UNIT :- 6

1. How is Apache Spark different from MapReduce?

	Spark MapReduce	Spark MapReduce
1.	It's a framework that is open-source which is used for writing data into the Hadoop Distributed file System	- It's an open-source framework used for faster data processing.
2.	It's having a very slow speed as compared to Apache Spark.	- It's much faster than MapReduce.
3.	It's unable to handle real-time processing.	- It can deal with real-time processing.
4.	It's difficult to program as you required code for every process.	- It's easy to program.
5.	It supports more security projects.	- It's security is not as good as MapReduce and continuously working on its security issues.

6.	For performing the task, it's unable to cache in memory.	- It can cache the memory data for processing its task.
7.	It's scalability is good as you can add up to n different nodes.	- It's having low scalability as compared to mapReduce
8.	It actually needs other queries to perform the task.	- It has Spark SQL as its very own query language.

2. What are the important components of the Spark ecosystem?

— * Spark Core:- The core engine that provides the basic functionality for distributed computation.
* Spark SQL:- A module for processing structured data using SQL-like queries.

* Spark Streaming:- A module for processing real-time streams of data.
* Spark MLlib:- A machine learning library with algorithms for classification, regression clustering, and more.
* GraphX:- A graph processing framework for analyzing graphs and networks.

3. What does a Spark Engine do?

— A Spark engine is responsible for:-

* Task Scheduling:- Assigning tasks to worker nodes based on resource availability and workload.

* Data Distribution:- Partitioning data and distributing it across worker nodes.

* Fault Tolerance:- Monitoring the health of worker nodes and recovering from failures.

* In-Memory Computation:- Executing computations on data stored in memory, whenever possible.

* DAG Execution:- Optimizing and executing the DAG of Computations.

4. Define Actions in Spark?

— Actions are operations in Spark that trigger the actual computation and produce a result.

count():- Counts the number of elements in an RDD.

collect():- Returns all elements of an RDD as a local collection.
first():- Returns the first element of an RDD.
take(n):- Returns the first n elements of an RDD.
saveAsTextFile():- Saves an RDD as a text file.

5. What is Immutable?

- Immutability means that data cannot be modified after is created.
- In Spark, RDDs are immutable, which ensures that transformations do not modify the original data and allows for efficient caching and reuse.
- When a transformation is applied to an RDD, a new RDD is created, preserving the original data.
- This immutability helps prevent unexpected side effects and makes Spark's operations more predictable.

UNIT:- 5

1. How HBase uses Zookeeper to Build Applications? Explain in detail.

- HBase, a distributed column-oriented database, heavily relies on Zookeeper for coordination and management.

- key Roles of Zookeeper in HBase:-

1. Name Service:- Zookeeper maintains a hierarchical namespace where HBase tables and regions are registered.
   - Clients use this namespace to locate the correct region servers for their requests.

2. Configuration Management:- HBase configuration settings are stored in Zookeeper.
   - Changes to these settings are propagated to all nodes in the cluster.

3. Master Elections:- When a HBase cluster starts, Zookeeper is used to elect a master node.
   - The master node is responsible for managing the cluster, assigning regions to region servers, and handling table operations.

4. Region Assignment:- The master node uses Zookeeper to keep track of the state of each region Server and assign regions to them based on load balancing and data distribution.

5. Failover:- If a region Server fails, Zookeeper is notified, and the master node reassigns the region to a healthy region Server.

6. Deadlock Prevention:- Zookeeper provides mechanisms to prevent deadlocks between HBase Components, ensuring the Cluster's stability.

- Zookeeper's Data model and operations:-

- ZNodes:- Zookeeper stores data in hierarchical ZNodes, similar to directories in a file system.
- Watches:- Client can register watches on ZNodes to be notified of changes.
- Atomic Operations:- Zookeeper provides atomic operations like create, delete, read and write, ensuring consistency and preventing data corruption.

2. Compare Hive and Pig query language.

	Hive	Pig
1.	SQL-like	- Scripting language (similar to Python)
2.	Easier for SQL users	- Requires scripting knowledge
3.	Generally faster	- Can be slower for complex transformations
4.	Less flexible for complex transformation	- More flexible for complex transformation.

3. Discuss on HiveQL data manipulation queries in detail.

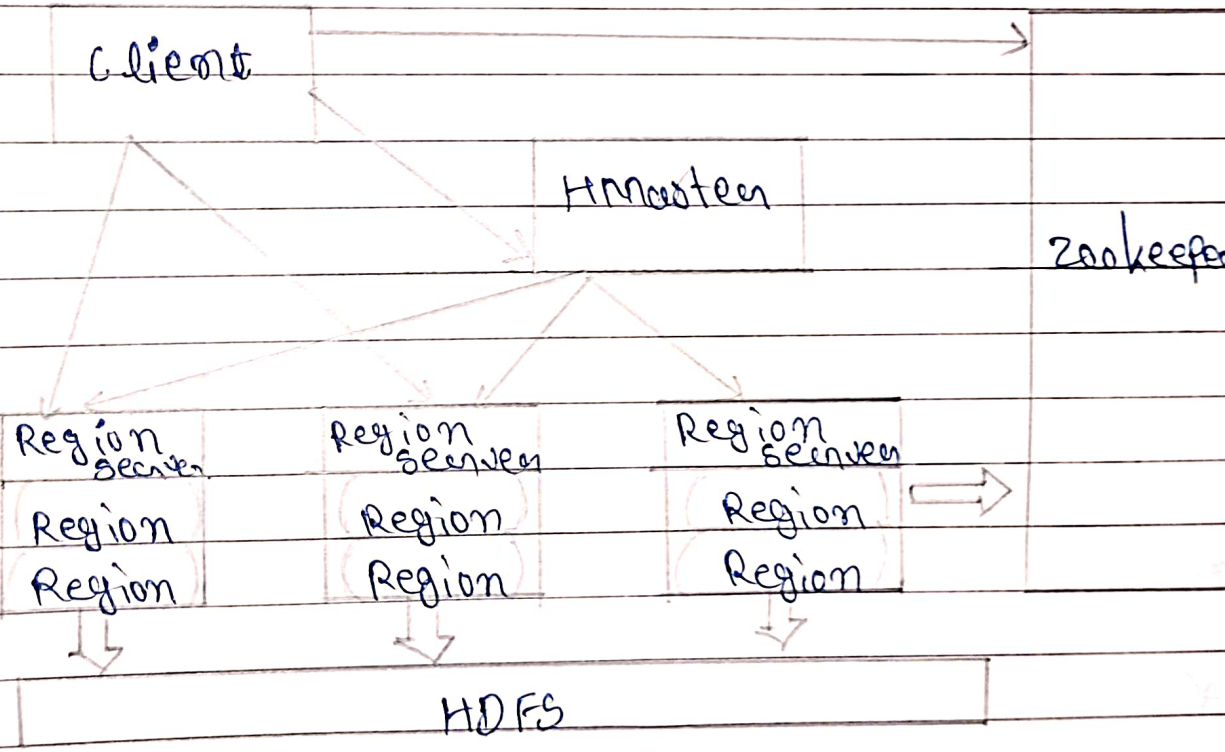
- Select :- Extracts data from tables.

```
SELECT * FROM customers WHERE age > 30;
```

- Insert :- Inserts data into tables.
- Update :- Updates existing data in tables.
- Delete :- Deletes data from tables.
- Create Table :- Creates new tables.

- Drop tables:- Deletes existing tables.
- Alter table:- Modifies existing tables.
- Join:- Combines data from multiple tables.
- Union:- Combines the result sets of multiple queries.

4. Explain briefly on HBase architecture with neat diagram.



- Clients:- Interacts with the HBase cluster to read and write data.
- Zookeeper:- Coordinates the cluster, manages regions, and handles failover.
- Master:- Manages the cluster, assigns regions, and handles table operations.

- Region Server:- Stores and serves data. Each region server is responsible for a subset of a table's data.
- HDFS:- The underlying distributed file system used to store Hbase data.

5. Generalize types of built in operator in HIVE.

- Hive provides a variety of built-in operators for data manipulation

- Arithmetic operators:- +, -, *, /, %
- Comparison operators:- =, !=, <, >, <=, >=
- Logical operators:- AND, OR, NOT
- String operators:- CONCAT, SUBSTRING, LENGTH
- Date and time operators:- YEAR, MONTH, DAY, HOUR, MINUTE, SECOND, etc.
- Aggregate functions:- COUNT, SUM, AVG, MIN, MAX
- window functions:- RANK, DENSE_RANK, ROW_NUMBER etc.

UNIT:- 4

1. Draw architecture of stream processing with brief explanations.

[Architecture]

- Stream processing involves processing data as it arrives in a continuous stream, rather than storing it and processing it in batches.

- Components:-

* Data Sources:-
  Generates the stream of data, such as sensors, social media feeds, or IoT devices.

* Ingestion layer:-
  Receives and processes incoming data, often performing tasks like normalization, filtering, and enrichment.

* Processing Engine:-
  Executes the stream processing logic, such as filtering, aggregation, and machine learning algorithms.

* Storage:-
  Stores processed data for further analysis or long-term retention.

* Output:-
  Delivers processed data to downstream systems or applications.

2. List various stream filters. Write short note on the Bloom filter.

— * Filtering:-
Removes irrelevant on redundant data from the stream.

* Aggregation:-
Combines data points into summary statistics.

* Windowing:-
Processes data within defined time intervals.

* Join:-
Combines data from multiple streams based on a common key.

* Machine learning:-
Applies machine learning algorithms to the stream to extract insights on make predictions.

⟹ Bloom filter:-
- A bloom filter is a probabilistic data structure used to test whether an element is a member of a set.
- It offers fast membership tests with a

false positive rate but no false negatives.

- Key characteristics:-

* Probabilistic:- It provides approximate membership queries, meaning there's a small chance of a false positive.

* Space-efficient:- It uses significantly less space compared to storing the entire set.

* Fast:- Membership tests are typically very fast.

- How it works:-

1. Initialization:- A Bloom filter is initialized with a bit array and a set of hash functions.

2. Insertion:- When an element is inserted it's hashed using all the hash functions.
- The resulting hash values are used to set corresponding bits in the bit array to 1.

3. Membership test:- To check if an element is present, it's hashed using the same hash functions.

- Applications:-

* Membership testing:- Efficiently checking if an item exists in a large set.

* Caching:- Avoiding unnecessary lookups in expensive storage systems.

* Counting unique elements:- Estimating the number of distinct elements in a stream

3. List various sampling methods for data stream. Explain any one in detail.

- Random sampling:- Selects elements randomly from the stream.
- Reservoir sampling:- Maintains a fixed-size reservoir and replaces elements with a certain probability.
- Periodic sampling:- Samples elements at regular intervals.
- Time-based sampling:- Samples elements based on their timestamps.
- Value-based sampling:- Samples elements based on their values.

- Reservoir Sampling :-

Reservoir sampling is a technique for selecting a fixed-size sample from a stream of unknown length.

- It works as follows:-

1. Initialize an empty reservoir of size k.
2. For each incoming element:
   - If the reservoir is not full, add the element.
   - Otherwise, generate a random number between 1 and the current element's index.
   - If the random number is 1, replace a random element in the reservoir with the current element.
3. The reservoir now contains a random sample of k elements from the stream.

4. Examine the need for RTAP.

- Real-time Analytics Processing
- RTAP is essential for applications that require immediate insights from streaming data. Such as:

* Fraud detection:- Identifying suspicious activities in real time.
* IOT monitoring:- Analyzing sensor data to detect anomalies or maintenance needs.
* Financial trading:- Making quick decisions based on market trends.
* Customer service:- Providing real-time support based on customer interactions.

5. Generalize how is data analysis used in
a) Stock market predictions
b) Weather forecasting predictions.

- a) Stock market predictions:-

* Historical data analysis:- Analyzing past stock prices, market trends, and economic indicators to identify patterns and correlations.

* Technical analysis:- Using technical indicators to analyze price charts and identify potential trading signals.

* Fundamental analysis:- Evaluating the financial health and prospects of companies to assess their investment value.

- * Machine learning :- Applying machine learning algorithms to predict future stock prices based on historical data and other factors.

- b) Weather forecasting :-

* Historical data analysis :- Analyzing past weather patterns, climate data, and atmospheric conditions to identify trends and correlations.

* Numerical modeling :- Using complex mathematical models to simulate the atmosphere and predict future weather conditions.

* Satellite imagery :- Analyzing satellite data to monitor weather systems and gather information about atmospheric conditions.

* Ensemble forecasting :- Combining multiple models to improve prediction accuracy and reduce uncertainty.

UNIT:-3

1. What is NoSQL?

- NoSQL, which stands for "Not only SQL" is a type of database that doesn't adhere strictly to the relational database model.
- It offers a more flexible and scalable approach to data storage and retrieval, making it suitable for handling large datasets and complex data structures.

2. List is the advantages of NoSQL.

- Advantages :-

* Scalability :-
NoSQL databases are designed to scale horizontally, meaning you can add more servers to handle increasing workloads without significant downtime or performance degradation.

* Flexibility :-
- NoSQL database can accommodate a wide range of data models, including key-value, document, columnar, and graph.
- This flexibility allows you to store and retrieve data in way that best suits your application requirements.

* Performance:-
NOSQL databases often outperform relational databases for certain types of workloads, especially those that involve large datasets or complex queries.

* Simplicity:-
NOSQL databases can be easier to set up and manage than relational databases, especially for smaller projects.

Analyze the reason behind why do we need NOSQL?

* Handling Large Datasets:-
NOSQL databases are well-suited for handling massive amounts of data that would be difficult or inefficient to store in a relational database.

* Complex Data Structures:-
NOSQL databases can accommodate complex data structures, such as hierarchical or graph-based data, that are challenging to represent in a relational model.

* High availability:-
NOSQL databases often provide built-in mechanisms for ensuring high availability and fault tolerance, making them suitable for mission-critical applications.

* Real-time Analytics:-
NOSQL database can be used to process and analyze large amounts of data in real time, enabling applications to make timely decisions.

4. How to Script NOSQL DB Configuration?

- The specific steps for scripting NOSQL DB Configuration will vary depending on the database system you're using.
- However, most NOSQL databases provide command line tools or APIs that allows you to configure various settings.

* Data storage:- specifying the location and size of data files.

* Indexing:- Creating indexes to improve query performance.

* Replications:- Configuring replication to ensure data redundancy and high availability.

* Security:- Setting up authentication and authorization mechanisms.

5. Difference between NOSQL Vs Relational database?

—	NOSQL	Relational database
1.	key-value, document, columnar, graph	- Tabular (rows and columns)
2.	Flexible, often schema less or dynamically typed	- Rigid, requiring a predefined schema
3.	Horizontal scaling (adding more nodes)	- Vertical scaling (upgrading hardware)
4.	Often better for specific workloads (eg:- large datasets, real-time analytics)	- Generally good for transactional workloads.

5.	Big data, real-time analytics, Content management, IoT	- online transcuctions, data coarehousing, reporting.

UNIT :- 2

1. Give the definition of Hadoop.

- Hadoop is an open-source framework designed to process and store massive amounts of data efficiently.
- It was developed by the Apache Software foundation and primarily used in big data analytics and distributed computing.
- Hadoop is based on the MapReduce programming model, which is a functional paradigm for processing large datasets in parallel.

2. Define how Map-Reduce computation is executed

- Map-Reduce computation involves two main phases:

- 1. Map Phase:-
  - In this phase, the input data is divided into smaller chunks, and a map function is applied to each chunk.
  - The map function transforms the data into key-value pairs.

- 2. Reduce Phase:-
  - The key-value pairs generated in the map phase are grouped by their keys, and a reduce function is applied to each group.
  - The reduce function combines the values associated with each key to produce a final result.

3. Point out the meaning of the term "Hadoop YARN".

- Hadoop YARN (Yet Another Resource Negotiator) is a resource management system that coordinates the allocation of resources to applications running on a Hadoop cluster.
- It acts as a general-purpose platform for running various types of distributed application not just MapReduce jobs.

4. Difference between Hadoop and MapReduce.

	Hadoop	MapReduce
1.	Hadoop is an open-source software framework that is used for storing and processing large amount of data in a distributed computing environment	- MapReduce is a programming model which is implemented for processing and generation Big data sets with distributed algorithm on a cluster.

2.	Hadoop was created by Doug Cutting and Mike Cafarella.	- MapReduce was created by Google.
3.	Hadoop has a storage framework which stores data and creates name nodes and data ~~result~~ nodes. It also has frameworks that includes MapReduce itself.	- MapReduce is a programming framework that has a key and value mappings to process the data.
4.	Hadoop is an open-source. The Hadoop Cluster is highly scalable.	- MapReduce provides a fault tolerance. It also provides with high availability.
5.	Hadoop is a multitude of modules and so may include other programming languages too.	- MapReduce is fundamentally written in the java Programming language.
6.	Hadoop works on the Hadoop distributed File System (HDFS)	- MapReduce can run on HDFS, GFS, NDFS or any other Distributed System.

5. Discuss the features of Hadoop and explain the functionalities of Hadoop clusters?

– Features of Hadoop

* Scalability :-
Hadoop can handel massive datasets and scale horizontally by adding more nodes to the clusters.

* Fault Tolerance :-
It's designed to be fault-tolerant meaning it can recover from node failures without losing data.

* Flexibility :-
Hadoop can be used for various big data applications, including data warehousing, data mining, and machine learning.

* Cost-Effective :-
It's a cost-effective solution for processing large datasets compared to traditional data processing systems.

- Hadoop cluster consists of multiple nodes.

* NameNode :- The master node that manages the Hadoop Distributed file system.

* DataNodes :- Slave nodes that store data blocked of HDFS.

* JobTracker :- The master node that coordinates the execution of mapReduce jobs.

* TaskTrackers :- slave nodes that execute map and Reduce tasks.

6. Express the various core components of the Hadoop.

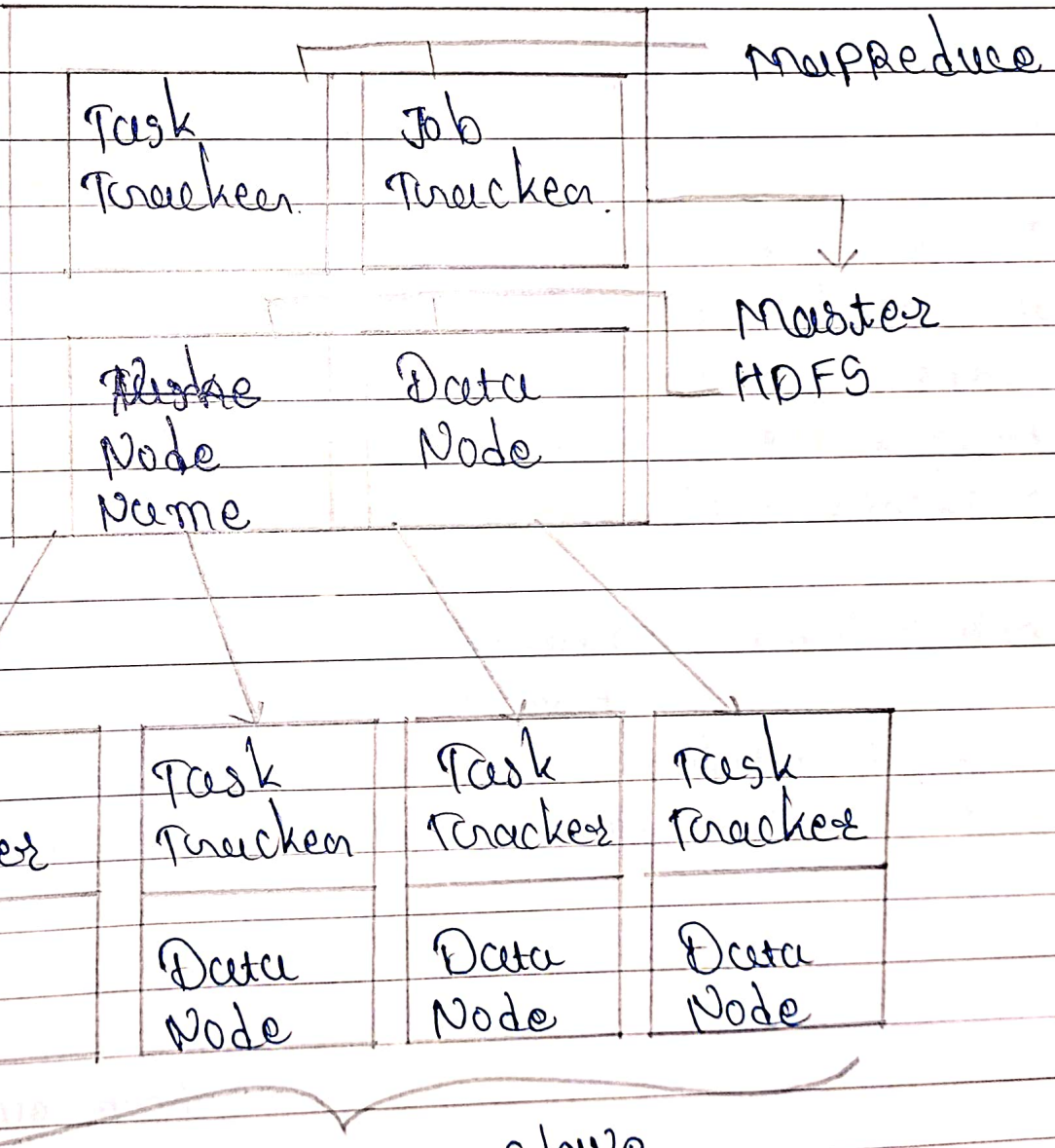
- HDFS (Hadoop distributed file system) :-
  The distributed file system used by Hadoop for storing large datasets.

- MapReduce :-
  The programming model for processing data in parallel.

- YARN :-
  The resource management system.

- Common :-
A set of utilities and libraries used by other components.

7. List about Hadoop distributed file system architecture with neat diagram.



- HDFS is a distributed file system designed to store massive amounts of data across multiple commodity servers.
- It's one of the core components of the Hadoop framework.

- Key Components of HDFS

1. Name Node :-

o Master node :-
- Responsible for managing the file system namespace, keeping track of file locations, and handling client requests.
- Maintains a metadata store that contains information about files, blocks, and their locations.
- Handles file creation, deletion, renaming and permission management.
- Periodically saves metadata to a secondary NameNode for backup and recovery.

2. DataNodes :-

o Slave nodes :-
- Store data blocks of files.
- Report block locations to the NameNode
- Handle block creation, deletion, and

replication.
- Replicate blocks to other DataNodes to ensure data redundancy and fault tolerance.

- Data Replication in HDFS

- Replication factor:-
  - The number of copies of a block stored across different DataNodes.

- Default replication factor:-
  - Typically 3.

- Replication factor can be adjusted:-
  - Depending on the importance of the data and the desired level of fault tolerance.

- Rack awareness:-
  HDFS tries to replicate blocks across different racks to minimize the impact of single rack failures.

— key features of HDFS

- Scalability:- HDFS can handle massive amounts of data by adding more DataNodes to the cluster.

- Fault tolerance:- Data replication ensures that data is not lost in case of node failures.

- High throughput:- HDFS is optimized for large data transfers.

- Write once, read many:- HDFS is designed for data that is written once and read many times.

- Commodity hardware:- HDFS can run on inexpensive commodity hardware.

8. Explain in detail about Hadoop distributed file Systems.

— HDFS write Process

1. Client submits a write request:- The client requests to write a file to HDFS.

2. NameNode assigns blocks:- The NameNode determines the location of the blocks based on the replication factor and rack awareness.

3. DataNode receives blocks:- The client sends the data to the assigned DataNodes.

4. DataNodes replicate blocks:- The DataNodes replicate the blocks to other DataNodes.

5. NameNode updates metadata:- The NameNode updates its metadata to reflect the new block locations.

HDFS Read Process

1. Client requests to read file:- The client requests to read a file from HDFS.

2. NameNode provides block locations:- The NameNode provides the client with the locations of the blocks.

3. Client reads from DataNodes:- The client reads the blocks from the DataNodes.

4. DataNodes may replicate blocks:- If a DataNode is unavailable, HDFS may replicate the block

9. Evaluate a procedure to find the number of occurrences of a word in a document.

— To find the number of occurrences of a word in a document using Hadoop, you could follow these steps:-

1. Split the document :-
   Divide the document into smaller chunks.

2. Map phase :-
   Apply a map function to each chunk to extract words and generate key-value pairs.

3. Combine phase :-
   Combine the key-value pairs within each DataNode to reduce the number of intermediate results.

4. Shuffle phase :-
   Shuffle the intermediate results to group them by key.

5. Reduce phase :-
   Apply a reduce function to each group to count the occurrences of each word.

10. Compile with a neat sketch about processing of a job in Hadoop.

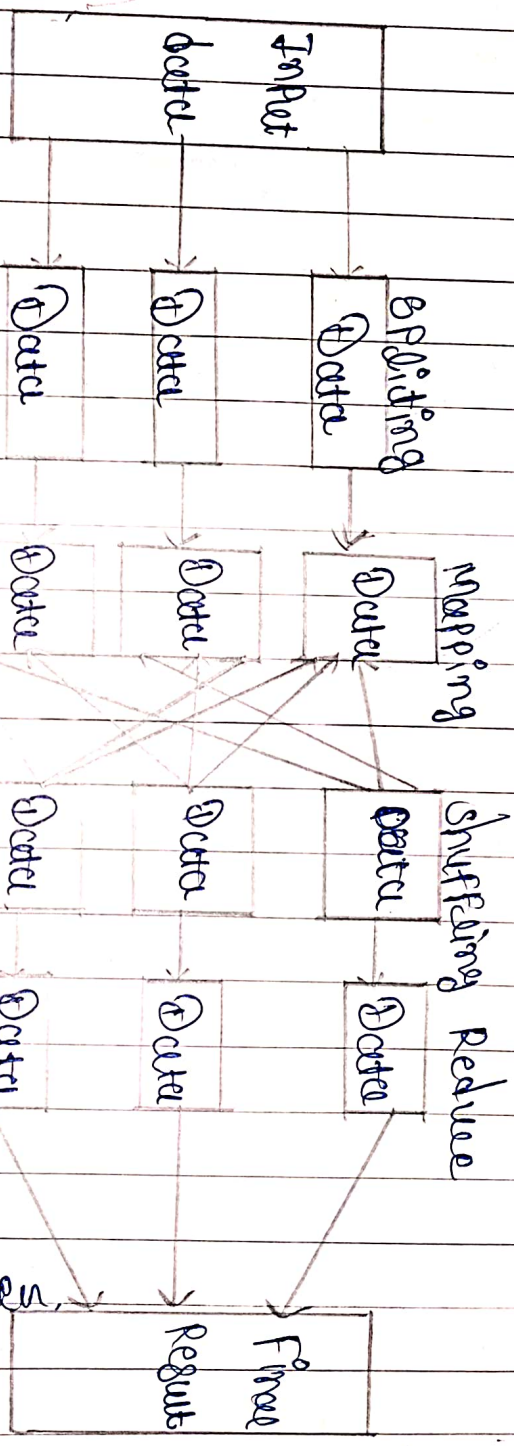
— Job submissions:-
A client submits a job to the Job Tracker.

— Job initialization:-
The JobTracker splits the job into tasks and assigns them to Task-Tracker.

— Task execution:-
Task Trackers execute Map and Reduce tasks.

— Task completion:-
Task Trackers report the status of tasks to the JobTracker.

— Job completion:-
The JobTracker determines when all tasks are ~~Comb~~ complete and the job is finished.



UNIT:-1

Write the defination of "big data" and under what conditions it's given that name.

Big data refers to extremely large datasets that are difficult to process using traditional data processing applications like relational database.
These datasets are characterized by their Volume, Velocity, Veracity and value.

Conditions for naming data as big data

- To be considered big data, a dataset should exhibit at least three of the following characteristics:-

* Volume:- The sheer quantity data is massiv
* Velocity:- The data is generated or collected at a high speed.
* Variety:- The data may contain errors, inconsistencies, or noise.
* Value:- The data has potential business value or can be used for insights.

2. Demonstrate the differences between big data and conventional data.

	Big data	Conventional data
1.	Extremely large	- Relatively small
2.	Generated at a high speed	- Generated at a slower pace
3.	Diverse sources and formates	- Primarily structured data
4.	May contain errors and inconsistencies	- Generally clean and accurate
5.	Requires specialized tools and techniques	- Can be processed using traditional database.

3. Examine the various dimensions of growth of big data.

- Dimensions of growth of big data.

1. Exponential Increase:-

The volume of data is growing exponentially due to factors like IoT devices, social media, and e-commerce.

2. Increasing Varietys-
Data is coming from various sources, including text, images, videos, audio, and sensor data

3. Higher Velocity:-
Data is generated and collected at a faster rate, making it difficult to process in real-time.

4. Increased Complexity:-
The complexity of data is increasing due to factors like unstructured data and data from multiple sources.

4. Difference between data analysis and data reporting

	Data analysis	Data reporting
1.	involves the process of examining data to discover patterns, trends and insights.	- Is the process of presenting data in a structured and informative way.
2.	It involves techniques like statistical analysis, data mining, and machine learning.	- It typically involves creating reports, dashboards, and visualizations.

5. List the risks involved in using big data.

- Data Quality issues:-
  Big data may contain errors, inconsistences, or noise which can lead to inaccurate results.

- Security Risks:-
  Protecting large datasets can be challenging, and data breaches can have serious consequences.

- Ethical Concerns:-
  The use of big data can raise ethical questions, such as privacy concerns and bias.

- Technical challenges:-
  Processing and analyzing big data requires specialized tools and techniques, which can be expensive and complex.

6. Explain the role of big data analytics.

- Big data analytics plays a crucial role in various industries by:-

- Improving Decision makings:-
  By providing insights into customer behaviour, market trends, and operational effciency.

- Optimizing Processes:-
  Identifying inefficiencies and opportunities for improvement.

- Developing New products and Services:-
  Discovering new market opportunities and customer needs.

- Personalizing Customer Experiences:-
  Tailoring products and services to individual preferences.

4. Identify the sources of big data.

- Social media:-
  Platforms like facebook, Twitter, and Instagram generate vast amounts of user-generated data.

- IoT devices:-
  Connected devices like smartphones, sensors and wearables generate data continuously.

E-commerce:-
Online transactions and customer interactions generate data on purchasing behavior, preferences, and demographics.

Government Data:-
Government agencies collect and analyze data on various aspects of society, such as healthcare, education, and transportation.

Scientific Research:-
Scientific experiments and simulations generate large datasets.

Analyze the challenges in big data.

Data Storage:-
Storing large datasets can be expensive and challenging.

Data Processing:-
Processing big data requires specialized hardware and software.

Data Quality:-
Ensuring data quality can be difficult especially for ~~under~~ unstructured data.

- Data Security:-
Protecting large datasets from unauthorized access and breaches is a major challenge.

Talent Shortage:-
There is a shortage of skilled professionals with the expertise to work with big data.

Summarize the reason for the domain expertise for any type of data analytics.

Understanding the data:-
Domain experts can provide context and insights into the data, helping to identify relevant patterns and trends.

Interpreting Results:-
Domain experts can interpret the results of data analysis and translate them into actionable insights.

Identifying Business Problems:-
Domain experts can identify business problems that can be addressed using data analytics.

- Validating Results:-
  Domain experts can validate the results of data analysis to ensure they are accurate and meaningful.

10. Analyze the list of data Analytical tools.

- Hadoop:-
  A framework for processing large datasets

- Spark:-
  A fast and general-purpose cluster computing system.

- NOSQL Databases:-
  Databases designed to handel large, unstructured datasets.

- Data visualization Tools:-
  Tools for creating charts, graphs, and other visualizations.

- Machine learning Libraries:-
  Libraries for building and ~~train~~ training machine learning models.