Dxc\_Assignment\_7-6-2022

Shaik Mohammed Raihan

Roll\_number: DXC262AB12036

1. Explain what are various components of SPARK with block diagram?

Sol:-

* Spark core
* Spark sql
* Spark streaming
* Mlib-
* Graphx

It also covers components of Spark ecosystem like Spark core component, Spark SQL, Spark Streaming, Spark MLlib, Spark GraphX and SparkR. We will also learn the features of Apache Spark ecosystem components in this Spark tutorial.

Cost based optimizer. Follow Spark SQL Optimization tutorial to learn more.

Mid query fault-tolerance: This is done by scaling thousands of nodes and multi-hour queries using the Spark engine. Follow this guide to Learn more about Spark fault tolerance.

Full compatibility with existing Hive data.

DataFrames and SQL provide a common way to access a variety of data sources. It includes Hive, Avro, Parquet, ORC, JSON, and JDBC.

Provision to carry structured data inside Spark programs, using either SQL or a familiar Data Frame API.

It is an add-on to core Spark API which allows scalable, high-throughput, fault-tolerant stream processing of live data streams. Spark can access data from sources like Kafka, Flume, Kinesis or TCP socket. It can operate using various algorithms. Finally, the data so received is given to file system, databases and live dashboards. Spark uses Micro-batching for real-time streaming.

Micro-batching is a technique that allows a process or task to treat a stream as a sequence of small batches of data. Hence Spark Streaming, groups the live data into small batches. It then delivers it to the batch system for processing. It also provides fault tolerance characteristics. Learn Spark Streaming in detail from this Apache Spark Streaming Tutorial.

How does Spark Streaming Works?

There are 3 phases of Spark Streaming:

a. GATHERING

The Spark Streaming provides two categories of built-in streaming sources:

Basic sources: These are the sources which are available in the StreamingContext API. Examples: file systems, and socket connections.

Advanced sources: These are the sources like Kafka, Flume, Kinesis, etc. are available through extra utility classes. Hence Spark access data from different sources like Kafka, Flume, Kinesis, or TCP sockets.

b. PROCESSING

The gathered data is processed using complex algorithms expressed with a high-level function. For example, map, reduce, join and window. Refer this guide to learn Spark Streaming transformations operations.

c. DATA STORAGE

The Processed data is pushed out to file systems, databases, and live dashboards.

Spark Streaming also provides high-level abstraction. It is known as discretized stream or DStream.

DStream in Spark signifies continuous stream of data. We can form DStream in two ways either from sources such as Kafka, Flume, and Kinesis or by high-level operations on other DStreams. Thus, DStream is internally a sequence of RDDs.

MLlib in Spark is a scalable Machine learning library that discusses both high-quality algorithm and high speed.

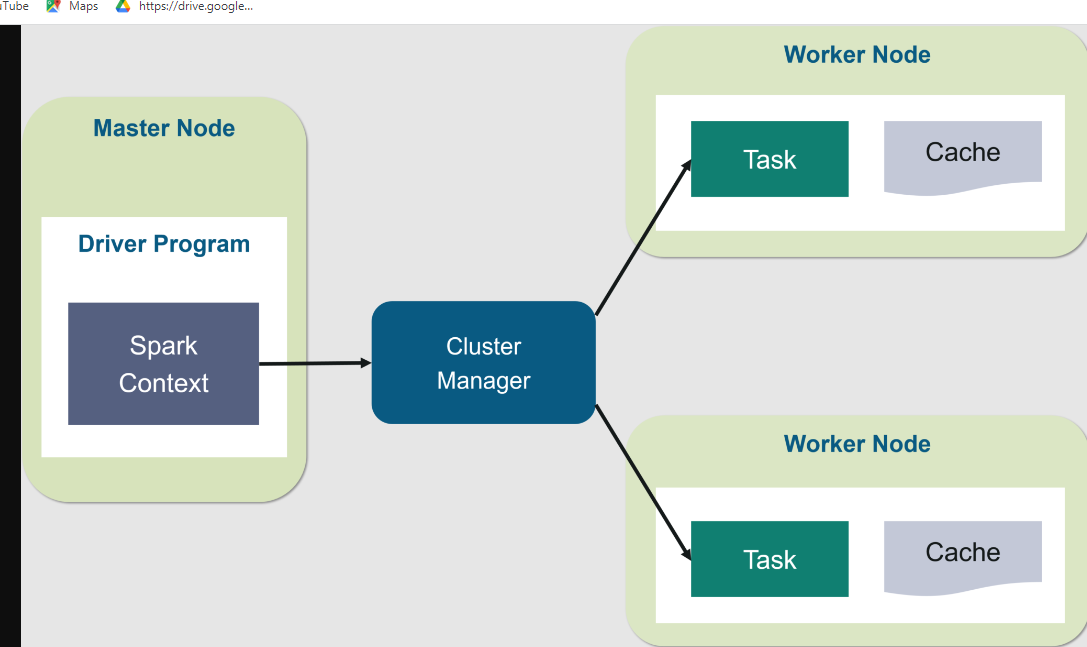
The motive behind MLlib creation is to make machine learning scalable and easy. It contains machine learning libraries that have an implementation of various machine learning algorithms. For example, clustering, regression, classification and collaborative filtering. Some lower level machine learning primitives like generic gradient descent optimization algorithm are also present in MLlib.

In Spark Version 2.0 the RDD-based API in spark.mllib package entered in maintenance mode. In this release, the DataFrame-based API is the primary Machine Learning API for Spark. So, from now MLlib will not add any new feature to the RDD based API.

GraphX in Spark is API for graphs and graph parallel execution. It is network graph analytics engine and data store. Clustering, classification, traversal, searching, and pathfinding is also possible in graphs. Furthermore, GraphX extends Spark RDD by bringing in light a new Graph abstraction: a directed multigraph with properties attached to each vertex and edge.

GraphX also optimizes the way in which we can represent vertex and edges when they are primitive data types. To support graph computation it supports fundamental operators (e.g., subgraph, join Vertices, and aggregate Messages) as well as an optimized variant of the Pregel API.

1. Explain Spark core in details & how RDD is related to Spark core - explain with Spark program ?
2. Spark core



Are RDDs being relegated as second class citizens? Are they being deprecated? The answer is a resounding NO! What’s more is you can seamlessly move between DataFrame or Dataset and RDDs at will—by simple API method calls—and DataFrames and Datasets are built on top of RDDs.

The key idea of spark is Resilient Distributed Datasets (RDD); it supports in-memory processing computation. This means, it stores the state of memory as an object across the jobs and the object is sharable between those jobs.

**Spark program:**

rdd = spark.sparkContext.parallelize([

      (1,2.,'string1',date(2022,6,6),datetime(2022,6,6,12,30)),

      (2,3.,'string2',date(2022,7,6),datetime(2022,6,7,12,30)),

      (3,4.,'string3',date(2022,8,6),datetime(2022,6,8,12,30)),

])

df=spark.createDataFrame(rdd, schema=['a','b','c','d','e'])

df

**3.Explain various Mlib algorithms Spark is supporting?**

The popular algorithms and utilities in Spark MLlib are:

1. Basic Statistics
2. Regression
3. Classification
4. Recommendation System
5. Clustering
6. Dimensionality Reduction
7. Feature Extraction
8. Optimization

**4.Explain benifits Spark SQL & how relational data will be inserted into SPARK?**

1. Support relational processing both within Spark programs (on native RDDs) and on external data sources using a programmerfriendly API.

2. Provide high performance using established DBMS techniques.

3. Easily support new data sources, including semi-structured data and external databases amenable to query federation.

4. Enable extension with advanced analytics algorithms such as graph processing and machine learning.

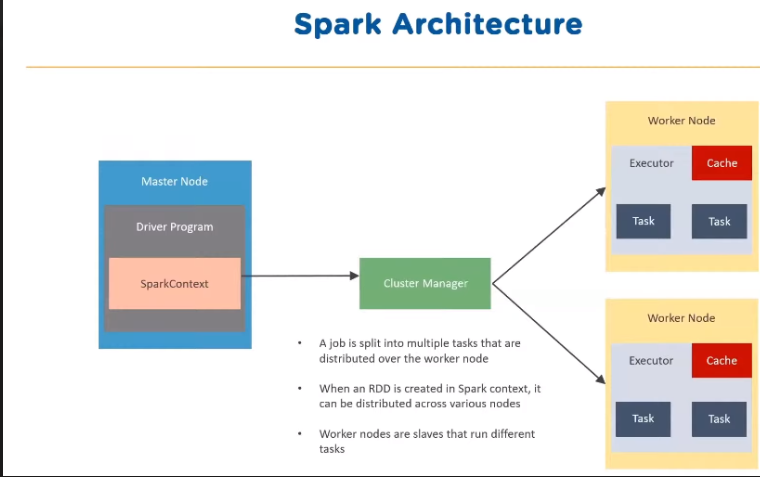
**5. Explain Spark streaming in detail ?**

Spark streaming is a light weight api that allow developers to perform batch processing and real time streaming of data with ease.

Input data->spark engine->batches of input data->spark engine->batches of processed data

It provides secure reliable and fast processing of live data streams

**6.Explain SPARK architecure? what is Master - Slave architecure ?**

****

Master slave is a master node and worker nodes are slaves that run different tasks.

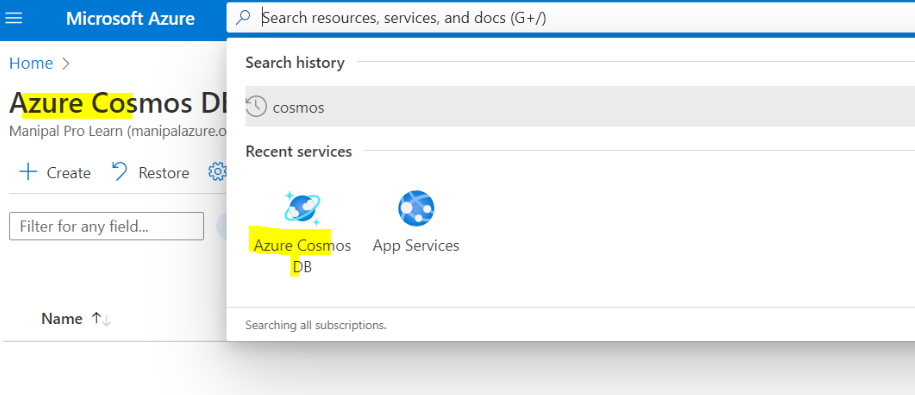
**7. Explain various cluster managers in SPARK?**

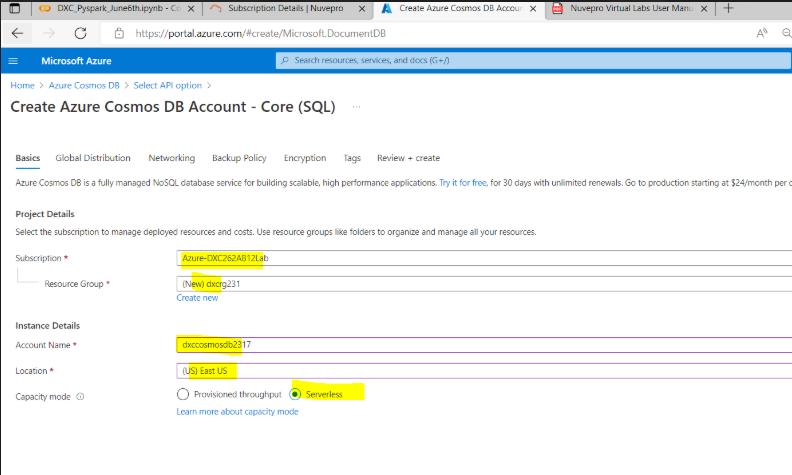
**There are four types of cluster managers**

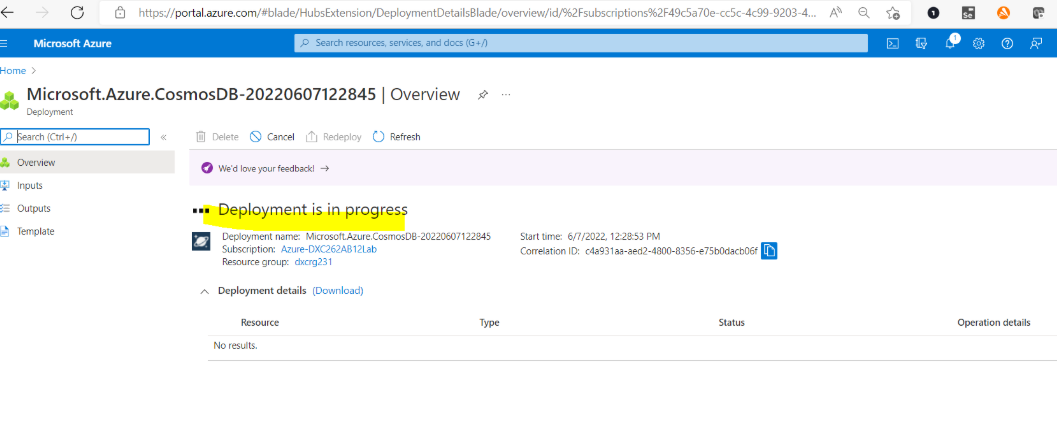
* **Sparklone:-by default applications submitted to the standalone mode cluster will run in fifo orderand each application will try to use all available nodes**
* **Mesos:-apache mesos is an open source project to manage computer clusters and can also run hadoop applications**
* **Hadoop yarn:-apache yarn is the cluster resource manager of hadoop 2 spark can be run on yarn**
* **Kubernetes**

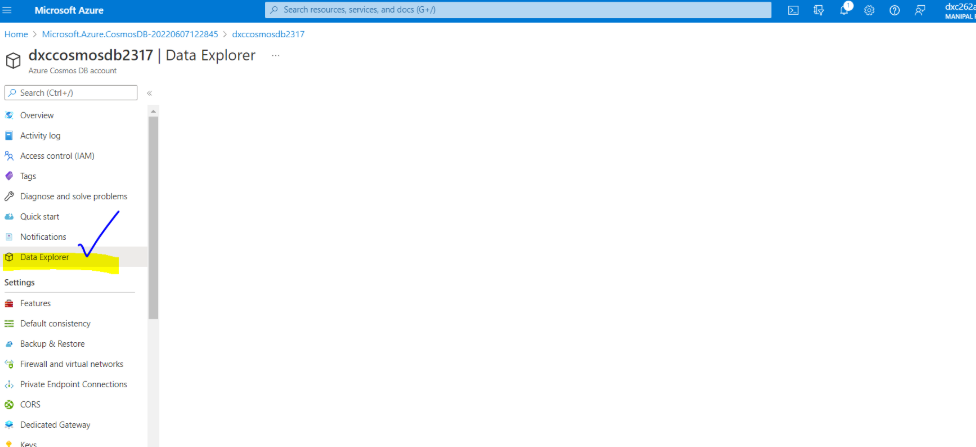
**9.Explain with sceenshots & steps how to create Cosmos DB ?**

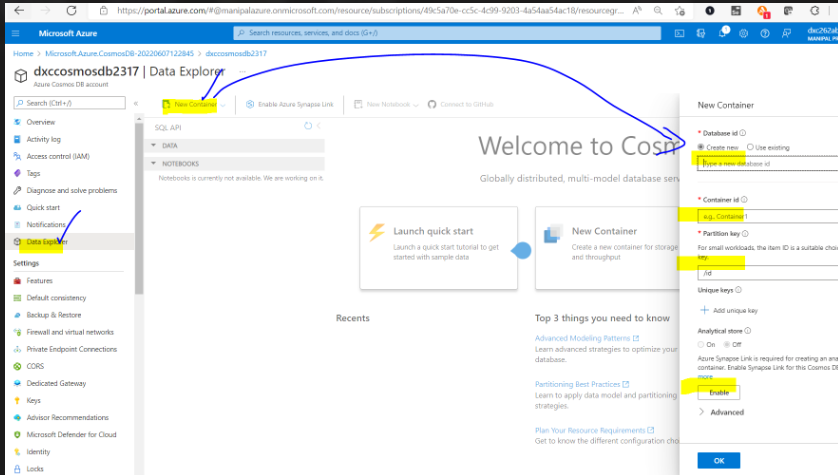
**Steps to create cosmos db?**

****

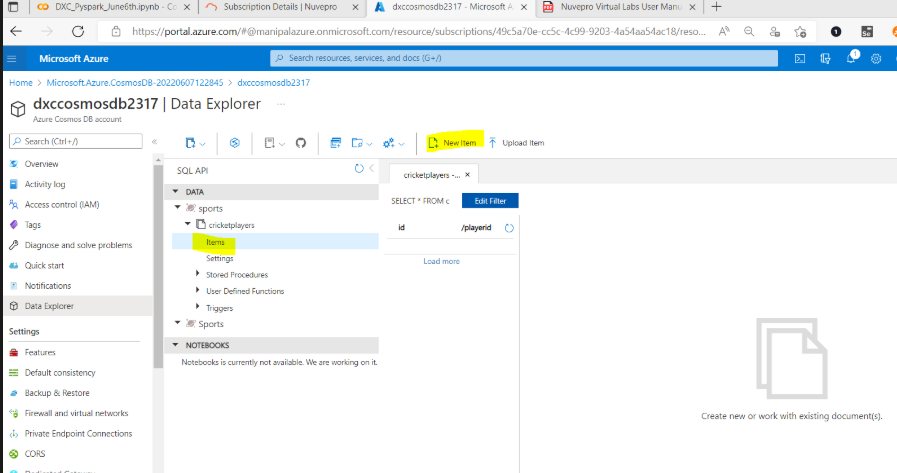
****

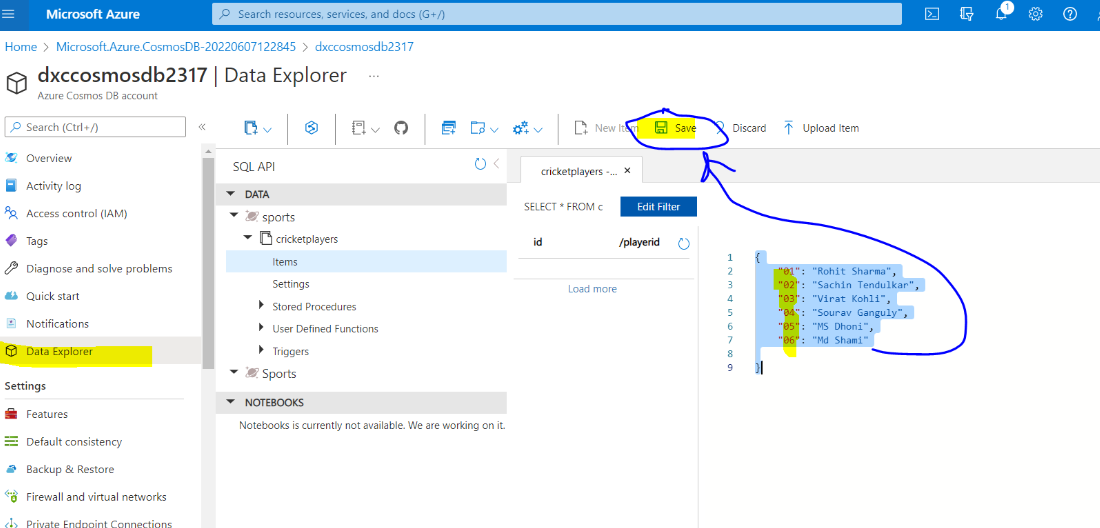
****

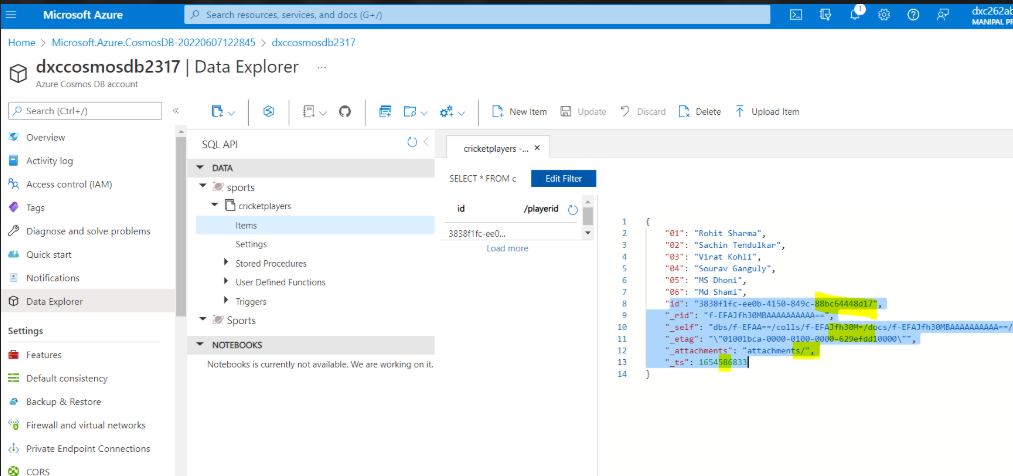
****

****

1. **Explain with sceenshots & step how to insert data into Cosmos DB?**

****





10.Explain with sceenshots & step how to create Azure SQL Db & also explain how toinsert data into Azure SQL D

