

Database Implementation System - Programming Assignment - Hadoop and MapReduce From Mrinal Monga

1. The tutorial provided in the Sakai resources was followed.
2. The MapReduce task that returns the average rating of movies in the Netflix dataset was implemented. Please see below for more details

Appropriate screenshots are given at the end of the document which make it easier to understand this report

Attachment contain the java programs and other files.

Submission related details -

1. Path to the s3 bucket with output - **s3://rutgers.monga.mrinal/**
2. For the Word Count example the output is at **s3://rutgers.monga.mrinal/WordCountTestOutput** and the cluster used was **WordCountTest**
3. For the Netflix Small dataset the output is at **s3://rutgers.monga.mrinal/Netflix_small_output** and the cluster used was **NetflixS**
For this part the jar file used was **movieratingaveragecalculator.jar** which is located at **s3://rutgers.monga.mrinal/movieratingaveragecalculator.jar** The Java program which was used for this is the **MovieRatingAverageCalculator.java** program. The detailed description is given below.

MovieRatingAverageCalculator.java program description

1. **The map method** - This method takes the second column of the input file as the id of the movie and the third column as the rating of the movie. Exception handling is done to take care of the situation when parsing a string from the tokenizer might not return a float value. The ratings are stored as float. This helps later during average calculation.
 2. **The reduce method** - This method has two variables of float type - sum and counter. sum stores the sum of all the ratings for a movie id. The counter stores the number of ratings. The average rating is arrived at by the formula sum/average for each movie id.
4. For the Netflix Large dataset the output is at **s3://rutgers.monga.mrinal/Netflix_large_output** the cluster used was **NetflixL**. For this the same jar file was used as in for Netflix_small.
 5. For sorting the movie average ratings the program used was SortingRating.java and the jar used was sortingrating.jar located at **s3://rutgers.monga.mrinal/sortingrating.jar**
Output is at - **s3://rutgers.monga.mrinal/SortedOutput** the cluster used was **Netflix LARGE Sorting Output**

SortingRating.java

1. The main method first takes the input provided and in a temporary folder writes the movie ids and the average ratings for the movies. Then it takes the input from the temporary folder and produces the required sorted output.
2. The SortingRating.java has **Map1, Reduce1, Map, Reduce, ReverseComparator**

classes.

3. In the main method of the `SortingRating.java` the first job is of calculating the average rating. For this the `Map1` and `Reduce1` classes are used. The map and reduce methods of these classes do the same job as that of the map and reduce methods of the `MovieAverageRatingCalculator.java`
4. The second job performed in the main method of the `SortingRating.java` is sorting. For this the `Reverse Comparator` class is used. This causes the data to be arranged in descending order of average movie rating.

SCREENSHOT 1 - CLUSTERS

The screenshot shows the AWS Elastic MapReduce Management Console. The top navigation bar includes the AWS logo, 'Services', 'Edit', and user information 'Mrinal Monga', 'Oregon', and 'Support'. The main header shows 'Elastic MapReduce' and 'Cluster List'. Below the header, there are buttons for 'Create cluster', 'View details', 'Clone', and 'Terminate'. A filter bar shows 'All clusters' and '20 clusters (all loaded)'. The main table lists the clusters with columns: Name, ID, Status, Creation time (UTC-5), Elapsed time, and Normalized instance hours. The 'NetflixS' cluster is highlighted in blue.

	Name	ID	Status	Creation time (UTC-5)	Elapsed time	Normalized instance hours
<input checked="" type="checkbox"/>	Netflix LARGE Sorting Output	j-2MG053JBO05CB	Terminated All steps completed	2014-12-19 15:55 (UTC-5)	8 minutes	6
<input type="checkbox"/>	Netflix LARGE Sorting Output	j-3EY4A18MD9H79	Terminated	2014-12-19 15:42 (UTC-5)	8 minutes	6
<input type="checkbox"/>	Netflix LARGE Sorting Output	j-14L2SV1TFSRMA	Terminated	2014-12-19 15:31 (UTC-5)	8 minutes	6
<input type="checkbox"/>	Netflix L Sorting	j-2F7914IT5KSBQ	Terminated All steps completed	2014-12-19 14:39 (UTC-5)	7 minutes	6
<input type="checkbox"/>	Netflix Small Sorting	j-2Z4EA76U59AK8	Terminated User request	2014-12-19 14:03 (UTC-5)	33 minutes	6
<input type="checkbox"/>	Word count	j-1B51QX7HJAOH2	Terminated All steps completed	2014-12-19 01:28 (UTC-5)	8 minutes	6
<input checked="" type="checkbox"/>	NetflixL	j-2OL6TJ0YEGJNY	Terminated All steps completed	2014-12-19 01:17 (UTC-5)	7 minutes	6
<input checked="" type="checkbox"/>	NetflixS	j-3B6GMMIWU8FC3	Terminated All steps completed	2014-12-19 01:13 (UTC-5)	8 minutes	6
<input checked="" type="checkbox"/>	WordCountTest	j-2HINFLD1OKEL1	Terminated All steps completed	2014-12-19 01:10 (UTC-5)	8 minutes	6
<input type="checkbox"/>	Jai Mata Di	j-GLVN5F5WQH34	Terminated User request	2014-12-19 00:46 (UTC-5)	18 minutes	6
<input type="checkbox"/>	Jai Mata Di	j-1G2RGII4COH80	Terminated User request	2014-12-19 00:28 (UTC-5)	16 minutes	6
<input type="checkbox"/>	Jai Mata Di	i-G3U56OXYTNRC	Terminated	2014-12-19 00:07 (UTC-5)	12 minutes	3

© 2008 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#) [Feedback](#)

SCREENSHOT - BUCKET AND DIRECTORIES

console.aws.amazon.com

AWS Elastic MapReduce Management ConsoleS3 Management Console

AWS

Services

Edit

Mrinal Monga

Global

Support

Upload

Create Folder

Actions

None

Properties

Transfers

All Buckets / rutgers.monga.mrinal

	Name	Storage Class	Size	Last Modified
<input checked="" type="checkbox"/>	NetfliX_large_Log	--	--	--
<input checked="" type="checkbox"/>	NetfliX_large_output	--	--	--
<input checked="" type="checkbox"/>	NetfliX_small_Log	--	--	--
<input checked="" type="checkbox"/>	NetfliX_small_output	--	--	--
<input checked="" type="checkbox"/>	SortedOutput	--	--	--
<input checked="" type="checkbox"/>	TempFolder	--	--	--
<input type="checkbox"/>	WordCount	--	--	--
<input checked="" type="checkbox"/>	WordCountTestOutput	--	--	--
<input checked="" type="checkbox"/>	WordCountTestOutputLog	--	--	--
<input type="checkbox"/>	WordCount_log	--	--	--
<input type="checkbox"/>	assignment	--	--	--
<input type="checkbox"/>	log	--	--	--
<input checked="" type="checkbox"/>	movieaverageratingcalculator.jar	Standard	3.4 KB	Fri Dec 19 00:06:59 GMT-500 2014
<input type="checkbox"/>	output	--	--	--
<input type="checkbox"/>	output2	--	--	--
<input checked="" type="checkbox"/>	sortingrating.jar	Standard	6 KB	Fri Dec 19 15:26:32 GMT-500 2014
<input type="checkbox"/>	temp	--	--	--
<input type="checkbox"/>	wordcount	--	--	--

© 2008 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved.

[Privacy Policy](#)

[Terms of Use](#)

Feedback

PERMISSIONS

The screenshot shows the AWS S3 Management Console interface. On the left, under 'All Buckets', the bucket 'rutgers.monga.mrinal' is listed. The main panel displays the 'Bucket: rutgers.monga.mrinal' details. The 'Permissions' section is expanded, showing a table of grants:

Grantee	Permissions	Actions
mrinalmonga9	<input checked="" type="checkbox"/> List <input checked="" type="checkbox"/> Upload/Delete <input checked="" type="checkbox"/> View Permissions	<input checked="" type="checkbox"/>
cyramel	<input checked="" type="checkbox"/> List <input type="checkbox"/> Upload/Delete <input type="checkbox"/> View Permissions	<input type="checkbox"/>

Below the table, there are links to 'Add more permissions', 'Edit bucket policy', and 'Add CORS Configuration'. At the bottom, there are buttons for 'Save' and 'Cancel'.

© 2008 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#) [Feedback](#)

The screenshot shows the same AWS S3 Management Console interface, but with the 'Bucket Policy Editor' dialog box open. The dialog box is titled 'Bucket Policy Editor' and contains the following text:

Policy for Bucket : "rutgers.monga.mrinal"

Add a new policy or edit an existing bucket policy in the text area below.

```
{
  "Version": "2008-10-17",
  "Statement": [
    {
      "Sid": "AllowArmelieToRead",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::880387136363:root"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::rutgers.monga.mrinal/**"
    }
  ]
}
```

At the bottom of the dialog box, there are links to 'AWS Policy Generator' and 'Sample Bucket Policies', and buttons for 'Save', 'Delete', and 'Close'.

© 2008 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#) [Feedback](#)

