*Spark*

**A C A D G I L D Page 1**

Session 6: In-Memory Management

Assignment 3*Spark*

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

*Session 6: In-Memory Management*

*Assignment 3*

**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 write the code for the given questions

**2. Objective**

This assignment will help you to understand Big Data basics.

**3. Prerequisites**

None

**4. Associated Data Files**

https://drive.google.com/open?id=0ByJLBTmJojjzYWxickJ5U3JHbGM

The above dataset consists of the details of movies. The dataset description is as follows

Metadata header: id, name, year, rating, duration

**5. Problem Statement**

1. Find the top 5 longest movies according to their duration

val textFile = sc.textFile("/tmp/movie.txt")

val split = textFile.map(lines=>lines.split(“,”)).map(x=>(x(1),1)).reduceByKey(\_+\_).map(item => item.swap).sortByKey(false).take(5)

2. Find the top 5 rated movies in the year 1995

val textFile = sc.textFile("/tmp/movie.txt")

val header = dataset.first()

val format = new java.text.SimpleDateFormat("MM/dd/yyyy")

var days =Array("Sun","Mon","Tue","Wed","Thu","Fri","Sat")

val eliminate = dataset.filter(line => line != header)

val split = eliminate.map(line => line.split(",")).map { x => (x(0),format.parse(x(1)),x(3)) }

val combine = split.map(x => (x.\_1+" "+days(x.\_2.getDay),x.\_3.toInt))

val arrange = combine.reduceByKey(\_+\_).map(item => item.swap).sortByKey(false).take(5).collect.foreach(println)

**6. Approximate Time to Complete Task**

30 min