211720104014

BIG DATA HOME WORK 4

Question No:1

1) Compare Hadoop and Spark:

A:

- Batch processing can be handled effectively by Hadoop. Spark is made to efficiently handle real-time data.

- Hadoop is a framework for high-latency computing that lacks an interactive mode.

- Spark is a low latency computing platform that supports interactive data processing.

Question No:2

2) What is Apache Spark?

- Big data workloads are processed using Apache Spark, an open-source distributed processing engine.

- It uses efficient query execution and in-memory caching for quick analytic queries against any size of data.

Question No:3

3) Explain the key features of Apache Spark.

A:

Features of Apache Spark

- Fault tolerance.
- Dynamic In Nature.
- Lazy Evaluation.
- Real-Time Stream Processing.
- Speed.
- Reusability.
- Advanced Analytics.
- In Memory Computing.

Question No:4

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

   - Java, Python, R, and Scala. Scala and Python have interactive Spark shells out of all of these.

   - Both the Python and Scala shells are accessible via pyspark and spark-shell, respectively.

   - Because Spark is developed in Scala and it is the most often utilized for Spark, Scala is the language that is used the most frequently.

Question No:5

5) What are benefits of Spark over MapReduce?

   A:

- Spark and MapReduce vary primarily in that Spark processes data in memory and keeps it there for following steps while MapReduce processes data on storage.

- As a result, Spark's data processing speeds are up to 100 times quicker than MapReduce's for lesser workloads.

Question No:6

6) Explain the concept of Resilient Distributed Dataset (RDD)

A:

- Since the beginning of Spark, RDD has been the main API that users interact with.

- An RDD's fundamental component is an immutable distributed collection of data components that is divided among cluster nodes and may be controlled concurrently using a low-level API that provides transformations and actions.

Question No:7

7) How do we create RDDs in Spark?

A. A dataset on an external storage system, such as a shared filesystem, HDFS, HBase, or any data source supplying a Hadoop InputFormat, can be referenced in order to create RDDs in one of two ways: by parallelizing an existing collection in your driver program, or by referencing a dataset there.

Question No:8

8) What is Executor Memory in a Spark application?

A:

- Yarn overhead memory and JVM heap memory are added together to create each executor memory.
- RDD Cache Memory is a component of JVM Heap Memory, Random Memory.

Question No:9

9)What do you understand by Transformations in Spark?

A:

- A function called Spark Transformation creates new RDDs from the existing RDDs.
- It generates one or more RDDs from an input of RDD. Every time we apply a transformation, a new RDD is created.
- As a result, since RDDs are immutable by nature, they cannot be modified.

Question No:10

10)Define Actions in Spark

A:

- RDD operates by returning value to the Spar driver programs, which in turn start a job to run on a cluster.
- The output of a transformation is an input of actions.
- Diminish, gather, and take Common Apache Spark actions include sample, take, first, saveAsTextfile, saveAsSequenceFile, countByKey, and foreach.