Chettinad College of Engineering & Technology
Approved by AICTE-New Delhi and Affiliated to Anna University-Chennai

Academic Year 2023 - 2024

Question Bank

Year/Semester: III/ V	**Department :CSE**	**Unit : I/II/III/IV/V**
Date:28/07/2023	**Subject Code/Title :CCS334- Big Data Analytics**	**Section : Part A/B**
	Faculty Name :Dr.T.Gobinath	

UNIT I
PART A

1. What you mean by bigdata?
   Big Data refers to extremely large datasets that cannot be easily managed, processed, or analyzed using traditional data processing tools. It involves high volume, velocity, and variety of data.

2. Name the types of bigdata.
   The types of Big Data include Structured Data, Unstructured Data, and Semi-Structured Data.

3. List out the characteristics of bigdata.
   The main characteristics of Big Data are Volume, Velocity, Variety, Veracity, and Value.

4. What is the advantage of bigdata?
   Big Data allows organizations to gain deep insights, improve decision-making, enhance operational efficiency, and create competitive advantages by analyzing vast amounts of data.

5. What you meant by unstructured data?
   Unstructured Data is data that lacks a predefined format or organization, making it difficult to process and analyze using traditional databases. Examples include text files, images, and videos.

6. Difference between structured and unstructured data.
   Structured Data is organized and easily searchable within databases (e.g., spreadsheets), while Unstructured Data lacks a defined structure, making it harder to analyze (e.g., emails, social media posts).

7. Define web analytics.
   Web Analytics is the measurement, collection, analysis, and reporting of web data to

understand and optimize web usage.

8. What are the data collection metrics in web analytics.
   Types include On-Site Web Analytics and Off-Site Web Analytics.

9. List out the types of web analytics.
   Benefits include improved user experience, enhanced marketing strategies, better content optimization, and increased conversion rates.

10. Name some benefits of web analytics.
    Applications include fraud detection, customer segmentation, predictive analytics, and real-time data processing.

11. List out some applications of bigdata.
    Examples include Hadoop, Spark, NoSQL databases, and Apache Flink.

12. Name some bigdata technologies.
    Examples include Hadoop, Spark, NoSQL databases, and Apache Flink.

13. What is Hadoop?
    Hadoop is an open-source framework for storing and processing large datasets in a distributed computing environment.

14. Explain the core components of Hadoop.
    The core components are HDFS (Hadoop Distributed File System) and YARN (Yet Another Resource Negotiator).

15. Explain the features of Hadoop.
    Features include scalability, fault tolerance, flexibility, and cost-effectiveness.

16. What you mean by HDFS?
    HDFS (Hadoop Distributed File System) is the storage system of Hadoop that stores large datasets across multiple machines in a distributed manner.

17. What you mean by YARN?
    YARN (Yet Another Resource Negotiator) is a resource management layer of Hadoop that allocates system resources and manages workloads.

18. List out the benefits of Hadoop.
    Benefits include scalability, cost-effectiveness, flexibility, and the ability to process vast amounts of data.

19. Define Open-source technology.
    Open-source technology refers to software with source code that is freely available for modification, enhancement, and redistribution.

20. How cloud technology impacts the bigdata?

Cloud technology provides scalable resources and storage, making it easier to manage, process, and analyze Big Data efficiently.

21. What do you mean by cloud computing?

Cloud computing is the delivery of computing services (e.g., storage, processing) over the internet, allowing on-demand access to resources.

22. List out the features of cloud computing.

Features include on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service.

23. What are the issues in using cloud services.

Issues include data security, privacy concerns, potential downtime, and dependency on the service provider.

24. Define mobile business intelligence.

Mobile Business Intelligence refers to the ability to access BI data, such as reports and dashboards, on mobile devices.

25. Justify the need for Business intelligence.

Business Intelligence is needed to improve decision-making, identify market trends, and optimize business processes by analyzing data.

26. What are the advantages of business intelligence.

Advantages include better decision-making, increased operational efficiency, and competitive advantage.

27. Define crowd sourcing.

Crowdsourcing is the practice of obtaining input, ideas, or services from a large group of people, typically via the Internet.

28. What are the types of crowd sourcing?

Types include crowdfunding, crowd voting, microtasking, and crowd contest.

29. What is inter firewall analytics?

Inter-Firewall Analytics involves analyzing data traffic and logs between firewalls to identify security threats and enhance network protection.

30. What do you mean by trans firewall analytics?

Trans Firewall Analytics involves the analysis of data and traffic that pass through a firewall from different networks to detect threats and anomalies across network boundaries.

31. Difference between inter and trans firewall analytics.

Inter Firewall Analytics focuses on analyzing traffic between firewalls within the same network to enhance internal security. In contrast, Trans Firewall Analytics examines traffic that crosses different network boundaries, focusing on external threats and ensuring security across interconnected networks.

PART B

1. What is Bigdata? Describe the main features of bigdata in detail. List out the characteristics of bigdata.
2. Discuss about web analytics in detail.
3. What do you mean by open-source technology? Discuss in brief. What is the relationship between cloud and bigdata?
4. Explain about Mobile BI and its types in detail.
5. What is Crowdsourcing? Discuss about it.
6. Compare and contrast the inter and trans firewall in detail.
7. Can you think of any bigdata application that impacts your daily life? How?
8. Discuss two bigdata applications in detail.

UNIT II

PART A

1. What is NOSQL?

NoSQL is a category of database management systems designed to handle large volumes of unstructured or semi-structured data, offering flexible schema design and horizontal scaling.

2. What are the features of NOSQL?

Key features include schema flexibility, horizontal scalability, high availability, and the ability to handle diverse data types (e.g., documents, graphs, key-value pairs).

3. What are the types of NOSQL databases?

The main types are Document Stores, Key-Value Stores, Column-Family Stores, and Graph Databases.

4. Difference between SQL and NOSQL?

SQL databases are relational, use structured query language, and have a fixed schema, whereas NoSQL databases are non-relational, offer flexible schemas, and are designed for distributed data storage.

5. List the advantages of NOSQL.

Advantages include scalability, flexibility in data modeling, fast performance for large datasets, and the ability to handle unstructured data.

6. List the disadvantages of NOSQL.

Disadvantages include eventual consistency, limited support for complex queries, lack of standardization, and potential difficulty in managing and maintaining the database.

7. Difference between Cassandra and MySQL.

Cassandra is a NoSQL database that provides high scalability and availability, while MySQL is an RDBMS known for ACID compliance and structured data storage with a fixed schema.

8. Difference between Cassandra and RDBMS.

Cassandra is a distributed NoSQL database designed for high availability and horizontal scalability, while RDBMS are relational databases that emphasize ACID transactions and a structured schema.

9. What is Apache Cassandra?

Apache Cassandra is an open-source, distributed NoSQL database designed to handle large amounts of data across many commodity servers with no single point of failure.

10. What is CQLSH? And why is it used?

CQLSH (Cassandra Query Language Shell) is an interactive command-line interface used to execute Cassandra Query Language (CQL) commands, manage data, and interact with a Cassandra cluster.

11. What are clusters in Cassandra?

A cluster in Cassandra is a collection of connected nodes that work together to store and manage data, ensuring high availability and fault tolerance.

12. What is a keyspace in Cassandra?

A Keyspace in Cassandra is a namespace that defines data replication and organization within the cluster, similar to a database in RDBMS.

13. What is a Column Family?

A Column Family in Cassandra is a structure that contains rows, each identified by a unique key, similar to a table in relational databases but with a flexible schema.

14. What is a Row in Cassandra? And what are the different elements of it?

A Row in Cassandra is a collection of related data identified by a unique key, consisting of columns (key-value pairs) and their associated timestamps.

15. Name some features of Apache Cassandra.

Features include decentralized architecture, horizontal scalability, high availability, fault tolerance, and tunable consistency.

16. List some of the components of Cassandra.

Components include Nodes, Clusters, Keyspaces, Column Families, Rows, Columns, Commit Logs, and SSTables.

17. Write some advantages of Cassandra.

Advantages include high scalability, fault tolerance, decentralized design, and the ability to handle large volumes of distributed data.

18. Define commit log.

A Commit Log in Cassandra is a write-ahead log that captures all write operations to ensure data durability and recovery in case of a failure.

19. Define composite key.

A Composite Key in Cassandra is a combination of multiple columns used to uniquely identify a row within a column family.

20. Define SSTable.

SSTable (Sorted String Table) is an immutable data file where Cassandra stores data after it is flushed from the MemTable, organized in a sorted format.

21. What is memtable?

A MemTable is an in-memory data structure where Cassandra stores write operations temporarily before flushing them to disk as SSTables.

22. How the SSTable is different from other relational tables?

SSTables are immutable and stored in a sorted, compressed format, whereas relational tables can be mutable and require a fixed schema.

23. What is data replication in Cassandra?

Data replication in Cassandra involves copying data across multiple nodes to ensure high availability, fault tolerance, and data durability across the cluster.

PART B:

1. Brief about Cassendra architecture with neat diagram
2. Explain about consistence
3. How schema less database improve the performance of database?
4. Brief about CAP Theorem
5. Explain Aggregated Data Models in NoSql
6. How Distribution model works in Nosql method

UNIT III
PART A

1. What do you mean by MapReduce?
   MapReduce is a programming model used for processing large datasets in parallel across a distributed cluster of computers. It breaks down tasks into a series of "Map" and "Reduce" functions.

2. What are the advantages of using using MapReduce with Hadoop.
   Advantages include parallel processing, fault tolerance, scalability, and the ability to handle large datasets efficiently across distributed systems.

3. Explain what is distributed cache in MapReduce framework?
   Distributed Cache is a mechanism in Hadoop that allows large files, which are needed by MapReduce tasks, to be cached and made available to all nodes in the cluster to improve processing efficiency.

4. Explain what is the function of MapReduce partitioner?
   The Partitioner in MapReduce determines how the output of the Mapper is distributed across the Reducers. It ensures that all related data ends up in the same Reducer based on the partitioning logic.

5. Mention what are the main configuration parameters that user need to specify to run MapReduce job?
   Key parameters include `input path`, `output path`, `Mapper class`, `Reducer class`, `input format`, `output format`, and `number of Reducers`.

6. List out the key concepts related to MapReduce.
   Key concepts include Mapper, Reducer, Combiner, Partitioner, InputFormat, OutputFormat, JobTracker, TaskTracker, and Distributed Cache.
7. Explain Map() and Reduce() function.
   The Map() function processes input data and generates key-value pairs. The Reduce() function then aggregates these key-value pairs based on keys to produce the final output.

8. Difference between job Tracker and Task Tracker.
   JobTracker is the master node that manages and schedules the execution of MapReduce jobs across the cluster. TaskTracker is a worker node that performs the tasks assigned by the JobTracker.

9. What are the stages in MapReduce workflow?
   The stages include Input Splitting, Mapping, Shuffling and Sorting, and Reducing.

10. What do you mean by MRUnit?
    MRUnit is a testing framework specifically for Hadoop MapReduce programs, allowing developers to write unit tests for MapReduce jobs to ensure correctness.

11. List down the steps to write unit test with MRUnit.
    Steps include setting up the MRUnit test environment, creating test input data, defining the Mapper and Reducer, running the MapReduce job, and validating the output against expected results.

12. What are the key components of MapReduce development that ensures the correctness of MapReduce program?
    Key components include unit testing (using MRUnit), data validation, correct partitioning and sorting logic, and comprehensive logging for debugging.

13. What is Test data?
    Test Data refers to the sample input data used in testing to validate that the MapReduce program processes data correctly.

14. What is the use of local test?
    Local Test allows developers to test MapReduce jobs on a local machine without deploying them on a full Hadoop cluster, enabling quicker and more efficient debugging.

15. Write the benefits of using test data and local test in MR development.
    Benefits include early detection of errors, reduced development time, easier debugging, and ensuring that the MapReduce program behaves correctly before deploying it on a full-scale cluster.

16. List the step involved in using test data and local test in MapReduce.
    Steps include:
    - Prepare sample test data.
    - Set up a local Hadoop environment or use MRUnit.
    - Run the MapReduce job on the test data.
    - Validate the output against expected results.
    - Debug and iterate as necessary.

17. What do you mean by Shuffle and Sort?
    Shuffle and Sort is a phase in the MapReduce process where the output from the Map phase is grouped by key and sorted before being passed to the Reducer. This ensures that all data associated with a particular key is available in a single Reducer task.

18. Infer your knowledge about YARN.
    YARN (Yet Another Resource Negotiator) is a resource management layer in Hadoop that allocates system resources and manages distributed applications. It separates the

resource management and job scheduling functions from the MapReduce engine.

19. List out the failures in MapReduce and YARN.

Failures include:

- Task failure (Mapper or Reducer tasks failing due to code errors or hardware issues).
- Node failure (failure of a node hosting TaskTracker or NodeManager).
- Job failure (the entire MapReduce job fails if critical tasks fail repeatedly).
- Resource contention or starvation in YARN.

20. List out the features of YARN.

Features include:

- Scalability
- Resource management
- Multi-tenancy support
- Fault tolerance
- Compatibility with other processing models (e.g., Spark, Hive)

21. Name the types of Hadoop Scheduler.

Types of Hadoop Schedulers include:

- FIFO Scheduler (First In, First Out)
- Capacity Scheduler
- Fair Scheduler

22. List out the types of MapReduce.

Types include:

- Traditional MapReduce (MR1)
- YARN-based MapReduce (MR2)

23. List out the stages in Task Execution.

Stages include:

- Task initialization
- Task assignment
- Data localization
- Execution (Map or Reduce)
- Shuffle and Sort (for Reduce tasks)
- Commit and cleanup

24. What do you mean by Job Scheduling?

Job Scheduling refers to the process of managing the execution order and allocation of resources to various MapReduce jobs in a Hadoop cluster, ensuring optimal utilization of resources and adherence to policies like priority, fairness, or capacity.

PART B

1.Explain in detail about MapReduce Workflows.

2. What is MRUnit? Explain in detail with a program.

3. Discuss YARN in detail.
4. Explain job scheduling and its types in detail.
5. Discuss about the input and output formats of MapReduce in detail.
6. Explain about MapReduce and its types.
7. Explain the anatomy of a MapReduce job run.

UNIT IV
PART A

1. What are the two main files stored in HDFS?
   The two main files stored in HDFS are:
   - **Blocks:** The actual data files split into blocks.
   - **Metadata (FSImage and EditLogs):** Stores information about the data, including the directory structure and file properties.

2. What are the steps in analyzing data with Hadoop?
   Steps include:
   - Data Ingestion (loading data into HDFS)
   - Data Processing (using MapReduce, Spark, etc.)
   - Data Storage (storing processed data in HDFS or another database)
   - Data Analysis and Visualization (using tools like Hive, Pig, or external tools).

3. What is data visualization?
   Data Visualization refers to the graphical representation of data to help understand trends, patterns, and outliers in data by using visual elements like charts, graphs, and maps.

4. What do you mean by sealing out?
   Scaling Out refers to adding more nodes to a distributed system, such as a Hadoop cluster, to increase processing power and storage capacity.

5. What are the bottlenecks that can be solved with sealing?
   Bottlenecks that can be solved include:
   - Insufficient storage capacity
   - High latency in data processing
   - Limited computational resources
   - Performance degradation due to high workload

6. Infer your knowledge on streaming.
   Streaming involves processing data in real-time as it is generated. In the context of Hadoop, tools like Apache Kafka, Apache Flink, and Apache Storm are often used for handling streaming data.

7. What is Hadoop pipes?
   Hadoop Pipes is a C++ API for Hadoop MapReduce that allows developers to write MapReduce jobs in C++ rather than Java, providing more flexibility in programming

language choice.

8. Name some key concepts of Hadoop.
   Key concepts include:
   - HDFS (Hadoop Distributed File System)
   - MapReduce
   - YARN (Yet Another Resource Negotiator)
   - Data Nodes and Name Nodes
   - JobTracker and TaskTracker

9. What is Java Interface?
   A Java Interface is an abstract type in Java that defines a set of methods that a class must implement. It allows for defining methods without their implementations, promoting code modularity and reusability.

10. Write down the basic dataflow of Hadoop system.
    The basic dataflow involves:
    - Data is split into blocks and stored in HDFS.
    - MapReduce jobs process the data in parallel across nodes.
    - Intermediate results are shuffled, sorted, and reduced.
    - Final results are stored back in HDFS or another storage system.

11. What do you mean by Data Integrity?
    Data Integrity refers to the accuracy, consistency, and reliability of data during its lifecycle. In Hadoop, checksums and replication help ensure data integrity.
12. Infer your knowledge on compression.
    Compression reduces the size of data to save storage space and improve processing speed. In Hadoop, data compression can be applied at various stages, including during storage and data transfer, using algorithms like GZIP, LZO, and Snappy.

13. What is serialization?
    Serialization is the process of converting an object or data structure into a byte stream for storage, transmission, or distribution, and later reconstructing it (deserialization) to its original form.

14. What do you mean by AVRO?
    AVRO is a row-based storage format for Hadoop that supports serialization and deserialization, making it easier to exchange data between programs regardless of the language in which they are written.

15. What are the commonly used file-based data structures Hadoop?
    Commonly used file-based data structures include:
    - AVRO
    - Parquet

- ORC (Optimized Row Columnar)
- Sequence Files

16. List out the features of Cassandra ?

Features include:

- Decentralized architecture
- Horizontal scalability
- High availability
- Tunable consistency
- Flexible schema design
- Fault tolerance

17. Infer your knowledge on Hadoop Integration.

Hadoop Integration refers to connecting Hadoop with other tools and platforms (e.g., Apache Spark, HBase, Hive, and traditional RDBMS) to enhance data processing capabilities, facilitate data migration, and improve analytics efficiency.

PART B

1. Explain how data is analyzed using Hadoop.
2. Explain Hadoop Streaming in detail with a neat sketch.
3. Discuss the design of HDFS in detail.
4. Explain briefly about Hadoop Distributed File System concepts in detail.
5. Discuss about Java Interface in Hadoop.
6. Briefly explain the Hadoop Input and Output operations.
7. What do you mean by AVRO? Explain in detail.
8. Explain the types of file-based data structures in detail.
9. Discuss Cassandra architecture in detail.
10. Explain about Hadoop integration in detail.

UNIT V

PART A

1. Explain what is HBase?

   HBase is an open-source, distributed, NoSQL database that runs on top of Hadoop. It is designed to provide real-time read/write access to large datasets, handling sparse data efficiently.

2. Explain why to use HBase?

   HBase is used for scenarios requiring random, real-time read/write access to Big Data. It is ideal for handling unstructured data, scaling horizontally, and managing large amounts of sparse data across clusters.

3. Mention what are the key components of HBase?

Key components include:
- HBase HMaster: Manages and coordinates the HBase cluster.
- RegionServer: Handles read, write, and update requests.
- Zookeeper: Coordinates and manages the distributed environment.
- Regions: Subsets of tables that store the actual data.

4. Explain what is the row key?
   A Row Key in HBase is a unique identifier for a row of data within a table. It is used to retrieve and store data efficiently, with rows sorted lexicographically by the row key.

5. Differentiate between HBase and HDFS?
   - HBase: NoSQL database that provides real-time read/write access and supports random data retrieval.
   - HDFS: A distributed file system that stores large datasets in a batch-oriented manner, optimized for sequential read/write operations.

6. Mention the difference between HBase and Relational database?
   - HBase: NoSQL, schema-less, supports sparse data, and excels in handling large, distributed datasets.
   - Relational Database: Structured schema, supports ACID transactions, and is optimized for structured data with complex queries.

7. What are the key components of Apache HBase?
   The key components include HMaster, RegionServer, Zookeeper, Regions, and HFile (the storage format for HBase data).

8. What is the use if HBase HMaster?
   HBase HMaster is responsible for managing the distribution of regions across RegionServers, handling administrative tasks like schema changes and balancing the load across the cluster.

9. What is the data model of Apache HBase?
   The HBase data model consists of tables with rows and columns, where each row is identified by a unique row key, and columns are grouped into column families. Each cell in a table has a timestamp, allowing for versioning of data.

10. What is the difference between RDBMS and HBase?
    - RDBMS: Supports structured data with predefined schemas and ACID properties.
    - HBase: NoSQL, schema-less, optimized for large-scale, sparse, and unstructured data with high write throughput.

11. What are the features of Apache HBase?

    Features include:
    - Scalability
    - Distributed architecture
    - Real-time read/write access

- Support for large datasets
- Automatic sharding and load balancing

12. What are some major advantages of Apache HBase?

Advantages include:
- Efficient handling of large, sparse datasets
- Horizontal scalability
- Real-time data access
- Integration with Hadoop ecosystem
- High availability and fault tolerance

13. List the features of pig Latin?

Features include:
- High-level scripting language for data processing
- Support for complex data types
- Extensibility through UDFs (User Defined Functions)
- Optimization through logical and physical plans

14. Difference between MapReduce and Pig.

- MapReduce: A low-level programming model for processing large datasets in parallel.
- Pig: A high-level data flow scripting language built on top of MapReduce, making it easier to write complex data transformations.

15. What are the different ways of executing Pig script?

Pig scripts can be executed:
- In Local Mode: On a single machine.
- In Hadoop Mode (MapReduce Mode): Distributed across a Hadoop cluster.
- Grunt Shell: An interactive shell for running Pig commands.
- Embedded in Java: Using the PigServer API.

16. List the applications of pig.

Applications include:
- Data processing and ETL (Extract, Transform, Load) workflows.
- Data analytics and aggregation.
- Log data analysis.
- Machine learning preprocessing.

17. What are the types of data models in Pig?

Types include:
- Atom: A single data item.
- Tuple: An ordered set of fields.
- Bag: A collection of tuples.
- Map: A set of key-value pairs.

18. How does the user communicate with shell in Apache Pig?

Users communicate with Apache Pig through the Grunt Shell, an interactive command-line interface where they can write and execute Pig Latin scripts.

19. What is Pigstorage?

PigStorage is a built-in storage function in Apache Pig that reads and writes data in a delimited text format, typically used for simple data loading and storing tasks.

20. What is Grunt Shell?

Grunt Shell is the interactive shell for Apache Pig where users can write, execute, and debug Pig Latin scripts in a command-line environment.

21. Explain the different ways to run pig scripts?

Pig scripts can be run in:

- Local Mode: Processing is done on the local machine.
- MapReduce Mode: Scripts are processed on a Hadoop cluster using MapReduce.
- Embedded Mode: Pig scripts are embedded within a Java program using the PigServer API.

22. What is the difference between Pig & SQL?

- Pig: A data flow language used for processing large datasets in Hadoop, allowing procedural programming.
- SQL: A declarative language used for querying and managing relational databases, focusing on structured data.

23. What are the operations supported by Pig?

Operations include:

- Load and Store Data: `LOAD`, `STORE`
- Filter and Transform Data: `FILTER`, `FOREACH`, `MAP`
- Grouping and Joining: `GROUP`, `COGROUP`, `JOIN`
- Sorting: `ORDER`
- Union and Cross: `UNION`, `CROSS`
- Splitting: `SPLIT`

24. What is the difference between Pig Latin and Pig Engine?

- Pig Latin: The scripting language used in Pig for specifying data processing tasks.
- Pig Engine: The execution framework that runs Pig Latin scripts on Hadoop.

25. What is pig storage?

PigStorage is a built-in function in Pig that reads and writes data in a delimited text format, allowing easy interaction with simple flat files.

26. What are the pig execution environment modes?

- Local Mode: Runs Pig on a single machine without requiring Hadoop.
- MapReduce Mode: Executes Pig scripts on a Hadoop cluster using the MapReduce framework.

27. Explain the features of Pig and Pig Latin.

- Pig: Provides a platform for analyzing large datasets using a high-level language and is built on top of Hadoop.

- Pig Latin: A high-level, procedural scripting language with a rich set of operators for performing data transformations, aggregations, and analysis.

28. List the limitations of Hive.
    - Performance: Slower than other query engines for small datasets due to reliance on MapReduce.
    - Schema Evolution: Limited support for changing schema without rewriting tables.
    - Real-Time Queries: Not suitable for real-time query processing.
    - ACID Compliance: Limited support for transactions.

29. What is the difference between Apache Hive and Apache Pig?
    - Hive: Designed for querying and managing structured data using SQL-like queries (HiveQL).
    - Pig: Used for processing large, semi-structured datasets with a more flexible data flow language (Pig Latin).

30. Define the difference between Hive and HBase.
    - Hive: A data warehousing tool that provides SQL-like query capabilities over structured data stored in Hadoop.
    - HBase: A NoSQL database providing real-time read/write access to large, unstructured datasets.
31. Explain what is Hive?
    Hive is a data warehousing and SQL-like query tool built on top of Hadoop, enabling the analysis and querying of large datasets stored in HDFS.

32. What is Hive QL?
    HiveQL (Hive Query Language) is a SQL-like language used in Hive to query and manage data stored in HDFS, with extensions for Hadoop's distributed environment.

33. When to use Hive?
    Hive is used when working with large datasets stored in HDFS that require SQL-like querying capabilities, especially for batch processing and data warehousing tasks.

34. Mention what are the different modes of Hive?
    - Local Mode: Runs Hive queries on local files.
    - MapReduce Mode: Executes Hive queries using Hadoop's MapReduce engine on a Hadoop cluster.
    - Tez Mode: Executes Hive queries using the Tez engine for faster processing.
    - Spark Mode: Executes Hive queries using the Apache Spark engine.

35. Mention key components of Hive Architecture.
    Key components include:
    - Metastore: Stores metadata about tables, columns, and partitions.
    - Driver: Manages the lifecycle of HiveQL statements.
    - Compiler: Converts HiveQL into execution plans.
    - Execution Engine: Executes the plans using Hadoop or other processing engines.
    - CLI/Thrift Server: Provides an interface for users to interact with Hive.

36. Mention what is Hive is composed of?

Hive is composed of the Metastore, Driver, Compiler, Execution Engine, and user interfaces like the CLI or Web UI.

37. Mention what Hive query processor does?

The Hive query processor parses, compiles, and optimizes HiveQL queries, converting them into execution plans that can be run on Hadoop.

38. Mention what are the components of a Hive query processor?

Components include:

- Parser: Parses the HiveQL query.
- Semantic Analyzer: Checks the query for logical correctness.
- Optimizer: Optimizes the query plan for efficient execution.
- Executor: Executes the query plan on Hadoop.

39. What is a metastore in Hive?

The Metastore is a central repository in Hive that stores metadata about the structure of tables, columns, partitions, and the schema of the data.

40. What is a partition in Hive?

A partition in Hive divides a table into segments based on the values of one or more columns, improving query performance by reducing the amount of data scanned.

41. What is indexing and why do we need it?

Indexing in Hive is used to speed up query processing by creating a pointer to specific data locations in a table, reducing the amount of data read during query execution.

42. Mention what is the difference between order by and sort by in Hive?

- ORDER BY: Sorts the entire dataset globally and guarantees a total order of the output.
- SORT BY: Sorts the data within each reducer, not guaranteeing a global order across all data.

Faculty Incharge **Head of the Department**

( ) ( )

HoD Remarks: