spark application?**

No not mandatory, but there is no separate storage in Spark, so it use local file system to store the data. You can load data from local system and process it, Hadoop or HDFS is not mandatory to run spark application.

**Q13) What is SparkContext?**

When a programmer creates a RDDs, SparkContext connect to the Spark cluster to create a new SparkContext object. SparkContext tell spark how to access the cluster. SparkConf is key factor to create programmer application.

**Q14) What is SparkCore functionalities?**

SparkCore is a base engine of apache spark framework. Memory management, fault tolarance, scheduling and monitoring jobs, interacting with store systems are primary functionalities of Spark.

**Q15) How SparkSQL is different from HQL and SQL?**

SparkSQL is a special component on the sparkCore engine that support SQL and HiveQueryLanguage without changing any syntax. It’s possible to join SQL table and HQL table.

**Q16) When did we use Spark Streaming?**

Spark Streaming is a real time processing of [streaming data](https://mindmajix.com/spark/streaming-big-data-with-apache-spark) API. Spark streaming gather streaming data from different resources like web server log files, social media data, stock market data or Hadoop ecosystems like Flume, and Kafka.

**Q17) How Spark Streaming API works?**

Programmer set a specific time in the configuration, with in this time how much data gets into the Spark, that data separates as a batch. The input stream (DStream) goes into spark streaming. Framework breaks up into small chunks called batches, then feeds into the spark engine for processing. Spark Streaming API passes that batches to the core engine. Core engine can generate the final results in the form of streaming batches. The output also in the form of batches. It can allows streaming data and batch data for processing.

**Q18) What is Spark MLlib?**

Mahout is a machine learning library for [Hadoop](https://mindmajix.com/hadoop-training), similarly MLlib is a Spark library. MetLib provides different algorithms, that algorithms scale out on the cluster for data processing. Most of the data scientists use this MLlib library.

**Q19) What is GraphX?**

GraphX is a Spark API for manipulating Graphs and collections. It unifies ETL, other analysis, and iterative graph computation. It’s fastest graph system, provides fault tolerance and ease of use without special skills.

**Q20) What is File System API?**

FS API can read data from different storage devices like HDFS, S3 or local FileSystem. Spark uses FS API to read data from different storage engines.

**Q21) Why Partitions are immutable?**

Every transformation generates new partition.  Partitions use HDFS API so that partition is immutable, distributed and fault tolerance. Partition also aware of data locality.

**Q22) What is Transformation in spark?**

Spark provides two special operations on RDDs called transformations and Actions. Transformation follows lazy operation and temporary hold the data until unless called the Action. Each transformation generates/return new RDD. Example of transformations: Map, flatMap, groupByKey, reduceByKey, filter, co-group, join, sortByKey, Union, distinct, sample are common spark transformations.

**Q23) What is Action in Spark?**

Actions are RDD’s operation, that value returns back to the spar driver programs, which kick off a job to execute on a cluster. Transformation’s output is an input of Actions. reduce, collect, takeSample, take, first, saveAsTextfile, saveAsSequenceFile, countByKey, foreach are common actions in Apache spark.

**Q24) What is RDD Lineage?**

Lineage is an RDD process to reconstruct lost partitions. Spark not replicate the data in memory, if data lost, Rdd use linege to rebuild lost data.Each RDD remembers how the RDD build from other datasets.

**Q25) What is Map and flatMap in Spark?**

The map is a specific line or row to process that data. In FlatMap each input item can be mapped to multiple output items (so the function should return a Seq rather than a single item). So most frequently used to return Array elements.

**Q26) What are broadcast variables?**

Broadcast variables let programmer keep a read-only variable cached on each machine, rather than shipping a copy of it with tasks. Spark supports 2 types of shared variables called broadcast variables (like Hadoop distributed cache) and accumulators (like Hadoop counters). Broadcast variables stored as Array Buffers, which sends read-only values to work nodes.

**Q27) What are Accumulators in Spark?**

Spark of-line debuggers called accumulators. Spark accumulators are similar to Hadoop counters, to count the number of events and what’s happening during job you can use accumulators. Only the driver program can read an accumulator value, not the tasks.

**Q28) How RDD persist the data?**

There are two methods to persist the data, such as persist() to persist permanently and cache() to persist temporarily in the memory. Different storage level options there such as MEMORY\_ONLY, MEMORY\_AND\_DISK, DISK\_ONLY and many more. Both persist() and cache() uses different options depends on the task.

**Q29) When do you use apache spark? OR  What are the benefits of Spark over Mapreduce?**

* Spark is really fast. As per their claims, it runs programs up to 100x faster than Hadoop MapReduce in memory, or 10x faster on disk. It aptly utilizes RAM to produce the faster results.
* In map reduce paradigm, you write many Map-reduce tasks and then tie these tasks together using Oozie/shell script. This mechanism is very time consuming and the map-reduce task has heavy latency.
* And quite often, translating the output out of one MR job into the input of another MR job might require writing another code because Oozie may not suffice.
* In Spark, you can basically do everything using single application/console (pyspark or scala console) and get the results immediately. Switching between ‘Running something on cluster’ and ‘doing something locally’ is fairly easy and straightforward. This also leads to less context switch of the developer and more productivity.
* Spark kind of equals to MapReduce and Oozie put together.

**Q30) Is there is a point of learning MapReduce, then?**

Yes. For the following reason:

* Mapreduce is a paradigm used by many big data tools including Spark. So, understanding the MapReduce paradigm and how to convert a problem into series of MR tasks is very important.
* When the data grows beyond what can fit into the memory on your cluster, the Hadoop Map-Reduce paradigm is still very relevant.
* Almost, every other tool such as Hive or Pig converts its query into MapReduce phases. If you understand the Mapreduce then you will be able to optimize your queries better.

**Q31) When running Spark on Yarn, do I need to install Spark on all nodes of Yarn Cluster?**

Since spark runs on top of Yarn, it utilizes yarn for the execution of its commands over the cluster’s nodes.  
So, you just have to install Spark on one node.

[Check Out Apache Spark Tutorials](https://mindmajix.com/apache-spark-tutorial)

**Q32) What are the downsides of Spark?**

Spark utilizes the memory. The developer has to be careful. A casual developer might make following mistakes:

* She may end up running everything on the local node instead of distributing work over to the cluster.
* She might hit some webservice too many times by the way of using multiple clusters.

The first problem is well tackled by Hadoop Map reduce paradigm as it ensures that the data your code is churning is fairly small a point of time thus you can make a mistake of trying to handle whole data on a single node.  
The second mistake is possible in Map-Reduce too. While writing Map-Reduce, user may hit a service from inside of map() or reduce() too many times. This overloading of service is also possible while using Spark.

**Q33) What is an RDD?**

The full form of RDD is resilience distributed dataset. It is a representation of data located on a network which is

* Immutable – You can operate on the rdd to produce another rdd but you can’t alter it.
* Partitioned / Parallel – The data located on RDD is operated in parallel. Any operation on RDD is done using multiple nodes.
* Resilience – If one of the node hosting the partition fails, another nodes takes its data.

RDD provides two kinds of operations: Transformations and Actions.

**Q34) What is Transformations?**

The transformations are the functions that are applied on an RDD (resilient distributed data set). The transformation results in another RDD. A transformation is not executed until an action follows.  
The example of transformations are:

1. map() – applies the function passed to it on each element of RDD resulting in a new RDD.
2. filter() – creates a new RDD by picking the elements from the current RDD which pass the function argument.

**Q35) What are Actions?**

An action brings back the data from the RDD to the local machine. Execution of an action results in all the previously created transformation. The example of actions are:

* reduce() – executes the function passed again and again until only one value is left. The function should take two argument and return one value.
* take() – take all the values back to the local node form RDD.

**Q36) 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);

**Q37) What is wrong with it? And How would you correct it?**

The average function is not commutative and associative;  
I would simply sum it and then divide by count.  
def sum(x, y):  
return x+y;  
total = myrdd.reduce(sum);  
avg = total / myrdd.count();  
The only problem with the above code is that the total might become very big thus over flow. So, I would rather divide each number by count and then sum in the following way.  
cnt = myrdd.count();  
def devideByCnd(x):  
return x/cnt;  
myrdd1 = myrdd.map(devideByCnd);  
avg = myrdd.reduce(sum);

**Q38) 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?**

# We would first load the file as RDD from HDFS on spark  
numsAsText = sc.textFile(“hdfs://hadoop1.knowbigdata.com/user/student/sgiri/mynumbersfile.txt”);  
# Define the function to compute the squares  
def toSqInt(str):  
v = int(str);  
return v\*v;  
#Run the function on spark rdd as transformation  
nums = numsAsText.map(toSqInt);  
#Run the summation as reduce action  
total = nums.reduce(sum)  
#finally compute the square root. For which we need to import math.  
import math;  
print math.sqrt(total);

**Q39) Is the following approach correct? Is the sqrtOfSumOfSq a valid reducer?**

numsAsText =sc.textFile(“hdfs://hadoop1.knowbigdata.com/user/student/sgiri/mynumbersfile.txt”);  
def toInt(str):  
return int(str);  
nums = numsAsText.map(toInt);  
def sqrtOfSumOfSq(x, y):  
return math.sqrt(x\*x+y\*y);  
total = nums.reduce(sum)  
import math;  
print math.sqrt(total);  
A: Yes. The approach is correct and **sqrtOfSumOfSq** is a valid reducer.

**Q40) Could you compare the pros and cons of the your approach (in Question 2 above) and my approach (in Question 3 above)?**

You are doing the square and square root as part of reduce action while I am squaring in map() and summing in reduce in my approach.  
My approach will be faster because in your case the reducer code is heavy as it is calling math.sqrt() and reducer code is generally executed approximately n-1 times the spark RDD.  
The only downside of my approach is that there is a huge chance of integer overflow because I am computing the sum of squares as part of map.

**Q41) If you have to compute the total counts of each of the unique words on spark, how would you go about it?**

#This will load the bigtextfile.txt as RDD in the spark lines = sc.textFile(“hdfs://hadoop1.knowbigdata.com/user/student/sgiri/bigtextfile.txt”);  
#define a function that can break each line into words  
def toWords(line):  
return line.split();  
# Run the toWords function on each element of RDD on spark as flatMap transformation.  
# We are going to flatMap instead of map because our function is returning multiple values.  
words = lines.flatMap(toWords);  
# Convert each word into (key, value) pair. Her key will be the word itself and value will be 1.  
def toTuple(word):  
return (word, 1);  
wordsTuple = words.map(toTuple);  
# Now we can easily do the reduceByKey() action.  
def sum(x, y):  
return x+y;  
counts = wordsTuple.reduceByKey(sum)  
# Now, print  
counts.collect()

**Q41) In a very huge text file, you want to just check if a particular keyword exists. How would you do this using Spark?**

lines = sc.textFile(“hdfs://hadoop1.knowbigdata.com/user/student/sgiri/bigtextfile.txt”);  
def isFound(line):  
if line.find(“mykeyword”) > -1:  
return 1;  
return 0;  
foundBits = lines.map(isFound);  
sum = foundBits.reduce(sum);  
if sum > 0:  
print “FOUND”;  
else:  
print “NOT FOUND”;

**Q42) Can you improve the performance of this code in previous answer?**

Yes. The search is not stopping even after the word we are looking for has been found. Our map code would keep executing on all the nodes which is very inefficient.  
We could utilize accumulators to report whether the word has been found or not and then stop the job. Something on these line:  
import thread, threading  
from time import sleep  
result = “Not Set”  
lock = threading.Lock()  
accum = sc.accumulator(0)  
def map\_func(line):  
#introduce delay to emulate the slowness  
sleep(1);  
if line.find(“Adventures”) > -1:  
accum.add(1);  
return 1;  
return 0;  
def start\_job():  
global result  
try:  
sc.setJobGroup(“job\_to\_cancel”, “some description”)  
lines = sc.textFile(“hdfs://hadoop1.knowbigdata.com/user/student/sgiri/wordcount/input/big.txt”);  
result = lines.map(map\_func);  
result.take(1);  
except Exception as e:  
result = “Cancelled”  
lock.release()  
def stop\_job():  
while accum.value < 3 :  
sleep(1);  
sc.cancelJobGroup(“job\_to\_cancel”)  
supress = lock.acquire()  
supress = thread.start\_new\_thread(start\_job, tuple())  
supress = thread.start\_new\_thread(stop\_job, tuple())  
supress = lock.acquire()  
[/tab]  
Facing technical problem in your current IT job, let us help you. MindMajix has highly technical people who can assist you in solving technical problems in your project.  
We have come across many developers in USA, Australia and other countries who have recently got the job but they are struggling to survive in the job because of less technical knowledge, exposure and the kind of work given to them.  
We are here to help you.

Let us know your profile and kind of help you are looking for and we shall do our best to help you out. The job support is provided by Mindmajix Technical experts who have more than 10 years of work experience on IT technologies landscape.  
How does the job support work?  
\* We see your project and technologies used, if we are 100% confident then we agree to support you.  
\* We work on the monthly basis  
\* No of hours of Support:  Based on customer need and the pricing also varies  
\* We support you to solve your technical problem and guide you in the right direction.

Explore Apache Spark Sample Resumes! Download & Edit, Get Noticed by Top Employers![**Download Now!**](https://mindmajix.com/apache-spark-sample-resumes)

#### **Subscribe For Free Demo**

Top of Form

Phone \*

E-mail Address \*

SUBSCRIBE

Bottom of Form

Free Demo for Corporate & Online Trainings.



##### **About The Author**

Ravindra Savaram is a Content Lead at [Mindmajix.com](https://mindmajix.com/). His passion lies in writing articles on the most popular IT platforms including Machine learning, DevOps, Data Science, Artificial Intelligence, RPA, Deep Learning, and so on. You can stay up to date on all these technologies by following him on [LinkedIn](https://www.linkedin.com/in/savaram-ravindra-48064641/)and [Twitter](https://twitter.com/s11ravindra).

[DMCA.com Protection Status](https://www.dmca.com/Protection/Status.aspx?ID=5b159fca-d195-4289-83e3-b69a0818b38d&refurl=https://mindmajix.com/apache-spark-interview-questions)

#### **Social Share**

[PREVIOUS](https://mindmajix.com/scrum-interview-questions)

[NEXT](https://mindmajix.com/mobile-application-testing-interview-questions)

#### **DROP US A QUERY**

Top of Form

Full Name

E-mail Address

Course Name



Phone Number \*

Message

CONTACT US

Bottom of Form

#### **Categories**

* [PHP](https://mindmajix.com/php)
* [SAP WM](https://mindmajix.com/sap-wm)
* [Data Modeling](https://mindmajix.com/data-modeling)
* [Cyber Security](https://mindmajix.com/cyber-security)
* [WORDPRESS](https://mindmajix.com/wordpress)
* [Git](https://mindmajix.com/git)
* [IBM Security QRadar SIEM](https://mindmajix.com/ibm-security-qradar-siem)
* [SAP FICA](https://mindmajix.com/sap-fica)
* [Tally](https://mindmajix.com/tally)
* [IBM Websphere Portal Developer](https://mindmajix.com/ibm-websphere-portal-developer)

[VIEW MORE](https://mindmajix.com/blog-category)

#### **Popular Courses in 2019**

#### [JENKINS Training](https://mindmajix.com/jenkins-training)

 921 Learners

#### [VMware Training](https://mindmajix.com/vmware-training)

 3402 Learners

#### [Selenium Training](https://mindmajix.com/selenium-training)

 4807 Learners

#### **Related Articles**

[How to Manipulate Structured Data Using Apache Spark SQL](https://mindmajix.com/spark/how-to-manipulate-structured-data-using-apache-spark-sql)May 10, 2017

[Apache Spark Tutorial](https://mindmajix.com/apache-spark-tutorial)May 16, 2017

[Streaming Big Data with Apache Spark](https://mindmajix.com/spark/streaming-big-data-with-apache-spark)May 24, 2017

[Exploring Apache Spark StandAlone Cluster](https://mindmajix.com/spark/exploring-apache-spark-standalone-cluster)May 17, 2017

Mindmajix  
  
Mindmajix - Online global training platform connecting individuals with the best trainers around the globe. With the diverse range of courses, Training Materials, Resume formats and On Job Support, we have it all covered to get into IT Career. Instructor Led Training - Made easy.

### Company

* [Home](https://mindmajix.com/)
* [About Us](https://mindmajix.com/about)
* [Courses](https://mindmajix.com/all-courses)
* [Sample Resumes](https://mindmajix.com/sample-resumes)
* [Blog](https://mindmajix.com/blog)
* [Contact Us](https://mindmajix.com/contact)
* [Reviews](https://mindmajix.com/reviews)
* [Write for us](https://mindmajix.com/write-for-us)

### Trending Courses

* [Tableau Training](https://mindmajix.com/tableau-training)
* [Oracle DBA training](https://mindmajix.com/oracle-dba-training)
* [Qlikview Training](https://mindmajix.com/qlikview-training)
* [Docker Training](https://mindmajix.com/docker-training)
* [JBoss Training](https://mindmajix.com/jboss-training)
* [Informatica Training](https://mindmajix.com/informatica-training)
* [Cassandra Training](https://mindmajix.com/apache-cassandra-training)
* [Blockchain Training](https://mindmajix.com/blockchain-training)

### Contact info

* 244 Fifth Avenue, Suite 1222 New York(NY) United States (US) - 10001
* USA : +1 917 456 8403
* [info@mindmajix.com](mailto:info@mindmajix.com)
* #811, 10th A Main, Suite No.506 1st Floor, Indira Nagar Bangalore, India - 560038
* India : +91 905 240 3388
* [info@mindmajix.com](mailto:info@mindmajix.com)

Copyright © 2019 Mindmajix Technologies Inc. All Rights Reserved

* [Privacy Policy](https://mindmajix.com/terms-of-service-and-privacy-policy)

* [Terms of use](https://mindmajix.com/terms-of-service-and-privacy-policy)

* [Refund Policy](https://mindmajix.com/terms-of-service-and-privacy-policy)
* [Call us on : USA - +1 201 378 0518 | IND - +91 9246 333 245](javascript:void(0);)
* [Drop us a Query](javascript:void(0);)
* [Call us on : USA - +1 201 378 0518 | IND - +91 9246 333 245](javascript:void(0);)
* [Drop us a Query](javascript:void(0);)

#### [**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

#### [**2.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

#### [**3.Repartition and coalesce difference?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

a. Using repartition spark can increase/decrease number of partitions of data.  
b. Using coalesce spark only can reduce the number of partitions of input data  
c. Reparition is not efficient than coalesce.

#### [**4.How to create a stream in spark**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

a. Dstream  
b. Structured stream.  
c. DirectStream

#### [**5.how to handle data shuffle in spark?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Using map partition and foreachpartition to replace all the collect methods in spark.

#### [**6. what are all the file formats supported by spark ?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Avro, parquest, json, xml, csv, tsv, snappy, orc, rc are the file formats supported by spark.  
Raw files as well as the structured file formats also supported by spark for efficient reading.

#### [**7. What are all the internal daemons used in spark?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

ACLs, BlockManager, Memestore, DAGScheduler, SparkContext, Driver, Worker,Executor, Tasks.

#### [**8. What is SPARK UI how to monitor a spark job?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Jobs- to view all the spark jobs  
Stages- to check the DAGs in spark  
Storages- to check all the cached RDDs  
Streaming- to check the cached RDDs  
Spark history server- to check all the logs of finished spark jobs.

#### [**9. Cluster manager in spark?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Standalone  
YARN-client and YARN-cluster (efficient for master-slave architecture)  
MESOS (Efficient for master master architecture container orchestration)  
KUBERNETES(container orchestration)

#### [**10. 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 is a cluster manager  
driver-memory is the actual memory size of the driver  
executor-memory is the actual memory size of the executor  
–num-executors is the total number of executors which are running at the worker nodes.  
–executor-cores number of individual processes that the executor memory can take up.

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

Dataframe is untyped (throw an exception at runtime in case of any error in the schema mismatch)  
Dataset is typed(throw an exception at compile time in case of any error in the schema mismatch)

#### [**12. what are all the memory tuning parameters and how to achieve parallelism in spark?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

a. leverage the Tungsten engine.  
b. spark job execution plan analysis.  
c. caching and data broadcasting and accumulating the data using multiple optimization techniques in spark.

#### [**13. spark history server how to start?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

./sbin/start-history-server.sh –properties-file history.properties  
Once you successfully start this server then you can check all the logs of all the containers in spark jobs.

#### [**14. 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)

#### [**15. what is udfs and how to use it ?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

UDFs are user defined functions and in which are used to make a certain changes across all the rows in a specific columns like timestamp to day conversion, timestamp to week conversion.

#### [**16. code sample to read a data from text file?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

from pyspark import SparkContext  
SparkContext.stop(sc)  
sc = SparkContext(“local”,”besant”) sqlContext = SQLContext(sc)  
sc.textFile(filename)

#### [**17. code sample to read a data from mysql ?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

spark.read.format(‘jdbc’).options(driver=’com.mysql.jdbc.Driver’,url=”””jdbc:mysql://<host>:3306/<>db?user=<usr>&password=<pass>”””,dbtable=’besant’,numPartitions=4 ).load()

#### [**18.What is Spark?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Start is a parallel information preparing structure. It permits to grow quick, brought together huge information application consolidate cluster, gushing and intuitive examination.

#### [**19.Why Spark?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Start is the third era circulated information preparing stage. It’s brought together huge information answer for every single enormous datum handling issues, for example, bunch , collaborating, gushing preparing. So it can ease numerous enormous information issues.

#### [**20. what is RDD?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Start’s essential center deliberation is called Resilient Distributed Datasets. RDD is an accumulation of apportioned information that fulfills these properties. Unchanging, conveyed, apathetically assessed, catchable are normal RDD properties.

#### [**21.What is Immutable?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Once made and relegate an esteem, it’s unrealistic to change, this property is called Immutability. Start is of course unchanging, it doesn’t permit updates and alterations. It would be ideal if you note information gathering isn’t unchanging, however information esteem is permanent.

#### [**22. What is Distributed?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

RDD can consequently the information is appropriated crosswise over various parallel processing hubs.

#### [**23.What is Lazy assessed?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

In the event that you execute a pack of projects, it’s not obligatory to assess instantly. Particularly in Transformations, this Laziness is a trigger.

#### [**24.What is Catchable?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Keep every one of the information in-memory for calculation, instead of heading off to the circle. So Spark can get the information multiple times quicker than Hadoop.

#### [**25.what is Spark motor duty?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Start in charge of planning, conveying, and checking the application over the group.

#### [**26.What are normal Spark Ecosystems?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

* Spark SQL(Shark) for SQL designers,
* Spark Streaming for spilling information,
* MLLib for machine learning calculations,
* GraphX for Graph calculation,
* SparkR to run R on Spark motor,
* BlinkDB empowering intelligent inquiries over gigantic information are normal Spark biological systems. GraphX, SparkR, and BlinkDB are in the brooding stage.

#### [**27. What is Partitions?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Parcel is a consistent division of the information, this thought got from Map-diminish (split). Consistent information explicitly inferred to process the information. Little lumps of information additionally it can bolster adaptability and accelerate the procedure. Information, moderate information, and yield information everything is Partitioned RDD.

#### [**28.How start parcel the information?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Start utilize outline API to do the segment the information. In Input design we can make number of allotments. As a matter of course HDFS square size is segment estimate (for best execution), however its conceivable to change parcel measure like Split.

#### [**29.How Spark store the information?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Start is a preparing motor, there is no capacity motor. It can recover information from any capacity motor like HDFS, S3 and other information assets.

#### [**30.is it compulsory to begin Hadoop to run start application?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

No not compulsory, but rather there is no different stockpiling in Spark, so it utilize nearby record framework to store the information. You can stack information from neighborhood framework and process it, Hadoop or HDFS isn’t required to run start application.

#### [**31.what is Spark Context?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

At the point when a software engineer makes a RDDs, Spark Context interface with the Spark group to make another Spark Context protest. Start Context advise start how to get to the bunch. SparkConf is key factor to make software engineer application.

#### [**32.What is Spark Core functionalities?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Start Core is a base motor of apache start structure. Memory the executives, blame tolarance, planning and observing occupations, associating with store frameworks are essential functionalities of Spark.

#### [**33.How SparkSQL is not the same as HQL and SQL?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

SparkSQL is a unique segment on the sparkCore motor that help SQL and HiveQueryLanguage without changing any sentence structure. It’s conceivable to join SQL table and HQL table.

#### [**34.When did we utilize Spark Streaming?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Start Streaming is a continuous handling of gushing information API. Start gushing assemble spilling information from various assets like web server log records, internet based life information, securities exchange information or Hadoop biological systems like Flume, and Kafka.

#### [**35.How Spark Streaming API works?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Software engineer set an explicit time in the design, with in this time how much information gets into the Spark, that information isolates as a bunch. The information stream (DStream) goes into start gushing. System separates into little pieces called groups, at that point encourages into the start motor for preparing. Start Streaming API passes that clumps profoundly motor. Center motor can create the last outcomes through spilling bunches. The yield likewise as groups. It can permits gushing information and clump information for handling.

#### [**36. What is Spark MLlib?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Mahout is a machine learning library for Hadoop, likewise MLlib is a Spark library. MetLib gives distinctive calculations, that calculations scale out on the bunch for information handling. The vast majority of the information researchers utilize this MLlib library.

#### [**37.What is GraphX?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

GraphX is a Spark API for controlling Graphs and accumulations. It brings together ETL, different investigation, and iterative chart calculation. It’s quickest chart framework, gives adaptation to non-critical failure and usability without uncommon abilities.

#### [**38.What is File System API?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

FS API can scrutinize data from different limit contraptions like HDFS, S3 or adjacent FileSystem. Begin uses FS API to scrutinize data from different limit engines.

#### [**39.Why Partitions are constant?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Each change creates new portion. Distributions use HDFS API so fragment is perpetual, flowed and adjustment to inner disappointment. Portion moreover aware of data region.

#### [**40.What is Transformation in begin?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Begin gives two one of a kind exercises on RDDs called changes and Actions. Change seeks after lazy assignment and short lived hold the data until the point that with the exception of whenever called the Action. Each change makes/return new RDD. Instance of changes: Map, flatMap, groupByKey, reduceByKey, channel, co-gathering, join, sortByKey, Union, specific, precedent are fundamental begin changes.

#### [**41.What is Action in Spark?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Exercises are RDD’s undertaking, that regard returns back to the battle driver programs, which kick off work to execute on a group. Change’s yield is a commitment of Actions. decrease, accumulate, takeSample, take, first, saveAsTextfile, saveAsSequenceFile, countByKey, foreach are ordinary exercises in Apache begin.

#### [**42.What is RDD Lineage?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Family is a RDD technique to reproduce lost bundles. Begin not reproduce the data in memory, if data lost, Rdd use linege to patch up lost data.Each RDD reviews how the RDD function from various datasets.

#### [**43.What is Map and flatMap in Spark?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

The guide is an unequivocal line or line to process that data. In FlatMap every data thing can be mapped to different yield things (so the limit ought to reestablish a Seq instead of a singular thing). So most a great part of the time used to return Array segments.

#### [**44.What are conveyed factors?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Convey factors let programming engineer keep a read-simply factor held on each machine, rather than conveyance a copy of it with assignments. Begin supports 2 sorts of shared variables called convey factors (like Hadoop flowed store) and aggregators (like Hadoop counters). Impart factors set away as Array Buffers, which sends read-only characteristics to work center points.

#### [**45. What are Accumulators in Spark?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Beginning of-line debuggers called gatherers. Begin aggregators resemble Hadoop counters, to check the amount of events and what’s happening in the midst of business you can use authorities. Simply the driver program can examine a gatherer regard, not the assignments.

#### [**46.How RDD hang on the data?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

There are two strategies to bear the data, for instance, hang on() to drive forward forever and hold() to proceed quickly in the memory. Unmistakable limit level decisions there, for instance, MEMORY\_ONLY, MEMORY\_AND\_DISK, DISK\_ONLY and some more. Both endure() and hold() uses assorted choices depends upon the task.

#### [**47.When do you use apache begin? Or on the other hand What are the benefits of Spark over Mapreduce?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

* Spark is amazingly speedy. As indicated by their cases, it runs programs up to 100x speedier than Hadoop MapReduce in memory, or 10x faster on circle. It reasonably utilizes RAM to make the snappier results.
* In diagram perspective, you make many Map-reduce errands and a while later incorporate these assignments using Oozie/shell content. This framework is particularly monotonous and the guide reduce task has significant torpidity.
* And consistently, decoding the yield out of one MR work into the commitment of another MR occupation may require creating another code in light of the way that Oozie may not work.
* In Spark, you can basically do everything using single application/bolster (pyspark or scala comfort) and get the results in a split second. Trading between ‘Running something on group’ and ‘achieving something locally’ is truly straightforward and clear. This also prompts less setting switch of the fashioner and more prominent benefit.
* Spark kind of reciprocals to MapReduce and Oozie set up together.

#### [**48.Is there is a point of learning MapReduce, by then?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Surely. For the going with reason:

* Mapreduce is a perspective used by various gigantic data instruments including Spark. Accordingly, understanding the MapReduce perspective and how to change over an issue into game plan of MR endeavors is basic.
* When the data creates past what can fit into the memory on your gathering, the Hadoop Map-Reduce perspective is still outstandingly noteworthy.
* Almost, each other instrument, for instance, Hive or Pig changes over its inquiry into MapReduce stages. If you understand the MapReduce, you will have the ability to streamline your request better.

#### [**49.When running Spark on Yarn, do I have to introduce Spark on every single focus purpose of Yarn Cluster?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Since start keeps running indeed Yarn, it uses the yarn for the execution of its directions over the group’s hubs.

#### [**50.What is DAG and Stage in spark processing?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

FYI the above program, the overall execution plan is as per the DAG scheduler.

For each and every method execution is optimized as per the stages.

#### [**51.code sample to read a data from mysql ?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

spark.read.format(&#39;jdbc&#39;).options(driver=&#39;com.mysql.jdbc.Driver&#39;,url=&quot;&quot;&quot;jdbc:mysql://&lt;host&gt;:3306/&lt;&gt;db

?user=&lt;usr&gt;&amp;password=&lt;pass&gt;&quot;&quot;&quot;,dbtable=’besant&#39;,numPartitions=4 ).load()

#### [**52. What is hang on() ?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

#### [**53. Write typical work procedure of a Spark program?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Each Spark program and shell session will fill in as seeks after:

* Create some data RDDs from external data.
* Transform them to describe new RDDs using changes like channel().
* Ask Spark to persevere through() any widely appealing RDDs that ought to be reused.
* Launch exercises, for instance, count() and first() to initiate a parallel figuring, which is then overhauled and executed by Spark.

#### [**54. Difference among cache() and persist()?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

With cache(), you utilize just the default stockpiling level MEMORY\_ONLY. With persist(), you can indicate which stockpiling level you want.So ache() is equivalent to calling hold on() with the default stockpiling level.Spark has numerous dimensions of determination to browse dependent on what our objectives are.The default continue() will store the information in the JVM pile as unserialized objects. When we work information out to circle, that information is additionally dependably serialized.Different dimensions of tirelessness are MEMORY\_ONLY, MEMORY\_ONLY\_SER, MEMORY\_AND\_DISK, MEMORY\_AND\_DISK\_SER, DISK\_ONLY.

#### [**55. What is lineage graph?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

As you get new RDDs from one another utilizing changes, Spark monitors the arrangement of conditions between various RDDs, called the ancestry chart. It utilizes this data to process each RDD on interest and to recoup lost information if part of a persevering RDD is lost.

#### [**56. Difference among map()and flatMap()?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

The map()change takes in a capacity and applies it to every component in the RDD with the consequence of the capacity being the new estimation of every component in the subsequent RDD. Some of the time we need to deliver numerous yield components for each information component. The task to do this is called flatMap(). Similarly as with guide(), the capacity we give to flatMap() is called independently for every component in our info RDD. Rather than restoring a solitary component, we return an iterator with our arrival esteems.

#### [**57. What is reduce() action?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

It takes a capacity that works on two components of the sort in your RDD and returns another component of a similar kind. A straightforward case of such a capacity is +, which we can use to whole our RDD. With lessen(), we can undoubtedly entirety the components of our RDD, tally the quantity of components, and perform different sorts of conglomerations.

#### [**58. What is Pair RDD?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Sparkle gives unique tasks on RDDs containing key/esteem sets. These RDDs are called pair RDDs. Pair RDDs are a valuable building obstruct in numerous projects, as they uncover tasks that enable you to follow up on each key in parallel.For precedent, pair RDDs have a reduceByKey() strategy that can total information independently for each key, and a join() technique that can consolidate two RDDs by gathering components with a similar key.

#### [**59. What are Accumulators?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Accumulators, gives a straightforward language structure to accumulating esteems from specialist hubs back to the driver program. A standout amongst the most widely recognized employments of aggregators is to tally occasions that happen amid occupation execution for investigating purposes.

#### [**60. What is Broadcast Variables?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Sparkle’s second kind of shared variable, communicate factors, enables the program to effectively send an expansive, read-just an incentive to all the laborer hubs for use in at least one Spark tasks. They prove to be useful, for instance, if your application needs to send an extensive, read-just query table to every one of the hubs.

#### [**61. What is Piping?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Spark gives a pipe() technique on RDDs. Sparkle’s pipe() gives us a chance to compose parts of occupations utilizing any language we need as long as it can peruse and keep in touch with Unix standard streams. With pipe(), you can compose a change of a RDD that peruses each RDD component from standard contribution as a String, controls that String anyway you like, and afterward composes the result(s) as Strings to standard yield.

#### [**62. What are advantages of Spark over MapReduce?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

* Due to the accessibility of in-memory preparing, Spark executes the handling around 10-100x quicker than Hadoop MapReduce. MapReduce makes utilization of diligence stockpiling for any of the information preparing undertakings.
* Unlike Hadoop, Spark gives in-manufactured libraries to play out different errands structure a similar center like group handling, Steaming, Machine learning, Interactive SQL questions. Be that as it may, Hadoop just backings group preparing.
* Hadoop is profoundly circle subordinate while Spark advances reserving and in-memory information stockpiling
* Spark is equipped for performing calculations on different occasions on the equivalent dataset. This is called iterative calculation while there is no iterative processing executed by Hadoop.

#### [**63. Is there any advantage of learning MapReduce, at that point?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Truly, MapReduce is a worldview utilized by numerous huge information devices including Spark too. It is amazingly pertinent to utilize MapReduce when the information becomes greater and greater. Most instruments like Pig and Hive convert their questions into MapReduce stages to streamline them better.

#### [**64. What is Spark Executor?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

At the point when SparkContext associate with a bunch chief, it obtains an Executor on hubs in the group. Agents are Spark forms that run calculations and store the information on the laborer hub. The last errands by SparkContext are exchanged to agents for their execution.

#### [**65. Name sorts of Cluster Managers in Spark?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

The Spark structure underpins three noteworthy kinds of Cluster Managers:

* Standalone: an essential administrator to set up a bunch
* Apache Mesos: summed up/ordinarily utilized group chief, likewise runs Hadoop MapReduce and different applications
* Yarn: in charge of asset the board in Hadoop

#### [**66. What are Executors?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Flash agents are specialist forms in charge of running the individual undertakings in a given Spark work. Agents are propelled once toward the start of a Spark application and commonly keep running for the whole lifetime of an application.Executors have two jobs. To begin with, they run the assignments that make up the application and return results to the driver.Second, they give in-memory stockpiling to RDDs that are stored by client programs.

#### [**67. What are the means that happen when you run a Spark application on a group?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

The client presents an application utilizing flash submit.

* Spark-submit dispatches the driver program and conjures the principle() technique indicated by the client.
* The driver program contacts the bunch chief to request assets to dispatch agents.
* The group director dispatches agents in the interest of the driver program.
* The driver process goes through the client application. In light of the RDD activities and changes in the program, the driver sends work to agents as errands.
* Tasks are kept running on agent procedures to register and spare outcomes.
* If the driver’s primary() technique ways out or it calls SparkContext.stop(),it will end the agents and discharge assets from the bunch director.

#### [**68. What is Spark SQL?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Sparkle SQL is a module in Apache Spark that incorporates social processing(e.g., decisive inquiries and advanced stockpiling) with Spark’s procedural programming API. Flash SQL makes two principle additions.First, it offers a lot more tightly joining among social and procedural handling, through a decisive DataFrame API.Second, it incorporates an exceptionally extensible analyzer, Catalyst.

Enormous information applications require a blend of preparing strategies, information sources and capacity groups. The most punctual frameworks intended for these remaining burdens, for example, MapReduce, gave clients an amazing, however low-level, procedural programming interface. Programming such frameworks was grave and required manual enhancement by the client to accomplish elite. Therefore, various new frameworks tried to give an increasingly profitable client experience by offering social interfaces to huge information. Frameworks like Pig, Hive and Shark all exploit revelatory inquiries to give more extravagant programmed improvements.

#### [**68. What is Spark SQL?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

Sparkle SQL is a module in Apache Spark that incorporates social processing(e.g., decisive inquiries and advanced stockpiling) with Spark’s procedural programming API. Flash SQL makes two principle additions.First, it offers a lot more tightly joining among social and procedural handling, through a decisive DataFrame API.Second, it incorporates an exceptionally extensible analyzer, Catalyst.

Enormous information applications require a blend of preparing strategies, information sources and capacity groups. The most punctual frameworks intended for these remaining burdens, for example, MapReduce, gave clients an amazing, however low-level, procedural programming interface. Programming such frameworks was grave and required manual enhancement by the client to accomplish elite. Therefore, various new frameworks tried to give an increasingly profitable client experience by offering social interfaces to huge information. Frameworks like Pig, Hive and Shark all exploit revelatory inquiries to give more extravagant programmed improvements.

#### [**69**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

#### [**69. What is a construction RDD/DataFrame?**](https://www.besanttechnologies.com/apache-spark-interview-questions-and-answers)

A SchemaRDD is a RDD made out of Row objects with extra construction data of the sorts in every segment. Column objects are only wrappers around varieties of fundamental sorts (e.g., whole numbers and strings).

Please highlight which part of the following code will be executed on the master, and which will be run on each worker node.

val formatter: DateTimeFormatter = DateTimeFormatter.ofPattern("yyyy/MM")

def getEventCountOnWeekdaysPerMonth(data: RDD[(LocalDateTime, Long)]): Array[(String, Long)] = {

val result = data

.filter(e => e.\_1.getDayOfWeek.getValue < DayOfWeek.SATURDAY.getValue)

.map(mapDateTime2Date)

.reduceByKey(\_ + \_)

.collect()

result

.map(e => (e.\_1.format(formatter), e.\_2))

}

private def mapDateTime2Date(v: (LocalDateTime, Long)): (LocalDate, Long) = {

(v.\_1.toLocalDate.withDayOfMonth(1), v.\_2)

}

View the answer →



Describe the following code and what the output will be.

case class User(userId: Long, userName: String)

case class UserActivity(userId: Long, activityTypeId: Int, timestampEpochSec: Long)

val LoginActivityTypeId = 0

val LogoutActivityTypeId = 1

private def readUserData(sparkSession: SparkSession): RDD[User] = {

sparkSession.sparkContext.parallelize(

Array(

User(1, "Doe, John"),

User(2, "Doe, Jane"),

User(3, "X, Mr."))

)

}

private def readUserActivityData(sparkSession: SparkSession): RDD[UserActivity] = {

sparkSession.sparkContext.parallelize(

Array(

UserActivity(1, LoginActivityTypeId, 1514764800L),

UserActivity(2, LoginActivityTypeId, 1514808000L),

UserActivity(1, LogoutActivityTypeId, 1514829600L),

UserActivity(1, LoginActivityTypeId, 1514894400L))

)

}

def calculate(sparkSession: SparkSession): Unit = {

val userRdd: RDD[(Long, User)] =

readUserData(sparkSession).map(e => (e.userId, e))

val userActivityRdd: RDD[(Long, UserActivity)] =

readUserActivityData(sparkSession).map(e => (e.userId, e))

val result = userRdd

.leftOuterJoin(userActivityRdd)

.filter(e => e.\_2.\_2.isDefined && e.\_2.\_2.get.activityTypeId == LoginActivityTypeId)

.map(e => (e.\_2.\_1.userName, e.\_2.\_2.get.timestampEpochSec))

.reduceByKey((a, b) => if (a < b) a else b)

result

.foreach(e => println(s"${e.\_1}: ${e.\_2}"))

}

View the answer →

**Find top Spark developers today.** Toptal can match you with the best engineers to finish your project.

[**HIRE TOPTAL’S SPARK DEVELOPERS**](https://www.toptal.com/hire?interested_in=developers&skill=spark)



The following code provides two prepared dataframes with the following structure:

DF1: userId, userName

DF2: userId, pageId, timestamp, eventType

Add the code to join the two dataframes and count the number of events per userName. It should output in the format userName; totalEventCount and only for users that have events.

def calculate(sparkSession: SparkSession): Unit = {

val UserIdColName = "userId"

val UserNameColName = "userName"

val CountColName = "totalEventCount"

val userRdd: DataFrame = readUserData(sparkSession)

val userActivityRdd: DataFrame = readUserActivityData(sparkSession)

val result = userRdd

.repartition(col(UserIdColName))

// ???????????????

.select(col(UserNameColName))

// ???????????????

result.show()

}

View the answer →



You have a cluster of 10 nodes with 24 CPU cores available on each node.

The following code works but might crash on large data sets, or at least will not leverage the full processing power of the cluster. Which is the problematic part and how would you adapt it?

def calculate(sparkSession: SparkSession): Unit = {

val NumNodes = 10

val userActivityRdd: RDD[UserActivity] =

readUserActivityData(sparkSession)

.repartition(NumNodes)

val result = userActivityRdd

.map(e => (e.userId, 1L))

.reduceByKey(\_ + \_)

result

.take(1000)

}

View the answer →



The following code registers a user-defined function (UDF) and uses it in a query. (The general business logic is irrelevant to the question.) What’s problematic about the code such that it might tear down the whole cluster, and how can it be solved?

(Hint: It has to do with the usage of the categoryNodesWithChildren Map variable.)

def calculate(sparkSession: SparkSession): Unit = {

val UserIdColumnName = "userId"

val CategoryIdColumnName = "categoryId"

val NumActionsColumnName = "numActions"

val OtherCategoryIdColumnName = "otherCategoryId"

val OtherNumActionsColumnName = "otherNumActions"

val categoryNodesWithChildren: Map[Int, Set[Int]] =

Map(0 -> Set(1, 2, 3),

1 -> Set(4, 5),

2 -> Set(6, 7),

3 -> Set(8),

7 -> Set(9, 10)

)

sparkSession.udf.register("isChildOf", (nodeId: Int, parentNodeId: Int) =>

nodeId != parentNodeId && categoryNodesWithChildren.getOrElse(nodeId, Set[Int]()).contains(parentNodeId))

val userCategoryActions = readUserCategoryActions(sparkSession)

val otherUserCategoryActions = userCategoryActions

.select(

col(UserIdColumnName),

col(CategoryIdColumnName).alias(OtherCategoryIdColumnName),

col(NumActionsColumnName).alias(OtherNumActionsColumnName)

)

val joinedUserActions = userCategoryActions

.join(otherUserCategoryActions, UserIdColumnName)

.where("!(isChildOf(categoryId,otherCategoryId) or isChildOf(otherCategoryId,categoryId))")

.groupBy(UserIdColumnName, CategoryIdColumnName, OtherCategoryIdColumnName)

.sum(OtherNumActionsColumnName)

.withColumnRenamed(s"sum($OtherNumActionsColumnName)", OtherNumActionsColumnName)

joinedUserActions.show()

}

View the answer →



Complete the missing SQL query to return the result as shown based on the example data:

case class User(userId: Long, userName: String)

case class UserActivity(userId: Long, activityTypeId: Int, timestampEpochMs: Long)

val LoginActivityTypeId = 0

val LogoutActivityTypeId = 1

private def readUserData(sparkSession: SparkSession): DataFrame = {

sparkSession.createDataFrame(

sparkSession.sparkContext.parallelize(

Array(

User(1, "Doe, John"),

User(2, "Doe, Jane"),

User(3, "X, Mr."))

)

)

}

private def readUserActivityData(sparkSession: SparkSession): DataFrame = {

sparkSession.createDataFrame(

sparkSession.sparkContext.parallelize(

Array(

UserActivity(1, LoginActivityTypeId, 1514764800000L),

UserActivity(2, LoginActivityTypeId, 1514808000000L),

UserActivity(1, LogoutActivityTypeId, 1514829600000L),

UserActivity(1, LoginActivityTypeId, 1514894400000L))

)

)

}

def calculate(sparkSession: SparkSession): Unit = {

val UserTableName = "user"

val UserActivityTableName = "userActivity"

val userDf: DataFrame = readUserData(sparkSession)

val userActivityDf: DataFrame = readUserActivityData(sparkSession)

userDf.createOrReplaceTempView(UserTableName)

userActivityDf.createOrReplaceTempView(UserActivityTableName)

val result = sparkSession

.sql(s"SELECT ...")

result.show()

}

The output should be this:

| **userName** | **firstLogin** |
| --- | --- |
| Doe, John | 1514764800000 |
| Doe, Jane | 1514808000000 |

View the answer →



Describe how model creation works with MLlib and how the model is applied.

View the answer →



What are the elements the GraphX library works with, and how are they created from an RDD? Complete the following code to calculate the page ranks.

def calculate(sparkSession: SparkSession): Unit = {

val pageRdd: RDD[(???, Page)] =

readPageData(sparkSession)

.map(e => (e.pageId, e))

.cache()

val pageReferenceRdd: RDD[???[PageReference]] = readPageReferenceData(sparkSession)

val graph = Graph(pageRdd, pageReferenceRdd)

val PageRankTolerance = 0.005

val ranks = graph.???

ranks.take(1000)

.foreach(println)

}

The result will be a list of tuples that looks like this:

(1,1.4537951595091907)

(2,0.7731024202454048)

(3,0.7731024202454048)

View the answer →



Compare Spark Streaming to Kafka Streams and Flink. Highlight the differences and advantages of each technology, and for which use cases each of the stream processing frameworks works best.

View the answer →

\* There is more to interviewing than tricky technical questions, so these are intended merely as a guide. Not every “A” candidate worth hiring will be able to answer them all, nor does answering them all guarantee an “A” candidate. At the end of the day, [hiring remains an art, a science — and a lot of work](https://www.toptal.com/freelance/in-search-of-the-elite-few-finding-and-hiring-the-best-developers-in-the-industry).

**Submit an interview**

## Top 20 Apache Spark Interview Questions

### 1. What is Apache Spark?

**A.**Apache Spark is a cluster computing framework which runs on a cluster of commodity hardware and performs data unification i.e., reading and writing of wide variety of data from multiple sources. In Spark, a task is an operation that can be a map task or a reduce task. Spark Context handles the execution of the job and also provides API’s in different languages i.e., Scala, Java and Python to develop applications and faster execution as compared to MapReduce.

### 2. How is Spark different from MapReduce? Is Spark faster than MapReduce?

**A.**Yes, Spark is faster than MapReduce. There are few important reasons why Spark is faster than MapReduce and some of them are below:

* There is no tight coupling in Spark i.e., there is no mandatory rule that reduce must come after map.
* Spark tries to keep the data “in-memory” as much as possible.

In MapReduce, the intermediate data will be stored in HDFS and hence takes longer time to get the data from a source but this is not the case with Spark.

### 3. Explain the Apache Spark Architecture. How to Run Spark applications?

* Apache Spark application contains two programs namely a Driver program and Workers program.
* A cluster manager will be there in-between to interact with these two cluster nodes. Spark Context will keep in touch with the worker nodes with the help of Cluster Manager.
* Spark Context is like a master and Spark workers are like slaves.
* Workers contain the executors to run the job. If any dependencies or arguments have to be passed then Spark Context will take care of that. RDD’s will reside on the Spark Executors.
* You can also run Spark applications locally using a thread, and if you want to take advantage of distributed environments you can take the help of S3, HDFS or any other storage system.

### 4. What is RDD?

**A.**RDD stands for Resilient Distributed Datasets (RDDs). If you have large amount of data, and is not necessarily stored in a single system, all the data can be distributed across all the nodes and one subset of data is called as a partition which will be processed by a particular task. RDD’s are very close to input splits in MapReduce.

More about [*RDD*](https://acadgild.com/blog/spark-rdd-introduction-basic-operations-rdd) Here.

### 5. What is the role of coalesce () and repartition () in Map Reduce?

**A.**Both coalesce and repartition are used to modify the number of partitions in an RDD but Coalesce avoids full shuffle.

If you go from 1000 partitions to 100 partitions, there will not be a shuffle, instead each of the 100 new partitions will claim 10 of the current partitions and this does not require a shuffle.

Repartition performs a coalesce with shuffle. Repartition will result in the specified number of partitions with the data distributed using a hash practitioner.

### 6. How do you specify the number of partitions while creating an RDD? What are the functions?

**A.**You can specify the number of partitions while creating a RDD either by using the sc.textFile or by using parallelize functions as follows:

Val rdd = sc.parallelize(data,4)

val data = sc.textFile(“path”,4)

### 7. What are actions and transformations?

**A.**Transformations create new RDD’s from existing RDD and these transformations are lazy and will not be executed until you call any action.

Eg: map(), filter(), flatMap(), etc.,

Actions will return results of an RDD.

Eg: reduce(), count(), collect(), etc.,

### 8. What is Lazy Evaluation?

**A.**If you create any RDD from an existing RDD that is called as transformation and unless you call an action your RDD will not be materialized the reason is Spark will delay the result until you really want the result because there could be some situations you have typed something and it went wrong and again you have to correct it in an interactive way it will increase the time and it will create un-necessary delays. Also, Spark optimizes the required calculations and takes intelligent decisions which is not possible with line by line code execution. Spark recovers from failures and slow workers.

### 9. Mention some Transformations and Actions

**A. Transformations**map (), filter(), flatMap()

**Actions**

reduce(), count(), collect()

### 10. What is the role of cache() and persist()?

**A.**Whenever you want to store a RDD into memory such that the RDD will be used multiple times or that RDD might have created after lots of complex processing in those situations, you can take the advantage of Cache or Persist.

You can make an RDD to be persisted using the persist() or cache() functions on it. The first time it is computed in an action, it will be kept in memory on the nodes.

When you call persist(), you can specify that you want to store the RDD on the disk or in the memory or both. If it is in-memory, whether it should be stored in serialized format or de-serialized format, you can define all those things.

cache() is like persist() function only, where the storage level is set to memory only.

### 11. What are Accumulators?

**A.**Accumulators are the write only variables which are initialized once and sent to the workers. These workers will update based on the logic written and sent back to the driver which will aggregate or process based on the logic.

Only driver can access the accumulator’s value. For tasks, Accumulators are write-only. For example, it is used to count the number errors seen in RDD across workers.

### 12. What are Broadcast Variables?

**A.**Broadcast Variables are the read-only shared variables. Suppose, there is a set of data which may have to be used multiple times in the workers at different phases, we can share all those variables to the workers from the driver and every machine can read them.

### 13. What are the optimizations that developer can make while working with spark?

**A.**Spark is memory intensive, whatever you do it does in memory.

Firstly, you can adjust how long spark will wait before it times out on each of the phases of data locality (data local –> process local –> node local –> rack local –> Any).

Filter out data as early as possible. For caching, choose wisely from various storage levels.

Tune the number of partitions in spark.

### 14. What is Spark SQL?

**A.**Spark SQL is a module for structured data processing where we take advantage of SQL queries running on the datasets.

### 15. What is a Data Frame?

**A.**A data frame is like a table, it got some named columns which organized into columns. You can create a data frame from a file or from tables in hive, external databases SQL or NoSQL or existing RDD’s. It is analogous to a table.

### 16. How can you connect Hive to Spark SQL?

**A.**The first important thing is that you have to place hive-site.xml file in conf directory of Spark.

Then with the help of Spark session object we can construct a data frame as,

result = spark.sql(“select \* from <hive\_table>”)

### 17. What is GraphX?

**A.**Many times you have to process the data in the form of graphs, because you have to do some analysis on it. It tries to perform Graph computation in Spark in which data is present in files or in RDD’s.

GraphX is built on the top of Spark core, so it has got all the capabilities of Apache Spark like fault tolerance, scaling and there are many inbuilt graph algorithms also. GraphX unifies ETL, exploratory analysis and iterative graph computation within a single system.

You can view the same data as both graphs and collections, transform and join graphs with RDD efficiently and write custom iterative algorithms using the pregel API.

GraphX competes on performance with the fastest graph systems while retaining Spark’s flexibility, fault tolerance and ease of use.

### 18. What is PageRank Algorithm?

**A.**One of the algorithm in GraphX is PageRank algorithm. Pagerank measures the importance of each vertex in a graph assuming an edge from u to v represents an endorsements of v’s importance by u.

For exmaple, in Twitter if a twitter user is followed by many other users, that particular will be ranked highly. GraphX comes with static and dynamic implementations of pageRank as methods on the pageRank object.

### 19. What is Spark Streaming?

**A.**Whenever there is data flowing continuously and you want to process the data as early as possible, in that case you can take the advantage of Spark Streaming. It is the API for stream processing of live data.

Data can flow for Kafka, Flume or from TCP sockets, Kenisis etc., and you can do complex processing on the data before you pushing them into their destinations. Destinations can be file systems or databases or any other dashboards.

### 20. What is Sliding Window?

**A.**In Spark Streaming, you have to specify the batch interval. For example, let’s take your batch interval is 10 seconds, Now Spark will process the data whatever it gets in the last 10 seconds i.e., last batch interval time.

But with Sliding Window, you can specify how many last batches has to be processed. In the below screen shot, you can see that you can specify the batch interval and how many batches you want to process.

Apart from this, you can also specify when you want to process your last sliding window. For example you want to process the last 3 batches when there are 2 new batches. That is like when you want to slide and how many batches has to be processed in that window.

Hope this post helped you know some important spark interview questions that are often asked in the Apache Spark topic.

# Spark Interview Questions

## Spark Interview Questions

**What is Spark?**

Spark is scheduling, monitoring and distributing engine for big data.It is a [cluster computing](http://en.wikipedia.org/wiki/Computer_cluster) platform designed to be fast and general purpose.Spark extends the popular [MapReduce](http://en.wikipedia.org/wiki/MapReduce) model.One of the main features Spark offers for speed is the ability to run computations in memory, but the system is also more efficient than MapReduce for complex applications running on disk.

**What is Standalone mode?**

In standalone mode, [Spark](https://tekslate.com/tutorials/spark/) uses a Master daemon which coordinates the efforts of the Workers, which run the executors. Standalone mode is the default, but it cannot be used on secure clusters.When you submit an application, you can choose how much memory its executors will use, as well as the total number of cores across all executors.

**What is YARN mode?**

In YARN mode, the YARN ResourceManager performs the functions of the Spark Master. The functions of the Workers are performed by the YARN NodeManager daemons, which run the executors. YARN mode is slightly more complex to set up, but it supports security.

**What is Apache Spark?**

Spark is a fast, easy-to-use and flexible [data processing](http://en.wikipedia.org/wiki/Computer_data_processing) framework. Most of the data users know only SQL and are not good at programming. Shark is a tool, developed for people who are from a database background - to access Scala MLib capabilities through Hive like [SQL](http://tekslate.com/sql-server-integration-in-ssis/) interface. Shark tool helps data users run Hive on Spark - offering compatibility with Hive metastore, queries and data.

**Explain key features of Spark.**

* Allows Integration with [Hadoop](http://tekslate.com/hadoop-tutorial-interview-questions-answers/) and files included in HDFS.
* Spark has an interactive language shell as it has an independent Scala (the language in which Spark is written) interpreter
* Spark consists of RDD’s (Resilient Distributed Datasets), which can be cached across computing nodes in a cluster.
* Spark supports multiple analytic tools that are used for interactive query analysis , real-time analysis and graph processing.

**What are client mode and cluster mode?**

Each application has a driver process which coordinates its execution. This process can run in the foreground (**client mode**) or in the background (**cluster mode**). Client mode is a little simpler, but cluster mode allows you to easily log out after starting a Spark application without terminating the application.

**Define RDD?**

RDDs (Resilient Distributed Datasets) are basic abstraction in [Apache Spark](http://tekslate.com/tutorials/apache-scala/) that represent the data coming into the system in object format. RDDs are used for in-memory computations on large clusters, in a fault tolerant manner. RDDs are read-only portioned, collection of records, that are –

Immutable – RDDs cannot be altered.

Resilient – If a node holding the partition fails the other node takes the data.

**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

**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

**How to run spark in YARN client mode?**

* spark-submit \
* class org.apache.spark.examples.SparkPi \
* deploy-mode client \
* master yarn \
* $SPARK\_HOME/examples/lib/spark-examples\_version.jar 10

**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

**What is Executor memory?**

You can configure this using the –executor-memory argument to sparksubmit. Each application will have at most one executor on each worker, so this setting controls how much of that worker’s memory the application will claim. By default, this setting is 1 GB—you will likely want to increase it on most servers.

**What is the maximum number of total cores?**

This is the total number of cores used across all executors for an application. By default, this is unlimited; that is, the application will launch executors on every available node in the cluster. For a multiuser workload, you should instead ask users to cap their usage. You can set this value through the –total-execution cores argument to spark-submit, or by configuring spark.cores.max in your Spark configuration file.

**What does a Spark Engine do?**

Spark Engine is responsible for scheduling, distributing and monitoring the data application across the cluster.

**Define Partitions?**

As the name suggests, partition is a smaller and logical division of data similar to ‘split’ in [MapReduce](http://tekslate.com/tutorials/mapreduce/). Partitioning is the process to derive logical units of data to speed up the processing process. Everything in Spark is a partitioned RDD.

**What operations RDD support?**

* [Transformations](http://tekslate.com/informatica-transformations-data-ware-houses/)
* Actions

**What do you understand by Transformations in Spark?**

Transformations are functions applied on RDD, resulting into another RDD. It does not execute until an action occurs. map() and filer() are examples of transformations, where the former applies the function passed to it on each element of RDD and results into another RDD. The filter() creates a new RDD by selecting elements form current RDD that pass [function argument](http://en.wikipedia.org/wiki/Parameter_%28computer_programming%29).

**Define Actions.**

An action helps in bringing back the data from RDD to the local machine. An action’s execution is the result of all previously created transformations. reduce() is an action that implements the function passed again and again until one value if left. take() action takes all the values from RDD to local node.

**Define functions of Spark Core?**

Spark Core performs various important functions like [memory management](http://tekslate.com/memory-management-teradata/), monitoring jobs, fault-tolerance, job scheduling and interaction with storage systems.

**What is RDD Lineage?**

Spark does not support [data replication](http://en.wikipedia.org/wiki/Replication_%28computing%29) in the memory and thus, if any data is lost, it is rebuild using RDD lineage. RDD lineage is a process that reconstructs lost data partitions. The best is that RDD always remembers how to build from other datasets.

**What is Spark**[**Driver**](http://en.wikipedia.org/wiki/Device_driver)**?**

Spark Driver is the program that runs on the master node of the machine and declares transformations and actions on data RDDs. In simple terms, driver in Spark creates SparkContext, connected to a given Spark Master. The driver also delivers the RDD graphs to Master, where the standalone [cluster manager](http://en.wikipedia.org/wiki/Cluster_manager) runs.

**What is Hive on Spark?**

Hive contains significant support for Apache Spark, wherein Hive execution is configured to Spark:

* hive> set spark.home=/location/to/sparkHome;
* hive> set hive.execution.engine=spark;
* Hive on Spark supports Spark on yarn mode by default.

**Name commonly-used Spark Ecosystems?**

* Spark SQL (Shark)- for developers
* Spark Streaming for processing live data streams
* GraphX for generating and computing graphs
* MLlib (Machine Learning Algorithms)
* SparkR to promote R Programming in Spark engine.

**What are the main components of Spark?**

* **Spark Core**: Spark Core contains the basic functionality of Spark, including components for task [scheduling](http://tekslate.com/scheduling-jobs-in-datastage/), memory management, fault recovery, interacting with storage systems, and more. Spark Core is also home to the API that defines RDDs,
* **Spark SQL**: Spark SQL is Spark’s package for working with structured data. It allows querying data via SQL as well as the HQL.
* **Spark Streaming:**Spark Streaming is a Spark component that enables processing of live streams of data. Examples of data streams include logfiles generated by production web servers.
* **MLlib**: Spark comes with a library containing common machine learning (ML) functionality, called MLlib. MLlib provides multiple types of machine learning algorithms.
* **GraphX**: GraphX is a library for manipulating graphs (e.g., a social network’s friend graph) and performing graph-parallel computations.

**How Spark Streaming works?**

Spark Streaming receives live input data streams and divides the data into batches, which are then processed by the Spark engine to generate the final stream of results in batches.Spark Streaming provides a high-level abstraction called discretized stream or DStream, which represents a continuous stream of data. DStreams can be created either from input data streams from sources such as [Kafka](http://tekslate.com/tutorials/apache-kafka-tutorials/), Flume, or by applying high-level operations on other DStreams. Internally, a DStream is represented as a sequence of RDDs.

**Define Spark Streaming.Spark supports stream processing?**

An extension to the Spark API , allowing stream processing of live data streams. The data from different sources like [Flume](http://tekslate.com/tutorials/apache-flume/), HDFS is streamed and finally processed to file systems, live dashboards and databases. It is similar tobatch processing as the input data is divided into streams like batches.

**What is GraphX?**

Spark uses GraphX for graph processing to build and transform interactive graphs. The GraphX component enables programmers to reason about structured data at scale.

**What does MLlib do?**

MLlib is scalable machine learning library provided by Spark. It aims at making machine learning easy and scalable with common learning algorithms and use cases like clustering, regression filtering, dimensional reduction, and alike.

**What is Spark SQL?**

SQL Spark, better known as Shark is a novel module introduced in Spark to work with structured data and perform structured data processing. Through this module, Spark executes relational SQL queries on the data. The core of the component supports an altogether different RDD called SchemaRDD, composed of rows objects and schema objects defining data type of each column in the row. It is similar to a table in relational database.

**What is a Parquet file?**

Parquet is a columnar format file supported by many other data processing systems. Spark SQL performs both read and write operations with Parquet file and consider it be one of the best big data analytics format so far.

**What file systems Spark support?**

* Hadoop Distributed File System (HDFS)
* Local File system
* S320.What is Yarn?Similar to Hadoop, Yarn is one of the key features in Spark, providing a central and resource management platform to deliver scalable operations across the cluster . Running Spark on Yarn necessitates a binary distribution of Spar as built on Yarn support.

**List the functions of Spark SQL?**

Spark SQL is capable of:

* Loading data from a variety of structured sources
* Querying data using SQL statements, both inside a Spark program and from external tools that connect to Spark SQL through standard database connectors ([JDBC](http://tekslate.com/jdbc-weblogic/)/ODBC). For instance, using business intelligence tools like Tableau
* Providing rich integration between SQL and regular Python/Java/Scala code, including the ability to join RDDs and SQL tables, expose custom functions in SQL, and more

**What is persist()?**

Spark’s RDDs are by default recomputed each time you run an action on them. If you would like to reuse an RDD in multiple actions, you can ask Spark to persist it using RDD.persist().After computing it the first time, [Spark](https://spark.com/) will store the RDD contents in memory (partitioned across the machines in your cluster), and reuse them in future actions. Persisting RDDs on disk instead of memory is also possible.

**Write common workflow of a Spark program?**

Every Spark program and shell session will work as follows:

* Create some input RDDs from external data.
* Transform them to define new RDDs using transformations like filter().
* Ask Spark to persist() any intermediate RDDs that will need to be reused.
* Launch actions such as count() and first() to kick off a parallel computation, which is then optimized and  executed by Spark.

**Difference between cache() and persist()?**

With cache(), you use only the default storage level MEMORY\_ONLY. With persist(), you can specify which storage level you want.So cache() is the same as calling persist() with the default storage level.Spark has many levels of persistence to choose from based on what our goals are.The default persist() will store the data in the JVM heap as unserialized objects. When we write data out to disk, that data is also always serialized.Different levels of persistence are MEMORY\_ONLY, MEMORY\_ONLY\_SER, MEMORY\_AND\_DISK, MEMORY\_AND\_DISK\_SER, DISK\_ONLY.

**What is lineage graph?**

As you derive new RDDs from each other using transformations, Spark keeps track of the set of dependencies between different RDDs, called the lineage graph. It uses this information to compute each RDD on demand and to recover lost data if part of a persistent RDD is lost.

**Difference between map() and flatMap()?**

The map() transformation takes in a function and applies it to each element in the RDD with the result of the function being the new value of each element in the resulting RDD. Sometimes we want to produce multiple output elements for each input element. The operation to do this is called flatMap(). As with map(), the function we provide to flatMap() is called individually for each element in our input RDD. Instead of returning a single element, we return an iterator with our return values.

**What is reduce() action?**

It takes a function that operates on two elements of the type in your RDD and returns a new element of the same type. A simple example of such a function is +, which we can use to sum our RDD. With reduce(), we can easily sum the elements of our RDD, count the number of elements, and perform other types of aggregations.

**What is Pair RDD?**

Spark provides special operations on RDDs containing key/value pairs. These RDDs are called pair RDDs. Pair RDDs are a useful building block in many programs, as they expose operations that allow you to act on each key in parallel.For example, pair RDDs have a reduceByKey() method that can aggregate data separately for each key, and a join() method that can merge two RDDs together by grouping elements with the same key.

**What are Accumulators?**

Accumulators, provides a simple [syntax](http://tekslate.com/sql-syntax/) for aggregating values from worker nodes back to the driver program. One of the most common uses of accumulators is to count events that occur during job execution for debugging purposes.

**What is Broadcast Variables?**

Spark’s second type of shared variable, broadcast variables, allows the program to efficiently send a large, read-only value to all the worker nodes for use in one or more Spark operations. They come in handy, for example, if your application needs to send a large, read-only lookup table to all the nodes.

**What is Piping?**

Spark provides a pipe() method on RDDs. Spark’s pipe() lets us write parts of jobs using any language we want as long as it can read and write to Unix standard streams. With pipe(), you can write a transformation of an RDD that reads each RDD element from standard input as a String, manipulates that String however you like, and then writes the result(s) as Strings to standard output.

**What are benefits of Spark over MapReduce?**

* Due to the availability of in-memory processing, Spark implements the processing around 10-100x faster than  Hadoop MapReduce. MapReduce makes use of persistence storage for any of the data processing tasks.
* Unlike Hadoop, Spark provides in-built libraries to perform multiple tasks form the same core like batch  processing, Steaming, Machine learning, Interactive SQL queries. However, Hadoop only supports batch     processing.
* Hadoop is highly disk-dependent whereas Spark promotes caching and in-memory data storage
* Spark is capable of performing computations multiple times on the same dataset. This is called iterative  computation while there is no iterative computing implemented by Hadoop.

**Is there any benefit of learning MapReduce, then?**

Yes, MapReduce is a paradigm used by many big data tools including Spark as well. It is extremely relevant to use MapReduce when the data grows bigger and bigger. Most tools like Pig and Hive convert their queries into MapReduce phases to optimize them better.

**What is Spark Executor?**

When SparkContext connect to a cluster manager, it acquires an Executor on nodes in the cluster. Executors are Spark processes that run computations and store the data on the worker node. The final tasks by SparkContext are transferred to executors for their execution.

**Name types of Cluster Managers in Spark?**

The Spark framework supports three major types of Cluster Managers:

* Standalone: a basic manager to set up a cluster
* Apache Mesos: generalized/commonly-used cluster manager, also runs Hadoop MapReduce and other applications
* Yarn: responsible for resource management in Hadoop

**What are Executors?**

Spark executors are worker processes responsible for running the individual tasks in a given Spark job. Executors are launched once at the beginning of a Spark application and typically run for the entire lifetime of an application.Executors have two roles. First, they run the tasks that make up the application and return results to the driver.Second, they provide in-memory storage for RDDs that are cached by user programs.

**What are the steps that occur when you run a Spark application on a cluster?**

The user submits an application using spark-submit.

* Spark-submit launches the driver program and invokes the main() method specified by the user.
* The driver program contacts the cluster manager to ask for resources to launch executors.
* The cluster manager launches executors on behalf of the driver program.
* The driver process runs through the user application. Based on the RDD actions and transformations in the program, the driver sends work to executors in the form of tasks.
* Tasks are run on executor processes to compute and save results.
* If the driver’s main() method exits or it calls SparkContext.stop(),it will terminate the executors and release resources from the cluster manager.

**What is Spark SQL?**

Spark SQL is a module in Apache Spark that integrates relational processing(e.g., declarative queries and optimized storage) with Spark’s procedural programming API. Spark SQL makes two main additions.First, it offers much tighter integration between relational and procedural processing, through a declarative DataFrame API.Second, it includes a highly extensible optimizer, Catalyst.

Big data applications require a mix of processing techniques, data sources and storage formats. The earliest systems designed for these workloads, such as MapReduce, gave users a powerful, but low-level, procedural programming interface. Programming such systems was onerous and required manual optimization by the user to achieve high performance. As a result, multiple new systems sought to provide a more productive user experience by offering relational interfaces to big data. Systems like Pig, Hive and Shark all take advantage of declarative queries to provide richer automatic optimizations.

**What is a schema RDD/DataFrame?**

A SchemaRDD is an RDD composed of Row objects with additional schema information of the types in each column. Row objects are just wrappers around arrays of basic types (e.g., integers and strings).

**What are Row objects?**

Row objects represent records inside SchemaRDDs, and are simply fixed-length arrays of fields.Row objects have a number of getter functions to obtain the value of each field given its index. The standard getter, get (or apply in Scala), takes a column number and returns an Object type (or Any in Scala) that we are responsible for casting to the correct type. For Boolean, Byte, Double, Float, Int, Long, Short, and String, there is a getType() method, which returns that type. For example, get String(0) would return field 0 as a string.

**What are DStreams?**

Much like Spark is built on the concept of RDDs, Spark Streaming provides an abstraction called DStreams, or discretized streams. A DStream is a sequence of data arriving over time. Internally, each DStream is represented as a sequence of RDDs arriving at each time step. DStreams can be created from various input sources, such as Flume, Kafka, or HDFS. Once built, they offer two types of operations: transformations, which yield a new DStream, and output operations, which write data to an external system.

**Explain Spark Streaming Architecture?**

Spark Streaming uses a “micro-batch” architecture, where Spark Streaming receives data from various input sources and groups it into small batches. New batches are created at regular time intervals. At the beginning of each time interval a new batch is created, and any data that arrives during that interval gets added to that batch. At the end of the time interval the batch is done growing. The size of the time intervals is determined by a parameter called the batch interval. Each input batch forms an RDD,  and is processed using Spark jobs to create other RDDs. The processed results can then be pushed out to external systems in batches.

[Spark Training](https://tekslate.com/spark-training)

**What are the types of Transformations on DStreams?**

* In **stateless transformations** the processing of each batch does not depend on the data of its previous batches. They include the common RDD transformations like map(), filter(), and reduceByKey().

• **Stateful transformations**, in contrast, use data or intermediate results from previous batches to compute the results of the current batch. They include transformations based on sliding windows and on tracking state across time.

**What is Receiver in Spark Streaming?**

Every input DStream is associated with a **Receiver**object which receives the data from a source and stores it in Spark’s memory for processing.

**How Spark achieves fault tolerance?**

Spark stores data in-memory whereas Hadoop stores data on disk. Hadoop uses replication to achieve fault tolerance whereas Spark uses different data storage model, RDD. RDDs achieve fault tolerance through a notion of lineage: if a partition of an RDD is lost, the RDD has enough information to rebuild just that partition.This removes the need for replication to achieve fault tolerance.

**What are Spark’s main features?**

* **Speed** : Spark enables applications in Hadoop clusters to run up to 100x faster in memory, and 10x faster even when running on disk. Spark makes it possible by reducing number of read/write to disc. It stores this intermediate processing data in-memory. It uses the concept of an Resilient Distributed Dataset (RDD), which allows it to transparently store data on memory and persist it to disc only it’s needed. This helps to reduce most of the disc read and write – the main time consuming factors – of data processing.
* **Combines SQL, streaming, and complex analytics:**In addition to simple “map” and “reduce” operations, Spark supports SQL queries, streaming data, and complex analytics such as machine learning and graph algorithms out-of-the-box. Not only that, users can combine all these capabilities seamlessly in a single workflow.
* **Ease of Use:**Spark lets you quickly write applications in Java, Scala, or Python. This helps developers to create and run their applications on their familiar programming languages and easy to build parallel apps.
* **Runs Everywhere:**Spark runs on Hadoop, Mesos, standalone, or in the [cloud](http://tekslate.com/cloud-foundry-training). It can access diverse data sources including HDFS, Cassandra, [HBase](http://tekslate.com/hbase-interview-questions-and-answers), S3.

**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

**Is Apache Spark a good fit for Reinforcement learning?**

No. Apache Spark works well only for simple machine learning algorithms like clustering, regression, classification.

**What is Spark Core?**

It has all the basic functionalities of Spark, like - memory management, fault recovery, interacting with storage systems, scheduling tasks, etc.

**How can you remove the elements with a key present in any other RDD?**

Use the subtractByKey () function

**What is the difference between persist() and cache()**

persist () allows the user to specify the storage level whereas cache () uses the default storage level.

**What are the various levels of persistence in Apache Spark?**

Apache Spark automatically persists the intermediary data from various shuffle operations, however it is often suggested that users call persist () method on the RDD in case they plan to reuse it. Spark has various persistence levels to store the RDDs on disk or in memory or as a combination of both with different replication levels.

The various storage/persistence levels in Spark are -

* MEMORY\_ONLY
* MEMORY\_ONLY\_SER
* MEMORY\_AND\_DISK
* MEMORY\_AND\_DISK\_SER, DISK\_ONLY
* OFF\_HEAP

**How Spark handles monitoring and logging in Standalone mode?**

Spark has a web based user interface for monitoring the cluster in standalone mode that shows the cluster and job statistics. The log output for each job is written to the work directory of the slave nodes.

**Does Apache Spark provide checkpointing?**

Lineage graphs are always useful to recover RDDs from a failure but this is generally time consuming if the RDDs have long lineage chains. Spark has an API for check pointing i.e. a REPLICATE flag to persist. However, the decision on which data to checkpoint - is decided by the user. Checkpoints are useful when the lineage graphs are long and have wide dependencies.

**How can you launch Spark jobs inside Hadoop MapReduce?**

Using SIMR (Spark in MapReduce) users can run any spark job inside MapReduce without requiring any admin rights.

[Spark Tutorials](https://tekslate.com/tutorials/spark/)

**How Spark uses Akka?**

Spark uses Akka basically for scheduling. All the workers request for a task to master after registering. The master just assigns the task. Here Spark uses Akka for messaging between the workers and masters.

**How can you achieve high availability in Apache Spark?**

Implementing single node recovery with local file system Using StandBy Masters with Apache ZooKeeper.

**Hadoop uses replication to achieve fault tolerance. How is this achieved in Apache Spark?**

Data storage model in Apache Spark is based on RDDs. RDDs help achieve fault tolerance through lineage. RDD always has the information on how to build from other datasets. If any partition of a RDD is lost due to failure, lineage helps build only that particular lost partition.

**Explain about the core components of a distributed Spark application.**

* Driver- The process that runs the main () method of the program to create RDDs and perform transformations and actions on them.
* Executor –The worker processes that run the individual tasks of a Spark job.

Cluster Manager-A pluggable component in Spark, to launch Executors and Drivers. The cluster manager allows Spark to run on top of other external managers like Apache Mesos or YARN.

**What do you understand by Lazy Evaluation?**

Spark is intellectual in the manner in which it operates on data. When you tell Spark to operate on a given dataset, it heeds the instructions and makes a note of it, so that it does not forget - but it does nothing, unless asked for the final result. When a transformation like map () is called on a RDD-the operation is not performed immediately. Transformations in Spark are not evaluated till you perform an action. This helps optimize the overall data processing workflow.

**Define a worker node?**

A node that can run the Spark application code in a cluster can be called as a worker node. A worker node can have more than one worker which is configured by setting the SPARK\_ WORKER\_INSTANCES property in the spark-env.sh file. Only one worker is started if the SPARK\_ WORKER\_INSTANCES property is not defined.

**What do you understand by SchemaRDD?**

An RDD that consists of row objects (wrappers around basic string or integer arrays) with schema information about the type of data in each column.

We invite the big data community to share the most frequently asked Apache Spark Interview questions and answers, in the comments below - to ease big data job interviews for all prospective analytics professionals.

#### **DROP US A QUERY**

Top of Form

Full Name

E-mail Address



Phone \*

Message

Contact Us

Bottom of Form

### About Us

#### **Q1). What are the key features of Apache Spark?**

Here is a list of the key features of Apache Spark:

* Hadoop Integration
* Lazy Evaluation
* Machine Learning
* Multiple Format Support
* Polyglot
* Real-Time Computation
* Speed

#### **Q2). What are the components of Spark Ecosystem?**

Here are the core components of the Spark ecosystem:

* **Spark Core**: a Base motor for vast scale parallel and appropriated information preparing
* **Spark Streaming**: Used for preparing ongoing gushing information
* **Spark SQL**: Integrates social preparing with Spark’s practical programming API
* **GraphX**: Graphs and diagram parallel calculation
* **MLlib**: Performs machine learning in Apache Spark

#### Q3). What are the languages supported by Apache Spark and which is the most popular one?

Apache Spark supports the accompanying four languages: Scala, Java, Python and R. Among these languages, Scala and Python have intuitive shells for Spark. The Scala shell can be gotten to through ./canister/start shell and the Python shell through ./receptacle/pyspark. Scala is the most utilized among them since Spark is composed in Scala and it is the most prominently utilized for Spark.

**[Read:   YARN- Empowering The Hadoop Functionalities](https://www.janbasktraining.com/blog/what-is-yarn/" \t "_blank)**

#### **Q4). What are the multiple data sources supported by Spark SQL?**

Apache Spark SQL is a popular ecosystem or interfaces to work with structured or semi-structured data. The multiple data sources supported by Spark SQL includethe text file, JSON file, Parquet file etc.

#### Q5). How is machine learning implemented in Spark?

MLlib is a versatile machine learning library given by Spark. It goes for making machine adapting simple and versatile with normal learning calculations and utilize cases like grouping, relapse separating, dimensional decrease, and alike.

#### Q6). What is YARN?

Like Hadoop, YARN is one of the key highlights in Spark, giving a focal and asset administration stage to convey versatile activities over the group. YARN is an appropriated compartment supervisor, as Mesos for instance, though Spark is an information preparing instrument. Spark can keep running on YARN, a similar way Hadoop Map Reduce can keep running on YARN. Running Spark on YARN requires a parallel dissemination of Spark as based on YARN support.

#### **Q7). Does Spark SQL help in big data analytics through external tools too?**

Yes, Spark SQL helps in big data analytics through external tools too. Let us see how it is done actually –

* It access data using SQL statements in both ways either data is stored inside the Spark program or data needs to access through external tools that are connected to Spark SQL through database connectors like JDBC or ODBC.
* It provides rich integration between a database and regular coding with RDDs and SQL tables. It is also able to expose custom SQL functions as needed.

#### **Q8). How is Spark SQL superior from others – HQL and SQL?**

Spark SQL is advance database component able to support multiple database tools without changing their syntax. This is the way how Spark SQL accommodates both HQL and SQL superiorly.

#### **Q9). Do real-time data processing is possible with Spark SQL?**

Real-time data processing is not possible directly but obviously, we can make it happen by registering existing RDD as a SQL table and trigger the SQL queries on priority.

#### **Q10). Explain the concept of Resilient Distributed Dataset (RDD).**

RDD is an abbreviation for Resilient Distribution Datasets. An RDD is a blame tolerant accumulation of operational components that keep running in parallel. The divided information in RDD is permanent and distributed in nature. There are fundamentally two sorts of RDD:

* Parallelized Collections: Here, the current RDDs run parallel with each other.
* Hadoop Datasets:
* They perform works on each document record in HDFS or other stockpiling frameworks.

RDDs are essential parts of information that are put away in the memory circulated crosswise over numerous hubs. RDDs are sluggishly assessed in Spark. This apathetic assessment is the thing that adds to Spark’s speed.

### ****Apache Spark SQL interview questions****

#### Q11). What kind of operations does RDD support?

There are two types of operations that RDDs support: transformations and actions.

* **Transformations**: Transformations make new RDD from existing RDD like guide, reduceByKey, and channel. Transformations are executed on interest. That implies they are registered lethargically.
* **Actions**: Actions return last aftereffects of RDD calculations. Actions trigger execution utilizing genealogy diagram to stack the information into unique RDD, carry out every single transitional change and return last outcomes to Driver program or compose it out to document framework.

#### Q12). What is a Parquet file?

Parquet is a columnar arrangement record upheld by numerous other information preparing frameworks. Start SQL performs both read and write operations with Parquet document and think of it as an extraordinary compared to other enormous information examination arranges up until this point.

#### **Q13). Why is Parquet file format taken best choice for various data processing systems?**

Parquet is a popular columnar file format compatible with almost all data processing systems. This is the reason why it is taken as one of the best choices for big data analytics so far. Spark SQL interface is able to perform read and write operation on Parquet file and it can be accessed quickly whenever required.

#### **Q14). Spark SQL is parallel or distributed data processing framework?**

Spark SQL is parallel data processing framework where batch streaming and interactive data analytics is performed altogether.

#### **Q15). What is the catalyst framework in Spark SQL?**

Catalyst framework is advanced functionality in Spark SQL for automatic transformation of SQL queries by addition of optimized functions that help in processing data faster and accurately than your expectations.

#### Q16). What is Executor Memory in a Spark application?

Each spark application has the same settled load estimate and settled a number of centers for a spark agent. The pile measure is the thing that alluded to as the Spark agent memory which is controlled with the spark.executor.memory property of the – agent memory signal. Each spark application will have one agent on every laborer hub. The agent memory is fundamentally a measure on how much memory of the specialist hub will the application use.

**[Read:   Kafka Interview Questions and Answers](https://www.janbasktraining.com/blog/kafka-interview-questions-answers/" \t "_blank)**

#### **Q17). How to balance query accuracy and response time in Spark SQL?**

To maintain query accuracy and response time in Spark SQL, you are advised to go with BlinkDB query engine. The engine renders queries with meaningful results and significant error to maintain the accuracy.

#### **Q18). Which framework is more preferable in terms of usage either Hadoop or Spark?**

The programming in Hadoop was really tough that has been made easier with Spark by usage of interactive APIs for the different programming language. Obviously, Spark is a preferable choice than Hadoop in terms of usage.

#### **Q19). Are there any benefits of Apache Spark over Hadoop MapReduce?**

Spark has the ability to perform data processing 100 times faster than MapReduce. Also, Spark has inbuilt memory processing and libraries to perform multiple tasks together like batch processing, streaming, interactive processing etc. The above discussion makes sure than Apache Spark is surely better than any other data processing frameworks exist as of now.

#### **Q20). How Array and List can be differentiated in Scala?**

The Array is a mutable data structure that is sequential in nature while Lists are immutable data structures that are recursive in nature.

Size of array is predefined while lists change its size based on operational requirements. In other words, Lists are variable in size while the array is fixed size data structure.

### Apache Spark Scala interview questions

#### **Q21). How to map data and forms together in Scala?**

The most wonderful solution to map data and forms together in Scala is “apply” and “unapply” methods. As the name suggests, the apply method is used to map data while the unapply method can be used to unmap the data. The unapply method follows the reverse operation of the apply method.

#### **Q22). 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.

#### **Q23). What is the significance of immutable design in Scala programming language?**

Every time when working with concurrent programs and other similar equality issues then immutable design in Scala programming language works amazingly. It helps in resolving coding related issues and makes programming easy for Scala developers.

#### **Q24). How can Auxiliary Constructors be defined in Scala?**

The keywords “def” and “this” is used to declare secondary or auxiliary constructors in Scala programming language. They are designed to overload constructors similar to Java. This is necessary to understand the working of each constructor deeply so that the right constructor can be invoked at the right time. Even declaration of constructor differs from each other in terms of data types or parameters.

#### **Q25). How will you explain yield keyword in Scala?**

Yield keyword can be used either before or after expressions. It is taken more useful when declared before expression. The return value from every expression will be stored as the collection. The returned value can either be used as a normal collection or iterate in another loop.

#### **Q26). How can functions be invoked silently without passing all the parameters?**

In case, when we want to invoke functions silently without passing all the parameters, we should use implicit parameters. The parameters that you want to use implicit, you need to provide default values for the same.

#### **Q27). What do you mean by Scala Traits and how it can be used in Scala programming language?**

Scala trait is an advanced class in Scala that enables the use of multiple inheritances and it can be extended to multiple classes together. In other words, one class can have multiple Scala traits based on requirement.

Traits are used commonly when you need dependency injection. You just need to initiate class with Scala traits and dependency will be injected immediately.

#### **Q28). Is there any difference between parallelism and concurrency in Scala programming language?**

Normal users are generally confused between two terms parallelism and concurrency in Scala programming language. Here, we will discuss in simple words how they are different from each other and their significance too. When processes are executed sequentially then it is termed as concurrency while processes are executed simultaneously then it is named as parallelism technology. There are several library functions available in Scala to achieve parallelism.

#### **Q29). How are Monads useful for Scala developers?**

If you want to understand Monads in simple words then it would not be wrong comparing them with a wrapper. As wrappers are used to protect any product and to make it attractive, Monads are used for the same purpose in Scala. They are used to wrap objects together and perform two important functions further. These functions are –

* Identity through “unit” in Scala
* Bind through “flatMap” in Scala

**[Read:   How to Install Apache Pig on Linux?](https://www.janbasktraining.com/blog/install-apache-pig-linux/" \t "_blank)**

#### **Q30). How can Transformations be defined in Apache Spark?**

Transformations are created early in programs and these are generally used along with RDD. These functions are applied on already existed RDD to make a new RDD. Transformations cannot be used without implementing actions in Apache Spark.

The most popular examples of transformation are amap () and filter () that helps to create new RDD by selecting elements in available RDD.

### Apache Spark Coding interview questions

#### **Q31). What is the meaning of “Actions” in Apache Spark?**

The data is taken back to the local machine from RDD with the help of “actions” in Apache Spark. The popular example of the action is folded () passes value again and again until the time it is left only one.

The actions are executed with the assistance of transformations that are created early in programs. The most popular examples of transformation are amap () and filter () that helps to create new RDD by selecting elements in available RDD.

#### **Q32). Define Spark Core and how it is useful for Scala Developers?**

Spark Core in Apache Spark is used for memory management, job monitoring, tolerate faults, scheduling jobs and interactive storage features. RDD is an advanced feature in Spark Core suitable for tolerating faults. RDD is a collection of distributed objects available across multiple nodes that are generally manipulated in parallel.

#### **Q33). Define data streaming in Apache Spark?**

No framework can come to the top without the functionality of live data streaming or handling live events. This is the reason why Apache Spark has used the most advanced techniques to allow the same. For this purpose, Apache uses complex algorithms and high-level functions like reduce, map, join or window etc. These functions push data to file systems and live dashboards further.

#### **Q34). How can graphs be processed in Apache Spark?**

Out of all, one attractive feature supported in Apache Spark includes graph processing. Spark uses advanced multimedia component GraphX to create or explore graphs used to explore data more wisely and accurately.

#### **Q35). Is there any library function to support machine learning algorithms?**

Spark MLib is a popular library function in Apache Spark to support machine learning algorithms. The common learning algorithms and utilities included in MLib library functions are a regression, clustering, classification, dimensional reduction, low-level optimization, advance level pipelining APIs, and collaborative filtering etc. The main objective of the machine learning algorithm is recommendations, predictions and similar other functions.

#### **Q36). Which File System is supported by Apache Spark?**

Apache Spark is an advanced data processing system that can access data from multiple data sources. It creates distributed datasets from the file system you use for data storage. The popular file systems used by Apache Spark include HBase, Cassandra, HDFS, and Amazon S3 etc.

#### **Q37). How many cluster modes are supported in Apache Spark?**

The three popular cluster modes supported in Apache Spark include – Standalone, Apache Mesos, and YARN cluster managers. YARN is the cluster management technology in Apache Spark stands for yet another resource negotiator. The idea was taken from Hadoop where YARN technology was specially introduced to reduce the burden on MapReduce function.

#### **Q38). Is there any cluster management technology in Apache Spark?**

Yes, the cluster management technology in Apache Spark is popular with the name YARN technology. YARN stands for yet another resource negotiator. The idea was taken from Hadoop where YARN technology was specially introduced to reduce the burden on MapReduce function.

#### **Q39). How can you create RDD in Apache Spark?**

There are two popular techniques that can be used to create RDD in Apache Spark – First is Parallelize and other is text File method. Here is a quick explanation of how both methods can be used for RDD creation.

val x= Array(5,7,8,9)

val y= sc.parallelize(x)

val input = sc.textFile(“input.txt”);

#### Q4). What is the key distinction between Hadoop and Spark?

The key distinction between Hadoop and Spark lies in the way to deal with processing: Spark can do it in-memory, while Hadoop MapReduce needs to peruse from and keep in touch with a disc. Thus, the speed of handling varies altogether – Spark might be up to 100 times quicker. Be that as it may, the volume of information prepared likewise varies: Hadoop MapReduce can work with far bigger informational indexes than Spark.

[](https://www.janbasktraining.com/blog/author/jbblog/)

### [JanBask Training](https://www.janbasktraining.com/blog/author/jbblog/)

JanBask Training is a leading Global Online Training Provider through Live Sessions. The Live classes provide a blended approach of hands on experience along with theoretical knowledge which is driven by certified professionals.

* [https://www.janbasktraining.com/assets/img/emails/fb.png](https://www.facebook.com/JanBasktraining/)
* [https://www.janbasktraining.com/assets/img/emails/twitter.png](https://twitter.com/janbasktraining)

* [https://www.janbasktraining.com/assets/img/emails/in.png](https://www.linkedin.com/company/janbask-training/)

### Write a Comment

Top of Form

avatar

Bottom of Form

### Question 1: What is Spark?

Obviously, the very first thing that your potential employers are going to ask you is going to be the definition of Spark. It would be surprising if they didn’t!

Now, this is a great example of the “definition-based” Spark interview questions that I mentioned earlier. Don’t just give a Wikipedia-type of an answer – try to formulate the definitions in your own words. This will show that you are actually trying to remember and thinking about what you say, not just mindlessly spilling random words out like a robot.

**Apache Spark** is an open-source framework used mainly for Big Data analysis, machine learning and real-time processing. The framework basically provides a fully-functional interface for programmers and developers – this interface does a great job in aiding in various complex cluster programming and machine learning tasks.

### Question 2: What are some of the more notable features of Spark?

This is one of the more opinion-based Spark interview questions – you probably won’t need to recite all of them one by one in alphabetical order, so just choose a few that you really like yourself and describe them.

To give you a few examples of what you could say, I’ve chosen three – **speed**, **multi-format support** and **inbuilt libraries.**

Since there is a minimal amount of networks processing the data, the Spark engine can achieve amazing speeds, especially when compared with Hadoop. On a side note, speed is super important if what you’re revising are Spark streaming interview questions.

In addition to that, Spark supports plenty of data sources (since it uses SparkSQL to integrate them) and has a great variety of different, default libraries that Big Data developers can utilize and use.

### Question 3: What is ‘SCC’?

===== Hidden Content. Only Registered users can view. Please [Sign Up](https://www.bitdegree.org/register) , it's FREE! =====

Although this abbreviation isn’t very commonly used (thus resulting in rather difficult surrounding interview questions on Spark), you might encounter such a question.

**SCC** stands for “Spark Cassandra Connector”. It is a tool that Spark uses to access the information (data) located in various Cassandra databases.

### Question 4: What’s ‘RDD’?

**RDD** stands for “Resilient Distribution Datasets”. These are basically operational elements that, when initiated, run in a parallel to one another. There are two types of known RDDs – **parallelized collections** and **Hadoop datasets.** Generally, RDDs support two types of operations – actions and transformations.

### Question 5: What is ‘immutability’?

As the name probably implies, when an item is **immutable**, it cannot be changed or altered in any way once it is fully created and has an assigned value.

This being one of the Apache Spark interview questions which allow some sort of elaboration, you could also add that by default, Spark (as a framework) has this feature. However, this does not apply to the processes of collecting data – only their assigned values.

### Question 6: What is YARN?

**YARN** is one of the core features of Spark. It is mainly concerned with resource management, but is also used to operate across Spark clusters – this is due to it being very scalable.

### Question 7: What is the most commonly used programming language used in Spark?

A great representation of the basic interview questions on Spark, this one should be a no-brainer. Even though there are plenty of developers that like to use Python, **Scala**still remains the most commonly used language for Spark.

### Question 8: How many cluster managers are available in Spark?

By default, there are **three** cluster managers that you can use in Spark. We’ve already talked about one of them in one of the previous Apache Spark interview questions – **YARN.** The other two are known as **Apache Mesos** and **standalone deployments.**

### Question 9: What are the responsibilities of the Spark engine?

===== Hidden Content. Only Registered users can view. Please [Sign Up](https://www.bitdegree.org/register) , it's FREE! =====

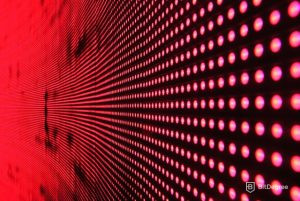
Generally, the Spark engine is concerned with establishing, spreading (distributing) and then monitoring the various sets of data spread around various clusters.

### Question 10: What are ‘lazy evaluations’?

If you think that this is one of the more fun-sounding interview questions on Spark, your completely right. As the name should imply, this type of evaluations is delayed up until the point that the value of the item is needed to be employed. Furthermore, lazy evaluations are only executed **once** – there are no repeat evaluations.

## Spark Interview Questions – Advanced

At this point in the tutorial, you should probably have a pretty good idea of what Spark interview questions actually are and what type of questions you should expect during the interview. Now that we’re warmed up, let’s transition and talk about some of the more popular Spark interview questions and answers for experienced Big Data developers.



Truth be told, the advanced versions of these questions are going to be very similar in nature to their basic counterparts. The only difference is that the advanced versions are going to require a little bit of knowledge and more research than the basic ones.

Not to worry, though – if you’ve already studied Apache Spark quite extensively, these questions should also feel like a breeze to you. Whether you haven’t started learning about Apache Spark or you’re already an expert – these Spark interview questions and answers for experienced developers are going to help you extend and further your knowledge in every step of your Spark journey.

### Question 1: What are ‘partitions’?

A **partition** is a super-small part of a bigger chunk of data. Partitions are based on **logic** – they are used in Spark to manage data so that the minimum network encumbrance would be achieved.

This being another one of those Spark interview questions that allow some sort of elaboration, you could also add that the process of **partitioning** is used to derive the before-mentioned small pieces of data from larger chunks, thus optimizing the network to run at the highest speed possible.

### Question 2: What is Spark Streaming used for?

===== Hidden Content. Only Registered users can view. Please [Sign Up](https://www.bitdegree.org/register) , it's FREE! =====

You should come to your interview prepared to receive a few Spark Streaming interview questions since it is quite a popular feature of Spark itself.

Basically, **Spark Streaming** is responsible for scalable and uninterruptable data streaming processes. It is an extension of the main Spark program, and is commonly used by Big Data developers and programmers alike.

### Question 3: Is it normal to run all of your processes on a localized node?

No, it is not. As a matter of fact, this is one of the most common mistakes that Spark developers make – especially when they’re just starting out. You should always try to distribute your data flow – this will both hasten the process and make it more fluid.

### Question 4: What is ‘SparkCore’ used for?

**SparkCore** is the main engine responsible for all of the processes happening within Spark. Keeping that in mind, you probably won’t be surprised to know that it has a bunch of duties – monitoring, memory and storage management, task scheduling, just to name a few.

### Question 5: Does the File System API have a usage in Spark?

Indeed, it does. This particular API allows Spark to read and compose the data from various different storage areas (devices).

## Summary

In this tutorial, we have talked about everything from the basics of Spark all the way to Spark interview questions and answers for experienced developers. Now you have at least a rough idea of what to expect from the job interview.

Try not to stress and overdo yourself before the interview. My guess is that you didn’t apply for a Spark developer’s job without even knowing what Spark is. Relax – you already know a lot! Try to focus all of your attention into these Spark interview questions – they will help you revise the most important information and prepare for the imminent interview.

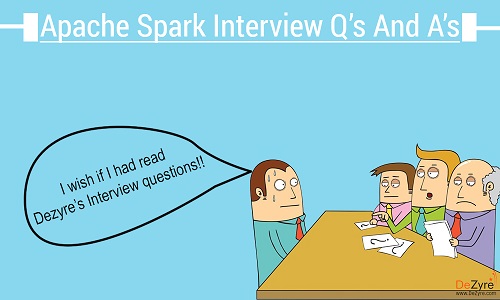


===== Hidden Content. Only Registered users can view. Please [Sign Up](https://www.bitdegree.org/register) , it's FREE! =====

When you’re already in there, try to listen to every question and think it through. Stress might lead to rambling and confusion – you don’t want that! That’s why you should trust your skills and try to keep a leveled head. One piece of advice that seems to work in these job interviews is to try and answer each question in as short and simple manner as possible, but then elaborate with two – three follow-up sentences – this will show your potential employers that you not only know the answers to their questions, but also possess additional knowledge on the topic at hand.

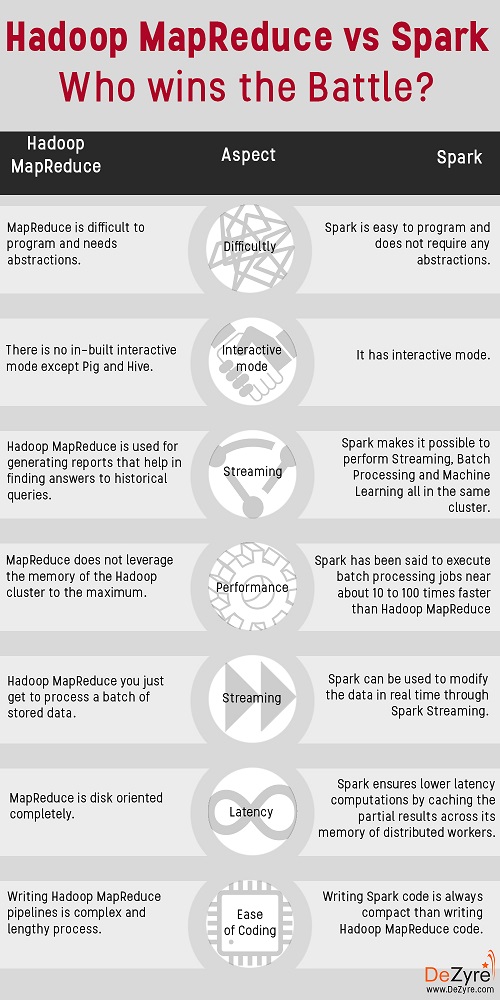
## **Top 50 Apache Spark Interview Questions and Answers**

Preparation is very important to reduce the nervous energy at any big data job interview. Regardless of the big data expertise and skills one possesses, every candidate dreads the face to face big data job interview. Though there is no way of predicting exactly what questions will be asked in any big data or spark developer job interview- these Apache spark interview questions and answers might help you prepare for these interviews better.

****

**1) Compare Spark vs Hadoop MapReduce**

|  |  |  |
| --- | --- | --- |
| ****Criteria**** | ****Hadoop MapReduce**** | ****Apache Spark**** |
| Memory | Does not leverage the memory of the hadoop cluster to maximum. | Let's save data on memory with the use of RDD's. |
| Disk usage | MapReduce is disk oriented. | Spark caches data in-memory and ensures low latency. |
| Processing | Only batch processing is supported | Supports real-time processing through spark streaming. |
| Installation | Is bound to hadoop. | Is not bound to Hadoop. |
| ****Spark vs Hadoop**** | | |



Simplicity, Flexibility and Performance are the major advantages of using Spark over Hadoop.

* Spark is 100 times faster than Hadoop for big data processing as it stores the data in-memory, by placing it in Resilient Distributed Databases (RDD).
* Spark is easier to program as it comes with an interactive mode.
* It provides complete recovery using lineage graph whenever something goes wrong.

Refer [Spark vs Hadoop](https://www.dezyre.com/article/hadoop-mapreduce-vs-apache-spark-who-wins-the-battle/83)

**2) What is Shark?**

Most of the data users know only SQL and are not good at programming. Shark is a tool, developed for people who are from a database background - to access Scala MLib capabilities through Hive like SQL interface. Shark tool helps data users run Hive on Spark - offering compatibility with Hive metastore, queries and data.

**3) List some use cases where Spark outperforms Hadoop in processing.**

1. Sensor Data Processing –Apache Spark’s ‘In-memory computing’ works best here, as data is retrieved and combined from different sources.
2. Spark is preferred over Hadoop for real time querying of data
3. Stream Processing – For processing logs and detecting frauds in live streams for alerts, Apache Spark is the best solution.

**4) What is a Sparse Vector?**

A sparse vector has two parallel arrays –one for indices and the other for values. These vectors are used for storing non-zero entries to save space.

**5) What is RDD?**

RDDs (Resilient Distributed Datasets) are basic abstraction in Apache Spark that represent the data coming into the system in object format. RDDs are used for in-memory computations on large clusters, in a fault tolerant manner. RDDs are read-only portioned, collection of records, that are –

* Immutable – RDDs cannot be altered.
* Resilient – If a node holding the partition fails the other node takes the data.

Build a Big Data Project Portfolio by working on [real-time apache spark projects](https://www.dezyre.com/projects/big-data-projects/apache-spark-projects?utm_source=DeZyreBlog&utm_medium=TextLink&utm_campaign=Blog_SprkInt208)

**6) Explain about transformations and actions in the context of RDDs.**

Transformations are functions executed on demand, to produce a new RDD. All transformations are followed by actions. Some examples of transformations include map, filter and reduceByKey.

Actions are the results of RDD computations or transformations. After an action is performed, the data from RDD moves back to the local machine. Some examples of actions include reduce, collect, first, and take.

**7) What are the languages supported by Apache Spark for developing big data applications?**

Scala, Java, Python, R and Clojure

**8) Can you use Spark to access and analyse data stored in Cassandra databases?**

Yes, it is possible if you use Spark Cassandra Connector.

**9) Is it possible to run Apache Spark on Apache Mesos?**

Yes, Apache Spark can be run on the hardware clusters managed by Mesos.

**10) Explain about the different cluster managers in Apache Spark**

The 3 different clusters managers supported in Apache Spark are:

* YARN
* Apache Mesos -Has rich resource scheduling capabilities and is well suited to run Spark along with other applications. It is advantageous when several users run interactive shells because it scales down the CPU allocation between commands.
* Standalone deployments – Well suited for new deployments which only run and are easy to set up.

**11) How can Spark be connected to Apache Mesos?**

To connect Spark with Mesos-

* Configure the spark driver program to connect to Mesos. Spark binary package should be in a location accessible by Mesos. (or)
* Install Apache Spark in the same location as that of Apache Mesos and configure the property ‘spark.mesos.executor.home’ to point to the location where it is installed.

[](https://goo.gl/PtGO0V)

**12) How can you minimize data transfers when working with Spark?**

Minimizing data transfers and avoiding shuffling helps write spark programs that run in a fast and reliable manner. The various ways in which data transfers can be minimized when working with Apache Spark are:

1. Using Broadcast Variable- Broadcast variable enhances the efficiency of joins between small and large RDDs.
2. Using Accumulators – Accumulators help update the values of variables in parallel while executing.
3. The most common way is to avoid operations ByKey, repartition or any other operations which trigger shuffles.

**13)  Why is there a need for broadcast variables when working with Apache Spark?**

These are read only variables, present in-memory cache on every machine. When working with Spark, usage of broadcast variables eliminates the necessity to ship copies of a variable for every task, so data can be processed faster. Broadcast variables help in storing a lookup table inside the memory which enhances the retrieval efficiency when compared to an RDD lookup ().

**14)  Is it possible to run Spark and Mesos along with Hadoop?**

Yes, it is possible to run Spark and Mesos with Hadoop by launching each of these as a separate service on the machines. Mesos acts as a unified scheduler that assigns tasks to either Spark or Hadoop.

**15)  What is lineage graph?**

The RDDs in Spark, depend on one or more other RDDs. The representation of dependencies in between RDDs is known as the lineage graph. Lineage graph information is used to compute each RDD on demand, so that whenever a part of persistent RDD is lost, the data that is lost can be recovered using the lineage graph information.

**16) How can you trigger automatic clean-ups in Spark to handle accumulated metadata?**

You can trigger the clean-ups by setting the parameter ‘spark.cleaner.ttl’ or by dividing the long running jobs into different batches and writing the intermediary results to the disk.

**17) Explain about the major libraries that constitute the Spark Ecosystem**

* **Spark MLib**- Machine learning library in Spark for commonly used learning algorithms like clustering, regression, classification, etc.
* **Spark Streaming**– This library is used to process real time streaming data.
* **Spark GraphX**– Spark API for graph parallel computations with basic operators like joinVertices, subgraph, aggregateMessages, etc.
* **Spark SQL**– Helps execute SQL like queries on Spark data using standard visualization or BI tools.

**18) What are the benefits of using Spark with Apache Mesos?**

It renders scalable partitioning among various Spark instances and dynamic partitioning between Spark and other big data frameworks.

**19) What is the significance of Sliding Window operation?**

Sliding Window controls transmission of data packets between various computer networks. Spark Streaming library provides windowed computations where the transformations on RDDs are applied over a sliding window of data. Whenever the window slides, the RDDs that fall within the particular window are combined and operated upon to produce new RDDs of the windowed DStream.

**20) What is a DStream?**

Discretized Stream is a sequence of Resilient Distributed Databases that represent a stream of data. DStreams can be created from various sources like Apache Kafka, HDFS, and Apache Flume. DStreams have two operations –

* Transformations that produce a new DStream.
* Output operations that write data to an external system.

**21) When running Spark applications, is it necessary to install Spark on all the nodes of YARN cluster?**

Spark need not be installed when running a job under YARN or Mesos because Spark can execute on top of YARN or Mesos clusters without affecting any change to the cluster.

**22) What is Catalyst framework?**

Catalyst framework is a new optimization framework present in Spark SQL. It allows Spark to automatically transform SQL queries by adding new optimizations to build a faster processing system.

**23) Name a few companies that use Apache Spark in production.**

Pinterest, Conviva, Shopify, Open Table

**24) Which spark library allows reliable file sharing at memory speed across different cluster frameworks?**

Tachyon

### ****Work On Interesting****[Data Science Projects](https://www.dezyre.com/data-science-projects?utm_source=TextLink&utm_medium=ProjectsLink&utm_campaign=Blog_SparkInterviewQuestions_208)****using Spark to build an impressive project portfolio!****

**25) Why is BlinkDB used?**

BlinkDB is a query engine for executing interactive SQL queries on huge volumes of data and renders query results marked with meaningful error bars. BlinkDB helps users balance ‘query accuracy’ with response time.

**26) How can you compare Hadoop and Spark in terms of ease of use?**

Hadoop MapReduce requires programming in Java which is difficult, though Pig and Hive make it considerably easier. Learning Pig and Hive syntax takes time. Spark has interactive APIs for different languages like Java, Python or Scala and also includes Shark i.e. Spark SQL for SQL lovers - making it comparatively easier to use than Hadoop.

**27) What are the common mistakes developers make when running Spark applications?**

Developers often make the mistake of-

* Hitting the web service several times by using multiple clusters.
* Run everything on the local node instead of distributing it.

Developers need to be careful with this, as Spark makes use of memory for processing.

**28) What is the advantage of a Parquet file?**

Parquet file is a columnar format file that helps –

* Limit I/O operations
* Consumes less space
* Fetches only required columns.

**29) What are the various data sources available in SparkSQL?**

* Parquet file
* JSON Datasets
* Hive tables

**30) How Spark uses Hadoop?**

Spark has its own cluster management computation and mainly uses Hadoop for storage.

For the complete list of big data companies and their salaries- [**CLICK HERE**](https://docs.google.com/forms/d/1LFuWEKQKCLR231qR9WE5PZakJj77fTDIW6ox5328HFM/viewform)

**31) What are the key features of Apache Spark that you like?**

* Spark provides advanced analytic options like graph algorithms, machine learning, streaming data, etc
* It has built-in APIs in multiple languages like Java, Scala, Python and R
* It has good performance gains, as it helps run an application in the Hadoop cluster ten times faster on disk and 100 times faster in memory.

**32) What do you understand by Pair RDD?**

Special operations can be performed on RDDs in Spark using key/value pairs and such RDDs are referred to as Pair RDDs. Pair RDDs allow users to access each key in parallel. They have a reduceByKey () method that collects data based on each key and a join () method that combines different RDDs together, based on the elements having the same key.

**33) Which one will you choose for a project –Hadoop MapReduce or Apache Spark?**

The answer to this question depends on the given project scenario - as it is known that Spark makes use of memory instead of network and disk I/O. However, Spark uses large amount of RAM and requires dedicated machine to produce effective results. So the decision to use Hadoop or Spark varies dynamically with the requirements of the project and budget of the organization.

**34) Explain about the different types of transformations on DStreams?**

* Stateless Transformations- Processing of the batch does not depend on the output of the previous batch. Examples – map (), reduceByKey (), filter ().
* Stateful Transformations- Processing of the batch depends on the intermediary results of the previous batch. Examples –Transformations that depend on sliding windows.

**35) 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

**36) Is Apache Spark a good fit for Reinforcement learning?**

No. Apache Spark works well only for simple machine learning algorithms like clustering, regression, classification.

**37) What is Spark Core?**

It has all the basic functionalities of Spark, like - memory management, fault recovery, interacting with storage systems, scheduling tasks, etc.

**38) How can you remove the elements with a key present in any other RDD?**

Use the subtractByKey () function

**39) What is the difference between persist() and cache()**

persist () allows the user to specify the storage level whereas cache () uses the default storage level.

**40) What are the various levels of persistence in Apache Spark?**

Apache Spark automatically persists the intermediary data from various shuffle operations, however it is often suggested that users call persist () method on the RDD in case they plan to reuse it. Spark has various persistence levels to store the RDDs on disk or in memory or as a combination of both with different replication levels.

The various storage/persistence levels in Spark are -

* MEMORY\_ONLY
* MEMORY\_ONLY\_SER
* MEMORY\_AND\_DISK
* MEMORY\_AND\_DISK\_SER, DISK\_ONLY
* OFF\_HEAP

**41) How Spark handles monitoring and logging in Standalone mode?**

Spark has a web based user interface for monitoring the cluster in standalone mode that shows the cluster and job statistics. The log output for each job is written to the work directory of the slave nodes.

**42) Does Apache Spark provide check pointing?**

Lineage graphs are always useful to recover RDDs from a failure but this is generally time consuming if the RDDs have long lineage chains. Spark has an API for check pointing i.e. a REPLICATE flag to persist. However, the decision on which data to checkpoint - is decided by the user. Checkpoints are useful when the lineage graphs are long and have wide dependencies.

**43) How can you launch Spark jobs inside Hadoop MapReduce?**

Using SIMR (Spark in MapReduce) users can run any spark job inside MapReduce without requiring any admin rights.

**44) How Spark uses Akka?**

Spark uses Akka basically for scheduling. All the workers request for a task to master after registering. The master just assigns the task. Here Spark uses Akka for messaging between the workers and masters.

**45) How can you achieve high availability in Apache Spark?**

* Implementing single node recovery with local file system
* Using StandBy Masters with Apache ZooKeeper.

**46) Hadoop uses replication to achieve fault tolerance. How is this achieved in Apache Spark?**

Data storage model in Apache Spark is based on RDDs. RDDs help achieve fault tolerance through lineage. RDD always has the information on how to build from other datasets. If any partition of a RDD is lost due to failure, lineage helps build only that particular lost partition.

**47) Explain about the core components of a distributed Spark application.**

* Driver- The process that runs the main () method of the program to create RDDs and perform transformations and actions on them.
* Executor –The worker processes that run the individual tasks of a Spark job.
* Cluster Manager-A pluggable component in Spark, to launch Executors and Drivers. The cluster manager allows Spark to run on top of other external managers like Apache Mesos or YARN.

**48) What do you understand by Lazy Evaluation?**

Spark is intellectual in the manner in which it operates on data. When you tell Spark to operate on a given dataset, it heeds the instructions and makes a note of it, so that it does not forget - but it does nothing, unless asked for the final result. When a transformation like map () is called on a RDD-the operation is not performed immediately. Transformations in Spark are not evaluated till you perform an action. This helps optimize the overall data processing workflow.

**49)  Define a worker node.**

A node that can run the Spark application code in a cluster can be called as a worker node. A worker node can have more than one worker which is configured by setting the SPARK\_ WORKER\_INSTANCES property in the spark-env.sh file. Only one worker is started if the SPARK\_ WORKER\_INSTANCES property is not defined.

**50) What do you understand by SchemaRDD?**

An RDD that consists of row objects (wrappers around basic string or integer arrays) with schema information about the type of data in each column.

**51) What are the disadvantages of using Apache Spark over Hadoop MapReduce?**

Apache spark does not scale well for compute intensive jobs and consumes large number of system resources. Apache Spark’s in-memory capability at times comes a major roadblock for cost efficient processing of big data. Also, Spark does have its own file management system and hence needs to be integrated with other cloud based data platforms or apache hadoop.

**52) Is it necessary to install spark on all the nodes of a YARN cluster  while running Apache Spark on YARN ?**

No , it is not necessary because Apache Spark runs on top of YARN.

**53) What do you understand by Executor Memory in a Spark application?**

Every spark application has same fixed heap size and fixed number of cores for a spark executor. The heap size is what referred to as the Spark executor memory which is controlled with the spark.executor.memory property of the –executor-memory flag. Every spark application will have one executor on each worker node. The executor memory is basically a measure on how much memory of the worker node will the application utilize.

**54) What does the Spark Engine do?**

Spark engine schedules, distributes and monitors the data application across the spark cluster.

**55) What makes Apache Spark good at low-latency workloads like graph processing and machine learning?**

Apache Spark stores data in-memory for faster model building and training. Machine learning algorithms require multiple iterations to generate a resulting optimal model and similarly graph algorithms traverse all the nodes and edges.These low latency workloads that need multiple iterations can lead to increased performance. Less disk access and  controlled network traffic make a huge difference when there is lots of data to be processed.

**56) Is it necessary to start Hadoop to run any Apache Spark Application ?**

Starting hadoop is not manadatory to run any spark application. As there is no seperate storage in Apache Spark, it uses Hadoop HDFS but it is not mandatory. The data can be stored in local file system, can be loaded from local file system and processed.

**57) What is the default level of parallelism in apache spark?**

If the user does not explicitly specify then the number of partitions are considered as default level of parallelism in Apache Spark.

**58) Explain about the common workflow of a Spark program**

* The foremost step in a Spark program involves creating input RDD's from external data.
* Use various RDD transformations like filter() to create new transformed RDD's based on the business logic.
* persist() any intermediate RDD's which might have to be reused in future.
* Launch various RDD actions() like first(), count() to begin parallel computation , which will then be optimized and executed by Spark.

**59) In a given spark program, how will you identify whether a given operation is Transformation or Action ?**

One can identify the operation based on the return type -

i) The operation is an action, if the return type is other than RDD.

ii) The operation is transformation, if the return type is same as the RDD.

**60) What according to you is a common mistake apache spark developers make when using spark ?**

* Maintaining the required size of shuffle blocks.
* Spark developer often make mistakes with managing directed acyclic graphs (DAG's.)

**61) Suppose that there is an RDD named DeZyrerdd that contains a huge list of numbers.  The following spark code is written to calculate the average -**

**def DeZyreAvg(x, y):  
return (x+y)/2.0;  
avg = DeZyrerdd.reduce(DeZyreAvg);**

**What is wrong with the above code and how will you correct it ?**

Average function is neither commutative nor associative. The best way to compute average is to first sum it and then divide it by count as shown below -

def sum(x, y):  
return x+y;  
total =DeZyrerdd.reduce(sum);  
avg = total / DeZyrerdd.count();

However, the above code could lead to an overflow if the total becomes big. So, the best way to compute average is divide each number by count and then add up as shown below -

cnt = DeZyrerdd.count();  
def divideByCnt(x):  
return x/cnt;  
myrdd1 = DeZyrerdd.map(divideByCnt);  
avg = DeZyrerdd.reduce(sum);

**Q. Say I have a huge list of numbers in a file in HDFS. Each line has one number.And I want to com**

### ****Spark SQL Interview Questions****

**1) Explain the difference between Spark SQL and Hive.**

* Spark SQL is faster than Hive.
* Any Hive query can easily be executed in Spark SQL but vice-versa is not true.
* Spark SQL is a library whereas Hive is a framework.
* It is not mandatory to create a metastore in Spark SQL but it is mandatory to create a Hive metastore.
* Spark SQL automatically infers the schema whereas in Hive schema needs to be explicitly declared..

### ****Spark Streaming Interview Questions****

**1) Name some sources from where Spark streaming component can process real-time data.**

Apache Flume, Apache Kafka, Amazon Kinesis

**2) Name some companies that are already using Spark Streaming.**

Uber, Netflix, Pinterest.

**3) What is the bottom layer of abstraction in the Spark Streaming API ?**

DStream.

**4) What do you understand by receivers in Spark Streaming ?**

Receivers are special entities in Spark Streaming that consume data from various data sources and move them to Apache Spark. Receivers are usually created by streaming contexts as long running tasks on various executors and scheduled to operate in a round robin manner with each receiver taking a single core.

We invite the big data community to share the most frequently asked Apache Spark Interview questions and answers, in the comments below - to ease big data job interviews for all prospective analytics professionals.

**5) How will you calculate the number of executors required to do real-time processing using Apache Spark? What factors need to be connsidered for deciding on the number of nodes for real-time processing?**

The number of nodes can be decided by benchmarking the hardware and considering multiple factors such as optimal throughput (network speed), memory usage, the execution frameworks being used (YARN, Standalone or Mesos) and considering the other jobs that are running within those execution frameworks along with spark.

**6) What is the difference between Spark Transform in DStream and map ?**

tranform function in spark streaming allows developers to use Apache Spark transformations on the underlying RDD's for the stream. map function in hadoop is used for an element to element transform and can be implemented using transform.Ideally , map works on the elements of Dstream and transform allows developers to work with RDD's of the DStream. map is an elementary transformation whereas transform is an RDD transformation.

Check Out Top [Scala Interview Questions](https://www.dezyre.com/article/scala-interview-questions-and-answers-for-spark-developers/302) for Spark Developers.

[Learn Apache Spark online](https://www.dezyre.com/apache-spark-scala-training-online/38) now and become a Certified Spark Developer!

|  |  |
| --- | --- |
| [PREVIOUS](https://www.dezyre.com/article/10-best-hadoop-articles-from-2015-that-you-should-read-/207) | [NEXT](https://www.dezyre.com/article/types-of-analytics-descriptive-predictive-prescriptive-analytics/209) |

[](https://goo.gl/8Tdu4V)

Follow

#### Q1 Name a few commonly used Spark Ecosystems?

Answer:

* Spark SQL (Shark)
* Spark Streaming
* GraphX
* MLlib
* SparkR

#### Q2 What is “Spark SQL”?

Answer:

Spark SQL is a Spark interface to work with structured as well as semi-structured data. It has the capability to load data from multiple structured sources like “text files”, JSON files, Parquet files, among others. Spark SQL provides a special type of RDD called SchemaRDD. These are row objects, where each object represents a record.

#### Q3 Can we do real-time processing using Spark SQL?

Answer:

Not directly but we can register an existing RDD as a SQL table and trigger SQL queries on top of that.

#### Q4 Explain about the major libraries that constitute the Spark Ecosystem

Answer:

Spark MLib- Machine learning library in Spark for commonly used learning algorithms like clustering, regression, classification, etc.

Spark Streaming – This library is used to process real time streaming data.

Spark GraphX – Spark API for graph parallel computations with basic operators like join Vertices, subgraph, aggregate Messages, etc.

Spark SQL – Helps execute SQL like queries on Spark data using standard visualization or BI tools.

#### Q5 What is Spark SQL?

Answer:

SQL Spark, better known as Shark is a novel module introduced in Spark to work with structured data and perform structured data processing. Through this module, Spark executes relational SQL queries on the data. The core of the component supports an altogether different RDD called SchemaRDD, composed of rows objects and schema objects defining data type of each column in the row. It is similar to a table in relational database.

#### Q6 What is a Parquet file?

Answer:

Parquet is a columnar format file supported by many other data processing systems. Spark SQL performs both read and write operations with Parquet file and consider it be one of the best big data analytics format so far.

#### Q7 List the functions of Spark SQL.

Answer:

Spark SQL is capable of:

* Loading data from a variety of structured sources
* Querying data using SQL statements, both inside a Spark program and from external tools that connect to Spark SQL through standard database connectors (JDBC/ODBC). For instance, using business intelligence tools like Tableau
* Providing rich integration between SQL and regular Python/Java/Scala code, including the ability to join RDDs and SQL tables, expose custom functions in SQL, and more

#### Q8 What is Spark?

Answer:

Spark is a parallel data processing framework. It allows to develop fast, unified big data application combine batch, streaming and interactive analytics.

#### Q9 What is Hive on Spark?

Answer:

Hive is a component of Hortonworks’ Data Platform (HDP). Hive provides an SQL-like interface to data stored in the HDP. Spark users will automatically get the complete set of Hive’s rich features, including any new features that Hive might introduce in the future.

The main task around implementing the Spark execution engine for Hive lies in query planning, where Hive operator plans from the semantic analyzer which is translated to a task plan that Spark can execute. It also includes query execution, where the generated Spark plan gets actually executed in the Spark cluster.

#### Q10 What is a “Parquet” in Spark?

Answer:

“Parquet” is a columnar format file supported by many data processing systems. Spark SQL performs both read and write operations with the “Parquet” file.

#### Q11 What is Catalyst framework?

Answer:

Catalyst framework is a new optimization framework present in Spark SQL. It allows Spark to automatically transform SQL queries by adding new optimizations to build a faster processing system.

#### Q12 Why is BlinkDB used?

Answer:

BlinkDB is a query engine for executing interactive SQL queries on huge volumes of data and renders query results marked with meaningful error bars. BlinkDB helps users balance ‘query accuracy’ with response time.

#### Q13 How can you compare Hadoop and Spark in terms of ease of use?

Answer:

Hadoop MapReduce requires programming in Java which is difficult, though Pig and Hive make it considerably easier. Learning Pig and Hive syntax takes time. Spark has interactive APIs for different languages like Java, Python or Scala and also includes Shark i.e. Spark SQL for SQL lovers – making it comparatively easier to use than Hadoop.

#### Q14 What are the various data sources available in SparkSQL?

Answer:

* Parquet file
* JSON Datasets
* Hive tables

SparkSQL is a Spark component that supports querying data either via SQL or via the Hive Query Language. It originated as the Apache Hive port to run on top of Spark (in place of MapReduce) and is now integrated with the Spark stack. In addition to providing support for various data sources, it makes it possible to weave SQL queries with code transformations which results in a very powerful tool. Below is an example of a Hive compatible query.

#### Q15 What are benefits of Spark over MapReduce?

Answer:

* Due to the availability of in-memory processing, Spark implements the processing around 10-100x faster than Hadoop MapReduce. MapReduce makes use of persistence storage for any of the data processing tasks.
* Unlike Hadoop, Spark provides in-built libraries to perform multiple tasks form the same core like batch processing, Steaming, Machine learning, Interactive SQL queries. However, Hadoop only supports batch processing.
* Hadoop is highly disk-dependent whereas Spark promotes caching and in-memory data storage
* Spark is capable of performing computations multiple times on the same dataset. This is called iterative computation while there is no iterative computing implemented by Hadoop.

#### Q16 How SparkSQL is different from HQL and SQL?

Answer:

SparkSQL is a special component on the spark Core engine that support SQL and Hive Query Language without changing any syntax. It’s possible to join SQL table and HQL table.

\*\*\*\*\*\*\*\*\*\*ALL The Best\*\*\*\*\*\*\*\*\*

**What is Spark?**

Answer :  
Apache Spark is an open source framework. It improves execution performance than [Map-Reduce process](https://www.educba.com/mapreduce-interview-questions/).  Its an open platform where we can use multiple programming languages like Java, Python, Scala, R . Spark provide in-memory execution which is 100 times faster than Map-Reduce. It uses the concept of RDD. RDD is a resilient distributed dataset which allows it to transparently store data on memory and persist it to disc only it’s needed. This is where it will reduce time to access the data from memory instead of Disk. Today Industry prefers Spark because of its processing power.

**2.Difference between Hadoop and Spark?**

Answer :

|  |  |  |
| --- | --- | --- |
| **Feature Criteria** | **Apache Spark** | **Hadoop** |
| **Speed** | 10 to 100 times faster than Hadoop | Normal speed |
| **Processing** | Real-time & Batch processing, In-memory, Caching | Batch processing only, Disk Dependent |
| **Difficulty** | Easy because of high-level modules | Difficult to learn |
| **Recovery** | Allows recovery of partitions using RDD | Fault-tolerant |
| **Interactivity** | Has interactive, interactive modes | No interactive mode except Pig & Hive, No iterative mode |

Normal Hadoop architecture follows basic Map-Reduce, For the same process spark provides in-memory execution. Instead of read-write from the hard drive for Map-Reduce, spark provide read-write from virtual memory.

Let us move to the next Spark Interview Questions

**3. What are the Features of Spark?**

Answer:

1. Provide integration facility with [Hadoop and Files on HDFS](https://www.educba.com/course/hadoop-beginner-training/). Spark can run on top of Hadoop using YARN resource clustering. Spark has the capacity to replace the Hadoop’s Map-Reduce engine.
2. **Polyglot:** Spark Provide high-level API for Java, Python, Scala and R. Spark Code can be written in any of these four languages. IT provides an independent shell for scale (the language in which Spark is written) and python interpreter. Which will help to interact with spark engine? Scala shell can be accessed through **./bin/spark-shell**and Python shell through **./bin/pyspark** from the installed directory.
3. **Speed:**Spark engine is 100 times faster than Hadoop Map-Reduce for large-scale data processing. Speed will be achieved through partitioning for parallelizing distributed data processing with minimal network traffic. Spark Provide RDD ’s (Resilient Distributed Datasets), which can be cached across computing nodes in a cluster
4. **Multiple Formats:** Spark has a data source API. It will provide a mechanism to access structured data through [spark SQL](https://www.educba.com/spark-sql-vs-presto/). Data Sources can be anything, Spark will just create a mechanism to convert the data and pull it to the spark. Spark supports multiple data source like Hive, HBase, Cassandra, Json, Parquet, ORC.
5. Spark provides some inbuilt libraries to perform multiple tasks from the same core like batch processing, Steaming, Machine learning, Interactive SQL queries. However, Hadoop only supports batch processing. Spark Provide MLIb (Machine learning libraries ) which will be helpful for [Big-Data Developer](https://www.educba.com/careers-in-big-data/) to process the data. This helps to remove dependencies on multiple tools for the different purpose. Spark provides a common powerful platform to data engineers and data scientists with both fast performance and easy to use.
6. Apache Spark delays the process execution until the action is necessary. This is one of the key features of spark. Spark will add each transformation to DAG (Direct Acyclic Graph) for execution, and when action wants to execute it will actually trigger the DAG to process.
7. **Real Time Streaming:** Apache Spark Provides real-time computations and low latency, Because of in-memory execution. Spark is designed for large scalabilities like thousand node of the cluster and several models for computations.

**4. What is YARN?**

Answer:  
This is the basic Spark Interview Questions asked in an interview. YARN (Yet Another Resource Negotiator) is Resource manager. Spark is a platform which provides in fast execution. Spark will use YARN for execution of the job to the cluster, rather than its own built-in manager. There are some configurations to run Yarn. They include master, deploy-mode, driver-memory, executor-memory, executor-cores, and queue. This is the common Spark Interview Questions that are asked in an interview below is the advantages of spark:

**Advantages of Spark over Map-Reduce**

Spark has advantages over Map-Reduce as follows:-  
Because of the ability of In-memory process, Spark able to execute 10 to 100 time faster than Map-Reduce. Where Map-Reduce can be used for persistence of data at Map and Reduce stage.

 Popular Course in this category

**Data Scientist Course**43 Online Courses | 170+ Hours | Verifiable Certificate of Completion | Lifetime Access   
4.8 (1,155 ratings)

Course Price   
₹16499 ~~₹39999~~   
[**View Course**](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner1)

Related Courses

[SAS Training Course](https://www.educba.com/data-science/courses/sas-training-course/?btnz=edu-blg-inline-banner1)[Machine Learning Course](https://www.educba.com/data-science/courses/machine-learning-course/?btnz=edu-blg-inline-banner1)[AWS Training Course](https://www.educba.com/data-science/courses/aws-training-course/?btnz=edu-blg-inline-banner1)

Apache Spark provides a high level of inbuilt libraries to process multiple tasks at the same time as batch processing, Real-time streaming, Spark-SQL, Structured Streaming, MLib etc. Same time Hadoop provide only batch processing.  
Hadoop Map-Reduce process will be disk dependent, where Spark provides Caching and In-Memory.

Spark has both iterative, perform computation multiple on the same dataset and interactive, perform computation between different dataset where Hadoop doesn’t support iterative computation.

**5. What is the language supported by Spark?**

Answer:  
Spark support scala, Python, R and Java. In the market, big data developer mostly prefers scala and python. For a scale to compile the code we need Set Path of scale/bin directory or to make a jar file.

**6. What is RDD?**

Answer:  
RDD is an abstraction of Resilient Distributed Dataset, which provides a collection of elements partitioned across the all nodes of the cluster which will help to execute multiple processes in parallel. Using RDD developer can store the data In-Memory or caching, to be reused efficiently for parallel execution of operations. RDD can be recovered easily from node failure.

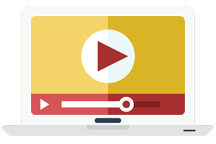
### Part 2 – Spark Interview Questions (Advanced)

Let us now have a look at the advanced Spark Interview Questions.

**7. What are the factors responsible for an execution of Spark?**

Answer:  
1. Spark provides in-memory execution instead of disk dependent like Hadoop Map-Reduce.  
2.RDD Resilient Distributed Dataset, which is a responsible parallel execution of multiple operations on all nodes of a cluster.  
3. Spark provides a shared variable feature for parallel execution. These variables help to reduce data transfer between nodes and share a copy to all nodes. There are two variables.  
4.Broadcast Variable: This variable can be used to cache a value in memory on all nodes  
5.Accumulators Variable: This variable that is only “added” to, such as counters and sums.

**[Data Scientist Course](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner2" \t "_blank)**[43 Online Courses | 170+ Hours | Verifiable Certificate of Completion | Lifetime Access](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner2" \t "_blank)**[Watch The Course Preview](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner2" \t "_blank)**

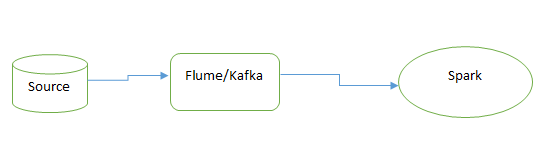
[[](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner2)](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner2" \t "_blank)

**8. What is executor Memory?**

Answer:  
This is the frequently asked Spark Interview Questions in an interview. It is heap size which allocated for spark executor. This property can be controlled by spark.executor.memory property of the –executor-memory flag. Each Spark application has at one executor for each worker node. This property refers to how much memory of worker node will be allocated for an application.

**9. How do you use Spark Stream? Explain One use case?**

Answer:  
Spark Stream is one of feature that useful for a real-time use case. We can use flume, Kafka with spark for this purpose. Flume will trigger the data from a source. Kafka will persist the data into Topic. From Kafka Spark will pull the data using the stream and it will D-stream the data and perform the transformation.



We can use this process for a real-time suspicious transaction, real-time offers etc.

Let us move to the next Spark Interview Questions

**10. Can we use Spark for ETL process?**

Answer:  
Yes, We can use spark platform for [ETL process](https://www.educba.com/etl-interview-questions/).

**11. What is Spark SQL?**

Answer:  
It is one special component of spark which will support SQL queries.

**12. What Lazy Evaluation?**

Answer:  
When we are working with a spark, Transformations are not evaluated until you perform an action. This helps optimize the overall data processing workflow. When defining transformation it will add to the DAG (Direct Acyclic Graph). And at action time it will start to execute stepwise transformations. This is the useful Spark Interview Question asked in an interview.

### Recommended Article

This has been a guide to List Of Spark Interview Questions and Answers so that the candidate can crackdown these Spark Interview Questions easily. You may also look at the following articles to learn more-

1. [Java vs Node JS simplifying the differences](https://www.educba.com/java-vs-node-js/)
2. [Mongo Database Interview Questions | Useful And Most Asked](https://www.educba.com/mongo-database-interview-questions/)
3. [15 Most Successful R Interview Questions and Answers](https://www.educba.com/r-interview-questions/)
4. [Perl Interview Questions And Answers](https://www.educba.com/perl-interview-questions/)
5. [SAS System Interview Questions – Top 10 Useful Questions](https://www.educba.com/sas-system-interview-questions/)

**DATA SCIENCE COURSE - ALL IN ONE BUNDLE**

 360+ Online Courses

 1500+ Hours

 Verifiable Certificates

 Lifetime Access

[**Learn More**](https://www.educba.com/bundles/data-science/data-science-all-in-one-bundle/?btnz=edu-blg-inline-banner3)

* Machine Learning Course
* Data Science with Python Course
* Data Scientist Course
* Deep Learning Course
* IoT Course
* [View More](https://www.educba.com/bundles/data-science/data-science-all-in-one-bundle/?btnz=edu-blg-inline-banner3)

**2SHARES**

[Share](https://www.facebook.com/share.php?u=https%3A%2F%2Fwww.educba.com%2Fspark-interview-questions%2F)

[Tweet](https://twitter.com/intent/tweet?text=Spark+Interview+Questions&url=https%3A%2F%2Fwww.educba.com%2Fspark-interview-questions%2F)

[Share](https://www.linkedin.com/cws/share?url=https%3A%2F%2Fwww.educba.com%2Fspark-interview-questions%2F)

## Reader Interactions

### Leave a Reply

**What is Spark?**

Answer :  
Apache Spark is an open source framework. It improves execution performance than [Map-Reduce process](https://www.educba.com/mapreduce-interview-questions/).  Its an open platform where we can use multiple programming languages like Java, Python, Scala, R . Spark provide in-memory execution which is 100 times faster than Map-Reduce. It uses the concept of RDD. RDD is a resilient distributed dataset which allows it to transparently store data on memory and persist it to disc only it’s needed. This is where it will reduce time to access the data from memory instead of Disk. Today Industry prefers Spark because of its processing power.

**2.Difference between Hadoop and Spark?**

Answer :

|  |  |  |
| --- | --- | --- |
| **Feature Criteria** | **Apache Spark** | **Hadoop** |
| **Speed** | 10 to 100 times faster than Hadoop | Normal speed |
| **Processing** | Real-time & Batch processing, In-memory, Caching | Batch processing only, Disk Dependent |
| **Difficulty** | Easy because of high-level modules | Difficult to learn |
| **Recovery** | Allows recovery of partitions using RDD | Fault-tolerant |
| **Interactivity** | Has interactive, interactive modes | No interactive mode except Pig & Hive, No iterative mode |

Normal Hadoop architecture follows basic Map-Reduce, For the same process spark provides in-memory execution. Instead of read-write from the hard drive for Map-Reduce, spark provide read-write from virtual memory.

Let us move to the next Spark Interview Questions

**3. What are the Features of Spark?**

Answer:

1. Provide integration facility with [Hadoop and Files on HDFS](https://www.educba.com/course/hadoop-beginner-training/). Spark can run on top of Hadoop using YARN resource clustering. Spark has the capacity to replace the Hadoop’s Map-Reduce engine.
2. **Polyglot:** Spark Provide high-level API for Java, Python, Scala and R. Spark Code can be written in any of these four languages. IT provides an independent shell for scale (the language in which Spark is written) and python interpreter. Which will help to interact with spark engine? Scala shell can be accessed through **./bin/spark-shell**and Python shell through **./bin/pyspark** from the installed directory.
3. **Speed:**Spark engine is 100 times faster than Hadoop Map-Reduce for large-scale data processing. Speed will be achieved through partitioning for parallelizing distributed data processing with minimal network traffic. Spark Provide RDD ’s (Resilient Distributed Datasets), which can be cached across computing nodes in a cluster
4. **Multiple Formats:** Spark has a data source API. It will provide a mechanism to access structured data through [spark SQL](https://www.educba.com/spark-sql-vs-presto/). Data Sources can be anything, Spark will just create a mechanism to convert the data and pull it to the spark. Spark supports multiple data source like Hive, HBase, Cassandra, Json, Parquet, ORC.
5. Spark provides some inbuilt libraries to perform multiple tasks from the same core like batch processing, Steaming, Machine learning, Interactive SQL queries. However, Hadoop only supports batch processing. Spark Provide MLIb (Machine learning libraries ) which will be helpful for [Big-Data Developer](https://www.educba.com/careers-in-big-data/) to process the data. This helps to remove dependencies on multiple tools for the different purpose. Spark provides a common powerful platform to data engineers and data scientists with both fast performance and easy to use.
6. Apache Spark delays the process execution until the action is necessary. This is one of the key features of spark. Spark will add each transformation to DAG (Direct Acyclic Graph) for execution, and when action wants to execute it will actually trigger the DAG to process.
7. **Real Time Streaming:** Apache Spark Provides real-time computations and low latency, Because of in-memory execution. Spark is designed for large scalabilities like thousand node of the cluster and several models for computations.

**4. What is YARN?**

Answer:  
This is the basic Spark Interview Questions asked in an interview. YARN (Yet Another Resource Negotiator) is Resource manager. Spark is a platform which provides in fast execution. Spark will use YARN for execution of the job to the cluster, rather than its own built-in manager. There are some configurations to run Yarn. They include master, deploy-mode, driver-memory, executor-memory, executor-cores, and queue. This is the common Spark Interview Questions that are asked in an interview below is the advantages of spark:

**Advantages of Spark over Map-Reduce**

Spark has advantages over Map-Reduce as follows:-  
Because of the ability of In-memory process, Spark able to execute 10 to 100 time faster than Map-Reduce. Where Map-Reduce can be used for persistence of data at Map and Reduce stage.

 Popular Course in this category

**Data Scientist Course**43 Online Courses | 170+ Hours | Verifiable Certificate of Completion | Lifetime Access   
4.8 (1,155 ratings)

Course Price   
₹16499 ~~₹39999~~   
[**View Course**](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner1)

Related Courses

[SAS Training Course](https://www.educba.com/data-science/courses/sas-training-course/?btnz=edu-blg-inline-banner1)[Machine Learning Course](https://www.educba.com/data-science/courses/machine-learning-course/?btnz=edu-blg-inline-banner1)[AWS Training Course](https://www.educba.com/data-science/courses/aws-training-course/?btnz=edu-blg-inline-banner1)

Apache Spark provides a high level of inbuilt libraries to process multiple tasks at the same time as batch processing, Real-time streaming, Spark-SQL, Structured Streaming, MLib etc. Same time Hadoop provide only batch processing.  
Hadoop Map-Reduce process will be disk dependent, where Spark provides Caching and In-Memory.

Spark has both iterative, perform computation multiple on the same dataset and interactive, perform computation between different dataset where Hadoop doesn’t support iterative computation.

**5. What is the language supported by Spark?**

Answer:  
Spark support scala, Python, R and Java. In the market, big data developer mostly prefers scala and python. For a scale to compile the code we need Set Path of scale/bin directory or to make a jar file.

**6. What is RDD?**

Answer:  
RDD is an abstraction of Resilient Distributed Dataset, which provides a collection of elements partitioned across the all nodes of the cluster which will help to execute multiple processes in parallel. Using RDD developer can store the data In-Memory or caching, to be reused efficiently for parallel execution of operations. RDD can be recovered easily from node failure.

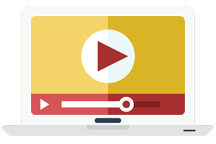
### Part 2 – Spark Interview Questions (Advanced)

Let us now have a look at the advanced Spark Interview Questions.

**7. What are the factors responsible for an execution of Spark?**

Answer:  
1. Spark provides in-memory execution instead of disk dependent like Hadoop Map-Reduce.  
2.RDD Resilient Distributed Dataset, which is a responsible parallel execution of multiple operations on all nodes of a cluster.  
3. Spark provides a shared variable feature for parallel execution. These variables help to reduce data transfer between nodes and share a copy to all nodes. There are two variables.  
4.Broadcast Variable: This variable can be used to cache a value in memory on all nodes  
5.Accumulators Variable: This variable that is only “added” to, such as counters and sums.

**[Data Scientist Course](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner2" \t "_blank)**[43 Online Courses | 170+ Hours | Verifiable Certificate of Completion | Lifetime Access](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner2" \t "_blank)**[Watch The Course Preview](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner2" \t "_blank)**

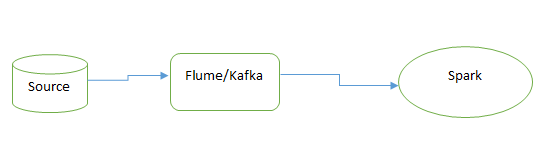
[[](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner2)](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner2" \t "_blank)

**8. What is executor Memory?**

Answer:  
This is the frequently asked Spark Interview Questions in an interview. It is heap size which allocated for spark executor. This property can be controlled by spark.executor.memory property of the –executor-memory flag. Each Spark application has at one executor for each worker node. This property refers to how much memory of worker node will be allocated for an application.

**9. How do you use Spark Stream? Explain One use case?**

Answer:  
Spark Stream is one of feature that useful for a real-time use case. We can use flume, Kafka with spark for this purpose. Flume will trigger the data from a source. Kafka will persist the data into Topic. From Kafka Spark will pull the data using the stream and it will D-stream the data and perform the transformation.



We can use this process for a real-time suspicious transaction, real-time offers etc.

Let us move to the next Spark Interview Questions

**10. Can we use Spark for ETL process?**

Answer:  
Yes, We can use spark platform for [ETL process](https://www.educba.com/etl-interview-questions/).

**11. What is Spark SQL?**

Answer:  
It is one special component of spark which will support SQL queries.

**12. What Lazy Evaluation?**

Answer:  
When we are working with a spark, Transformations are not evaluated until you perform an action. This helps optimize the overall data processing workflow. When defining transformation it will add to the DAG (Direct Acyclic Graph). And at action time it will start to execute stepwise transformations. This is the useful Spark Interview Question asked in an interview.

### Recommended Article

This has been a guide to List Of Spark Interview Questions and Answers so that the candidate can crackdown these Spark Interview Questions easily. You may also look at the following articles to learn more-

1. [Java vs Node JS simplifying the differences](https://www.educba.com/java-vs-node-js/)
2. [Mongo Database Interview Questions | Useful And Most Asked](https://www.educba.com/mongo-database-interview-questions/)
3. [15 Most Successful R Interview Questions and Answers](https://www.educba.com/r-interview-questions/)
4. [Perl Interview Questions And Answers](https://www.educba.com/perl-interview-questions/)
5. [SAS System Interview Questions – Top 10 Useful Questions](https://www.educba.com/sas-system-interview-questions/)

**DATA SCIENCE COURSE - ALL IN ONE BUNDLE**

 360+ Online Courses

 1500+ Hours

 Verifiable Certificates

 Lifetime Access

[**Learn More**](https://www.educba.com/bundles/data-science/data-science-all-in-one-bundle/?btnz=edu-blg-inline-banner3)

* Machine Learning Course
* Data Science with Python Course
* Data Scientist Course
* Deep Learning Course
* IoT Course
* [View More](https://www.educba.com/bundles/data-science/data-science-all-in-one-bundle/?btnz=edu-blg-inline-banner3)

**2SHARES**

[Share](https://www.facebook.com/share.php?u=https%3A%2F%2Fwww.educba.com%2Fspark-interview-questions%2F)

[Tweet](https://twitter.com/intent/tweet?text=Spark+Interview+Questions&url=https%3A%2F%2Fwww.educba.com%2Fspark-interview-questions%2F)

[Share](https://www.linkedin.com/cws/share?url=https%3A%2F%2Fwww.educba.com%2Fspark-interview-questions%2F)

## Reader Interactions

### Leave a Reply

**. What is Apache Spark and what are the benefits of Spark over MapReduce?**

* Spark is really fast. If run in-memory it is 100x faster than Hadoop MapReduce.
* In Hadoop MapReduce, you write many MapReduce jobs and then tie these jobs together using Oozie/shell script. This mechanism is very time consuming and MapReduce tasks have heavy latency. Between two consecutive MapReduce jobs, the data has to be written to HDFS and read from HDFS. This is time-consuming. In case of Spark, this is avoided using RDDs and utilizing memory (RAM). And quite often, translating the output of one MapReduce job into the input of another MapReduce job might require writing another code because Oozie may not suffice.
* In Spark, you can basically do everything from single code or console (PySpark or Scala console) and get the results immediately. Switching between ‘Running something on cluster’ and ‘doing something locally’ is fairly easy and straightforward. This also leads to less context switch of the developer and more productivity.
* Spark kind of equals to MapReduce and Oozie put together.

[Watch this video](https://cloudxlab.com/assessment/slide/apache-spark-basics/379/spark-introduction-why-spark) to learn more about benefits of using Apache Spark over MapReduce.

**2. Is there any point of learning MapReduce, then?**

* MapReduce is a paradigm used by many big data tools including Spark. So, understanding the MapReduce paradigm and how to convert a problem into series of MapReduce tasks is very important.
* Many organizations have already written a lot of code in MapReduce. For legacy reasons, it is required.
* Almost, every other tool such as Hive or Pig converts its query into MapReduce phases. If you understand the MapReduce then you will be able to optimize your queries better.

**3. What are the downsides of Spark?**

Spark utilizes the memory. So, in a shared environment, it might consume little more memory for longer durations.

The developer has to be careful. A casual developer might make following mistakes:

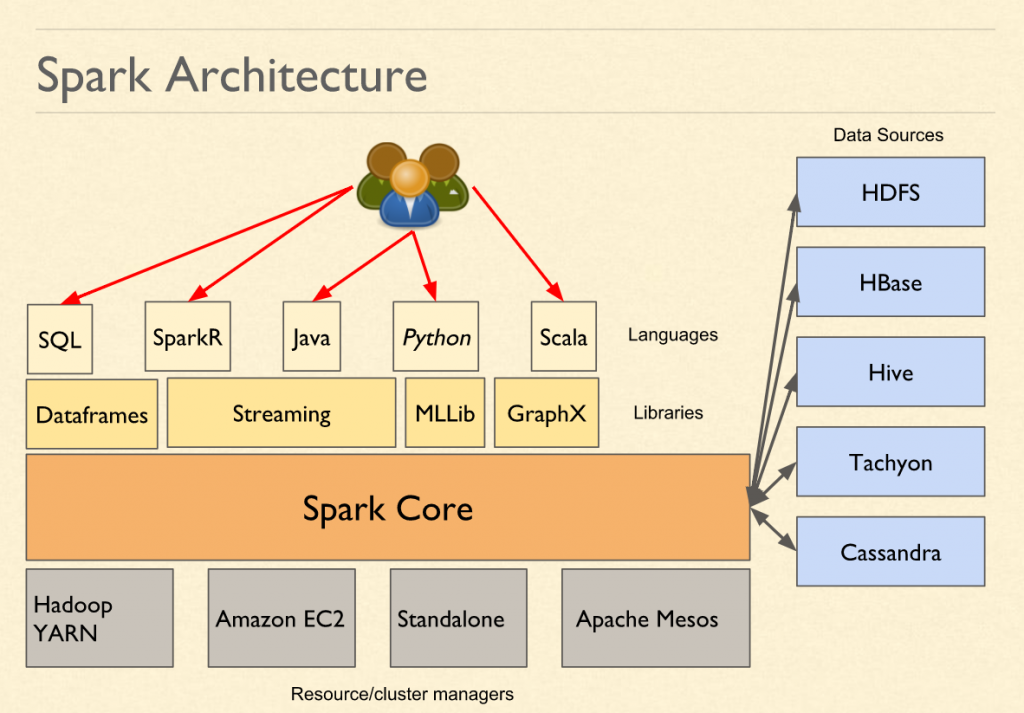
* She may end up running everything on the local node instead of distributing work over to the cluster.
* She might hit some web service too many times by the way of using multiple clusters.

The first problem is well tackled by Hadoop MapReduce paradigm as it ensures that the data your code is churning is fairly small a point of time thus you can make a mistake of trying to handle whole data on a single node.

The second mistake is possible in MapReduce too. While writing MapReduce, a user may hit a service from inside of ***map()*** or ***reduce()*** too many times. This overloading of service is also possible while using Spark.

[Learn Spark From Experts! Enroll Now>>](https://cloudxlab.com/course/4/big-data-with-spark)

**4. Explain in brief what is the architecture of Spark?**

Spark Interview Questions – Spark Architecture

At the architecture level, from a macro perspective, the Spark might look like this:

Spark Architecture

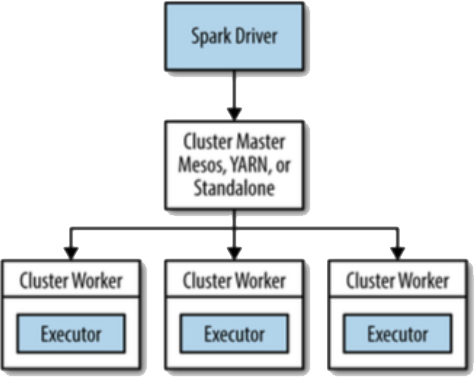
|  |
| --- |
| 5) Interactive Shells or Job Submission Layer |
| 4) API Binding: Python, Java, Scala, R, SQL |
| 3) Libraries: MLLib, GraphX, Spark Streaming |
| 2) Spark Core (RDD & Operations on it) |
| 1) Spark Driver -> Executor |
| 0) Scheduler or Resource Manager |

**0) Scheduler or Resource Manager:**

At the bottom is the resource manager. This resource manager could be external such YARN or Mesos. Or it could be internal if the Spark is running in standalone mode. The role of this layer is to provide a playground in which the program can run distributively. For example, YARN (Yet Another Resource Manager) would create application master, executors for any process.

**1) Spark Driver -> Executor:**

One level above scheduler is the actual code by the Spark which talks to the scheduler to execute. This piece of code does the real work of execution. The Spark Driver that would run inside the application master is part of this layer. Spark Driver dictates what to execute and executor executes the logic.

Spark Interview Questions – Spark Driver and Executors

**2) Spark Core (RDD & Operations on it):**

Spark Core is the layer which provides maximum functionality. This layer provides abstract concepts such as RDD and the execution of the transformation and actions.

**3) Libraries: MLLib,, GraphX, Spark Streaming, Dataframes:**

The additional vertical wise functionalities on top of Spark Core is provided by various libraries such as MLLib, Spark Streaming, GraphX, Dataframes or SparkSQL etc.

**4)** API Bindings are internally calling the same API from different languages.

**5) Interactive Shells or Job Submission Layer:**

The job submission APIs provide a way to submit bundled code. It also provides interactive programs (PySpark, SparkR etc.) that are also called REPL or Read-Eval-Print-Loop to process data interactively.

[Watch this video](https://cloudxlab.com/assessment/slide/apache-spark-basics/126/apache-spark-ecosystem-walkthrough) to learn more about Spark architecture.

**5. On which all platform can Apache Spark run?**

Spark can run on the following platforms:

* **YARN (Hadoop):** Since yarn can handle any kind of workload, the spark can run on Yarn. Though there are two modes of execution. One in which the Spark driver is executed inside the container on node and second in which the Spark driver is executed on the client machine. This is the most common way of using Spark.
* **Apache Mesos:** Mesos is an open source good upcoming resource manager. Spark can run on Mesos.
* **EC2:** If you do not want to manage the hardware by yourself, you can run the Spark on top of Amazon EC2. This makes spark suitable for various organizations.
* **Standalone:**If you have no resource manager installed in your organization, you can use the standalone way. Basically, Spark provides its own resource manager. All you have to do is install Spark on all nodes in a cluster, inform each node about all nodes and start the cluster. It starts communicating with each other and run.

**6. What are the various programming languages supported by Spark?**

Though Spark is written in Scala, it lets the users code in various languages such as:

* Scala
* Java
* Python
* R (Using SparkR)
* SQL (Using SparkSQL)

Also, by the way of piping the data via other commands, we should be able to use all kinds of programming languages or binaries.

[Learn Scala For Free>>](https://cloudxlab.com/assessment/slide/scala)

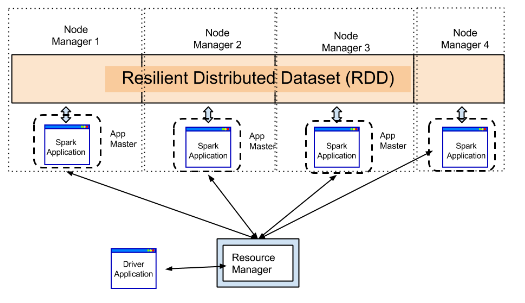
**7. What are the various modes in which Spark runs on YARN? (Local vs Client vs Cluster Mode)**

Apache Spark has two basic parts:

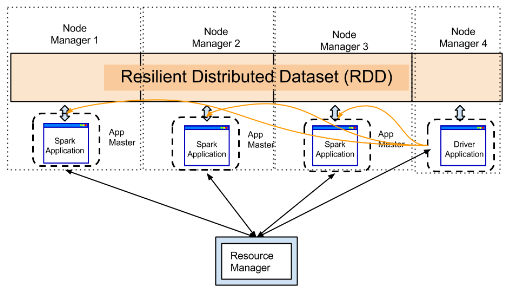
1. **Spark Driver:** Which controls what to execute where
2. **Executor:** Which actually executes the logic

While running Spark on YARN, though it is very obvious that executor will run inside containers, the driver could be run either on the machine which user is using or inside one of the containers. The first one is known as Yarn client mode while second is known as Cluster-Mode. The following diagrams should give you a good idea:

**YARN client mode:** The driver is running on the machine from which client is connected

Spark Interview Questions – Spark RDD Client Mode

**YARN cluster mode:** The driver runs inside the cluster.

Spark Interview Questions – Spark RDD Cluster-Mode

[Watch this video](https://cloudxlab.com/assessment/slide/spark-on-cluster/502/apache-spark-running-on-cluster-cluster-mode-yarn) to learn more about cluster mode.

**Local mode:** It is only for the case when you do not want to use a cluster and instead want to run everything on a single machine. So Driver Application and Spark Application are both on the same machine as the user.

[Watch this video](https://cloudxlab.com/assessment/slide/spark-on-cluster/500/apache-spark-running-on-cluster-local-mode) to learn more about local mode.

**8. What are the various storages from which Spark can read data?**

Spark has been designed to process data from various sources. So, whether you want to process data stored in HDFS, Cassandra, EC2, Hive, HBase, and Alluxio (previously Tachyon). Also, it can read data from any system that supports any Hadoop data source.

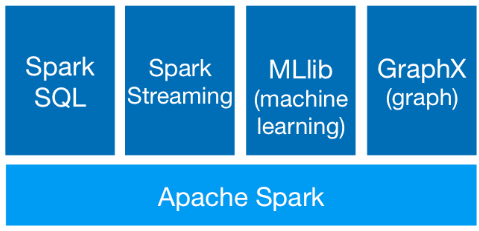
**9. While processing data from HDFS, does it execute code near data?**

Yes, it does in most cases. It creates the executors near the nodes that contain data.

**10. What are the various libraries available on top of Apache Spark?**

Spark powers a stack of libraries including [SQL and DataFrames](http://spark.apache.org/sql/), [MLlib](http://spark.apache.org/mllib/) for machine learning, [GraphX](http://spark.apache.org/graphx/), and [Spark Streaming](http://spark.apache.org/streaming/). You can combine these libraries seamlessly in the same application.

* **MLlib:** It is machine learning library provided by Spark. It basically has all the algorithms that internal are wired to use Spark Core (RDD Operations) and the data structures required. For example, it provides ways to translate the Matrix into RDD and recommendation algorithms into sequences of transformations and actions. MLLib provides the machine learning algorithms that can run parallelly on many computers.
* **GraphX:** GraphX provides libraries which help in manipulating huge graph data structures. It converts graphs into RDD internally. Various algorithms such PageRank on graphs are internally converted into operations on RDD.
* **Spark Streaming:** It is a very simple library that listens on unbounded data sets or the datasets where data is continuously flowing. The processing pauses and waits for data to come if the source isn’t providing data. This library converts the incoming data streaming into RDDs for the “n” seconds collected data aka batch of data and then run the provided operations on the RDDs.

Spark Interview Questions – Spark Libraries[Learn Spark Streaming For Free>>](https://cloudxlab.com/assessment/slide/spark-streaming)

**11. Does Spark provide the storage layer too?**

No, it doesn’t provide storage layer but it lets you use many data sources. It provides the ability to read from almost every popular file systems such as HDFS, Cassandra, Hive, HBase, SQL servers.

**12. Where does Spark Driver run on Yarn?**

If you are submitting a job with –master client, the Spark driver runs on the client’s machine. If you are submitting a job with –master yarn-cluster, the Spark driver would run inside a YARN container.

[Watch this video](https://cloudxlab.com/assessment/slide/spark-on-cluster/501/apache-spark-running-on-cluster-deployment-modes) to learn more about deployment modes in Spark

**13. To use Spark on an existing Hadoop Cluster, do we need to install Spark on all nodes of Hadoop?**

**See question above: What are the various modes in which Spark runs on YARN? (Client vs Cluster Mode)**

Since Spark runs as an application on top of Yarn, it utilizes yarn for the execution of its commands over the cluster’s nodes. So, you do not need to install the Spark on all nodes. When a job is submitted, the Spark will be installed temporarily on all nodes on which execution is needed.

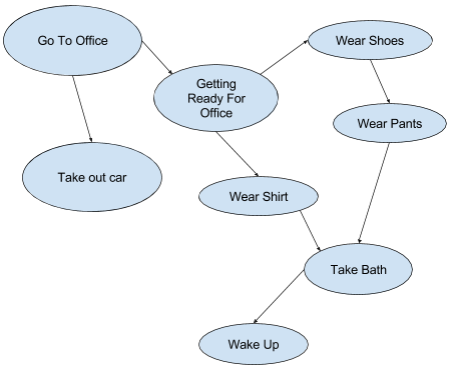
**14. What is sparkContext?**

SparkContext is the entry point to Spark. Using sparkContext you create RDDs which provided various ways of churning data.

**15. What is DAG – Directed Acyclic Graph?**

Directed Acyclic Graph – DAG is a graph data structure having edges which are directional and do not have any loops or cycles.

People use DAG almost all the time. Let’s take an example of getting ready for office.

Spark Interview Questions – DAG

DAG is a way of representing dependencies between objects. It is widely used in computing. The examples where it is used in computing are:

1. Build tools such Apache Ant, Apache Maven, make, sbt
2. Tasks Dependencies in project management – Microsoft Project
3. The data model of Git

**16. What is an RDD?**

The full form of RDD is a resilient distributed dataset. It is a representation of data located on a network which is

* **Immutable –** You can operate on the RDD to produce another RDD but you can’t alter it.
* **Partitioned / Parallel –** The data located on RDD is operated in parallel. Any operation on RDD is done using multiple nodes.
* **Resilience –** If one of the nodes hosting the partition fails, other nodes takes its data.

You can always think of RDD as a big array which is under the hood spread over many computers which are completely abstracted. So, RDD is made up many partitions each partition on different computers.

RDD provides two kinds of operations: Transformations and Actions.

RDD can hold data of any type from any supported programming language such as Python, Java, Scala. The case where RDD’s each element is a tuple – made up of (key, value) is called Pair RDD. PairRDDs provides extra functionalities such as “group by” and joins.

RDD is generally lazily computed i.e. it is not computed unless an action on it is called. So, RDD is either prepared out of another RDD or it is loaded from a data source. In case, it is loaded from another data source it has a binding between the actual data storage and partitions. So, RDD is essentially a pointer to actual data, not data unless it is cached.

If a machine that holds a partition of RDD dies, the same partition is regenerated using the lineage graph of RDD.

If there is a certain RDD that you require very frequently, you can cache it so that it is readily available instead of re-computation every time. Please note that the cached RDD will be available only during the lifetime of the application. If it is costly to recreate the RDD every time, you may want to persist it to the disc.

RDD can be stored at various data storage (such as HDFS, database etc.) in many formats.

[Watch this video](https://cloudxlab.com/assessment/slide/apache-spark-basics/380/spark-introduction-what-is-rdd) to learn more about Spark RDD

**17. What is lazy evaluation and how is it useful?**

Imagine there are two restaurants I (immediate) and P (patient).

In a restaurant I, the waiters are very prompt – as soon as you utter the order they run to the kitchen and place an order to the chef. If you have to order multiple things, the waiter will make multiple trips to the kitchen.

In P, the waiter patiently hears your orders and once you confirm your orders they go to the chef and place the orders. The waiter might combiner multiple dishes into one and prepare. This could lead to tremendous optimization.

While in the restaurant I, the work appears to happen immediately, in restaurant P the work would be actually fast because of clubbing multiple items together for preparation and serving. Restaurant P is doing we call ‘Lazy Evaluation’.

Examples of lazy evaluations are Spark, Pig (Pig Latin). The example of immediate execution could be Python interactive shell, SQL etc.

[Watch this video](https://cloudxlab.com/assessment/slide/apache-spark-basics/385/apache-spark-lazy-evaluation-lineage-graph) to learn more about lazy evaluation and lineage graph in Spark.

**18. How to create an RDD?**

You can create an RDD from an in-memory data or from a data source such as HDFS.

You can load the data from memory using parallelize method of Spark Context in the following manner, in python:

|  |  |
| --- | --- |
| 1 | myrdd = sc.parallelize([1,2,3,4,5]); |

Here myrdd is the variable that represents an RDD created out of an in-memory object. “sc” is the sparkContext which is readily available if you are running in interactive mode using PySpark. Otherwise, you will have to import the SparkContext and initialize.

And to create RDD from a file in HDFS, use the following:

|  |  |
| --- | --- |
| 1 | linesrdd = sc.textFile("/data/file\_hdfs.txt"); |

This would create linesrdd by loading a file from HDFS. Please note that this will work only if your Spark is running on top of Yarn. In case, you want to load the data from external HDFS cluster, you might have to specify the protocol and name node:

|  |  |
| --- | --- |
| 1 | linesrdd = sc.textFile("hdfs://namenode\_host/data/file\_hdfs.txt"); |

In the similar fashion, you can load data from third-party systems.

[Watch this video](https://cloudxlab.com/assessment/slide/apache-spark-basics/381/apache-spark-creating-rdd) to learn more about how to create an RDD in Spark.

**19. When we create an RDD, does it bring the data and load it into the memory?**

No. An RDD is made up of partitions which are located on multiple machines. The partition is only kept in memory if the data is being loaded from memory or the RDD has been cached/persisted into the memory. Otherwise, an RDD is just mapping of actual data and partitions.

**20. If there is certain data that we want to use again and again in different transformations, what should improve the performance?**

RDD can be persisted or cached. There are various ways in which it can be persisted: in-memory, on disc etc. So, if there is a dataset that needs a good amount computing to arrive at, you should consider caching it. You can cache it to disc if preparing it again is far costlier than just reading from disc or it is very huge in size and would not fit in the RAM. You can cache it to memory if it can fit into the memory.

[Watch this video](https://cloudxlab.com/assessment/slide/adv-spark-programming/553/adv-spark-programming-understanding-persistence) to learn more about persisting Spark RDD.

**21. What happens to RDD when one of the nodes on which it is distributed goes down?**

Since Spark knows how to prepare a certain data set because it is aware of various transformations and actions that have lead to the dataset, it will be able to apply the same transformations and actions to prepare the lost partition of the node which has gone down.

**22. How to save RDD?**

There are few methods provided by Spark:

* **saveAsTextFile**: Write the elements of the RDD as a text file (or set of text files) to the provided directory. The directory could be in the local filesystem, HDFS or any other file system. Each element of the dataset will be converted to text using toString() method on every element. And each element will be appended with newline character “\n”
* **saveAsSequenceFile:** Write the elements of the dataset as a Hadoop SequenceFile. This works only on the key-value pair RDD which implement Hadoop’s Writeable interface. You can load sequence file using sc.sequenceFile().
* **saveAsObjectFile:** This simply saves data by serializing using standard java object serialization.

**23. What do we mean by Paraquet?**

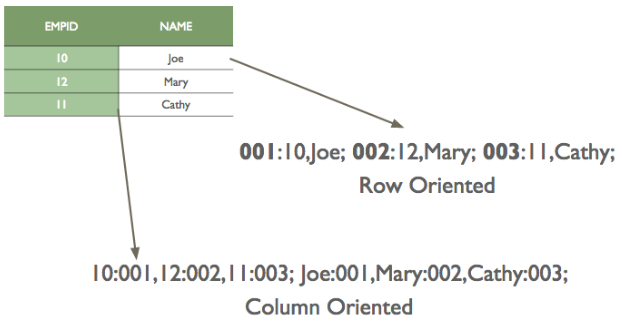
Apache Paraquet is a columnar format for storage of data available in Hadoop ecosystem. It is space efficient storage format which can be used in any programming language and framework.

Apache Spark supports reading and writing data in Paraquet format.

**24. What does it mean by Columnar Storage Format?**

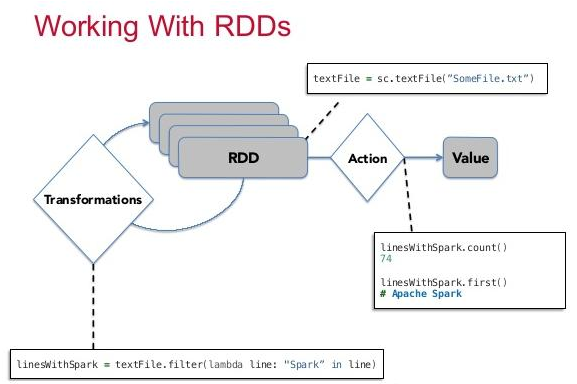
While converting a tabular or structured data into the stream of bytes we can either store row-wise or we could store column-wise.

In row-wise, we first store the first row and then store the second row and so on. In column-wise, we first store first column and second column.

Spark Interview Questions – Columnar Storage Format

**25. When creating an RDD, what goes on internally?**

There are two ways to create RDD. One while loading data from a source. Second, by operating on existing RDD. And an action causes the computation from an RDD to yield the result. The diagram below shows the relationship between RDD, transformations, actions and value/result.

Spark Interview Questions – Working With RDD

* **While loading Data from Source –**When an RDD is prepared by loading data from some source (HDFS, Cassandra, in-memory), the machines which exist nearer to the data are assigned for the creation of partitions. These partitions would hold the parts of mappings or pointers to the actual data. When we are loading data from the memory (for example, by using parallelize), the partitions would hold the actual data instead of pointers or mapping
* **By converting an in-memory array of objects –**An in-memory object can be converted to an RDD using parallelize.
* **By operating on existing RDD –**An RDD is immutable. We can’t change an existing RDD. We can only form a new RDD based on the previous RDD by operating on it. When operating on existing RDD, a new RDD is formed. These operations are also called transformations. The operation could also result in shuffling – moving data across the nodes. Some operations that do not cause shuffling: map, flatMap and filter. Examples of the operations that could result in shuffling are groupByKey, repartition, sortByKey, aggregateByKey, reduceByKey, distinct.Spark maintains the relationship between the RDD in the form of a DAG (Directed Acyclic Graph). When an action such reduce() or saveAsTextFile() is called, the whole graph is evaluated and the result is returned to the driver or saved to the location such as HDFS.

**26. What do we mean by Partitions or slices?**

Partitions (also known as slices earlier) are the parts of RDD. Each partition is generally located on a different machine. Spark runs a task for each partition during the computation.

If you are loading data from HDFS using textFile(), it would create one partition per block of HDFS(64MB typically). Though you can change the number of partitions by specifying the second argument in the textFile() function.

If you are loading data from an existing memory using sc.parallelize(), you can enforce your number of partitions by passing the second argument.

You can change the number of partitions later using repartition().

If you want certain operations to consume the whole partitions at a time, you can use map partition().

**27. What is meant by Transformation? Give some examples.**

The transformations are the functions that are applied on an RDD (resilient distributed dataset). The transformation results in another RDD. A transformation is not executed until an action follows.

Some examples of transformation are:

1. **map() –** applies the function passed to it on each element of RDD resulting in a new RDD.
2. **filter() –** creates a new RDD by picking the elements from the current RDD which pass the function provided as an argument

[Watch this video](https://cloudxlab.com/assessment/slide/apache-spark-basics/383/apache-spark-transformations-map-filter) to learn more about transformations in Spark.

**28. What does map transformation do? Provide an example.**

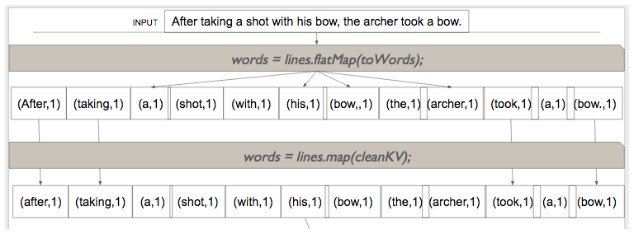
Map transformation on an RDD produces another RDD by translating each element. It translates each element by executing the function provided by the user.

**29. What is the difference between map and flatMap?**

Map and flatMap both functions are applied to each element of RDD. The only difference is that the function that is applied as part of the map must return only one value while flatMap can return a list of values.So, flatMap can convert one element into multiple elements of RDD while map can only result in an equal number of elements.

So, flatMap can convert one element into multiple elements of RDD while map can only result in an equal number of elements.

So, if we are loading RDD from a text file, each element is a sentence. To convert this RDD into an RDD of words, we will have to apply using flatMap a function that would split a string into an array of words. If we have just to clean up each sentence or change case of each sentence, we would be using the map instead of flatMap. See the diagram below.

Spark Interview Questions – flatMap

[Watch this video](https://cloudxlab.com/assessment/slide/apache-spark-basics/386/apache-spark-more-operations-transformations-actions) to learn more about differences between the map and flatMap transformations in Spark.

**30. What are Actions? Give some examples.**

An action brings back the data from the RDD to the local machine. Execution of an action results in all the previously created transformation. The example of actions are:

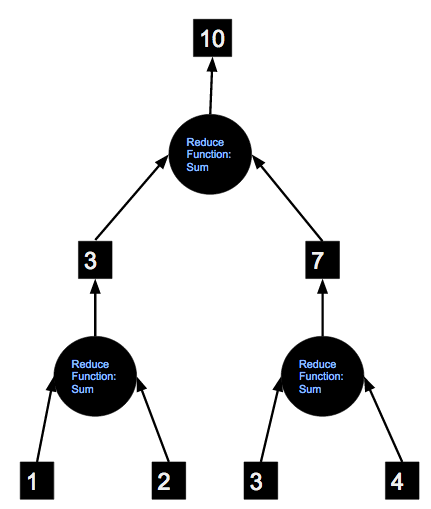
* **reduce() –** executes the function passed again and again until only one value is left. The function should take two arguments and return one value.
* **take() –** take all the values back to the local node from RDD.

[Watch this video](https://cloudxlab.com/assessment/slide/apache-spark-basics/384/apache-spark-actions-take-savetextfile) to learn more about actions in Spark.

**31. What does reduce action do?**

A reduce action converts an RDD to a single value by applying recursively the provided (in argument) function on the elements of an RDD until only one value is left. The provided function must be commutative and associative – the order of arguments or in what way we apply the function should not make difference.

The following diagram shows the process of applying “sum” reduce function on an RDD containing 1, 2, 3, 4.

Spark Interview Questions – Reduce Action

[Watch this video](https://cloudxlab.com/assessment/slide/apache-spark-basics/387/apache-spark-reduce-commutative-associative) to learn more about the reduce action in Spark.

**32. What is broadcast variable?**

Quite often we have to send certain data such as a machine learning model to every node. The most efficient way of sending the data to all of the nodes is by the use of broadcast variables.

Even though you could refer an internal variable which will get copied everywhere but the broadcast variable is far more efficient. It would be loaded into the memory on the nodes only where it is required and when it is required not all the time.

It is sort of a read-only cache similar to distributed cache provided by Hadoop MapReduce.

[Watch this video](https://cloudxlab.com/assessment/slide/adv-spark-programming/560/adv-spark-programming-broadcast-variables) to learn more about broadcast variables in Spark.

**33. What is accumulator?**

An accumulator is a good way to continuously gather data from a Spark process such as the progress of an application. The accumulator receives data from all the nodes in parallel efficiently. Therefore, only the operations in order of operands don’t matter are valid accumulators. Such functions are generally known as associative operations.

An accumulator a kind of central variable to which every node can emit data.

[Watch this video](https://cloudxlab.com/assessment/slide/adv-spark-programming/558/adv-spark-programming-accumulators) to learn more about accumulators in Spark.

**34. Say I have a huge list of numbers in RDD(say myRDD). And I wrote the following code to compute average:**

|  |  |
| --- | --- |
| 1  2  3 | def myAvg(x, y):      return (x+y)/2.0;  avg = myrdd.reduce(myAvg); |

**What is wrong with it and how would you correct it?**

The average function is not commutative and associative. I would simply sum it and then divide by count.

|  |  |
| --- | --- |
| 1  2  3  4 | def sum(x, y):      return x+y;  total = myrdd.reduce(sum);  avg = total / myrdd.count(); |

The only problem with the above code is that the total might become very big thus overflow. So, I would rather divide each number by count and then sum in the following way.

|  |  |
| --- | --- |
| 1  2  3  4  5 | cnt = myrdd.count();  def devideByCnd(x):      return x/cnt;  myrdd1 = myrdd.map(devideByCnd);  avg = myrdd.reduce(sum); |

The problem with above code is that it uses two jobs – one for the count and other for the sum. We can do it in a single shot as follows:

|  |  |
| --- | --- |
| 1  2  3  4 | myrdd = sc.parallelize([1.1, 2.4, 5, 6.0, 2, 3, 7, 9, 11, 13, 10])  sumcount\_rdd = myrdd.map(lambda n : (n, 1))  (total, counts) = sumcount\_rdd.reduce(lambda a,b: (a[0]+b[0], a[1]+b[1]))  avg = total/counts |

Again, it might cause a number overflow because we are summing a huge number of values. We could instead keep the average values and keep computing the average from the average and counts of two parts getting reduced.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | myrdd = sc.parallelize([1.1, 2.4, 5, 6.0, 2, 3, 7, 9, 11, 13, 10])  sumcount\_rdd = myrdd.map(lambda n : (n, 1))  def avg(A, B):      R = 1.0\*B[1]/A[1]      Ri = 1.0/(1+R);      av = A[0]\*Ri + B[0]\*R\*Ri      return (av, B[1] + A[1]);    (av, counts) = sumcount\_rdd.reduce(avg)  print(av) |

If you have two parts having average and counts as (a1, c1) and (a2, c2), the overall average is:  
total/counts = (total1 + total2)/ (count1 + counts2) = (a1\*c1 + a2\*c2)/(c1+c2)

If we mark R = c2/c1, It can be re-written further as a1/(1+R) + a2\*R/(1+R)  
If we further mark Ri as 1/(1+R), we can write it as a1\*Ri + a2\*R\*Ri

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | myrdd = sc.parallelize([1.1, 2.4, 5, 6.0, 2, 3, 7, 9, 11, 13, 10])  sumcount\_rdd = myrdd.map(lambda n : (n, 1))  def avg(A, B):      R = 1.0\*B[1]/A[1]      Ri = 1.0/(1+R);      av = A[0]\*Ri + B[0]\*R\*Ri      return (av, B[1] + A[1]);    (av, counts) = sumcount\_rdd.reduce(avg)  print(av) |

[Watch this video](https://cloudxlab.com/assessment/slide/apache-spark-basics/388/apache-spark-problem-solving-compute-average) to learn more about computing average in Spark.

**35. 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 the sum of squares of these numbers. How would you do it?**

|  |  |
| --- | --- |
| 1  2  3 | # We would first load the file as RDD from HDFS on Spark    numsAsText = sc.textFile("hdfs:////user/student/sgiri/mynumbersfile.txt"); |
| 1  2  3  4 | # Define the function to compute the squares  def toSqInt(str):      v = int(str);      return v\*v; |

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | #Run the function on Spark rdd as transformation  nums = numsAsText.map(toSqInt);    #Run the summation as reduce action  total = nums.reduce(sum)  #finally compute the square root. For which we need to import math.  import math;  print math.sqrt(total); |

**36. Is the following approach correct? Is the sqrtOfSumOfSq a valid reducer?**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | numsAsText = sc.textFile("hdfs:///user/student/sgiri/mynumbersfile.txt");    def toInt(str):      return int(str);    nums = numsAsText.map(toInt);    def sqrtOfSumOfSq(x, y):      return math.sqrt(x\*x+y\*y);    total = nums.reduce(sum)  import math;  print math.sqrt(total); |

Yes. The approach is correct and sqrtOfSumOfSq is a valid reducer.

**37. In a very huge text file, you want to just check if a particular keyword exists. How would you do this using Spark?**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | lines = sc.textFile("hdfs:///user/student/sgiri/bigtextfile.txt");  def isFound(line):      if line.find("mykeyword") > -1:          return 1;      return 0;  foundBits = lines.map(isFound);  sum = foundBits.reduce(sum);  if sum > 0:      print “FOUND”;  else:      print “NOT FOUND”; |

**38. Can you improve the performance of the code in the previous answer?**

Yes. The search is not stopping even after the word we are looking for has been found. Our map code would keep executing on all the nodes which is very inefficient.

We could utilize accumulators to report whether the word has been found or not and then stop the job. Something on these lines.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34 | import thread, threading  from time import sleep  result = "Not Set"  lock = threading.Lock()  accum = sc.accumulator(0)    def map\_func(line):      #introduce delay to emulate the slowness      sleep(1);      if line.find("Adventures") > -1:          accum.add(1);  return 1;      return 0;    def start\_job():      global result      try:  sc.setJobGroup("job\_to\_cancel", "some description")  lines = sc.textFile("hdfs:///user/student/sgiri/wordcount/input/big.txt");  result = lines.map(map\_func);  result.take(1);      except Exception as e:  result = "Cancelled"  lock.release()    def stop\_job():      while accum.value < 3 :          sleep(1);  sc.cancelJobGroup("job\_to\_cancel")    supress = lock.acquire()  supress = thread.start\_new\_thread(start\_job, tuple())  supress = thread.start\_new\_thread(stop\_job, tuple())  supress = lock.acquire() |

[Become Spark Certified Today>>](https://cloudxlab.com/course/4/big-data-with-spark)

This concludes our Spark interview questions guide. I hope these Spark interview questions will help you in preparing for your next interview.

At CloudxLab, we provide free projects on Spark to all our learners so that they can learn by doing. These projects will help you to apply your Spark knowledge in real-world scenarios. Please find the lists of Spark projects below

* [Building Real-time Analytics Dashboard](https://cloudxlab.com/blog/real-time-analytics-dashboard-with-apache-spark-kafka/) – In this project, you will build a real-time analytics dashboard for an e-commerce company using Spark, Kafka, Node.js, Socket.IO and Highcharts.
* [Writing Spark Applications](https://cloudxlab.com/assessment/slide/writing-spark-applications) – In this project, you will learn how to write Spark project on your local machine, unit test it, commit it using Git, build on the production and deploy it. This is an end-to-end project to help you understand how to deploy a Spark project in production following the best practices.
* [Apache Log Parsing](https://cloudxlab.com/assessment/slide/spark-project-log-parsing) – In this project, you will parse Apache logs using Spark to generate meaningful insights from logs.

Below is the list of our free topics on Scala, Linux, Java and Spark. We highly recommend our learners to go through these free topics before your next Spark interview

* [Linux Basics for BigData](https://cloudxlab.com/assessment/slide/linux-basics)
* [Java Essentials](https://cloudxlab.com/assessment/slide/java-essentials)
* [Scala](https://cloudxlab.com/assessment/slide/scala)
* [Apache Spark Basics (Includes Kafka)](https://cloudxlab.com/assessment/slide/apache-spark-basics)
* [Spark Streaming](https://cloudxlab.com/assessment/slide/spark-streaming)

Please feel free to leave your comments in the comment box so that we can improve the guide and serve you better. Also, Follow [CloudxLab on Twitter](https://twitter.com/CloudxLab)to get updates on new blogs and videos.

If you wish to learn Spark technologies such as RDD, Spark Streaming, Kafka, Dataframes, SparkSQL, SparkR, MLlib, GraphX and build a career in BigData and Spark domain then check out our [signature course on Apache Spark](https://cloudxlab.com/course/4/big-data-with-spark) which comes with

* Online instructor-led training by professionals having years of experience in building world-class BigData products
* High-quality learning content including videos and quizzes
* Automated hands-on assessments
* 90 days of lab access so that you can learn by doing
* 24×7 support and [forum access](https://discuss.cloudxlab.com/) to answer all your queries throughout your learning journey
* Real-world projects
* Certificate which you can share on LinkedIn

**About author**

[](https://www.linkedin.com/in/girisandeep)

[**Sandeep Giri**](https://www.linkedin.com/in/girisandeep)

Seasoned hands-on technical architect with years of experience in building world-class software products at Amazon and InMobi

https://secure.gravatar.com/avatar/1393214840cf7455bb4cba055cb30468?s=49&d=mm&r=gAuthor[Sandeep Giri](https://cloudxlab.com/blog/author/sandeepgiri/)Posted on[September 15, 2017](https://cloudxlab.com/blog/spark-interview-questions/)Categories[General](https://cloudxlab.com/blog/category/general/)

* 1. **Question 1. What Is Shark?**

**Answer :**

Most of the data users know only SQL and are not good at programming. Shark is a tool, developed for people who are from a database background - to access Scala MLib capabilities through Hive like SQL interface. Shark tool helps data users run Hive on Spark - offering compatibility with Hive metastore, queries and data.

* 1. **Question 2. Most Of The Data Users Know Only Sql And Are Not Good At Programming. Shark Is A Tool, Developed For People Who Are From A Database Background - To Access Scala Mlib Capabilities Through Hive Like Sql Interface. Shark Tool Helps Data Users Run Hive On Spark - Offering Compatibility With Hive Metastore, Queries And Data.**

**Answer :**

* + 1. Sensor Data Processing –Apache Spark’s ‘In-memory computing’ works best here, as data is retrieved and combined from different sources.
    2. Spark is preferred over Hadoop for real time querying of data
    3. Stream Processing – For processing logs and detecting frauds in live streams for alerts, Apache Spark is the best solution.
  1. **Question 3. What Is A Sparse Vector?**

**Answer :**

sparse vector has two parallel arrays –one for indices and the other for values. These vectors are used for storing non-zero entries to save space.

* 1. **Question 4. What Is Rdd?**

**Answer :**

RDDs (Resilient Distributed Datasets) are basic abstraction in Apache Spark that represent the data coming into the system in object format. RDDs are used for in-memory computations on large clusters, in a fault tolerant manner. RDDs are read-only portioned, collection of records, that are –

* + 1. Immutable – RDDs cannot be altered.
    2. Resilient – If a node holding the partition fails the other node takes the data.
  1. **Question 5. Explain About Transformations And Actions In The Context Of Rdds.**

**Answer :**

Transformations are functions executed on demand, to produce a new RDD. All transformations are followed by actions. Some examples of transformations include map, filter and reduceByKey.

Actions are the results of RDD computations or transformations. After an action is performed, the data from RDD moves back to the local machine. Some examples of actions include reduce, collect, first, and take.

* 1. **Question 6. What Are The Languages Supported By Apache Spark For Developing Big Data Applications?**

**Answer :**

Scala, Java, Python, R and Clojure

* 1. **Question 7. Can You Use Spark To Access And Analyse Data Stored In Cassandra Databases?**

**Answer :**

Yes, it is possible if you use Spark Cassandra Connector.

* 1. **Question 8. Is It Possible To Run Apache Spark On Apache Mesos?**

**Answer :**

Yes, Apache Spark can be run on the hardware clusters managed by Mesos.

* 1. **Question 9. Explain About The Different Cluster Managers In Apache Spark**

**Answer :**

The 3 different clusters managers supported in Apache Spark are:

* + 1. YARN
    2. Apache Mesos -Has rich resource scheduling capabilities and is well suited to run Spark along with other applications. It is advantageous when several users run interactive shells because it scales down the CPU allocation between commands.
    3. Standalone deployments – Well suited for new deployments which only run and are easy to set up.
  1. **Question 10. How Can Spark Be Connected To Apache Mesos?**

**Answer :**

To connect Spark with Mesos-

* + 1. Configure the spark driver program to connect to Mesos. Spark binary package should be in a location accessible by Mesos. (or)
    2. Install Apache Spark in the same location as that of Apache Mesos and configure the property ‘spark.mesos.executor.home’ to point to the location where it is installed.
  1. **Question 11. How Can You Minimize Data Transfers When Working With Spark?**

**Answer :**

Minimizing data transfers and avoiding shuffling helps write spark programs that run in a fast and reliable manner. The various ways in which data transfers can be minimized when working with Apache Spark are:

* + 1. Using Broadcast Variable- Broadcast variable enhances the efficiency of joins between small and large RDDs.
    2. Using Accumulators – Accumulators help update the values of variables in parallel while executing.
    3. The most common way is to avoid operations ByKey, repartition or any other operations which trigger shuffles.
  1. **Question 12. Why Is There A Need For Broadcast Variables When Working With Apache Spark?**

**Answer :**

These are read only variables, present in-memory cache on every machine. When working with Spark, usage of broadcast variables eliminates the necessity to ship copies of a variable for every task, so data can be processed faster. Broadcast variables help in storing a lookup table inside the memory which enhances the retrieval efficiency when compared to an RDD lookup ().

* 1. **Question 13. Is It Possible To Run Spark And Mesos Along With Hadoop?**

**Answer :**

Yes, it is possible to run Spark and Mesos with Hadoop by launching each of these as a separate service on the machines. Mesos acts as a unified scheduler that assigns tasks to either Spark or Hadoop.

* 1. **Question 14. What Is Lineage Graph?**

**Answer :**

The RDDs in Spark, depend on one or more other RDDs. The representation of dependencies in between RDDs is known as the lineage graph. Lineage graph information is used to compute each RDD on demand, so that whenever a part of persistent RDD is lost, the data that is lost can be recovered using the lineage graph information.

* 1. **Question 15. How Can You Trigger Automatic Clean-ups In Spark To Handle Accumulated Metadata?**

**Answer :**

You can trigger the clean-ups by setting the parameter ‘spark.cleaner.ttl’ or by dividing the long running jobs into different batches and writing the intermediary results to the disk.

* 1. **Question 16. Explain About The Major Libraries That Constitute The Spark Ecosystem**

**Answer :**

* + 1. Spark MLib- Machine learning library in Spark for commonly used learning algorithms like clustering, regression, classification, etc.
    2. Spark Streaming – This library is used to process real time streaming data.
    3. Spark GraphX – Spark API for graph parallel computations with basic operators like joinVertices, subgraph, aggregateMessages, etc.
    4. Spark SQL – Helps execute SQL like queries on Spark data using standard visualization or BI tools.
  1. **Question 17. What Are The Benefits Of Using Spark With Apache Mesos?**

**Answer :**

It renders scalable partitioning among various Spark instances and dynamic partitioning between Spark and other big data frameworks.

* 1. **Question 18. What Is The Significance Of Sliding Window Operation?**

**Answer :**

Sliding Window controls transmission of data packets between various computer networks. Spark Streaming library provides windowed computations where the transformations on RDDs are applied over a sliding window of data. Whenever the window slides, the RDDs that fall within the particular window are combined and operated upon to produce new RDDs of the windowed DStream.

* 1. **Question 19. What Is A Dstream?**

**Answer :**

Discretized Stream is a sequence of Resilient Distributed Databases that represent a stream of data. DStreams can be created from various sources like Apache Kafka, HDFS, and Apache Flume. DStreams have two operations –

* + 1. Transformations that produce a new DStream.
    2. Output operations that write data to an external system.
  1. **Question 20. When Running Spark Applications, Is It Necessary To Install Spark On All The Nodes Of Yarn Cluster?**

**Answer :**

Spark need not be installed when running a job under YARN or Mesos because Spark can execute on top of YARN or Mesos clusters without affecting any change to the cluster.

* 1. **Question 21. What Is Catalyst Framework?**

**Answer :**

Catalyst framework is a new optimization framework present in Spark SQL. It allows Spark to automatically transform SQL queries by adding new optimizations to build a faster processing system.

* 1. **Question 22. Name A Few Companies That Use Apache Spark In Production.**

**Answer :**

Pinterest, Conviva, Shopify, Open Table

* 1. **Question 23. Which Spark Library Allows Reliable File Sharing At Memory Speed Across Different Cluster Frameworks?**

**Answer :**

Tachyon

Work On Interesting Data Science Projects using Spark to build an impressive project portfolio!

* 1. **Question 24. Why Is Blinkdb Used?**

**Answer :**

BlinkDB is a query engine for executing interactive SQL queries on huge volumes of data and renders query results marked with meaningful error bars. BlinkDB helps users balance ‘query accuracy’ with response time.

* 1. **Question 25. How Can You Compare Hadoop And Spark In Terms Of Ease Of Use?**

**Answer :**

Hadoop MapReduce requires programming in Java which is difficult, though Pig and Hive make it considerably easier. Learning Pig and Hive syntax takes time. Spark has interactive APIs for different languages like Java, Python or Scala and also includes Shark i.e. Spark SQL for SQL lovers - making it comparatively easier to use than Hadoop.

* 1. **Question 26. What Are The Common Mistakes Developers Make When Running Spark Applications?**

**Answer :**

Developers often make the mistake of-

* + 1. Hitting the web service several times by using multiple clusters.
    2. Run everything on the local node instead of distributing it.

Developers need to be careful with this, as Spark makes use of memory for processing.

* 1. **Question 27. What Is The Advantage Of A Parquet File?**

**Answer :**

Parquet file is a columnar format file that helps –

* + 1. Limit I/O operations
    2. Consumes less space
    3. Fetches only required columns.
  1. **Question 28. What Are The Various Data Sources Available In Sparksql?**

**Answer :**

* + 1. Parquet file
    2. JSON Datasets
    3. Hive tables
  1. **Question 29. How Spark Uses Hadoop?**

**Answer :**

Spark has its own cluster management computation and mainly uses Hadoop for storage.

* 1. **Question 30. What Are The Key Features Of Apache Spark That You Like?**

**Answer :**

* + 1. Spark provides advanced analytic options like graph algorithms, machine learning, streaming data, etc
    2. It has built-in APIs in multiple languages like Java, Scala, Python and R
    3. It has good performance gains, as it helps run an application in the Hadoop cluster ten times faster on disk and 100 times faster in memory.
  1. **Question 31. What Do You Understand By Pair Rdd?**

**Answer :**

Special operations can be performed on RDDs in Spark using key/value pairs and such RDDs are referred to as Pair RDDs. Pair RDDs allow users to access each key in parallel. They have a reduceByKey () method that collects data based on each key and a join () method that combines different RDDs together, based on the elements having the same key.

* 1. **Question 32. Which One Will You Choose For A Project –hadoop Mapreduce Or Apache Spark?**

**Answer :**

As it is known that Spark makes use of memory instead of network and disk I/O. However, Spark uses large amount of RAM and requires dedicated machine to produce effective results. So the decision to use Hadoop or Spark varies dynamically with the requirements of the project and budget of the organization.

* 1. **Question 33. Explain About The Different Types Of Transformations On Dstreams?**

**Answer :**

* + 1. Stateless Transformations- Processing of the batch does not depend on the output of the previous batch. Examples – map (), reduceByKey (), filter ().
    2. Stateful Transformations- Processing of the batch depends on the intermediary results of the previous batch. Examples –Transformations that depend on sliding windows.
  1. **Question 34. Explain About The Popular Use Cases Of Apache Spark**

**Answer :**

Apache Spark is mainly used for

* + 1. Iterative machine learning.
    2. Interactive data analytics and processing.
    3. Stream processing
    4. Sensor data processing
  1. **Question 35. Is Apache Spark A Good Fit For Reinforcement Learning?**

**Answer :**

No. Apache Spark works well only for simple machine learning algorithms like clustering, regression, classification.

* 1. **Question 36. What Is Spark Core?**

**Answer :**

It has all the basic functionalities of Spark, like - memory management, fault recovery, interacting with storage systems, scheduling tasks, etc.

* 1. **Question 37. How Can You Remove The Elements With A Key Present In Any Other Rdd?**

**Answer :**

Use the subtractByKey () function

* 1. **Question 38. What Is The Difference Between Persist() And Cache()**

**Answer :**

persist () allows the user to specify the storage level whereas cache () uses the default storage level.

* 1. **Question 39. What Are The Various Levels Of Persistence In Apache Spark?**

**Answer :**

Apache Spark automatically persists the intermediary data from various shuffle operations, however it is often suggested that users call persist () method on the RDD in case they plan to reuse it. Spark has various persistence levels to store the RDDs on disk or in memory or as a combination of both with different replication levels.

The various storage/persistence levels in Spark are -

* + 1. MEMORY\_ONLY
    2. MEMORY\_ONLY\_SER
    3. MEMORY\_AND\_DISK
    4. MEMORY\_AND\_DISK\_SER, DISK\_ONLY
    5. OFF\_HEAP
  1. **Question 40. How Spark Handles Monitoring And Logging In Standalone Mode?**

**Answer :**

Spark has a web based user interface for monitoring the cluster in standalone mode that shows the cluster and job statistics. The log output for each job is written to the work directory of the slave nodes.

* 1. **Question 41. Does Apache Spark Provide Check Pointing?**

**Answer :**

Lineage graphs are always useful to recover RDDs from a failure but this is generally time consuming if the RDDs have long lineage chains. Spark has an API for check pointing i.e. a REPLICATE flag to persist. However, the decision on which data to checkpoint - is decided by the user. Checkpoints are useful when the lineage graphs are long and have wide dependencies.

* 1. **Question 42. How Can You Launch Spark Jobs Inside Hadoop Mapreduce?**

**Answer :**

Using SIMR (Spark in MapReduce) users can run any spark job inside MapReduce without requiring any admin rights.

* 1. **Question 43. How Spark Uses Akka?**

**Answer :**

Spark uses Akka basically for scheduling. All the workers request for a task to master after registering. The master just assigns the task. Here Spark uses Akka for messaging between the workers and masters.

* 1. **Question 44. How Can You Achieve High Availability In Apache Spark?**

**Answer :**

* + 1. Implementing single node recovery with local file system
    2. Using StandBy Masters with Apache ZooKeeper.
  1. **Question 45. Hadoop Uses Replication To Achieve Fault Tolerance. How Is This Achieved In Apache Spark?**

**Answer :**

Data storage model in Apache Spark is based on RDDs. RDDs help achieve fault tolerance through lineage. RDD always has the information on how to build from other datasets. If any partition of a RDD is lost due to failure, lineage helps build only that particular lost partition.

* 1. **Question 46. Explain About The Core Components Of A Distributed Spark Application.**

**Answer :**

* + 1. Driver- The process that runs the main () method of the program to create RDDs and perform transformations and actions on them.
    2. Executor –The worker processes that run the individual tasks of a Spark job.
    3. Cluster Manager-A pluggable component in Spark, to launch Executors and Drivers. The cluster manager allows Spark to run on top of other external managers like Apache Mesos or YARN.
  1. **Question 47. What Do You Understand By Lazy Evaluation?**

**Answer :**

Spark is intellectual in the manner in which it operates on data. When you tell Spark to operate on a given dataset, it heeds the instructions and makes a note of it, so that it does not forget - but it does nothing, unless asked for the final result. When a transformation like map () is called on a RDD-the operation is not performed immediately. Transformations in Spark are not evaluated till you perform an action. This helps optimize the overall data processing workflow.

* 1. **Question 48. Define A Worker Node.**

**Answer :**

A node that can run the Spark application code in a cluster can be called as a worker node. A worker node can have more than one worker which is configured by setting the SPARK\_ WORKER\_INSTANCES property in the spark-env.sh file. Only one worker is started if the SPARK\_ WORKER\_INSTANCES property is not defined.

* 1. **Question 49. What Do You Understand By Schemardd?**

**Answer :**

An RDD that consists of row objects (wrappers around basic string or integer arrays) with schema information about the type of data in each column.

* 1. **Question 50. What Are The Disadvantages Of Using Apache Spark Over Hadoop Mapreduce?**

**Answer :**

Apache spark does not scale well for compute intensive jobs and consumes large number of system resources. Apache Spark’s in-memory capability at times comes a major roadblock for cost efficient processing of big data. Also, Spark does have its own file management system and hence needs to be integrated with other cloud based data platforms or apache hadoop.

* 1. **Question 51. Is It Necessary To Install Spark On All The Nodes Of A Yarn Cluster While Running Apache Spark On Yarn ?**

**Answer :**

No , it is not necessary because Apache Spark runs on top of YARN.

* 1. **Question 52. What Do You Understand By Executor Memory In A Spark Application?**

**Answer :**

Every spark application has same fixed heap size and fixed number of cores for a spark executor. The heap size is what referred to as the Spark executor memory which is controlled with the spark.executor.memory property of the –executor-memory flag. Every spark application will have one executor on each worker node. The executor memory is basically a measure on how much memory of the worker node will the application utilize.

* 1. **Question 53. What Does The Spark Engine Do?**

**Answer :**

Spark engine schedules, distributes and monitors the data application across the spark cluster.

* 1. **Question 54. What Makes Apache Spark Good At Low-latency Workloads Like Graph Processing And Machine Learning?**

**Answer :**

Apache Spark stores data in-memory for faster model building and training. Machine learning algorithms require multiple iterations to generate a resulting optimal model and similarly graph algorithms traverse all the nodes and edges.These low latency workloads that need multiple iterations can lead to increased performance. Less disk access and  controlled network traffic make a huge difference when there is lots of data to be processed.

* 1. **Question 55. Is It Necessary To Start Hadoop To Run Any Apache Spark Application ?**

**Answer :**

Starting hadoop is not manadatory to run any spark application. As there is no seperate storage in Apache Spark, it uses Hadoop HDFS but it is not mandatory. The data can be stored in local file system, can be loaded from local file system and processed.

* 1. **Question 56. What Is The Default Level Of Parallelism In Apache Spark?**

**Answer :**

If the user does not explicitly specify then the number of partitions are considered as default level of parallelism in Apache Spark.

* 1. **Question 57. Explain About The Common Workflow Of A Spark Program**

**Answer :**

* + 1. The foremost step in a Spark program involves creating input RDD's from external data.
    2. Use various RDD transformations like filter() to create new transformed RDD's based on the business logic.
    3. persist() any intermediate RDD's which might have to be reused in future.
    4. Launch various RDD actions() like first(), count() to begin parallel computation , which will then be optimized and executed by Spark.
  1. **Question 58. Name A Few Commonly Used Spark Ecosystems.**

**Answer :**

Spark SQL (Shark)

Spark Streaming

GraphX

MLlib

SparkR

* 1. **Question 59. What Is “spark Sql”?**

**Answer :**

Spark SQL is a Spark interface to work with structured as well as semi-structured data. It has the capability to load data from multiple structured sources like “text files”, JSON files, Parquet files, among others. Spark SQL provides a special type of RDD called SchemaRDD. These are row objects, where each object represents a record.

* 1. **Question 60. Can We Do Real-time Processing Using Spark Sql?**

**Answer :**

Not directly but we can register an existing RDD as a SQL table and trigger SQL queries on top of that.

* 1. **Question 61. What Is Spark Sql?**

**Answer :**

SQL Spark, better known as Shark is a novel module introduced in Spark to work with structured data and perform structured data processing. Through this module, Spark executes relational SQL queries on the data. The core of the component supports an altogether different RDD called SchemaRDD, composed of rows objects and schema objects defining data type of each column in the row. It is similar to a table in relational database.

* 1. **Question 62. What Is A Parquet File?**

**Answer :**

Parquet is a columnar format file supported by many other data processing systems. Spark SQL performs both read and write operations with Parquet file and consider it be one of the best big data analytics format so far.

* 1. **Question 63. List The Functions Of Spark Sql.**

**Answer :**

Spark SQL is capable of:

* + 1. Loading data from a variety of structured sources
    2. Querying data using SQL statements, both inside a Spark program and from external tools that connect to Spark SQL through standard database connectors (JDBC/ODBC). For instance, using business intelligence tools like Tableau
    3. Providing rich integration between SQL and regular Python/Java/Scala code, including the ability to join RDDs and SQL tables, expose custom functions in SQL, and more
  1. **Question 64. What Is Spark?**

**Answer :**

Spark is a parallel data processing framework. It allows to develop fast, unified big data application combine batch, streaming and interactive analytics.

* 1. **Question 65. What Is Hive On Spark?**

**Answer :**

Hive is a component of Hortonworks’ Data Platform (HDP). Hive provides an SQL-like interface to data stored in the HDP. Spark users will automatically get the complete set of Hive’s rich features, including any new features that Hive might introduce in the future.

The main task around implementing the Spark execution engine for Hive lies in query planning, where Hive operator plans from the semantic analyzer which is translated to a task plan that Spark can execute. It also includes query execution, where the generated Spark plan gets actually executed in the Spark cluster.

* 1. **Question 66. What Is A “parquet” In Spark?**

**Answer :**

“Parquet” is a columnar format file supported by many data processing systems. Spark SQL performs both read and write operations with the “Parquet” file.

* 1. **Question 67. What Are Benefits Of Spark Over Mapreduce?**

**Answer :**

Due to the availability of in-memory processing, Spark implements the processing around 10-100x faster than Hadoop MapReduce. MapReduce makes use of persistence storage for any of the data processing tasks.

* + 1. Unlike Hadoop, Spark provides in-built libraries to perform multiple tasks form the same core like batch processing, Steaming, Machine learning, Interactive SQL queries. However, Hadoop only supports batch processing.
    2. Hadoop is highly disk-dependent whereas Spark promotes caching and in-memory data storage
    3. Spark is capable of performing computations multiple times on the same dataset. This is called iterative computation while there is no iterative computing implemented by Hadoop.
  1. **Question 68. How Sparksql Is Different From Hql And Sql?**

**Answer :**

SparkSQL is a special component on the spark Core engine that support SQL and Hive Query Language without changing any syntax. It’s possible to join SQL table and HQL table.

| **Spark SQL Programming Related Tutorials** | |
| --- | --- |
| [Python Tutorial](https://www.wisdomjobs.com/e-university/python-tutorial-176.html) | [Core Java Tutorial](https://www.wisdomjobs.com/e-university/core-java-tutorial-231.html) |
| [JDBC Tutorial](https://www.wisdomjobs.com/e-university/jdbc-tutorial-278.html) | [DB2 Using SQL Tutorial](https://www.wisdomjobs.com/e-university/db2-using-sql-tutorial-283.html) |
| [Hibernate Tutorial](https://www.wisdomjobs.com/e-university/hibernate-tutorial-879.html) | [Java Swing Tutorial](https://www.wisdomjobs.com/e-university/java-swing-tutorial-1081.html) |
| [Scala Tutorial](https://www.wisdomjobs.com/e-university/scala-tutorial-1348.html) | [Apache ZooKeeper Tutorial](https://www.wisdomjobs.com/e-university/apache-zookeeper-tutorial-1739.html) |
| [Hyper SQL Database Tutorial](https://www.wisdomjobs.com/e-university/hyper-sql-database-tutorial-1770.html) |  |
| **Spark SQL Programming Related Interview Questions** | |
| [Python Interview Questions](https://www.wisdomjobs.com/e-university/python-interview-questions.html) | [Core Java Interview Questions](https://www.wisdomjobs.com/e-university/core-java-interview-questions.html) |
| [JDBC Interview Questions](https://www.wisdomjobs.com/e-university/jdbc-interview-questions.html) | [DB2 Using SQL Interview Questions](https://www.wisdomjobs.com/e-university/db2-using-sql-interview-questions.html) |
| [Domain Name System(DNS) Interview Questions](https://www.wisdomjobs.com/e-university/domain-name-system-dns-interview-questions.html) | [Hibernate Interview Questions](https://www.wisdomjobs.com/e-university/hibernate-interview-questions.html) |
| [MYSQL DBA Interview Questions](https://www.wisdomjobs.com/e-university/mysql-dba-interview-questions.html) | [Java Swing Interview Questions](https://www.wisdomjobs.com/e-university/java-swing-interview-questions.html) |
| [SQLite Interview Questions](https://www.wisdomjobs.com/e-university/sqlite-interview-questions.html) | [Apache Spark Interview Questions](https://www.wisdomjobs.com/e-university/apache-spark-interview-questions.html) |
| [Scala Interview Questions](https://www.wisdomjobs.com/e-university/scala-interview-questions.html) | [Apache ZooKeeper Interview Questions](https://www.wisdomjobs.com/e-university/apache-zookeeper-interview-questions.html) |
| [Hyper SQL Database Interview Questions](https://www.wisdomjobs.com/e-university/hyper-sql-database-interview-questions.html) | [Apache Tomcat Interview Questions](https://www.wisdomjobs.com/e-university/apache-tomcat-interview-questions.html) |

| **Spark SQL Programming Related Practice Tests** | |
| --- | --- |
| [Python Practice Tests](https://www.wisdomjobs.com/e-university/python-practice-tests-176-326595) | [Core Java Practice Tests](https://www.wisdomjobs.com/e-university/core-java-practice-tests-231-327027) |
| [JDBC Practice Tests](https://www.wisdomjobs.com/e-university/jdbc-practice-tests-278-326781) | [DB2 Using SQL Practice Tests](https://www.wisdomjobs.com/e-university/db2-using-sql-practice-tests-283-327162) |
| [Domain Name System(DNS) Practice Tests](https://www.wisdomjobs.com/e-university/domain-name-system-dns-practice-tests-791-327358) | [Hibernate Practice Tests](https://www.wisdomjobs.com/e-university/hibernate-practice-tests-879-327399) |
| [MYSQL DBA Practice Tests](https://www.wisdomjobs.com/e-university/mysql-dba-practice-tests-1000-327486) | [Scala Practice Tests](https://www.wisdomjobs.com/e-university/scala-practice-tests-1348-327870) |

[Spark Sql Programming Tutorial](https://www.wisdomjobs.com/e-university/spark-sql-programming-tutorial-1687.html)

**Spark Sql Programming**

* [Spark Introduction](https://www.wisdomjobs.com/e-university/spark-sql-programming-tutorial-1687/spark-introduction-19159.html)
* [Spark Rdd](https://www.wisdomjobs.com/e-university/spark-sql-programming-tutorial-1687/spark-rdd-19164.html)
* [Spark Installation](https://www.wisdomjobs.com/e-university/spark-sql-programming-tutorial-1687/spark-installation-19173.html)
* [Spark Sql Introduction](https://www.wisdomjobs.com/e-university/spark-sql-programming-tutorial-1687/spark-sql-introduction-19174.html)
* [Spark Sql Dataframes](https://www.wisdomjobs.com/e-university/spark-sql-programming-tutorial-1687/spark-sql-dataframes-19230.html)
* [Spark Sql Data Sources](https://www.wisdomjobs.com/e-university/spark-sql-programming-tutorial-1687/spark-sql-data-sources-19231.html)

[All Tutorials](https://www.wisdomjobs.com/e-university/all-skillsets.html)

[All Practice Tests](https://www.wisdomjobs.com/e-university/onlineexam.html)

**List of Tutorials**

* [Developers Best Practices Tutorial https://www.wisdomjobs.com/img/new_icon.gif](https://www.wisdomjobs.com/e-university/developers-best-practices-tutorial-3129.html)
* [YAML Tutorial https://www.wisdomjobs.com/img/new_icon.gif](https://www.wisdomjobs.com/e-university/yaml-tutorial-3120.html)
* [Salesforce Tutorial https://www.wisdomjobs.com/img/new_icon.gif](https://www.wisdomjobs.com/e-university/salesforce-tutorial-3117.html)
* [Adobe Robohelp Tutorial](https://www.wisdomjobs.com/e-university/adobe-robohelp-tutorial-3114.html)
* [Sublime Text Tutorial](https://www.wisdomjobs.com/e-university/sublime-text-tutorial-3104.html)
* [GItlab Tutorial](https://www.wisdomjobs.com/e-university/gitlab-tutorial-3096.html)
* [Adobe InDesign CC Tutorial](https://www.wisdomjobs.com/e-university/adobe-indesign-cc-tutorial-3092.html)
* [SaltStack Tutorial](https://www.wisdomjobs.com/e-university/saltstack-tutorial-3063.html)

[**Read More**](https://www.wisdomjobs.com/e-university/all-skillsets.html)

**List of Topics**

* [Summary https://www.wisdomjobs.com/img/new_icon.gif](https://www.wisdomjobs.com/e-university/developers-best-practices-tutorial-3129/summary-28360.html)
* [Career Planning in best practice https://www.wisdomjobs.com/img/new_icon.gif](https://www.wisdomjobs.com/e-university/developers-best-practices-tutorial-3129/career-planning-in-best-practice-28359.html)
* [Managing Managers https://www.wisdomjobs.com/img/new_icon.gif](https://www.wisdomjobs.com/e-university/developers-best-practices-tutorial-3129/managing-managers-28358.html)
* [Stress Management https://www.wisdomjobs.com/img/new_icon.gif](https://www.wisdomjobs.com/e-university/developers-best-practices-tutorial-3129/stress-management-28357.html)
* [Eager to Learn](https://www.wisdomjobs.com/e-university/developers-best-practices-tutorial-3129/eager-to-learn-28356.html)

**Q1. What is Apache Spark ?**  
Apache Spark is a lightning-fast cluster computing technology, designed for fast computation. It is based on Hadoop MapReduce and it extends the MapReduce model to efficiently use it for more types of computations, which includes interactive queries and stream processing. The main feature of Spark is its **in-memory cluster computing** that increases the processing speed of an application.  
Spark is designed to cover a wide range of workloads such as batch applications, iterative algorithms, interactive queries and streaming. Apart from supporting all these workload in a respective system, it reduces the management burden of maintaining separate tools.

**Q2. What is sparkContext?**  
SparkContext is the entry point to Spark. Using sparkContext you create RDDs which provided various ways of churning data.

**Q3. Why is Spark faster than MapReduce?**  
**A.**There are few important reasons why Spark is faster than MapReduce and some of them are below:

* There is no tight coupling in Spark i.e., there is no mandatory rule that reduce must come after map.
* Spark tries to keep the data “in-memory” as much as possible.

In MapReduce, the intermediate data will be stored in HDFS and hence takes longer time to get the data from a source but this is not the case with Spark.

**Q4. Explain the Apache Spark Architecture.**

* Apache Spark application contains two programs namely a Driver program and Workers program.
* A cluster manager will be there in-between to interact with these two cluster nodes. Spark Context will keep in touch with the worker nodes with the help of Cluster Manager.
* Spark Context is like a master and Spark workers are like slaves.
* Workers contain the executors to run the job. If any dependencies or arguments have to be passed then Spark Context will take care of that. RDD’s will reside on the Spark Executors.
* You can also run Spark applications locally using a thread, and if you want to take advantage of distributed environments you can take the help of S3, HDFS or any other storage system

**Q5. What are the key features of Spark.**

* Allows Integration with Hadoop and files included in HDFS.
* Spark has an interactive language shell as it has an independent Scala (the language in which Spark is written) interpreter.
* Spark consists of RDD’s (Resilient Distributed Datasets), which can be cached across computing nodes in a cluster.
* Spark supports multiple analytic tools that are used for interactive query analysis , real-time analysis and graph processing

**Q6. What is Shark?**  
Most of the data users know only SQL and are not good at programming. Shark is a tool, developed for people who are from a database background – to access Scala MLib capabilities through Hive like SQL interface. Shark tool helps data users run Hive on Spark – offering compatibility with Hive metastore, queries and data.

**Q7. On which all platform can Apache Spark run?**  
Spark can run on the following platforms:

* **YARN (Hadoop):** Since yarn can handle any kind of workload, the spark can run on Yarn. Though there are two modes of execution. One in which the Spark driver is executed inside the container on node and second in which the Spark driver is executed on the client machine. This is the most common way of using Spark.
* **Apache Mesos:** Mesos is an open source good upcoming resource manager. Spark can run on Mesos.
* **EC2:** If you do not want to manage the hardware by yourself, you can run the Spark on top of Amazon EC2. This makes spark suitable for various organizations.
* **Standalone:**If you have no resource manager installed in your organization, you can use the standalone way. Basically, Spark provides its own resource manager. All you have to do is install Spark on all nodes in a cluster, inform each node about all nodes and start the cluster. It starts communicating with each other and run.

**Q8. What are the various programming languages supported by Spark?**  
Though Spark is written in Scala, it lets the users code in various languages such as:

* Scala
* Java
* Python
* R (Using SparkR)
* SQL (Using SparkSQL)

Also, by the way of piping the data via other commands, we should be able to use all kinds of programming languages or binaries.

**Q9. Compare Spark vs Hadoop MapReduce**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Hadoop MapReduce** | **Apache Spark** |
| **Memory** | Does not leverage the memory of the hadoop cluster to maximum. | Let’s save data on memory with the use of RDD’s. |
| **Disk usage** | MapReduce is disk oriented. | Spark caches data in-memory and ensures low latency. |
| **Processing** | Only batch processing is supported | Supports real-time processing through spark streaming. |
| **Installation** | Is bound to hadoop. | Is not bound to Hadoop. |
| **Spark vs Hadoop** | | |

**Q10. What are actions and transformations?**  
Transformations create new RDD’s from existing RDD and these transformations are lazy and will not be executed until you call any action.  
Eg: map(), filter(), flatMap(), etc.,  
Actions will return results of an RDD.  
Eg: reduce(), count(), collect(), etc.,

**Q11. What are the various storages from which Spark can read data?**  
Spark has been designed to process data from various sources. So, whether you want to process data stored in HDFS, Cassandra, EC2, Hive, HBase, and Alluxio (previously Tachyon). Also, it can read data from any system that supports any Hadoop data source.

**Q12. List some use cases where Spark outperforms Hadoop in processing.**

1. Sensor Data Processing –Apache Spark’s ‘In-memory computing’ works best here, as data is retrieved and combined from different sources.
2. Spark is preferred over Hadoop for real time querying of data
3. Stream Processing – For processing logs and detecting frauds in live streams for alerts, Apache Spark is the best solution.

**Q13. What is Spark Driver?**  
Spark Driver is the program that runs on the master node of the machine and declares transformations and actions on data RDDs. In simple terms, driver in Spark creates SparkContext, connected to a given Spark Master.The driver also delivers the RDD graphs to Master, where the standalone cluster manager runs.

**Q14. What are Accumulators?**  
Accumulators are the write only variables which are initialized once and sent to the workers. These workers will update based on the logic written and sent back to the driver which will aggregate or process based on the logic.  
Only driver can access the accumulator’s value. For tasks, Accumulators are write-only. For example, it is used to count the number errors seen in RDD across workers.

**Q15 . What is Hive on Spark?**  
Hive contains significant support for Apache Spark, wherein Hive execution is configured to Spark:

hive> set spark.home=/location/to/sparkHome;

hive> set hive.execution.engine=spark;

Hive on Spark supports Spark on yarn mode by default.

**Q16. What are Broadcast Variables?**  
Broadcast Variables are the read-only shared variables. Suppose, there is a set of data which may have to be used multiple times in the workers at different phases, we can share all those variables to the workers from the driver and every machine can read them.

**Q17. What are the optimizations that developer can make while working with spark?**  
1.Spark is memory intensive, whatever you do it does in memory.  
2.Firstly, you can adjust how long spark will wait before it times out on each of the phases of data locality (data local –> process local –> node local –> rack local –> Any)  
3.Filter out data as early as possible. For caching, choose wisely from various storage levels.  
4.Tune the number of partitions in spark.

**Q18. What is Spark SQL?**  
Spark SQL is a module for structured data processing where we take advantage of SQL queries running on the datasets.

**Q19. What is Spark Streaming?**  
Whenever there is data flowing continuously and you want to process the data as early as possible, in that case you can take the advantage of Spark Streaming. It is the API for stream processing of live data. Data can flow for Kafka, Flume or from TCP sockets, Kenisis etc., and you can do complex processing on the data before you pushing them into their destinations. Destinations can be file systems or databases or any other dashboards.

**Q20. What is Sliding Window?**  
In Spark Streaming, you have to specify the batch interval. For example, let’s take your batch interval is 10 seconds, Now Spark will process the data whatever it gets in the last 10 seconds i.e., last batch interval time.But with Sliding Window, you can specify how many last batches has to be processed. In the below screen shot, you can see that you can specify the batch interval and how many batches you want to process. Apart from this, you can also specify when you want to process your last sliding window. For example you want to process the last 3 batches when there are 2 new batches. That is like when you want to slide and how many batches has to be processed in that window.

**Q21. What does MLlib do?**  
MLlib is scalable machine learning library provided by Spark. It aims at making machine learning easy and scalable with common learning algorithms and use cases like clustering, regression filtering, dimensional reduction, and alike.

**Q22. List the functions of Spark SQL.?**  
Spark SQL is capable of:

* Loading data from a variety of structured sources.
* Querying data using SQL statements, both inside a Spark program and from external tools that connect to Spark SQL through standard database connectors (JDBC/ODBC). For instance, using business intelligence tools like Tableau.
* Providing rich integration between SQL and regular Python/Java/Scala code, including the ability to join RDDs and SQL tables, expose custom functions in SQL, and more.

**Q23. How can Spark be connected to Apache Mesos?**  
To connect Spark with Mesos-

* Configure the spark driver program to connect to Mesos. Spark binary package should be in a location accessible by Mesos. (or)
* Install Apache Spark in the same location as that of Apache Mesos and configure the property ‘spark.mesos.executor.home’ to point to the location where it is installed.

**Q24. Is it possible to run Spark and Mesos along with Hadoop?**  
Yes, it is possible to run Spark and Mesos with Hadoop by launching each of these as a separate service on the machines. Mesos acts as a unified scheduler that assigns tasks to either Spark or Hadoop.

**Q25. When running Spark applications, is it necessary to install Spark on all the nodes of YARN cluster?**  
Spark need not be installed when running a job under YARN or Mesos because Spark can execute on top of YARN or Mesos clusters without affecting any change to the cluster.

**Q26. What is Catalyst framework?**  
Catalyst framework is a new optimization framework present in Spark SQL. It allows Spark to automatically transform SQL queries by adding new optimizations to build a faster processing system.

**Q27. Name a few companies that use Apache Spark in production.**  
Pinterest, Conviva, Shopify, Open Table

**Q28.** **Why is BlinkDB used?**  
BlinkDB is a query engine for executing interactive SQL queries on huge volumes of data and renders query results marked with meaningful error bars. BlinkDB helps users balance ‘query accuracy’ with response time.

**Q29. What are the common mistakes developers make when running Spark applications?**  
Developers often make the mistake of-

* Hitting the web service several times by using multiple clusters.
* Run everything on the local node instead of distributing it.

Developers need to be careful with this, as Spark makes use of memory for processing.

**Q30. What is the advantage of a Parquet file?**  
Parquet file is a columnar format file that helps –

* Limit I/O operations
* Consumes less space
* Fetches only required columns.

**Summary**

**Reviewer**

Tilak Sha

**Review Date**

2018-04-02

**Reviewed Item**

Excellent!! Collection of Apache Spark Interview Questions and Answers for freshers and experienced. Thank you mytectra.com.

**Author Rating**

1star1star1star1star1star

 Post Views: 2,173

**Share**

[APACHE SPARK INTERVIEW QUESTIONS AND ANSWERS](https://www.mytectra.com/interview-question/tag/apache-spark-interview-questions-and-answers/)

### RECOMMENDED POSTS

[](https://www.mytectra.com/interview-question/blockchain-interview-questions/)

##### [Blockchain Interview Questions](https://www.mytectra.com/interview-question/blockchain-interview-questions/)

January 29, 2019

[](https://www.mytectra.com/interview-question/crm-interview-questions/)

##### [CRM Interview Questions](https://www.mytectra.com/interview-question/crm-interview-questions/)

May 26, 2018

[](https://www.mytectra.com/interview-question/sap-bibw-frequently-asked-interview-questions/)

##### [SAP BI/BW Frequently asked Interview Questions.](https://www.mytectra.com/interview-question/sap-bibw-frequently-asked-interview-questions/)

Apache Spark Streaming - Interview Questions

### What is Apache Spark Streaming?

**FAQ**

Spark Streaming is a library provided in Apache Spark for processing live data streams that is scalable, has high-throughput and is fault-tolerant. Spark Streaming can ingest data from multiple sources such as Kafka, Flume, Kinesis or TCP sockets; and process this data using complex algorithms provided in the Spark API including algorithms provided in the Spark MLlib and GraphX libraries. Processed data can be pushed to live dashboards, file systems and databases.

### Describe how Spark Streaming processes data?

**FAQ**

Apache Spark Streaming component receives live data streams from input sources such as Kafka, Flume, Kinesis etc. and divides them into batches. The Spark engine processes these input batches and produces the final stream of results in batches.

### What are DStreams?

**FAQ**

DStreams, or discretized streams, are high-level abstractions provided in Spark Streaming that represents a continuous stream of data. DStreams can be either created from input sources such as Kafka, Flume or Kinesis; or by applying high-level operations on existing DStreams.

Internally, a DStream is represented by a continuous series of RDDs. Each RDD in a DStream contains data from a certain interval.

### What is a StreamingContext object?

**FAQ**

StreamingContext object represents the entry point to a Spark streaming application and contains the batch interval for the Spark streaming program and the SparkConf object. The SparkConf object and the batch interval are set when creating a new instance of the StreamingContext object.

### What are the two different types of built-in streaming sources provided by Spark Streaming?

**FAQ**

Spark Streaming provides two different categories of built-in streaming sources.

**Basic Sources** - These sources are directly available in the StreamingContext API. Example: file systems and socket connections.

**Advanced Sources** - These are sources such as Kafka, Flume etc. that are provided in the Spark Streaming library through extra utility classes.

**Big Data Interview Guide has over 150+ interview questions and answers. Get the guide for $44.95 only.**

[BUY EBOOK](https://gumroad.com/l/MChNX)

### What are some of the common transformations on DStreams supported by Spark Streaming ?

**FAQ**

**map(func)** - map() transformation returns a new distributed dataset from a source dataset formed by passing each element of the source through a function func.

**filter()** - filter() transformation returns a new distributed dataset from a source dataset formed by selecting the elements of the source on which func returns true.

**flatMap()** - flatMap() transformation is similar to map() function. In flatMap() each input item can be mapped to 0 or more output items.

**union()** - union() transformation returns a new dataset that contains the union of the elements in the source dataset and the dataset that is passed as argument to the function.

**intersection()** - intersection() transformation returns a new distributed dataset that contains the intersection of elements in the source dataset and the dataset that is passed as argument to the function.

**distinct()** - distinct() transformation returns a new distributed dataset that contains the distinct elements of the source dataset.

**groupByKey()** - groupByKey(func) transformation called on a dataset of (K, V) pairs and returns a dataset of (K, Iterable) pairs.

**reduceByKey(func)** - reduceByKey(func) transformation is called on a dataset of (K, V) pairs, and returns a dataset of (K, V) pairs where the values for each key are aggregated using the given reduce function func.

**aggregateByKey(func)** - aggregateByKey() transformation is called on a dataset of (K, V) pairs, returns a dataset of (K, U) pairs where the values for each key are aggregated using the given combine functions and a neutral zero value.

**sortByKey()** - sortByKey() transformation is called on a dataset of (K, V) pairs which returns a dataset of (K, V) pairs sorted by keys in ascending or descending order, as specified in the boolean ascending argument.

### What are the output operations that can be performed on DStreams?

**FAQ**

Output operations on DStreams pushes the DStream's data to external systems like a database or a file system. Following are the key operations that can be performed on DStreams.

**saveAsTextFiles()** - Saves the DStream's data as text file.

**saveAsObjectFiles()** - Saves the DStreams data as serialized Java objects.

**saveAsHadoopFiles()** - Saves the DStream's data as Hadoop files.

**foreachRDD()** - A generic output operator that applies a function, func, to each RDD generated from the DStream.

Important Keywords to Remember 

Apache Spark Streaming

DStreams

StreamingContext

Transformations

Output operations - saveAsTextFiles(), saveAsObjectFiles(), saveAsHadoopFiles(), foreachRDD()



# $49.95

[BUY EBOOK](https://gumroad.com/l/MChNX)

  SSL Secure Payment

Java Interview Quesiuons - Secure Payment

[DOWNLOAD FREE CHAPTER](http://eepurl.com/dslSzn)

**Thank You For Reading**

### This site tries to follow the principle of 'Master the Concepts -> Ace the Interview'. Focus on mastering the concepts while preparing for the interview, Success will follow naturally in interviews as well as in your career.

### This same philosophy is followed in the Big Data Interview Guide. The guide has 150 plus interview questions, separated into key chapters or focus areas. Each of the questions has detailed answers and most with code snippets that will help you in white-boarding interview sessions. The guide is structured to give you a definite and focused edge over other candidates.

Good Luck!

Apache Spark Streaming - Interview Questions

### What is Apache Spark Streaming?

**FAQ**

Spark Streaming is a library provided in Apache Spark for processing live data streams that is scalable, has high-throughput and is fault-tolerant. Spark Streaming can ingest data from multiple sources such as Kafka, Flume, Kinesis or TCP sockets; and process this data using complex algorithms provided in the Spark API including algorithms provided in the Spark MLlib and GraphX libraries. Processed data can be pushed to live dashboards, file systems and databases.

### Describe how Spark Streaming processes data?

**FAQ**

Apache Spark Streaming component receives live data streams from input sources such as Kafka, Flume, Kinesis etc. and divides them into batches. The Spark engine processes these input batches and produces the final stream of results in batches.

### What are DStreams?

**FAQ**

DStreams, or discretized streams, are high-level abstractions provided in Spark Streaming that represents a continuous stream of data. DStreams can be either created from input sources such as Kafka, Flume or Kinesis; or by applying high-level operations on existing DStreams.

Internally, a DStream is represented by a continuous series of RDDs. Each RDD in a DStream contains data from a certain interval.

### What is a StreamingContext object?

**FAQ**

StreamingContext object represents the entry point to a Spark streaming application and contains the batch interval for the Spark streaming program and the SparkConf object. The SparkConf object and the batch interval are set when creating a new instance of the StreamingContext object.

### What are the two different types of built-in streaming sources provided by Spark Streaming?

**FAQ**

Spark Streaming provides two different categories of built-in streaming sources.

**Basic Sources** - These sources are directly available in the StreamingContext API. Example: file systems and socket connections.

**Advanced Sources** - These are sources such as Kafka, Flume etc. that are provided in the Spark Streaming library through extra utility classes.

**Big Data Interview Guide has over 150+ interview questions and answers. Get the guide for $44.95 only.**

[BUY EBOOK](https://gumroad.com/l/MChNX)

### What are some of the common transformations on DStreams supported by Spark Streaming ?

**FAQ**

**map(func)** - map() transformation returns a new distributed dataset from a source dataset formed by passing each element of the source through a function func.

**filter()** - filter() transformation returns a new distributed dataset from a source dataset formed by selecting the elements of the source on which func returns true.

**flatMap()** - flatMap() transformation is similar to map() function. In flatMap() each input item can be mapped to 0 or more output items.

**union()** - union() transformation returns a new dataset that contains the union of the elements in the source dataset and the dataset that is passed as argument to the function.

**intersection()** - intersection() transformation returns a new distributed dataset that contains the intersection of elements in the source dataset and the dataset that is passed as argument to the function.

**distinct()** - distinct() transformation returns a new distributed dataset that contains the distinct elements of the source dataset.

**groupByKey()** - groupByKey(func) transformation called on a dataset of (K, V) pairs and returns a dataset of (K, Iterable) pairs.

**reduceByKey(func)** - reduceByKey(func) transformation is called on a dataset of (K, V) pairs, and returns a dataset of (K, V) pairs where the values for each key are aggregated using the given reduce function func.

**aggregateByKey(func)** - aggregateByKey() transformation is called on a dataset of (K, V) pairs, returns a dataset of (K, U) pairs where the values for each key are aggregated using the given combine functions and a neutral zero value.

**sortByKey()** - sortByKey() transformation is called on a dataset of (K, V) pairs which returns a dataset of (K, V) pairs sorted by keys in ascending or descending order, as specified in the boolean ascending argument.

### What are the output operations that can be performed on DStreams?

**FAQ**

Output operations on DStreams pushes the DStream's data to external systems like a database or a file system. Following are the key operations that can be performed on DStreams.

**saveAsTextFiles()** - Saves the DStream's data as text file.

**saveAsObjectFiles()** - Saves the DStreams data as serialized Java objects.

**saveAsHadoopFiles()** - Saves the DStream's data as Hadoop files.

**foreachRDD()** - A generic output operator that applies a function, func, to each RDD generated from the DStream.

Important Keywords to Remember 

Apache Spark Streaming

DStreams

StreamingContext

Transformations

Output operations - saveAsTextFiles(), saveAsObjectFiles(), saveAsHadoopFiles(), foreachRDD()



# $49.95

[BUY EBOOK](https://gumroad.com/l/MChNX)

  SSL Secure Payment

Java Interview Quesiuons - Secure Payment

[DOWNLOAD FREE CHAPTER](http://eepurl.com/dslSzn)

**Thank You For Reading**

### This site tries to follow the principle of 'Master the Concepts -> Ace the Interview'. Focus on mastering the concepts while preparing for the interview, Success will follow naturally in interviews as well as in your career.

### This same philosophy is followed in the Big Data Interview Guide. The guide has 150 plus interview questions, separated into key chapters or focus areas. Each of the questions has detailed answers and most with code snippets that will help you in white-boarding interview sessions. The guide is structured to give you a definite and focused edge over other candidates.

Good Luck!

Apache\_Spark\_Interview\_Questions\_Book.pdf

Apache\_Spark\_Interview\_Questions\_Book.pdf

### [1. What do you understand by the partitions in spark?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled1)

Partitions are done in order to simplify the data as they are the logical distribution of entire data. It is similar to the split in MapReduce. In order to enhance the processing speed, this logical distribution is carried out. Each and every association in Apache Spark is a partitioned RDD.

### [2. How can we create RDDs in Apache spark?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled2)

In order to create RDD there are basically two methods –

* In a driver program parallelizes a collection. This will use the sparkcontext's “Parallelize”.
* From the external storage, an external data sheet can be loaded into the file system.

### [3. What do you understand about yarn?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled3)

The YARN is a key feature in Spark which provides a central resource management for most of the operational deliveries across a cluster. It is also a container manager like Mesos. Spark can easily run on YARN which eventually emphasizes a binary distribution of Apache Spark built on its support.

### [4. Can you define RDD?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled4)

The acronym for Resale in Distributed Datasheets is RDD. It is a fault-tolerant collection for all of the elements that run parallel. The sorted data in RDD is immutable and primarily of two types –

* Parallelized collections
* Hadoop datasets

### [5. What are accumulators in spark?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled5)

The write only variables which are initially executed once and send to the workers are accumulators. On the basis of the logic written, these workers will be updated and sent back to the driver which will process it on the basis of logic. A driver has the potential to exercise accumulator’s value.

### [6. Can you explain about the cluster manager of Apache spark?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled6)

There are three different cluster manager in Spark which are as-

* YARN
* Apache Mesos - Spark along with several other applications can easily be scheduled and run over it. Its priority is to scale down the allocations between several commands in order to provide interfaces when several users run their shells.
* Standalone deployments - For the easy setup and convenience, for the new deployments.

### [7. List few benefits of spark over map reduce?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled7)

The benefits are –

* With the in-memory processing, Spark implements it around 100 times faster than the head of MapReduce.
* Spark provides inbuilt libraries for most of the multidimensional task as compared to map reduce.
* Spark is independent of the disk storage on the other hand Hadoop is highly dependent.
* Spark can perform multiple computations within the same datasheet.

### [8. If map reduce is inferior to Spark then is there any benefit of learning it?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled8)

Apache Spark is far better than MapReduce but still learning MapReduce is essential. MapReduce is a paradigm which is even used by Spark as big data tools. When the data is large and grows bigger, in that case, MapReduce is much relevant. Data tools like pig and hive convert their message queries into MapReduce in order to optimize them properly.

### [9. Do you know the comparative differences between Apache Spark and Hadoop?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled9)

Yes there are several segments on which they can be differentiated. Few of them are-

|  |  |  |
| --- | --- | --- |
| Feature | Apache spark | Hadoop |
| Speed | It is almost 100 times faster than Hadoop | It has moderate speed |
| Processing | Offers real time and batch processing functionality | It offers batch processing only |
| Difficulty | It has high level modules hence it is easy | It is tough to learn |
| Recovery | It allows the partition recovery | MapReduce |
| Interactivity | It has interactive modes | Other than Pig and Hive, it has no interactive mode |

### [10. What is Spark SQL?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled10)

Spark SQL is basically a module which is formulated to provide structured data processing. The advantage of SQL message queries running on the datasheets can be taken from it.

### [11. Can you mention some features of spark?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled11)

### [12. What is Apache Spark?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled12)

Apache Spark is basically a processing framework which is extremely fast and convenient to use. With an advanced execution engine supporting it offers the cyclic data flow and in-memory computation. Apache spark can also run on Hadoop, cloud or standalone. It is capable to access the diverse data including Cassandra, HDFS, HBase.

### [13. Name the languages which are supported by Apache Spark and which one is most popular?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled13)

Apache Spark supports the languages Java, Python, Scala and R. among them Scala and Python have interactive shares for Apache Spark and Scala shell can be easily accessed through the ./bin/spark-shell and Python can be accessed through ./bin/pyspark. Among them, Scala is the most popular because Apache Spark is written in Scala.

### [14. Name few companies that are the uses of Apache spark?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled14)

The companies that are using Apache Spark with the production are Pinterest, Shopify, Open Table, ComViva.

### [15. Name the operations supported by RDD?](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/" \l "collapseUnfiled15)

As the major logical data units in Apache Spark, RDD possesses a distributed collection of data. It is a read-only data structure and you cannot change the original format but it can always be transformed into a different form with the changes. The two operations which are supported by RDD are -

* Transformation - It creates a new RDD from the former one. They are executed only on demand.
* Actions - The final outcomes of the RDD computations are returned by actions.
* ‹
* 1
* [2](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/page/2/)
* [›](https://www.onlineinterviewquestions.com/apache-spark-interview-questions/page/2/)

##### **Got a Questions? Share with us**

Top of Form

 POST

Bottom of Form

## **Features, Pros and Cons of Apache Spark**

Apache Spark is a tool for large data processing and execution. It also offers high-level operators which effortlessly develop a parallel application for the processing. The most prominent Apache Spark features are:

* Polyglot – multiple languages platform
* It is proficient in speed
* It also supports the multiple formats
* Provides real-time computation in data
* Efficient machine learning

## **Pros associated**

* In memory computation of spark.
* Reusability of the spark code.
* Offers fault tolerance.

## **Cons associated**

* Sometimes it becomes a bottleneck when it comes to cost efficiency.
* As compared to Apache Flink, Apache spark has higher latency.

| **Also Read Related Apache Spark Interview Questions** | | |
| --- | --- | --- |
| [**A+ interview Questions**](https://www.onlineinterviewquestions.com/a-plus-interview-questions-answers/) | [**Git Interview Questions**](https://www.onlineinterviewquestions.com/git-interview-questions/) |  |
| [**GWT interview questions**](https://www.onlineinterviewquestions.com/gwt-interview-questions/) | [**IELTS Interview Questions**](https://www.onlineinterviewquestions.com/ielts-interview-questions/) |  |
| [**Interview Questions for Hiring managers**](https://www.onlineinterviewquestions.com/interview-questions-hiring-managers/) |  |  |

#### **. What is Apache Spark?**

Answer: Spark is an in-memory parallel data processing framework. It support batch, streaming processing also interactive analytics.

#### **Q2. What are the three ways to create RDD in Spark?**

Answer: The three ways to create RDD in Spark is:

1. By using parallelized collection
2. By loading an external dataset
3. From an existing RDD

#### **Q3. Can we create RDD from existing RDD?**

Answer:  Yes, by applying transformations on RDD we can create new RDD.

#### **Q4. In how many ways we can create RDD?**

Answer: There are three possible ways to create RDD.

#### **Q5. Can we create RDD using Dataset like .txt file?**

Answer: Yes, by loading dataset we can create RDD.

#### **Q6. Can we run Spark without using HDFS?**

Answer: Yes, we need HDFS just for storage purpose.

#### **Q7. Does spark support stand alone mode?**

Answer: Yes it supports standalone mode

#### **Q8. What are the types of transformations in Spark?**

Answer: Narrow and Wide Transformation are available in Spark.

#### **Q9. Give some example for Narrow transformation?**

Answer: Map and Filter.

#### **Q10. Give some example of wide transformations?**

Answer: GroupByKey and ReduceByKey

[*Become a Spark Certified Expert in 25 Hrs*](https://www.gangboard.com/big-data-training/apache-spark-training?utm_source=GB_blog)

#### **Q11. What are the Components in Spark?**

Answer: Spark SQL, Spark Streaming, Mlib And Graph X

#### **Q12. What is Spark SQL?**

Answer: It is a component of Spark which provides support for structured and semi-structured data. Data Frame appeared in Spark Release 1.3.0

#### **Q13. What is Spark Dataset?**

Answer: Dataset is an extension of Data Frame API which provides type-safe, object-oriented programming interface.

#### **Q14. What are the limitations of Data frame?**

Answer: Data Frame does not have provision for compile-time type safety.

#### **Q15. What is the transformation in Spark?**

Answer: Transformation is a function that produces new RDD from the existing RDDs. It takes RDD as input and produces output as one or more RDD.

#### **Q16. What is Action in Spark?**

Answer: Actions return final results of RDD computations. It triggers execution using lineage graph and after carry out all intermediate transformations return the final results to the Driver.

#### **Q17. Give some examples of Transformation in Spark?**

Answer: Map, flatmap and filter.

#### **Q18. Give some example of Action in Spark?**

Answer: Count(),Collect() and reduce(func).

#### **Q19. What collect does in Spark?**

Answer: It returns all the elements in the RDD to Driver.

#### **Q20. What are the different storage levels in Spark?**

Answer: Memory\_Only, MEMORY\_AND\_DISK, MEMORY\_ONLY\_SER , MEMORY\_AND\_DISK\_SER  & DISK\_Only

#### **Q21. What is the default storage level in Spark?**

Answer:  Memory\_Only

#### **Q22. What is coalesce ()?**

Answer: It is used to decrease the number of partitions in an RDD. It avoids full shuffle of RDD.

#### **Q23. What is repartition ()?**

Answer: It is used to increase the number of Partitions. It creates a new partition from the existing partition by shuffling of data.

#### **Q24. What is the role of Driver in Spark?**

Answer: The driver is the program which creates the Spark Context, connecting to a given Spark Master. It declares the transformations and actions on RDDs and submits such requests to the master.

#### **Q25. What are the deployment modes in Spark?**

Answer: Cluster mode and Client Mode

#### **Q26. Why Spark is faster than Hadoop?**

Answer: Spark is faster than Hadoop because it does processing in memory.

#### **Q27. What is accumulator?**

Answer: An accumulator is a shared variable which is used for aggregating information across the cluster.

#### **Q28. What are the two types of shared variable available in Apache Spark?**

Answer: Broadcast Variable and Accumulator.

#### **Q29. What is the Broadcast variable?**

Answer: It allows the programmer to keep a read-only variable cached on each machine instead of shipping a copy of it with tasks.

#### **Q30. What is Spark D Stream?**

Answer: Spark DStream is the basic abstraction of Spark Streaming. It is a continuous stream of data.

[*Become a Spark Expert with Certification in 25hours*](https://www.gangboard.com/big-data-training/apache-spark-training?utm_source=GB_blog)

#### **Q31.What is map transformation in Spark?**

Answer: Map transformation takes a function as input after applying that function to each RDD return another RDD. Its return type can be different from its input type.

#### **Q32. What is a flatmap transformation in Spark?**

Answer: Flatmap is used when we want to produce multiple elements for each input element. The output of the flatmap is a List of the element through which we can iterate.

#### **Q33. What is action Reduce in Spark?**

Answer: Reduce takes a function as an input which has two parameters which are of same type and output a single value of the input type.

#### **Q34. What is lazy evaluation in Spark?**

Answer: When we apply transformation on RDD it does not immediately gives output it will make DAG of all transformation. Transformations in Spark are evaluated after you perform an action. This is called Lazy Evaluation.

#### **Q35. What is MLib in Spark?**

Answer: MLlib is a distributed machine learning framework built on top of Spark.

#### **Q36. What is GraphX in Spark?**

Answer: GraphX is a distributed graph-processing framework built top of Spark. It provides different APIs for expressing graph computation.

#### **Q37. 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.

#### **Q38. 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.

#### **Q39. Write some function of Spark Executor?**

Answer: To run a task that makeup application and to return the result to Driver. It Provides in-memory storage for RDDs cached by user.

#### **Q40. Which are the Programming languages supported by Spark?**

Answer: Java, Python, Scala, SQL and R.

#### **Q41. What is DAG in Spark?**

Answer: DAG is a set of Vertices and Edges, where vertices represent the RDDs and the edges represent the Operation to be applied on RDD.

#### **Q42. What is Caching in Spark Streaming?**

Answer: Caching Streaming is storing streaming RDD in memory. It is a mechanism to speed up applications that access the same RDD multiple times.

#### **Q43. Which file systems are supported by Spark?**

Answer: HDFS, Local File system & Amazon S3.

#### **Q44. What is RDD?**

Answer: Resilient Distribution Datasets (RDD) is a fault-tolerant collection of partitioned data that run in parallel. RDD is immutable and distributed in nature.

#### **Q45. Write some input sources for Spark Streaming.**

Answer: TCP Sockets, Stream of files, Apache Kafka, Apache Flume, Kinesis etc.

#### **Q46. Can we use Hive on Spark?**

Answer: Yes, by creating Hive context

#### **Q47. What is a pipe () operation in Spark?**

Answer: Spark is using Scala, Java, and Python to write the program. However, if one wants to pipe (inject) the data

which is written in other languages Spark provides a general mechanism for that in the form of pipe() method.

#### **Q48. What are the data sources available in Spark SQL?**

Answer: Parquet, Avro, JSON and Hive tables

#### **Q49. What is a partition in Spark?**

Answer: A partition in spark is a logical division of data stored on a Node in the cluster. Partitions are basic units of parallelism in Apache Spark.

#### **Q50.What are the types of Partitioning in Apache Spark?**

Answer: The types of portioning in Apache Spark are as follows:

* Hash Partitioning
* Range Partitioning

#### **Q51. What are the types of Cluster managers in Spark?**

Answer: Standalone,Yarn & Mesos.

[*Enroll Now !*](https://www.gangboard.com/big-data-training/apache-spark-training?utm_source=GB_blog)

[*Share on Facebook*](http://www.facebook.com/sharer/sharer.php?u=https%3A%2F%2Fwww.gangboard.com%2Fblog%2Fapache-spark-interview-questions-and-answers%2F)[*Share on Twitter*](http://twitter.com/intent/tweet/?text=Apache+Spark+Interview+Questions+and+Answers&url=https%3A%2F%2Fwww.gangboard.com%2Fblog%2Fapache-spark-interview-questions-and-answers%2F)[*Share on Linkedin*](http://www.linkedin.com/shareArticle?mini=true&url=https%3A%2F%2Fwww.gangboard.com%2Fblog%2Fapache-spark-interview-questions-and-answers%2F&title=Apache+Spark+Interview+Questions+and+Answers)[*Share on Pinterest*](http://pinterest.com/pin/create/button/?url=https%3A%2F%2Fwww.gangboard.com%2Fblog%2Fapache-spark-interview-questions-and-answers%2F&media=https%3A%2F%2Fwww.gangboard.com%2Fblog%2Fwp-content%2Fuploads%2F2019%2F04%2FApache-Spark-Interview-Questions-and-Answers-300x149.jpg&description=Apache+Spark+Interview+Questions+and+Answers)

## NO COMMENTS

### LEAVE A REPLY

Top of Form

*Your email address will not be published Required fields are marked \**

Comment

You may use these HTML tags and attributes: <a href="" title=""> <abbr title=""> <acronym title=""> <b> <blockquote cite=""> <cite> <code> <del datetime=""> <em> <i> <q cite=""> <s> <strike> <stro

Bottom of Form

99 Apache Spark Interview Questions for Professionals

Page

**7**

 of

**82**

**1.**

**What is the difference between Spark and Hadoop?**

**Features SPARK HadoopInspiration**

Hadoop Map-Reduce and Scala programming language,developed by UC-Berkeley's AMPLab in 2009, usegeneralized computation instead of MapReduceQuery optimization - RDBMSReal time processing capabilityGoogle, papers in 2004outlining MapReduce No optimizationBatch Processing

**Speed**

100X in-memory and10X on DiskHeavy Disk read I/Ointensive

**Ease of Use**

Easily to write application using Java, Scala, Python, R(Functional programming style)Interactive Shell available with Scala and PythonHigh level simple map-reduce OperationsJava Imperative programming style No shellcomplex map-reduceoperations

**Iterative Workflow**

Great at Iterative workloads (Machine learning ..etc) Not ideal for iterative work

**Tools**

Well integrated tools (Spark SQL, Streaming, Mlib andGraphX) to develop complex analytical applicationLoosely coupled large set oftools, but matured

**Deployment**

Hadoop YARN, Mesos, Amazon-EC2

Usually use Oozie andAzkaban to create workflow

**Data Source**

HDFS(Hadoop), HBase, Cassandra, MongoDB,Amazon-S3, RDBMS, file, socket, Twitter

RDBMS (using sqoop),streaming using FLUME

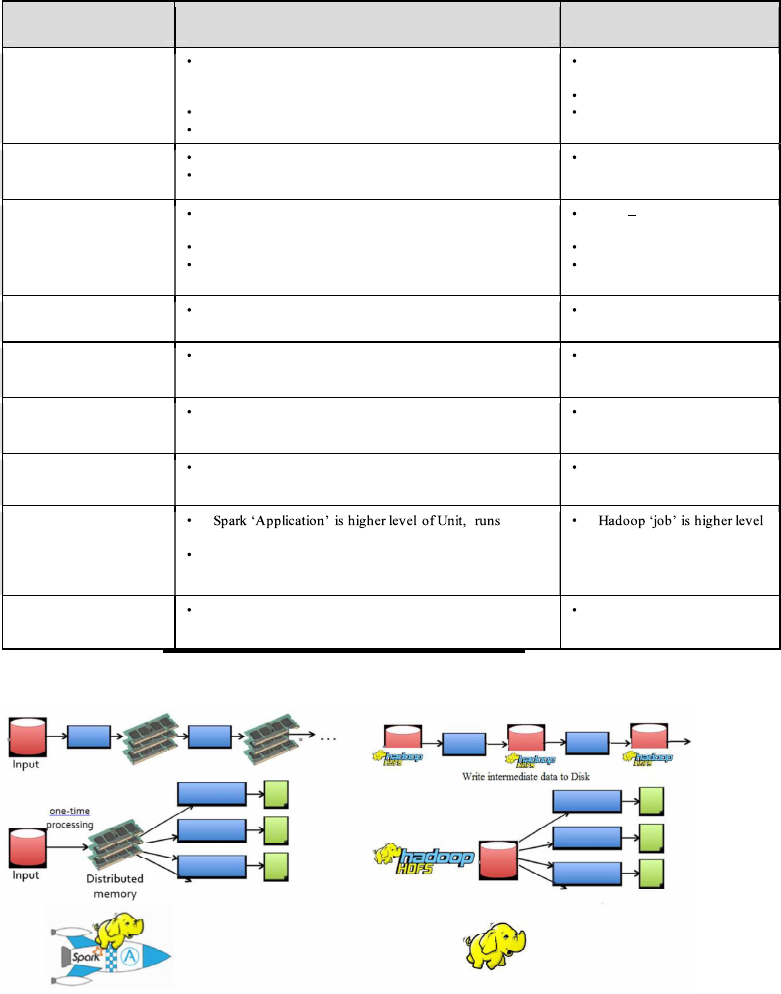
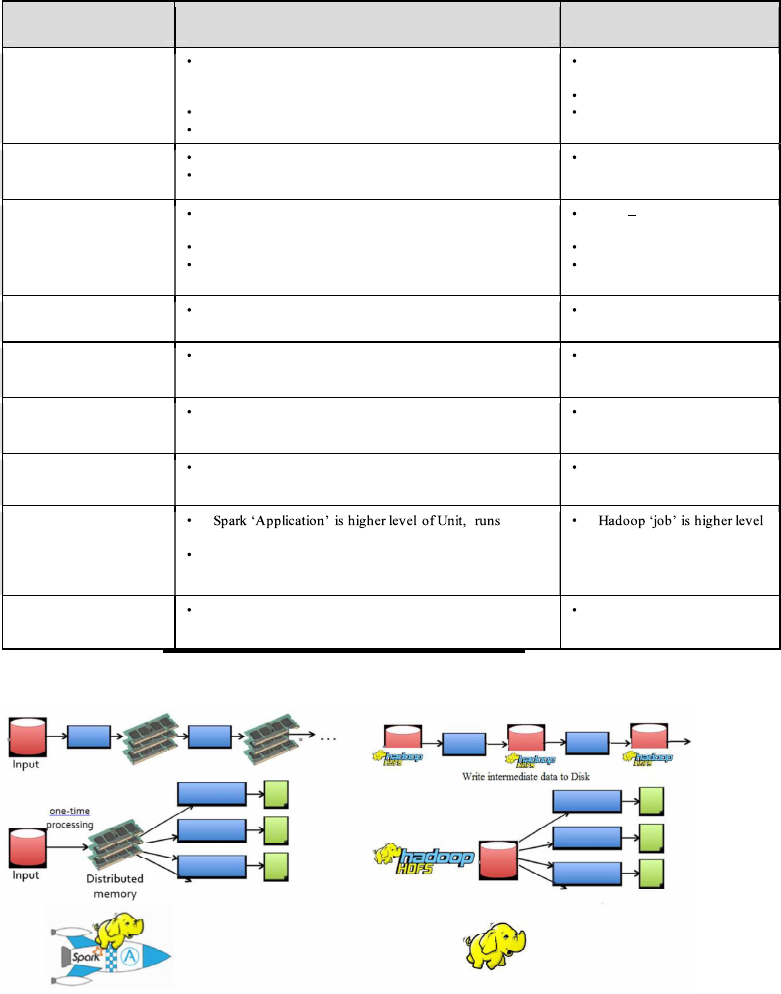
**Applications**

multiple jobs in sequence or parallel

Application processes are called executors, run onclusters(workers)unit; Processes data withMapReduce and writes datato storage

**Executors**

Executors can run multiple tasks in a single processor Each MapReduce runs in itsown processor



Apache Spark Interview Questions for Professionals

Page

8

 of

82

**Shuffle**

above the configured threshold (200 by default)Always sorts its partitionduring shuffle

**Shared Variable**

Broadcast variables: Read-only(look-up) variable, shipsonly once to workerAccumulators: Workers add values and driver reads thedata, and fault tolerant

Hadoop counter hasadditional (system ) metric

**Persisting/CachingRDD**

Cached RDDs can be used & reused across theoperation, thus increasing the processing speed None

**Lazy Evaluation**

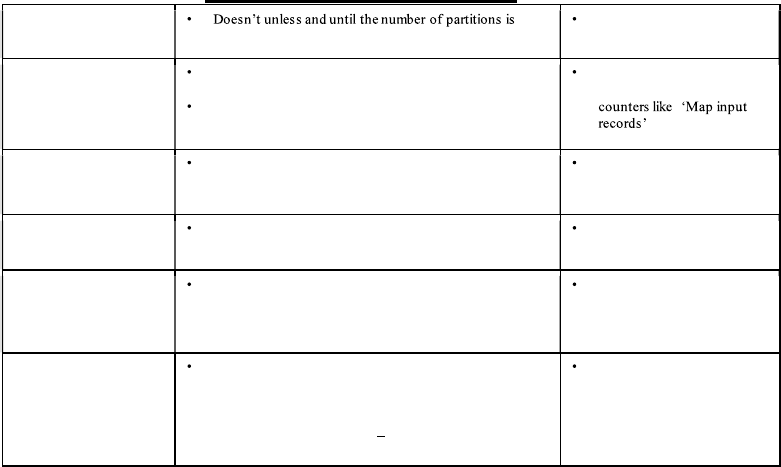
Transformation functions and execution plan bundledtogether and execute only with RDD action function None

**Memory Managementand Compression**

Memory is conserved, because of the compact format.Speed is improved by custom code-generation.Custom compression can beachieved using AVRO,Kyro; no memorymanagement

**Optimizer and QueryPlanning**

Optimizer is a Rule Executor for logical plans. It uses acollection of logical plan optimizations. Generatesencoders via runtime code-generation. The generatedcode can operate directly on the Tungsten compactformat. Query is optimized logical and physical plan(inspired by RDBMS query planning and optimization) None



Apache Spark Interview Questions for Professionals

Page

9

 of

82

**2.**

**What are the differences between functional and imperative languages, and why is functional programming important?**

Following features of Scala makes it uniquely suitable for Spark.

**Immutability**

- Immutable means that you can't change your variables; you mark them as final in Java, or use the valkeyword in Scala

**Higher order functions**

 - These are functions that take other functions as parameters, or whose result is a function. Here is afunction apply which takes another function f and a value v and applies function f to v: example - def apply(f: Int => String,v: Int) = f(v)

**Lazy loading**

- Lazy val is executed when it is accessed the first time else no execution.

**Pattern matching**

 - Scala has a built-in general pattern matching mechanism. It allows to match on any sort of data with afirst-match policy

**Currying**

- If we turn this into a function object that we can assign or pass around, the signature of that function looks likethis: val sizeConstraintFn: IntPairPred => Int => Email => Boolean = sizeConstraint \_ Such a chain of one-parameterfunctions is called a curried function

**Partial application**

 - When applying the function, you do not pass in arguments for all of the parameters defined by thefunction, but only for some of them, leaving the remaining ones blank. What you get back is a new function whose parameterlist only contains those parameters from the original function that were left blank.

**Monads**

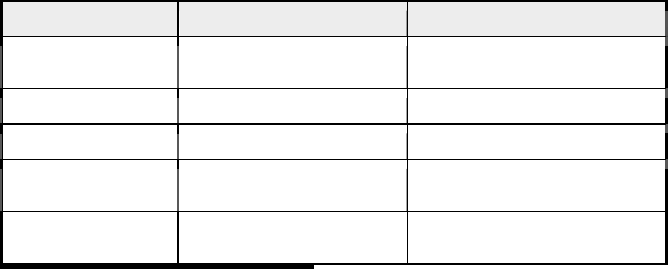
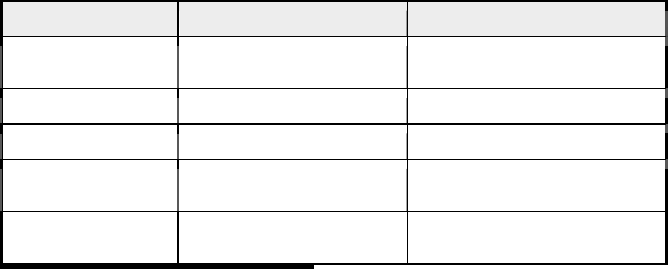
 - Most Scala collections are monadic, and operating on them using map and flatMap operations, or using for-comprehensions is referred to as monadic-style.

**Programming approach difference**

:

**Characteristic Imperative approach Functional approach**

Programmer focus How to perform tasks (algorithms)and how to track changes in state.What information is desired and whattransformations are required.State changes Important.Non-existent.Order of execution Important.Low importance.Primary flow control Loops, conditionals, and function(method) calls.Function calls, including recursion.Primary manipulation unit Instances of structures or classes. Functions as first-class objects and datacollections.



Apache Spark Interview Questions for Professionals

Page

10

 of

82

**3.**

**What is a resilient distributed dataset (RDD), explain showing diagrams?Resilient distributed dataset (RDD)**

is a read-only and fault-tolerant collection of objects partitioned across a cluster ofcomputers that can be operated on in parallelwith one another. There are two ways to create RDDs: parallelizing an existingcollection in your driver program, or referencing a dataset in an external storage system, such as a shared filesystem, HDFS,HBase, S3, Cassandra or RDBMS.

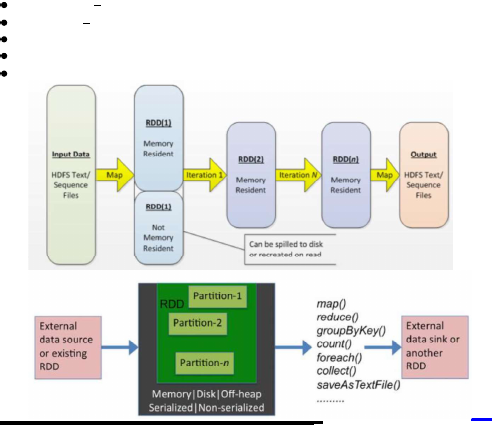
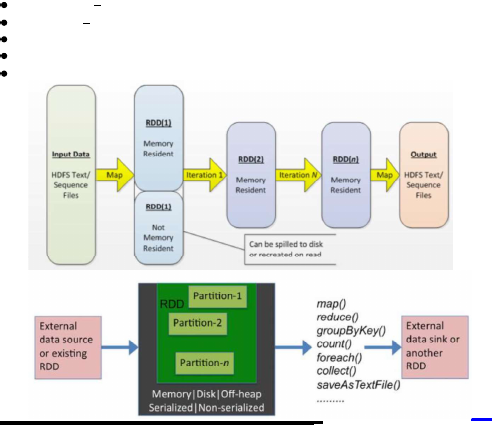
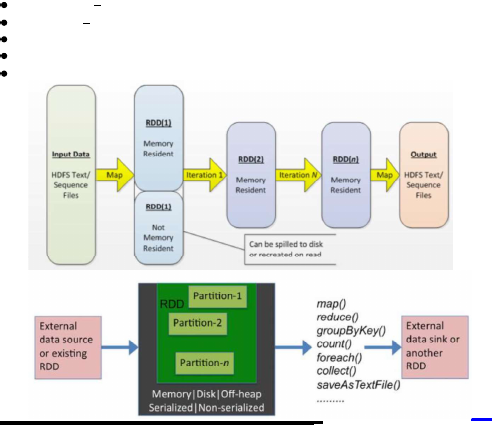
RDDs (Resilient Distributed Datasets) are basic abstractionsin Apache Spark that represent the data coming into the systemin object format. RDDs are used for in-memory computations on large clusters, in a fault tolerant manner. RDDs are read-only, portioned, collection of records, which areImmutable RDDs cannot be altered.

Resilient If a node holding the partition fails the other node takes the data.

Lazy evaluated

Cacheable

Type inferredRef



Apache Spark Interview Questions for Professionals

Page

11

 of

8

## **op 11 Apache Spark Interview Questions are**

#### **1. What is Apache Spark?**

**Answer:** Apache Spark is a big-data cluster computing framework used for real-time processing. It is open-source and has a huge open source community to back it up. With Apache Spark, anyone can program entire cluster with the help of API interface. The framework also supports fault-tolerance and implicit data parallelism.

Apache Spark went open source in 2010 and now more than 1000 companies are actively contributing to its core. Its success can easily be gauged by the fact that it is used by big players such as eBay, Amazon, and Microsoft!

#### **2.  What are the key features of Spark?**

**Answer:** There are many key features of Spark. Some of the features are as below:

* It works seamlessly with Hadoop and HDFS.
* It comes with independent Scala interpreter.
* Spark’s speed is one of its key features. It beats Hadoop in large-scale data processing.
* Supports Resilient Distributed Datasets which can easily be used in a cluster.
* It also supports Real-Time Computation, Machine Learning, and Multiple Format.

#### **3. What are the main components of Spark ecosystem?**

**Answer:**The main components of Spark ecosystem are as follows:

* Spark core: The base engine that offers large-scale distributed and parallel data processing.
* Spark SQL: Relational functionality that works in tandem with Spark’s API.
* GraphX: offers all graph related features.
* MLib: Machine learning library.
* Spark Streaming: Offers real-time streaming data processing.

Preparing for Big Data Interview? Here are the Top 25 [Big Data Interview Questions and Answers](https://www.whizlabs.com/blog/big-data-interview-questions/) that will help you crack the Big Data Interview!

#### **4. What is the function of SparkCore?**

**Answer:**SparkCore is the heart of Apache Spark. It handles important functions such as fault-tolerance, memory management, job scheduling, storage system interaction, and so on. The engine offers large-scale distributed and parallel processing. The SparkCore functionality can easily be accessed with the help of available Scala, Java, And Python APIs.

#### 5. **Which language is supported by Apache Spark?**

**Answer:**Apache Spark is written using Scala. It means that Spark supports Scala and Java by default. Other than Scala and Java, R and Python can also be used efficiently. The choice for the programming language depends on the project requirement.

Here’s [5 Best Apache Spark Certification To Boost Your Career](https://www.whizlabs.com/blog/5-best-apache-spark-certification/)

#### 6. **Can you explain RDD?**

**Answer:** The fundamental data structure for Spark is Resilient Distributed Datasets (RDD). Data in the network is stored in this format. It has many features including immutability, resilience, and Parallel/Partitioned. As data is stored in the cluster, the above features help the data to be run parallelly and on multiple nodes.

RDD offers two operations: actions and transformations, and can be used to store any data. If there is a key associated with a value, it is known as Pair RDD.

#### **7. Compare Hadoop MapReduce and Spark.**

**Answer:** There is a lot of difference between these two frameworks even if they are used for same purpose, i.e., big data processing. However, there is some key difference between them.

* **Speed:**Spark is created with speed in mind. Hence, Apache Spark can go up to 100x than MapReduce on memory.
* **Implementation:**MapReduce is comparatively harder to work and maintain than Spark.
* **Real-time analysis:**Spark can do real-time analysis whereas MapReduce cannot.
* **Security:**Spark in comparison is less secure than MapReduce as it only supports secret password authentication. Hadoop, in addition to secret password authentication, supports ACLs.

Find yourself prepared for Hadoop interview? Try out how many of these [Hadoop Interview Questions](https://www.whizlabs.com/blog/top-50-hadoop-interview-questions/) you can answer!

#### **8. Define Actions.**

**Answer:** Actions is one of the operations offered by RDD. When “Action” operation takes place, the local machine receives the data from RDD. Action’s execution is delayed to ensure that operations run smoothly. Two good example of Actions are – ***reduce( )*** and ***take( )***.

* reduce( ): it is executed until only one value is left. The function takes two arguments.
* take( ): take( ) method act as a data carrier from RDD to the local node.

#### **9. Define Partitions.**

**Answer:** Partitions are used to split the data into logical units. It helps the Spark to run computations parallelly on different nodes. Partitions are very similar to “split” in MapReduce. The partition created is located on different machine. Spark is designed to run data from different machine efficiently, and that’s why it is so popular among the big data community.

[](https://www.whizlabs.com/spark-developer-certification/)

#### **10. How is machine learning implemented in Spark?**

**Answer:**With Spark, anyone can use the MLib to do machine learning. With MLib, common algorithms can easily be implemented. MLib library also supports common machine learning cases such as dimensional reduction, regression filtering, clustering, etc.

#### **11. What is GraphX?**

**Answer:**GraphX in Spark allows its user to create interactive graphs from scratch. With GraphX, building and transforming graphs is easy. It is a crucial part of the Spark ecosystem as it gives the necessary tools to visualize data.

These top 11 Apache Spark interview questions will surely help you in the Apache Spark interview. You can also choose to do a[*Spark Developer Certification*](https://www.whizlabs.com/spark-developer-certification/)(HDPCD) to make your CV strong.

## **op 11 Apache Spark Interview Questions are**

#### **1. What is Apache Spark?**

**Answer:** Apache Spark is a big-data cluster computing framework used for real-time processing. It is open-source and has a huge open source community to back it up. With Apache Spark, anyone can program entire cluster with the help of API interface. The framework also supports fault-tolerance and implicit data parallelism.

Apache Spark went open source in 2010 and now more than 1000 companies are actively contributing to its core. Its success can easily be gauged by the fact that it is used by big players such as eBay, Amazon, and Microsoft!

#### **2.  What are the key features of Spark?**

**Answer:** There are many key features of Spark. Some of the features are as below:

* It works seamlessly with Hadoop and HDFS.
* It comes with independent Scala interpreter.
* Spark’s speed is one of its key features. It beats Hadoop in large-scale data processing.
* Supports Resilient Distributed Datasets which can easily be used in a cluster.
* It also supports Real-Time Computation, Machine Learning, and Multiple Format.

#### **3. What are the main components of Spark ecosystem?**

**Answer:**The main components of Spark ecosystem are as follows:

* Spark core: The base engine that offers large-scale distributed and parallel data processing.
* Spark SQL: Relational functionality that works in tandem with Spark’s API.
* GraphX: offers all graph related features.
* MLib: Machine learning library.
* Spark Streaming: Offers real-time streaming data processing.

Preparing for Big Data Interview? Here are the Top 25 [Big Data Interview Questions and Answers](https://www.whizlabs.com/blog/big-data-interview-questions/) that will help you crack the Big Data Interview!

#### **4. What is the function of SparkCore?**

**Answer:**SparkCore is the heart of Apache Spark. It handles important functions such as fault-tolerance, memory management, job scheduling, storage system interaction, and so on. The engine offers large-scale distributed and parallel processing. The SparkCore functionality can easily be accessed with the help of available Scala, Java, And Python APIs.

#### 5. **Which language is supported by Apache Spark?**

**Answer:**Apache Spark is written using Scala. It means that Spark supports Scala and Java by default. Other than Scala and Java, R and Python can also be used efficiently. The choice for the programming language depends on the project requirement.

Here’s [5 Best Apache Spark Certification To Boost Your Career](https://www.whizlabs.com/blog/5-best-apache-spark-certification/)

#### 6. **Can you explain RDD?**

**Answer:** The fundamental data structure for Spark is Resilient Distributed Datasets (RDD). Data in the network is stored in this format. It has many features including immutability, resilience, and Parallel/Partitioned. As data is stored in the cluster, the above features help the data to be run parallelly and on multiple nodes.

RDD offers two operations: actions and transformations, and can be used to store any data. If there is a key associated with a value, it is known as Pair RDD.

#### **7. Compare Hadoop MapReduce and Spark.**

**Answer:** There is a lot of difference between these two frameworks even if they are used for same purpose, i.e., big data processing. However, there is some key difference between them.

* **Speed:**Spark is created with speed in mind. Hence, Apache Spark can go up to 100x than MapReduce on memory.
* **Implementation:**MapReduce is comparatively harder to work and maintain than Spark.
* **Real-time analysis:**Spark can do real-time analysis whereas MapReduce cannot.
* **Security:**Spark in comparison is less secure than MapReduce as it only supports secret password authentication. Hadoop, in addition to secret password authentication, supports ACLs.

Find yourself prepared for Hadoop interview? Try out how many of these [Hadoop Interview Questions](https://www.whizlabs.com/blog/top-50-hadoop-interview-questions/) you can answer!

#### **8. Define Actions.**

**Answer:** Actions is one of the operations offered by RDD. When “Action” operation takes place, the local machine receives the data from RDD. Action’s execution is delayed to ensure that operations run smoothly. Two good example of Actions are – ***reduce( )*** and ***take( )***.

* reduce( ): it is executed until only one value is left. The function takes two arguments.
* take( ): take( ) method act as a data carrier from RDD to the local node.

#### **9. Define Partitions.**

**Answer:** Partitions are used to split the data into logical units. It helps the Spark to run computations parallelly on different nodes. Partitions are very similar to “split” in MapReduce. The partition created is located on different machine. Spark is designed to run data from different machine efficiently, and that’s why it is so popular among the big data community.

[](https://www.whizlabs.com/spark-developer-certification/)

#### **10. How is machine learning implemented in Spark?**

**Answer:**With Spark, anyone can use the MLib to do machine learning. With MLib, common algorithms can easily be implemented. MLib library also supports common machine learning cases such as dimensional reduction, regression filtering, clustering, etc.

#### **11. What is GraphX?**

**Answer:**GraphX in Spark allows its user to create interactive graphs from scratch. With GraphX, building and transforming graphs is easy. It is a crucial part of the Spark ecosystem as it gives the necessary tools to visualize data.

These top 11 Apache Spark interview questions will surely help you in the Apache Spark interview. You can also choose to do a[*Spark Developer Certification*](https://www.whizlabs.com/spark-developer-certification/)(HDPCD) to make your CV strong.

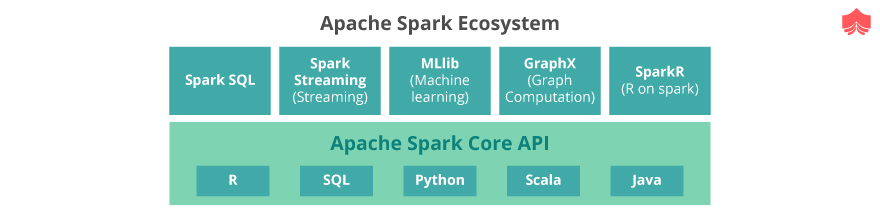
#### [What do the features of Spark provide, which is not available to the Map-Reduce?](https://www.knowledgehut.com/interview-questions/spark#collapse-beginner-941)

Spark API provides various key features, which is very useful for spark real time processing, most of the features has a well support library along with real time processing capability.

Below are the key features providing by spark framework:

* Spark Core
* Spark Streaming.
* Spark SQL
* GrasphX
* MLib

Spark core is a heart of spark framework and well support capability for functional programing practice for the language like Java, Scala, Python, however most of the new release come for JVM language first and then later on introduced for python.



#### [How spark core fit into the picture to solving the big data use case?](https://www.knowledgehut.com/interview-questions/spark#collapse-beginner-942)

Reduce, collection, aggregation API, stream, parallel stream, optional which can easily handle to all the use case where we are dealing volume of data handling.

**Bullet points are as follows:**

* Spark core is the distributed execution engine for large-scala parallel and distributed data processing.
* Spark core provide a real time processing for large data set.
* Handle memory management and fault recovery.
* Scheduling, distributing and monitoring jobs on a cluster.
* Spark core comes with map, flatmap, reduce, reducebykey, groupbykey which handling the key value pair-based data processing for large data set.
* Spark core also support aggregation operation.
* Spark core support Java, Scala and Python.
* Code snippet: val counts = textReader.flatMap(line => line.split(",")).map(word => (word, 1)).reduceByKey(\_ + \_).

Apparently spark use for data processing framework, however we can also use to perform the data analysis and data science.

#### [What are the benefits of using Spark streaming for real time processing instead of other framework and tools?](https://www.knowledgehut.com/interview-questions/spark#collapse-beginner-943)

Spark Streaming supports micro-batch-oriented stream processing engine, Spark has a capability to allow the data can be ingested from many sources like Kafka, Flume, Kinesis, or TCP sockets,

and can be processed using complex algorithms expressed with high-level functions like map, reduce, join and window.

**Below are the other key benefits which Spark streaming support.**

* Spark streaming is one of features of Spark used to process the real time data efficiently.
* Spark Streaming implement using Kafka and Zookeeper messaging API, which is again a fault tolerant messaging container can create a messaging cluster.
* Provide high-throughput and fault-tolerant stream processing
* Provide DStream data structure which is a basically a stream of RDD to process the real-time data.
* Spark Streaming fits for scenario where interaction require Kafka to  Database or Kafka to Data science model type of context.

Spark work on batches which receives an input data stream and divided into the micro batches, which is further processed by the spark engine to generate the final stream of result in the batches.  
  
**Below diagram clearly illustrated the workflow of Spark streaming. **

#### c[What Spark-SQL does, how it’s benefits to programmer to interact with database? And Syntax of creating SQL Context?](https://www.knowledgehut.com/interview-questions/spark#collapse-beginner-944)

Spark SQL provides programmatic abstraction in the form of data frame and data set which can work the principal of distributed SQL query engine. Spark SQL simplify the interaction to the large amount of data through the dataframe and dataset.

* Spark-SQL provide a relation processing along with spark functional programming.
* Support querying data using SQL and HIVE query language.
* Support Datasource API, Dataframe API, Interpreter & Optimizer, SQL Service.
* Spark-SQL also providing the new API called Dataset which has capability of both Dataframe and core.
* Spark-SQL I much optimize to perform SQL query-based operation on flat file, json.
* Spark SQL support variety of language like: Java, Scala, Python and R.
* Code Snippet: val sqlContext = new SQLContext( sc: SparkContext)
* Dataframe can be create using below approach:
* Structured data files:
* Tables in Hive:
* External databases:
* Using existing RDD:

Spark SQL plays a vital role on optimization technique using Catalyst optimizer, Spark SQL also support UDF, built in function and aggregates function.

#### [What are the key component of spark which internally spark require to execute the job?](https://www.knowledgehut.com/interview-questions/spark#collapse-beginner-945)

* Spark follows a master/slave architecture.
  + Master Daemon: (Master Drive process)
  + Worker Daemon: (Slave process)
* Spark cluster has a single Master
* No. of Slave worked as a commodity server.
* When we submit the spark job it triggers the spark driver.
* Getting the current status of spark application
* Canceling the job
* Canceling the Stage
* Running job synchronously
* Running job asynchronously
* Accessing persistent RDD
* Un-persisting RDD
* Programmable dynamic allocation

#### [Why we need the master driver in spark?](https://www.knowledgehut.com/interview-questions/spark#collapse-beginner-946)

Master driver is central point and the entry point of the Spark Shell which is supporting this language (Scala, Python, and R). Below is the sequential process, which driver follows to execute the spark job.

* Driver runs the main () function of the application which create the spark context.
* Driver program that runs on the master node of the spark cluster schedules the job execution.
* Translates the RDD’s into the execution graph and splits the graph into multiple stages.
* Driver stores the metadata about all the Resilient Distributed Databases and their partitions.
* Driver program converts a user application into smaller execution units known as tasks which is also as a stage.
* Tasks are then executed by the executors i.e. the worker processes which run individual tasks.

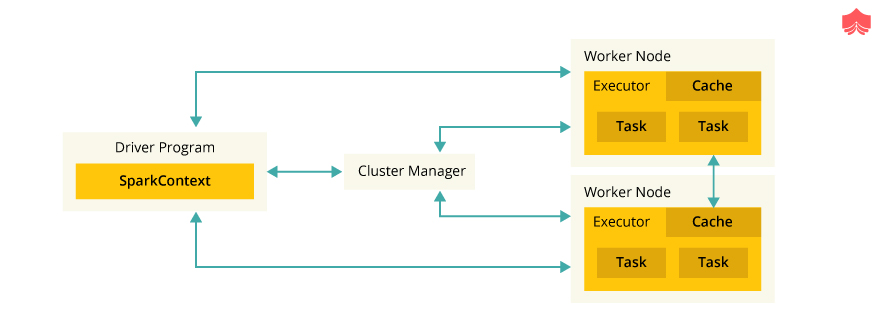
The complete process can track by cluster manager user interface. Driver exposes the information about the running spark application through a Web UI at port 4040

#### [7.What is an executor in spark and how its support to perform the operation on volume of data?](https://www.knowledgehut.com/interview-questions/spark#collapse-beginner-947)

Executors are worker nodes' processes in charge of running individual tasks when Spark job get submitted. They are launched at the beginning of a Spark application and typically run for the entire lifetime of an application. Once they have run the task, they send the results to the driver. They also provide in-memory storage for RDDs that are cached by user programs through Block Manager.

**Below are the key points on executors:**

* Every spark application has its own executor process.
* Executor performs all the data processing.
* Reads from and Writes data to external sources.
* Executor stores the computation results data in-memory, cache or on hard disk drives.



Executor also work as a distributed agent responsible for the execution of tasks. When the job getting launched, spark trigger the executor, which act as a worker node which responsible for running individual task, which is assigned by spark driver.

#### [.What happens when a Spark Job is submitted?](https://www.knowledgehut.com/interview-questions/spark#collapse-beginner-948)

**Below is the step which spark job follows once job get submitted:**

* A standalone application starts and instantiates a SparkContext instance and it is only then when you can call the application a driver.
* The driver program asks for resources to the cluster manager to launch executors.
* The cluster manager launches executors.
* The driver process runs through the user application.
* Depending on the actions and transformations over RDDs task are sent to executors.
* Executors run the tasks and save the results.
* If any worker crashes, its tasks will be sent to different executors to be processed again.
* Driver implicitly converts the code containing transformations and actions into a logical
* directed acyclic graph (DAG).

Spark automatically deals with failed or slow machines by re-executing failed or slow tasks. For example, if the node running a partition of a map () operation crashes, Spark will rerun it on another node; and even if the node does not crash but is simply much slower than other nodes, Spark can preemptively launch a “speculative” copy of the task on another node and take its result if that finishes.

#### [What is RDD? How does spark RDD works? What are the various ways to create the RDD?](https://www.knowledgehut.com/interview-questions/spark#collapse-beginner-949)

Resilient distributed dataset (RDD) is a core of Spark framework, which is a fault-tolerant collection of elements that can be operated on in parallel.

**Below are the key points on RDD:**

* RDD is an immutable distributed collection of objects.
* RDD works on in-memory computation paradigm.
* RDD is divided into logical partitions, which computed in different worker nodes.
* Stores the state of memory as an object across the jobs and the object is sharable between those jobs.
* Data sharing using RDD faster than the I/O and disk, because its use the in – memory computation.
* The working of RDD is:
  + Resilient handling a fault-tolerant with the help of RDD spark able to recover or recompute the missing or damaged partitions due to node failures.
  + Distributed mechanism handling data residing on multiple nodes in a cluster.
  + Dataset is a collection of partitioned data with primitive values or values of values, e.g. tuples or other objects

**We can create the RDD using below approach:**

**By Referring a dataset:**

* Val byTextFile = sc.textFile(hdfs:// or s3:// )

**By Parallelizing a dataset:**

* Val byParalizeOperation = sc.paralize( Seq(DataFrame or Dataset), numSlices: Integer)

**By converting dataframe to rdd.**

* Val byDF = df.filter().toRDD

RDDs predominately support two types of operations: transformations, which create a new dataset from an existing one, and actions, which return a value to the driver program after running a computation on the dataset.

#### [What is the other notable feature of RDD and ways to create the RDD?](https://www.knowledgehut.com/interview-questions/spark#collapse-beginner-950)

* **In-Memory:**Ability to perform operation in the primary memory not in the disk
* **Immutable or Read-Only:**Emphasize in creating the immutable data set.
* **Lazy evaluated:**Spark computing the record when the action is going to perform, not in transformation level.
* **Cacheable:**We can cache the record, for faster processing.
* **Parallel:**Spark has an ability to parallelize the operation on data, saved in     RDD.
* **Partitioned of records:**Spark has ability to partition the record, by default its support 128 MB of partition.
* **Parallelizing:** an existing collection in your driver program.
* **Referencing a dataset** in an external storage system, such as a shared file system, HDFS, HBase

## Spark Interview Questions

**What is Shark?**

**Most of the data users know only SQL and are not good at programming. Shark is a tool, developed for people who are from a database background – to access Scala MLib capabilities through Hive like SQL interface. Shark tool helps data users run Hive on Spark – offering compatibility with Hive metastore, queries and data.**

**List some use cases where Spark outperforms Hadoop in processing.**

1. **Sensor Data Processing –Apache Spark’s ‘In-memory computing’ works best here, as data is retrieved and combined from different sources.**
2. **Spark is preferred over Hadoop for real time querying of data**
3. **Stream Processing – For processing logs and detecting frauds in live streams for alerts, Apache Spark is the best solution.**

**What is a Sparse Vector?**

**A sparse vector has two parallel arrays –one for indices and the other for values. These vectors are used for storing non-zero entries to save space.**

**What is RDD?**

**RDDs (Resilient Distributed Datasets) are basic abstraction in Apache Spark that represent the data coming into the system in object format. RDDs are used for in-memory computations on large clusters, in a fault tolerant manner. RDDs are read-only portioned, collection of records, that are –**

**Immutable – RDDs cannot be altered.**

**Resilient – If a node holding the partition fails the other node takes the data.**

**Explain about transformations and actions in the context of RDDs.**

**Transformations are functions executed on demand, to produce a new RDD. All transformations are followed by actions. Some examples of transformations include map, filter and reduceByKey.**

**What are the languages supported by Apache Spark for developing big data applications?**

**Scala, Java, Python, R and Clojure**

**Can you use Spark to access and analyse data stored in Cassandra databases?**

**Yes, it is possible if you use Spark Cassandra Connector.**

**Is it possible to run Apache Spark on Apache Mesos?**

**Yes, Apache Spark can be run on the hardware clusters managed by Mesos.**

**Is it possible to run Spark and Mesos along with Hadoop?**

**Yes, it is possible to run Spark and Mesos with Hadoop by launching each of these as a separate service on the machines. Mesos acts as a unified scheduler that assigns tasks to either Spark or Hadoop.**

**What is lineage graph?**

**The RDDs in Spark, depend on one or more other RDDs. The representation of dependencies in between RDDs is known as the lineage graph. Lineage graph information is used to compute each RDD on demand, so that whenever a part of persistent RDD is lost, the data that is lost can be recovered using the lineage graph information.**

**How can you trigger automatic clean-ups in Spark to handle accumulated metadata?**

**You can trigger the clean-ups by setting the parameter ‘spark.cleaner.ttl’ or by dividing the long running jobs into different batches and writing the intermediary results to the disk.**

**What are the benefits of using Spark with Apache Mesos?**

**It renders scalable partitioning among various Spark instances and dynamic partitioning between Spark and other big data frameworks.**

**What is Catalyst framework?**

**Catalyst framework is a new optimization framework present in Spark SQL. It allows Spark to automatically transform SQL queries by adding new optimizations to build a faster processing system.**

## Spark Interview Questions for 3 years Experience

**What are the key features of Apache Spark?**

**Here is a list of the key features of Apache Spark:**

**Hadoop Integration**

**Lazy Evaluation**

**Machine Learning**

**Multiple Format Support**

**Polyglot**

**Real-Time Computation**

**Speed**

**What are the components of Spark Ecosystem?**

**Here are the core components of the Spark ecosystem:**

**Spark Core: a Base motor for vast scale parallel and appropriated information preparing**

**Spark Streaming: Used for preparing ongoing gushing information**

**Spark SQL: Integrates social preparing with Spark’s practical programming API**

**GraphX: Graphs and diagram parallel calculation**

**MLlib: Performs machine learning in Apache Spark**

**– Spark interview questions and answers for 3 years experience**

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

**Apache Spark supports the accompanying four languages: Scala, Java, Python and R. Among these languages, Scala and Python have intuitive shells for Spark. The Scala shell can be gotten to through ./canister/start shell and the Python shell through ./receptacle/pyspark. Scala is the most utilized among them since Spark is composed in Scala and it is the most prominently utilized for Spark.**

**What are the multiple data sources supported by Spark SQL?**

**Apache Spark SQL is a popular ecosystem or interfaces to work with structured or semi-structured data. The multiple data sources supported by Spark SQL includethe text file, JSON file, Parquet file etc.**

**– Spark interview questions and answers for 3 years experience**

**How is machine learning implemented in Spark?**

**MLlib is a versatile machine learning library given by Spark. It goes for making machine adapting simple and versatile with normal learning calculations and utilize cases like grouping, relapse separating, dimensional decrease, and alike.**

**What is YARN?**

**Like Hadoop, YARN is one of the key highlights in Spark, giving a focal and asset administration stage to convey versatile activities over the group. YARN is an appropriated compartment supervisor, as Mesos for instance, though Spark is an information preparing instrument. Spark can keep running on YARN, a similar way Hadoop Map Reduce can keep running on YARN. Running Spark on YARN requires a parallel dissemination of Spark as based on YARN support.**

**Does Spark SQL help in big data analytics through external tools too?**

**Yes, Spark SQL helps in big data analytics through external tools too. Let us see how it is done actually.**

**It access data using SQL statements in both ways either data is stored inside the Spark program or data needs to access through external tools that are connected to Spark SQL through database connectors like JDBC or ODBC.**

**It provides rich integration between a database and regular coding with RDDs and SQL tables. It is also able to expose custom SQL functions as needed.**

**– Spark interview questions and answers for 3 years experience**

**How is Spark SQL superior from others – HQL and SQL?**

**Spark SQL is advance database component able to support multiple database tools without changing their syntax. This is the way how Spark SQL accommodates both HQL and SQL superiorly.**

**Do real-time data processing is possible with Spark SQL?**

**Real-time data processing is not possible directly but obviously, we can make it happen by registering existing RDD as a SQL table and trigger the SQL queries on priority.**

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

**RDD is an abbreviation for Resilient Distribution Datasets. An RDD is a blame tolerant accumulation of operational components that keep running in parallel. The divided information in RDD is permanent and distributed in nature. There are fundamentally two sorts of RDD:**

**Parallelized Collections: Here, the current RDDs run parallel with each other.**

**Hadoop Datasets:**

**They perform works on each document record in HDFS or other stockpiling frameworks.**

**RDDs are essential parts of information that are put away in the memory circulated crosswise over numerous hubs. RDDs are sluggishly assessed in Spark. This apathetic assessment is the thing that adds to Spark’s speed.**

## spark sql interview questions for experienced

**Q. What is Catalyst framework?**

**Answer: Catalyst framework is a new optimization framework present in Spark SQL. It allows Spark to automatically transform SQL queries by adding new optimizations to build a faster processing system.**

**Q. Why is BlinkDB used?**

**Answer: BlinkDB is a query engine for executing interactive SQL queries on huge volumes of data and renders query results marked with meaningful error bars. BlinkDB helps users balance ‘query accuracy’ with response time.**

**– Spark interview questions and answers for 3 years experience**

**Q. How can you compare Hadoop and Spark in terms of ease of use?**

**Answer: Hadoop MapReduce requires programming in Java which is difficult, though Pig and Hive make it considerably easier. Learning Pig and Hive syntax takes time. Spark has interactive APIs for different languages like Java, Python or Scala and also includes Shark i.e. Spark SQL for SQL lovers – making it comparatively easier to use than Hadoop.**

**Q. What are the various data sources available in SparkSQL?**

**Answer: Parquet file**

**JSON Datasets**

**Hive tables**

**SparkSQL is a Spark component that supports querying data either via SQL or via the Hive Query Language. It originated as the Apache Hive port to run on top of Spark (in place of MapReduce) and is now integrated with the Spark stack. In addition to providing support for various data sources, it makes it possible to weave SQL queries with code transformations which results in a very powerful tool. Below is an example of a Hive compatible query:**

**– Spark interview questions and answers for 3 years experience**

**Q. Name a few commonly used Spark Ecosystems.**

**Answer: Spark SQL (Shark)**

**Spark Streaming**

**GraphX**

**MLlib**

**SparkR**

**Q. What is “Spark SQL”?**

**Answer: Spark SQL is a Spark interface to work with structured as well as semi-structured data. It has the capability to load data from multiple structured sources like “text files”, JSON files, Parquet files, among others. Spark SQL provides a special type of RDD called SchemaRDD. These are row objects, where each object represents a record.**

**Q. Can we do real-time processing using Spark SQL?**

**Answer: Not directly but we can register an existing RDD as a SQL table and trigger SQL queries on top of that.**

**Q. Explain about the major libraries that constitute the Spark Ecosystem**

**Answer: Spark MLib- Machine learning library in Spark for commonly used learning algorithms like clustering, regression, classification, etc.**

**Spark Streaming – This library is used to process real time streaming data.**

**Spark GraphX – Spark API for graph parallel computations with basic operators like join Vertices, subgraph, aggregate Messages, etc.**

**Spark SQL – Helps execute SQL like queries on Spark data using standard visualization or BI tools.**

**– Spark interview questions and answers for 3 years experience**

**Q. What is Spark SQL?**

**Answer: SQL Spark, better known as Shark is a novel module introduced in Spark to work with structured data and perform structured data processing. Through this module, Spark executes relational SQL queries on the data. The core of the component supports an altogether different RDD called SchemaRDD, composed of rows objects and schema objects defining data type of each column in the row. It is similar to a table in relational database.**

**Q. What is a Parquet file?**

**Answer: Parquet is a columnar format file supported by many other data processing systems. Spark SQL performs both read and write operations with Parquet file and consider it be one of the best big data analytics format so far.**

**Q. List the functions of Spark SQL.**

**Answer: Spark SQL is capable of:**

**Loading data from a variety of structured sources**

**Querying data using SQL statements, both inside a Spark program and from external tools that connect to Spark SQL through standard database connectors (JDBC/ODBC). For instance, using business intelligence tools like Tableau**

**Providing rich integration between SQL and regular Python/Java/Scala code, including the ability to join RDDs and SQL tables, expose custom functions in SQL, and more**

## Apache spark scala interview questions: Shyam Mallesh

**Q. Who designed Scala? Which is the latest version?**

**At the time of writing, Scala 2.12.6 is the latest version. The interviewer may ask you this to find out whether you keep yourself updated. Martin Odersky, a German computer scientist, began designing it in 2001 at EPFL, Switzerland.**

**Q. What are the advantages of Scala?**

**Among various other benefits of the language, here are a few:**

**It is highly scalable**

**It is highly testable**

**It is highly maintainable and productive**

**It facilitates concurrent programming**

**It is both object-oriented and functional**

**It has no boilerplate code**

**Singleton objects are a cleaner solution than static**

**Scala arrays use regular generics**

**Scala has native tuples and concise code**

**Q. What is ofDim in Scala?**

**ofDim() is a method in Scala that lets us create multidimensional arrays. Since these let us store data in more than one dimension, we can store data like in a matrix. Let’s take an example.**

**scala> import Array.ofDim**

**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**

**k: Int = 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)).**

**Q. What do you have to say about exception propagation in Scala?**

**When a function experiences an exception, it looks for a handler to deal with it. When it fails to find one, it searches for one in the caller method. Failing there, it looks for yet another in the next caller in the chain. Whenever it does find a handler, it makes it catch the exception. This is exception propagation.**

**Q. What is a BitSet?**

**A bitset is a set of non-negative integers depicted as arrays. These arrays are variable in size and packed into 64-bit words. The largest number in a bitset determines its memory footprint. Let’s take an example.**

**scala> import scala.collection.immutable.\_**

**import scala.collection.immutable.\_**

**scala> var nums=BitSet(7,2,4,3,1)**

**nums: scala.collection.immutable.BitSet = BitSet(1, 2, 3, 4, 7)**

**scala> nums+=9  //Adding an element**

**scala> nums**

**res14: scala.collection.immutable.BitSet = BitSet(1, 2, 3, 4, 7, 9)**

**scala> nums-=4  //Deleting an element**

**scala> nums**

**res16: scala.collection.immutable.BitSet = BitSet(1, 2, 3, 7, 9)**

**scala> nums-=0  //Deleting an element that doesn’t exist**

**scala> nums**

**res18: scala.collection.immutable.BitSet = BitSet(1, 2, 3, 7, 9)**

**Q. What is a vector in Scala?**

**A vector is a general-purpose data structure that is immutable. We can use it when we want to hold a huge number of elements and want random access to them. This data structure extends the trait IndexedSeq and the abstract class AbstractSeq.**

**scala> import scala.collection.immutable.\_**

**import scala.collection.immutable.\_**

**scala> var v1=Vector.empty**

**v1: scala.collection.immutable.Vector[Nothing] = Vector()**

**scala> var v2=Vector(7,2,4,3,1)**

**v2: scala.collection.immutable.Vector[Int] = Vector(7, 2, 4, 3, 1)**

**scala> var v3:Vector[Int]=Vector(8,2,6,5,9)**

**v3: scala.collection.immutable.Vector[Int] = Vector(8, 2, 6, 5, 9)**

**scala> v3=v3 :+7  //Adding a new element**

**v3: scala.collection.immutable.Vector[Int] = Vector(8, 2, 6, 5, 9, 7)**

**scala> v2++v3  //Merging two vectors**

**res19: scala.collection.immutable.Vector[Int] = Vector(7, 2, 4, 3, 1, 8, 2, 6, 5, 9, 7)**

**scala> v3.reverse  //Reversing a vector**

**res20: scala.collection.immutable.Vector[Int] = Vector(7, 9, 5, 6, 2, 8)**

**scala> v3.sorted  //Sorting a vector**

**res21: scala.collection.immutable.Vector[Int] = Vector(2, 5, 6, 7, 8, 9)**

**In results 20 and 21, we do not assign the expression to any variable, so not that this doesn’t change the original vectors.**

## Cts spark interview questions

**Q. What do you understand by Lazy Evaluation?**

**Spark is intellectual in the manner in which it operates on data. When you tell Spark to operate on a given dataset, it heeds the instructions and makes a note of it, so that it does not forget – but it does nothing, unless asked for the final result. When a transformation like map () is called on a RDD-the operation is not performed immediately. Transformations in Spark are not evaluated till you perform an action. This helps optimize the overall data processing workflow.**

**Q. Define a worker node?**

**A node that can run the Spark application code in a cluster can be called as a worker node. A worker node can have more than one worker which is configured by setting the SPARK\_ WORKER\_INSTANCES property in the spark-env.sh file. Only one worker is started if the SPARK\_ WORKER\_INSTANCES property is not defined.**

**– Spark interview questions and answers for 3 years experience**

**Q. What do you understand by SchemaRDD?**

**An RDD that consists of row objects (wrappers around basic string or integer arrays) with schema information about the type of data in each column.**

**Q. How Spark uses Akka?**

**Spark uses Akka basically for scheduling. All the workers request for a task to master after registering. The master just assigns the task. Here Spark uses Akka for messaging between the workers and masters.**

**– Spark interview questions and answers for 3 years experience**

**Q. How can you achieve high availability in Apache Spark?**

**Implementing single node recovery with local file system Using StandBy Masters with Apache ZooKeeper.**

**Q. Hadoop uses replication to achieve fault tolerance. How is this achieved in Apache Spark?**

**Data storage model in Apache Spark is based on RDDs. RDDs help achieve fault tolerance through lineage. RDD always has the information on how to build from other datasets. If any partition of a RDD is lost due to failure, lineage helps build only that particular lost partition.**

**Q. How Spark handles monitoring and logging in Standalone mode?**

**Spark has a web based user interface for monitoring the cluster in standalone mode that shows the cluster and job statistics. The log output for each job is written to the work directory of the slave nodes.**

**– Spark interview questions and answers for 3 years experience**

**Q. Does Apache Spark provide checkpointing?**

**Lineage graphs are always useful to recover RDDs from a failure but this is generally time consuming if the RDDs have long lineage chains. Spark has an API for check pointing i.e. a REPLICATE flag to persist. However, the decision on which data to checkpoint – is decided by the user. Checkpoints are useful when the lineage graphs are long and have wide dependencies.**

**– Spark interview questions and answers for 3 years experience**

**Q. How can you launch Spark jobs inside Hadoop MapReduce?**

**Using SIMR (Spark in MapReduce) users can run any spark job inside MapReduce without requiring any admin rights.**

## Times Spark Questions

**Question. What Is Spark Core?**

**Answer : It has all the basic functionalities of Spark, like – memory management, fault recovery, interacting with storage systems, scheduling tasks, etc.**

**Question. How Can You Remove The Elements With A Key Present In Any Other Rdd?**

**Answer : Use the subtractByKey () function**

**Question. What Is The Difference Between Persist() And Cache()**

**Answer : persist () allows the user to specify the storage level whereas cache () uses the default storage level.**

**Question. What Are The Various Levels Of Persistence In Apache Spark?**

**Answer : Apache Spark automatically persists the intermediary data from various shuffle operations, however it is often suggested that users call persist () method on the RDD in case they plan to reuse it. Spark has various persistence levels to store the RDDs on disk or in memory or as a combination of both with different replication levels.**

**The various storage/persistence levels in Spark are –**

**MEMORY\_ONLY**

**MEMORY\_ONLY\_SER**

**MEMORY\_AND\_DISK**

**MEMORY\_AND\_DISK\_SER, DISK\_ONLY**

**OFF\_HEAP**

**Question. How Spark Handles Monitoring And Logging In Standalone Mode?**

**Answer : Spark has a web based user interface for monitoring the cluster in standalone mode that shows the cluster and job statistics. The log output for each job is written to the work directory of the slave nodes.**

**Question. How Can You Launch Spark Jobs Inside Hadoop Mapreduce?**

**Answer : Using SIMR (Spark in MapReduce) users can run any spark job inside MapReduce without requiring any admin rights.**

**Question. How Spark Uses Akka?**

**Answer : Spark uses Akka basically for scheduling. All the workers request for a task to master after registering. The master just assigns the task. Here Spark uses Akka for messaging between the workers and masters.**

**Question. How Can You Achieve High Availability In Apache Spark?**

**Answer : Implementing single node recovery with local file system**

**Using StandBy Masters with Apache ZooKeeper.**

**Question. Hadoop Uses Replication To Achieve Fault Tolerance. How Is This Achieved In Apache Spark?**

**Answer : Data storage model in Apache Spark is based on RDDs. RDDs help achieve fault tolerance through lineage. RDD always has the information on how to build from other datasets. If any partition of a RDD is lost due to failure, lineage helps build only that particular lost partition.**

Category: [questions](https://questionsgems.com/category/questions/)[By Shyami Goyal](https://questionsgems.com/author/ssghtc/)[December 21, 2018](https://questionsgems.com/2018/12/21/)[Leave a comment](https://questionsgems.com/spark-interview-questions/#respond)

#### Author: Shyami Goyal

## **Q1.  What is RDD?**

**Ans.** RDD (Resilient Distribution Datasets) is a fault-tolerant collection of operational elements that run parallel. The partitioned data in RDD is immutable and distributed.

## **Q2.  Name the different types of RDD**

**Ans.** There are primarily two types of RDD – parallelized collection and Hadoop datasets.

## **Q3.  What are the methods of creating RDDs in Spark?**

**Ans.** There are two methods –

1. By parallelizing a collection in your Driver program.
2. By loading an external dataset from external storage like HDFS, HBase, shared file system.

## **Q4.  What is a Sparse Vector?**

**Ans.** A sparse vector has two parallel arrays –one for indices and the other for values.

## **Q5.  Mention some of the areas where Spark outperforms Hadoop in processing**

**Ans.** Sensor data processing, real-time querying of data, and stream processing.

## **Q6.  What are the languages supported by Apache Spark and which is the most popular one?**

**Ans.** There are four languages supported by Apache Spark – Scala, Java, Python, and R. Scala is the most popular one.

## **Q7.  What is Yarn?**

**Ans.** Yarn is one of the key features in Spark, providing a central and resource management platform to deliver scalable operations across the cluster.

### Also Read>>[Top Hadoop Interview Questions & Answers](https://learning.naukri.com/articles/hadoop-interview-questions-answers/)

## **Q8.  Do you need to install Spark on all nodes of Yarn cluster? Why?**

**Ans.** No, because Spark runs on top of Yarn.

## **Q9.  Is it possible to run Apache Spark on Apache Mesos?**

**Ans.** Yes.

**SPARK TECHNICAL INTERVIEW**

**How to create array in Spark scala?**

var sarray = Array(1,23,14,23,100,111);

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | sarray.size; | |  | // printing the distinct elements | |  | sarray.distinct; | |  | sarray.map(\_/2); | |  | sarray.filter(\_>3); | |  | sarray(0) | |  | sarray(1); | |  | sarray(0) =1; | |  | // we cannot update the array sarray(11) = 2; | |  | sarray(5) = 111; | |  | sarray.indexOf(3); | |  | (1 until 9).toArray | |  | ( 'a' to 'h').toArray; | |  | Array.range(1,99); | |  | Array.tabulate(4)(n => n \* n) |   How to create arraybuffer in Spark scala?   |  | | --- | | var empty\_arraybuffer = new ArrayBuffer[Int]()// | |  | empty\_arraybuffer += 3; | |  | empty\_arraybuffer += 4; |   What is syntax of the forloop in scala? | |
| package Loops |
|  |  |
|  | object ForLoop { |
|  | def main(args:Array[String]) |
|  | { |
|  | var a=0 |
|  | var b=0 |
|  | var c=0 |
|  | for(a<-1 to 10; b<-4 until 11;if a > b) |
|  | { |
|  | println(a + "|" + b); |
|  |  |
|  | } |
|  | } |
|  |  |
|  | } |

**Syntax of List , Map and Set**

**List:-**

|  |
| --- |
| var l = List(1 , 3 , 4 , 5, 6 ,7); |
|  |  |
|  | for (index <- 0 until l.length) |
|  | { |
|  | if (l(index)%2 != 0) |
|  | println("odd element : " + l(index)); |
|  | else |
|  | println("even element: " + l(index)); |
|  | } |
|  |  |
|  | for ( a <- l) |
|  | println("\t" + a) |
|  | // for populating the another list |
|  |  |
|  | val duplicate\_list = List.tabulate(9)(n=>n); |
|  | println("duplicate elements" + duplicate\_list); |

**Set :-**

|  |
| --- |
| package Collection |
|  | import Set.\_ |
|  |  |
|  | object Set\_Initial { |
|  | def main(args:Array[String]) { |
|  | var s = Set(11,23,13,14,15); |
|  | println("Values are:" + s); |
|  |  |
|  |  |
|  | } |
|  | } |

**Map**

|  |
| --- |
| 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 sTuple = (1,2,3,5,6) |
|  |  |
|  | // var aTuple = scala.io.Source.fromFile("C:\\Users\\jabin\\Desktop\\sample\_test.txt"); |
|  |  |
|  | println(sTuple.\_2); |
|  | println(sTuple.\_3); |
|  |  |

**How will you handle error in spark?**

|  |
| --- |
| 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"); |
|  |  |
|  | 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") |
|  |  |
|  | } |
|  |  |
|  | } |
|  | } |

**Can we perform RDD operations on the Collections?**

**Yes , but for that we need toconvert the Collection into the sequence .**

**What is difference between map and FlatMap**

**What is case classes and why we use case classes**

**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) }; |
|  |  |

**Which are print operations in Spark ?**

**Dump**

**Println**

**Take**

**First**

**Top**

**Which operation we use for combining and subtracting items?**

**Union and Intersection**

**Explain the ways of reading files in spark scala?**

**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\_handling\_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" + |

**What is registerTemp Table?**

**‘**