SPARK Exercise

1. Explain Architecture of Spark?

- Transformation in Spark refers to the process of taking an RDD, DataFrame, or Dataset and applying some operation to it to produce another RDD, DataFrame, or Dataset.

- These operations are lazy, meaning they don't execute immediately but rather create a plan of execution. Examples of transformations include map, filter, flatMap, groupBy, and join.

2. Difference between Hadoop and spark.

- Hadoop and Spark are both big data processing frameworks, but Spark offers advantages such as in-memory processing, support for real-time and batch processing, and a wider range of APIs (including SQL, streaming, machine learning, and graph processing) compared to Hadoop's primarily batch-oriented MapReduce paradigm.

3. Difference between RDD, Dataframe, Dataset.

- RDD (Resilient Distributed Dataset) is the basic abstraction in Spark, representing a distributed collection of objects.
- DataFrame is a distributed collection of data organized into named columns, providing structured data processing capabilities.
- Dataset is a distributed collection of data with strong typing, introduced as an extension of DataFrame API, offering benefits of both RDDs and DataFrames.

4. Explain the similarities in all API of Spark.

- All APIs in Spark offer ways to manipulate distributed data and perform operations like transformations and actions. They leverage parallelism, fault tolerance, and support various data sources, allowing integration with external libraries for advanced analytics.

5. What is Transformation? Explain in detail.

- Transformation in Spark refers to operations applied to RDDs, DataFrames, or Datasets to produce another RDD, DataFrame, or Dataset. Transformations are lazy and create a plan of execution. Examples include map, filter, and groupBy.

6. What are Actions in spark? Explain in detail.

- Actions in Spark are operations that trigger the execution of transformations, causing actual computation to occur. Unlike transformations, actions are eager and return results to the driver program or write them to external storage. Examples include collect, count, and save.

7. What is the Wide Transformation? explain with example

- Wide transformations involve shuffling data across partitions, typically after operations requiring data from different partitions to be combined. Example: reduceByKey, where data is shuffled across the cluster to combine values with the same key.

8. What is Narrow Transformation? Explain with examples.

- Narrow transformations involve operations where each partition of the parent RDD contributes to only one partition of the child RDD, without shuffling data across partitions. Example: map, where each partition of the input RDD is processed independently.

9. Write down the query of wide n narrow transformation with example?

- Wide Transformation Example (shuffle operation):

```
# Wide transformation: reduceByKey

rdd = sc.parallelize([(1, 2), (2, 3), (1, 4)])
result = rdd.reduceByKey(lambda x, y: x + y)
```

- Narrow Transformation Example:

```
# Narrow transformation: map

rdd = sc.parallelize([1, 2, 3, 4, 5])
result = rdd.map(lambda x: x * 2)
```

10. Explain Kerberos Architecture.

- Kerberos is a network authentication protocol with components like Client, Key Distribution Center (KDC) comprising Authentication Server (AS) and Ticket Granting Server (TGS), and Service Server. It enables secure authentication and authorization without transmitting passwords over the network.