# Big Data Hadoop Training

Session 12 Assignment 1 Solution:

1. **What is in-memory processing in Spark?**

a) Processing data in each node

b) Storing data in RDBMS during processing

c) Maximum effective utilization of RAM during the processing

d) Using more no of CPU threads

A. **c) Maximum effective utilization of RAM during the processing**

**2. What are the features of Apache Spark?**

a) In-memory processing

b) Ease of use APIs

c) Unified high level tools

d) Runs Everywhere (Hadoop, Mesos, standalone, or in the cloud. It can access diverse data sources including HDFS, Cassandra, HBase, S3.)

e) All the above

f) None of the above

A. **e) All the above**

In-memory processing, Ease of use APIs, Unified high level tools, Runs Everywhere (Hadoop, Mesos, standalone, or in the cloud. It can access diverse data sources including HDFS, Cassandra, HBase, S3) All these are features of Apache Spark.

**3. Apache Hadoop is more active project than Apache Spark in open source community in the last year.**

a) true

b) false

A. **b) false**

Apache Spark is the more active project since 2014

**4. Driver program will be launched in every node of the worker.**

a) true

b) false

A. **b) false**

The Spark Driver is actually a process running the main () function of the application and creating the SparkContext. The driver program can be launched on the Master node or if the driver program is submitted to a YARN cluster, the Driver may be scheduled to a Worker node. If the driver program is launched on a local PC, then it will run atop the local PC. However, this approach is not recommended since the local PC may not be in the same network with the Worker nodes, which will slow down the communication between the driver and the executors.

Therefore, driver program is not launched in every node of the worker.

**5. Spark only supports Stream processing.**

a. Yes

b. No

A. **b) No**

Spark supports other components as well like Spark SQL, MLlib (Machine Learning Library) , GraphX in addition to Stream processing