Spark Assignt.

1) What is Spark and why do we need apache spark

→ Spark is an open source, fast, general purpose in memory process for big data process.

→ Spark is a unify frawork which provide all kinds of data proces

→ It is fast

→ Provide high-level operator

→ Support both real time and batch process

→ Provide an iterative shell to see & learn about the data.

2) Difference between Hadoop map reduce & Spark.

Spark	Hadoop
→ Process speed is fast	→ Slower when compared to spark
→ In memory process	→ Read/write from disk
→ Provide high level operator	→ Doesnt provide high level operator
→ Provide a Interactive shell	→ Dont provide Interactive Shell.

3) List the areas where Spark & Hadoop map Reduce are good.

→ Task where map Reduce is good for:

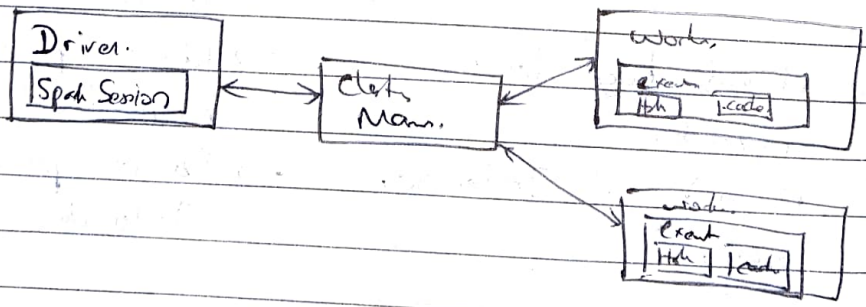
Linear proces of huge data set, Hadoop map Reduce allow parallel proces of huge amount of data

Task where Spark is Good for:

→ Fast data Proces
→ Iterative proces
→ Real time proces
→ Machine Learning
→ Join Datasets

4) How Spark manage the Cluster & Resources

→ Spark is Designed to proces a large volum of data efficiently and quickly. This distributed system is typically deployed onto a collection of machine which is called as Spark cluster. In Cluster, to efficient & intelligent manage a collection of Machine, cluster intelligent manage a collection of mach.



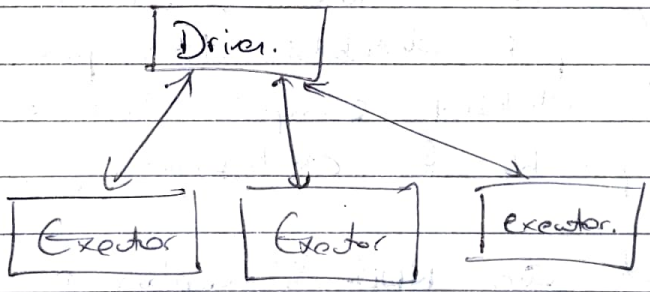
5) Write a note on Spark driver & executor

→ Spark driver:

A spark application consists of two part. The first is application data process logic expressed using spark APIs and other spark driver. The Application data Process logic can be simple as few lines of code to perform.

Spark Executors:

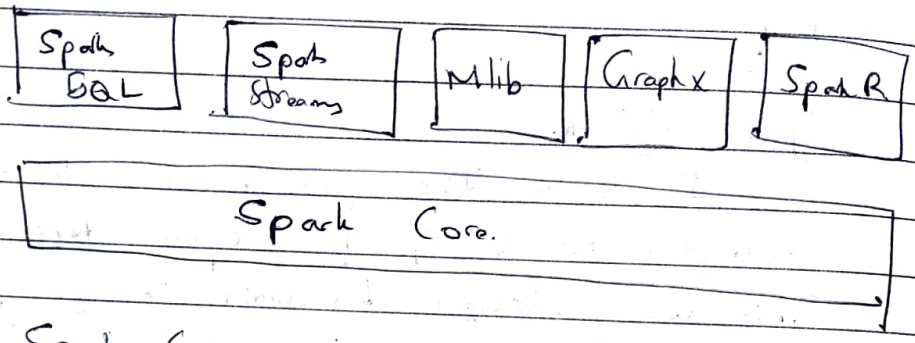
It is a JVM process and is exclusive allocated to a Specific spark Application. This is designed to avoid sharing Spark executors between multiple application. Lifetime of a Spark executor is the lifetime of the Application. Spark follows master-slave Architecture.



6) Write a short note on Spark unified stack & explain Spark core.

→ Spark provides an unified data process engine known as the Spark stack. This stack is built on top of a strong foundation called spark core. which provides all the necessary functions to Manage and run distributed application such as Scheduling, coordination & fault tolerance.

In addition it provid powerful generic program arbitration pro data proc called Resilient Distributed Datasets (RDD's)



Spark Core:-

It provids all necessary functions to manage and run Distributed Appliation such as schedul coordination & fault tolerance. It is the bedrock of spark distributed data proc eqm. The distributed compute Infrastructure is responsible for distribution.

7) What are RDD's & Explain Propert of RDD

→ RDD's are immutable fault tlen, parallel data structure that let user explicit persist intermediate result in memory, control their partition to optim data placement.

→ Immutable RDD's are designed to be immutable which mean you cat specifically modify a particular row in the dataset represented by RDD.

→ Fault tolerant:- The ability to process multiple data in parallel which requires a cluster of Machines to host & execute the computational logic. Spark automatically take care of Handling the failures.

→ Parallel Data Structure:- It is dividing the 1TB file into several chunks and executing on each chunk in a parallelized Manner.

→ In Memory Computation:- RDD power the speed boundary by introducing the ability to perform in-memory computation.

a) What are transformation & Action?

→ In Spark the core data structures are immutable, So if we want to perform any changes, the changes are defined by the transformations.

The actions are the triggers to start the transformations. The ReduceByKey, map, flatMap are the transformations and collect, show are the actions.

a) What is lazy Copy evaluation?

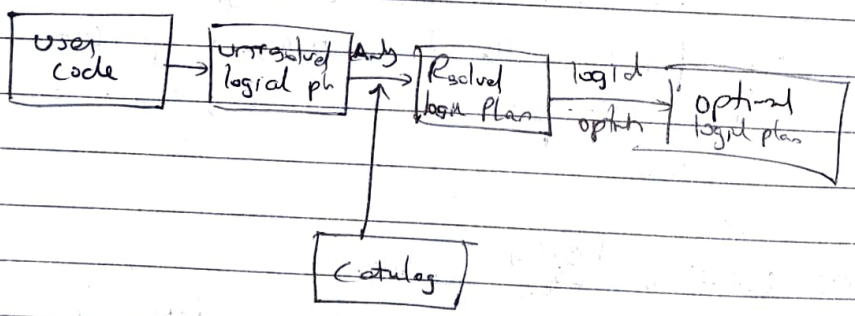
→ Lazy evaluation means that Spark would until the very last moment to execute the graph of computational instructions. In Spark, instead of modifying the data immediately when you express some operations, you build up a plan of transformations that you would like to apply to your source data.

By waiting until the last minute to execute the code, Spark compiles this plan from your raw dataframe transformations to a streamlined physical plan that will run as efficiently as possible across the cluster.

b) What is logical & Physical plans and explain how structured API's execute in Spark

→ Logical Planning:

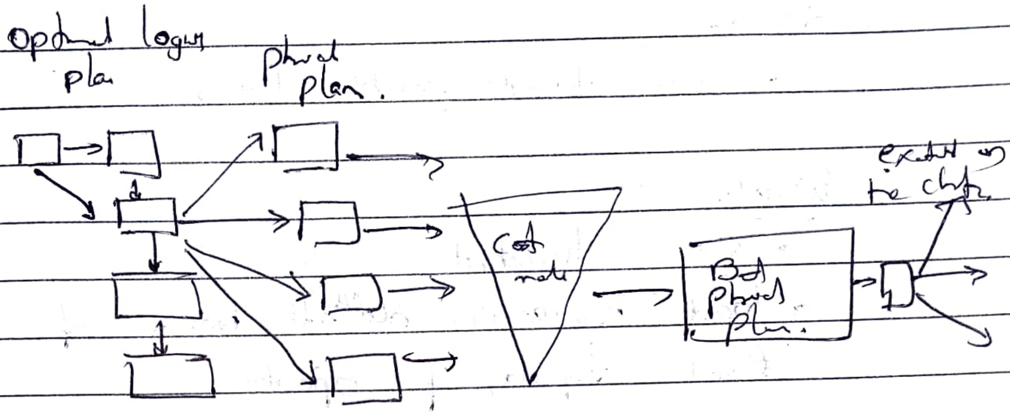
The first phase of execution is meant to take user code and convert it into a logical plan. This logical plan only represents a set of abstract transformations that do not refer to executors or drivers, its purely to convert the user set of expressions into the most optimized version.



Physical Planning:

After creating an optimized logical plan, Spark then begins physical planning process. Physical plan often called as a Spark plan specifies how the logical plan will

execute on cluster by generating different physical execution.



11) Explain use-cases of stream processing
→ Notifications & Alerts
→ Real time reports
→ Realtime decision making
→ Update the data to serve in real time
→ Online machine learning

12) List out the challenges of stream processing.
→ Handle & process out of order data
→ Maintain large amount of state
→ Support high-data throughput
→ Handle load imbalance
→ Respond to events at low latency
→ Join with external data in other storage system.

13) Explain continuous stream processing model, what are its advantages & Disadvantages
→ In continuous processing based system each node in the system is continuously listen to System is continuous listen to messages.

from other nodes and outputs new updates to its child nodes

Advantages
* offers lowest possible latency

Disadvantages
* lower throughput
* load balancing difficult because of fixed topology of operators

14) Explain Micro-Batch Stream processing, what are its advantages & disadvantages
→ It waits to accumulate small batches of input data then processes each batch in parallel using a distributed collection of tasks.

Advantages:
→ High throughput per node
→ Dynamic load Balancing

Disadvantages:
→ Higher latency due to wait to accumulate a micro-batch.

15) Write a short note on DStream & unified stream
→ DStream:
→ Spark original DStream API has been used broadly for stream processing since its first release in 2012

→ API is purely based on processing time to handle event time operation

Structured Streaming:

→ Higher-level stream API built from the ground up on Spark SQL API
→ Runs on both continuous and micro batch execution models.

16) Write short note on Data sources & Data sink of structured streaming in Spark

→ Data Sources:

→ Apache Kafka
→ Files on distributed file system
→ A socket source for testing

Sinks

→ Apache Kafka
→ Almost any file format
→ A console sink for testing
→ A memory sink for debugging