Session 2: Introduction to Spark

Assignment 2*Spark*

**A C A D G I L D Page 2**

*Session 2: Introduction to Spark*

*Assignment 2*

**Table of Contents**

1. Introduction ........................................................................................................................................ 3

2. Objective ............................................................................................................................................. 3

3. Prerequisites ....................................................................................................................................... 3

4. Associated Data Files ............................................................................................................................ 3

5. Problem Statement ............................................................................................................................... 3

6. Approximate Time to Complete Task ................................................................................................... 3 *Spark*

**A C A D G I L D Page 3**

**1. Introduction**

In this assignment you need to answer the given questions.

**2. Objective**

This assignment will help you to understand Big Data basics.

**3. Prerequisites**

None

**4. Associated Data Files**

N/A

**5. Problem Statement**

1. How are worker, executor and task related to each other?

Worker – Slave daemon in Spark responsible for computation

• Workers contain executors for driving out the computations.

Worker nodes are machines that run executors

• Host one or multiple Workers

• One JVM (= 1 UNIX process) per Worker

• Each Worker can spawn one or more Executors

• Executors run tasks

• Run in child JVM (= 1 UNIX process)

• Execute one or more task using threads in a ThreadPool

TASK:

Unit of work to execute, in an executor thread

• Unlike MR, there is no “map” vs “reduce” task

• Each task, either partitions its output for “shuffle”, or sends the output back to

the driver

2. What are the key features of Spark?

Allows Integration with [Hadoop](https://tekslate.com/hadoop-tutorial-interview-questions-answers/) and files included in HDFS.

Spark has an interactive language shell as it has an independent Scala (the language in which Spark is written) interpreter .

Spark consists of RDD’s (Resilient Distributed Datasets), which can be cached across computing nodes in a cluster.

Spark supports multiple analytic tools that are used for interactive query analysis , real-time analysis and graph processing.

3. What is Spark Driver?

The driver performs the following:

1. Connects to a cluster manager to allocate resources across applications

2. Acquires executors on cluster nodes processes run compute tasks, cache data

3. Sends app code to the executors

4. Sends tasks for the executors to run

4. What are the benefits of Spark over MapReduce?

|  |  |  |
| --- | --- | --- |
| **Criteria** | **MapReduce** | **Spark** |
| Processing Speeds | Good | Excellent (up to 100 times faster) |
| Data caching | Hard disk | In-memory |
| Perform iterative jobs | Average | Excellent |
| Independent of Hadoop | No | Yes |
| Machine learning applications | Average | Excellent |

5. What is Spark Executor?

Executors are worker nodes' processes in charge of running individual tasks in a given **Spark** job. They are launched at the beginning of a **Spark** application and typically run for the entire lifetime of an application

**6. Approximate Time to Complete Task**

30 min

*Session 2: Introduction to Spark*

*Assignment 2*

**Table of Contents**

1. Introduction ........................................................................................................................................ 3

2. Objective ............................................................................................................................................. 3

3. Prerequisites ....................................................................................................................................... 3

4. Associated Data Files ............................................................................................................................ 3

5. Problem Statement ............................................................................................................................... 3

6. Approximate Time to Complete Task ................................................................................................... 3 *Spark*

**A C A D G I L D Page 3**

**1. Introduction**

In this assignment you need to answer the given questions.

**2. Objective**

This assignment will help you to understand Big Data basics.

**3. Prerequisites**

None

**4. Associated Data Files**

N/A

**5. Problem Statement**

1. How are worker, executor and task related to each other?

Worker – Slave daemon in Spark responsible for computation

• Workers contain executors for driving out the computations.

Worker nodes are machines that run executors

• Host one or multiple Workers

• One JVM (= 1 UNIX process) per Worker

• Each Worker can spawn one or more Executors

• Executors run tasks

• Run in child JVM (= 1 UNIX process)

• Execute one or more task using threads in a ThreadPool

TASK:

Unit of work to execute, in an executor thread

• Unlike MR, there is no “map” vs “reduce” task

• Each task, either partitions its output for “shuffle”, or sends the output back to

the driver

2. What are the key features of Spark?

Allows Integration with Hadoop and files included in HDFS.

Spark has an interactive language shell as it has an independent Scala (the language in which Spark is written) interpreter .

Spark consists of RDD’s (Resilient Distributed Datasets), which can be cached across computing nodes in a cluster.

Spark supports multiple analytic tools that are used for interactive query analysis , real-time analysis and graph processing.

3. What is Spark Driver?

The driver performs the following:

1. Connects to a cluster manager to allocate resources across applications

2. Acquires executors on cluster nodes processes run compute tasks, cache data

3. Sends app code to the executors

4. Sends tasks for the executors to run

4. What are the benefits of Spark over MapReduce?

|  |  |  |
| --- | --- | --- |
| **Criteria** | **MapReduce** | **Spark** |
| Processing Speeds | Good | Excellent (up to 100 times faster) |
| Data caching | Hard disk | In-memory |
| Perform iterative jobs | Average | Excellent |
| Independent of Hadoop | No | Yes |
| Machine learning applications | Average | Excellent |

5. What is Spark Executor?

**Executors** are worker nodes' processes in charge of running individual tasks in a given **Spark** job. They are launched at the beginning of a **Spark** application and typically run for the entire lifetime of an application

**6. Approximate Time to Complete Task**

30 min