USECASE :- MOVIE RATING

Problem Statement:

What are the movie titles that the user has rated?

How many times a movie has been rated by the user?

In question 2 above, what is the average rating given for a movie?

Dataset:

Movies :

```
movieId,title,genres
1,Toy Story (1995),Adventure|Animation|Children|Comedy|Fantasy
2,Jumanji (1995),Adventure|Children|Fantasy
3,Grumpier Old Men (1995),Comedy|Romance
4,Waiting to Exhale (1995),Comedy|Drama|Romance
5,Father of the Bride Part II (1995),Comedy
6,Heat (1995),Action|Crime|Thriller
7,Sabrina (1995),Comedy|Romance
8,Tom and Huck (1995),Adventure|Children
9,Sudden Death (1995),Action
10,GoldenEye (1995),Action|Adventure|Thriller
11,"American President, The (1995)",Comedy|Drama|Romance
12,Dracula: Dead and Loving It (1995),Comedy|Horror
13,Balto (1995),Adventure|Animation|Children
14,Nixon (1995),Drama
15,Cutthroat Island (1995),Action|Adventure|Romance
16,Casino (1995),Crime|Drama
17,Sense and Sensibility (1995),Drama|Romance
18,Four Rooms (1995),Comedy
19,Ace Ventura: When Nature Calls (1995),Comedy
20,Money Train (1995),Action|Comedy|Crime|Drama|Thriller
21,Get Shorty (1995),Comedy|Crime|Thriller
22,Copycat (1995),Crime|Drama|Horror|Mystery|Thriller
23,Assassins (1995),Action|Crime|Thriller
24,Powder (1995),Drama|Sci-Fi
25,Leaving Las Vegas (1995),Drama|Romance
26,Othello (1995),Drama
27,Now and Then (1995),Children|Drama
28,Persuasion (1995),Drama|Romance
29,"City of Lost Children, The (Cité des enfants perdus, La) (1995)",Adventure|Drama|Fantasy|Mystery|Sci-Fi
30,Shanghai Triad (Yao a yao yao dao waipo qiao) (1995),Crime|Drama
31,Dangerous Minds (1995),Drama
32,Twelve Monkeys (a.k.a. 12 Monkeys) (1995),Mystery|Sci-Fi|Thriller
34,Babe (1995),Children|Drama
35,Carrington (1995),Drama|Romance
36,Dead Man Walking (1995),Crime|Drama
37,Across the Sea of Time (1995),Documentary|IMAX
```

Ratings:

```
userId,movieId,rating,timestamp
1,31,2.5,1260759144
1,1029,3.0,1260759179
1,1061,3.0,1260759182
1,1129,2.0,1260759185
1,1172,4.0,1260759205
1,1263,2.0,1260759151
1,1287,2.0,1260759187
1,1293,2.0,1260759148
1,1339,3.5,1260759125
1,1343,2.0,1260759131
1,1371,2.5,1260759135
1,1405,1.0,1260759203
1,1953,4.0,1260759191
1,2105,4.0,1260759139
1,2150,3.0,1260759194
1,2193,2.0,1260759198
1,2294,2.0,1260759108
1,2455,2.5,1260759113
1,2968,1.0,1260759200
1,3671,3.0,1260759117
2,10,4.0,835355493
2,17,5.0,835355681
2,39,5.0,835355604
2,47,4.0,835355552
2,50,4.0,835355586
2,52,3.0,835356031
2,62,3.0,835355749
2,110,4.0,835355532
2,144,3.0,835356016
2,150,5.0,835355395
2,153,4.0,835355441
2,161,3.0,835355493
2,165,3.0,835355441
2,168,3.0,835355710
2,185,3.0,835355511
2,186,3.0,835355664
```

Solution:

DriverClass:

```
0  public class CaseStudyIUseCasesDriver {
1
2⊖     @SuppressWarnings("deprecation")
3      public static void main(String[] args) throws Exception {
4      if (args.length != 3) {
5        System.err.println("Usage: CaseStudyIUseCase2Driver <input path1> <input path2> <output path>");
6        System.exit(-1);
7      }
8
9      //Job Related Configurations
0      Configuration conf = new Configuration();
1      Job job = new Job(conf, "CaseStudyIUseCase2Driver");
2      job.setJarByClass(CaseStudyIUseCasesDriver.class);
3
4      job.setNumReduceTasks(2);
5
6      //Since there are multiple input, there is a slightly different way of specifying input path, input format
7      MultipleInputs.addInputPath(job, new Path(args[0]),TextInputFormat.class, CaseStudyIUseCasesMoviesMapper.
8      MultipleInputs.addInputPath(job, new Path(args[1]),TextInputFormat.class, CaseStudyIUseCasesRatingsMapper.
9
0      //Set the reducer
1      job.setReducerClass(CaseStudyIUseCasesReducer.class);
2
3      //set the out path
4      Path outputPath = new Path(args[2]);
5      FileOutputFormat.setOutputPath(job, outputPath);
6      outputPath.getFileSystem(conf).delete(outputPath, true);
7
8      //set up the output key and value classes
9      job.setOutputKeyClass(Text.class);
0      job.setOutputValueClass(Text.class);
1
2      //execute the job
3      System.exit(job.waitForCompletion(true) ? 0 : 1);
```

Code :

```
import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.io.Text;

//import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;

import org.apache.hadoop.mapreduce.Job;

import org.apache.hadoop.mapreduce.lib.input.MultipleInputs;

import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;


public class CaseStudyIUseCasesDriver {


        @SuppressWarnings("deprecation")

        public static void main(String[] args) throws Exception {

   if (args.length != 3) {

      System.err.println("Usage: CaseStudyIUseCase2Driver <input path1> <input path2> <output path>");
```

```java
    System.exit(-1);

  }


        //Job Related Configurations

        Configuration conf = new Configuration();

        Job job = new Job(conf, "CaseStudyIUseCase2Driver");

        job.setJarByClass(CaseStudyIUseCasesDriver.class);


        job.setNumReduceTasks(2);


        //Since there are multiple input, there is a slightly different way of specifying input path,
input format and mapper

        MultipleInputs.addInputPath(job, new Path(args[0]),TextInputFormat.class,
CaseStudyIUseCasesMoviesMapper.class);

        MultipleInputs.addInputPath(job, new Path(args[1]),TextInputFormat.class,
CaseStudyIUseCasesRatingsMapper.class);


        //Set the reducer

        job.setReducerClass(CaseStudyIUseCasesReducer.class);


  //set the out path

        Path outputPath = new Path(args[2]);

        FileOutputFormat.setOutputPath(job, outputPath);

        outputPath.getFileSystem(conf).delete(outputPath, true);


  //set up the output key and value classes

  job.setOutputKeyClass(Text.class);

  job.setOutputValueClass(Text.class);


  //execute the job

  System.exit(job.waitForCompletion(true) ? 0 : 1);

  }
```

}

Movies Mapper Class:

```
import java.io.IOException;
public class CaseStudyIUseCasesMoviesMapper extends
        Mapper<LongWritable, Text, Text, Text> {

    public void map(LongWritable key, Text value, Context context)
            throws IOException, InterruptedException {

        try {
            if (key.get() == 0 && value.toString().contains("movieId")){
                return;
            } else {
                String record = value.toString();
                String[] parts = record.split(",");
                context.write(new Text(parts[0]), new Text("movies\t" + parts[1]));
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

import java.io.IOException;

import org.apache.hadoop.io.LongWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapreduce.Mapper;

public class CaseStudyIUseCasesMoviesMapper extends

Mapper<LongWritable, Text, Text, Text> {

public void map(LongWritable key, Text value, Context context)

throws IOException, InterruptedException {

try {

if (key.get() == 0 && value.toString().contains("movieId")){

```java
                return;
        } else {
            String record = value.toString();
                        String[] parts = record.split(",");
                        context.write(new Text(parts[0]), new Text("movies\t" + parts[1]));
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
        }
}
```

Rating Mapper Class:

```java
import java.io.IOException;

import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

public class CaseStudyIUseCasesRatingsMapper extends
            Mapper<LongWritable, Text, Text, Text> {

    public void map(LongWritable key, Text value, Context context)
                throws IOException, InterruptedException {

        try {
            if (key.get() == 0 && value.toString().contains("userId")){
                return;
            } else {
                String record = value.toString();
                String[] parts = record.split(",");
                context.write(new Text(parts[1]), new Text("ratings\t" + parts[2]));
            }
        } catch (Exception e) {
            e.printStackTrace();
        }

    }
}
```

```java
import java.io.IOException;

import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

public class CaseStudyIUseCasesRatingsMapper extends
                    Mapper<LongWritable, Text, Text, Text> {

    public void map(LongWritable key, Text value, Context context)
                        throws IOException, InterruptedException {

                try {
```

```java
                    if (key.get() == 0 && value.toString().contains("userId")){
                        return;
                    } else {
                            String record = value.toString();
                            String[] parts = record.split(",");
                            context.write(new Text(parts[1]), new Text("ratings\t"
+ parts[2]));
                    }
                } catch (Exception e) {
                    e.printStackTrace();
                }

        }
}
```

Reducer Class:

```java
1  import java.io.IOException;
2
3  import org.apache.hadoop.io.Text;
4  import org.apache.hadoop.mapreduce.Reducer;
5  |
6  public class CaseStudyIUseCasesReducer extends
7             Reducer<Text, Text, Text, Text> {
8
9         public void reduce(Text key, Iterable<Text> values, Context context)
0                   throws IOException, InterruptedException {
1               String titles = "";
2               double total = 0.0;
3               int count = 0;
4               System.out.println("Text Key      =>"+key.toString());
5               for (Text t : values) {
6                   String parts[] = t.toString().split("\t");
7                   System.out.println("Text values =>"+t.toString());
8                   if (parts[0].equals("ratings")) {
9                       count++;
0                       String rating = parts[1].trim();
1                       System.out.println("Rating is =>"+rating);
2                       total += Double.parseDouble(rating);
3                   } else if (parts[0].equals("movies")) {
4                       titles = parts[1];
5                   }
6               }
7
8               double average = total / count;
9               String str = String.format("%d\t%f", count, average);
0               //String str = String.format("%d", count);
1
2               context.write(new Text(titles), new Text(str));
3          }
4      }
```

```java
import java.io.IOException;

import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

public class CaseStudyIUseCasesReducer extends
                Reducer<Text, Text, Text, Text> {
```

```java
    public void reduce(Text key, Iterable<Text> values, Context context)
            throws IOException, InterruptedException {
        String titles = "";
        double total = 0.0;
        int count = 0;
        System.out.println("Text Key     =>"+key.toString());
        for (Text t : values) {
                String parts[] = t.toString().split("\t");
                System.out.println("Text values =>"+t.toString());
                if (parts[0].equals("ratings")) {
                        count++;
                        String rating = parts[1].trim();
                        System.out.println("Rating is =>"+rating);
                        total += Double.parseDouble(rating);
                } else if (parts[0].equals("movies")) {
                        titles = parts[1];
                }
        }

        double average = total / count;
        String str = String.format("%d\t%f", count, average);
    //String str = String.format("%d", count);

        context.write(new Text(titles), new Text(str));
    }
}
```

Output:

```
drwxr-xr-x   - acadgild supergroup          0 2018-07-01 11:22 /user
drwxr-xr-x   - acadgild supergroup          0 2018-09-27 11:21 /wordcount
[acadgild@localhost Desktop]$ hadoop fs -ls /hadoopdata
18/10/02 11:44:26 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform..
. using builtin-java classes where applicable
Found 7 items
drwxr-xr-x   - acadgild supergroup          0 2018-09-23 18:37 /hadoopdata/casestudy
drwxr-xr-x   - acadgild supergroup          0 2018-08-22 00:50 /hadoopdata/flume
drwxr-xr-x   - acadgild supergroup          0 2018-07-15 17:22 /hadoopdata/hive
drwxr-xr-x   - acadgild supergroup          0 2018-09-23 19:24 /hadoopdata/out
drwxr-xr-x   - acadgild supergroup          0 2018-08-19 15:37 /hadoopdata/pig
drwxr-xr-x   - acadgild supergroup          0 2018-06-30 08:58 /hadoopdata/wordcount
drwxr-xr-x   - acadgild supergroup          0 2018-09-13 23:25 /hadoopdata/wordmean
[acadgild@localhost Desktop]$ hadoop fs -ls /hadoopdata/casestudy
18/10/02 11:44:39 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform..
. using builtin-java classes where applicable
Found 2 items
-rw-r--r--   1 acadgild supergroup     458390 2018-09-23 18:36 /hadoopdata/casestudy/movies_small.csv
-rw-r--r--   1 acadgild supergroup    2438266 2018-09-23 18:37 /hadoopdata/casestudy/ratings_small.csv
[acadgild@localhost Desktop]$ hadoop fs -ls /hadoopdata/out
18/10/02 11:44:55 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform..
. using builtin-java classes where applicable
[acadgild@localhost Desktop]$ pwd
```

```
        at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
[acadgild@localhost Desktop]$ hadoop jar casestudy.jar com.acadgild.hadoop.casestudy.CaseStudyIUseCase
sDriver /hadoopdata/casestudy/movies_small.csv /hadoopdata/casestudy/ratings_small.csv /hadoopdata/out
```