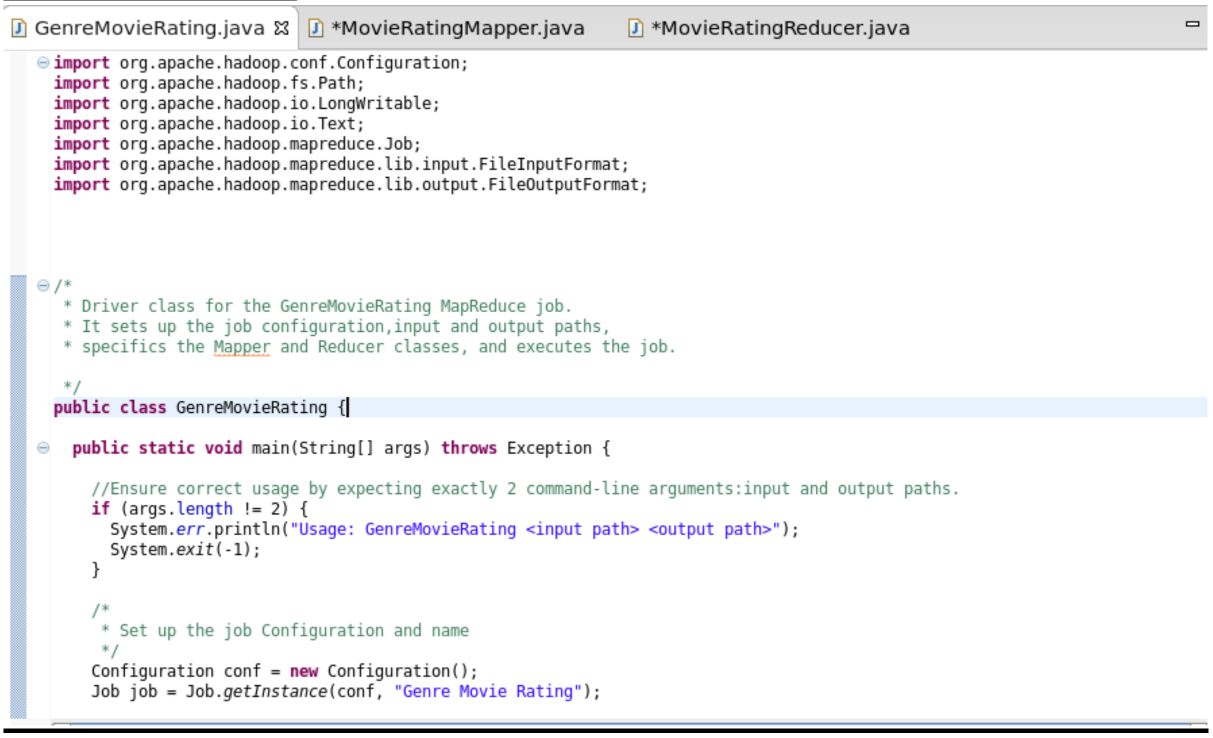
Assignment 1: MapReduce Program Application

COS20028 – Big data Architecture and Applications

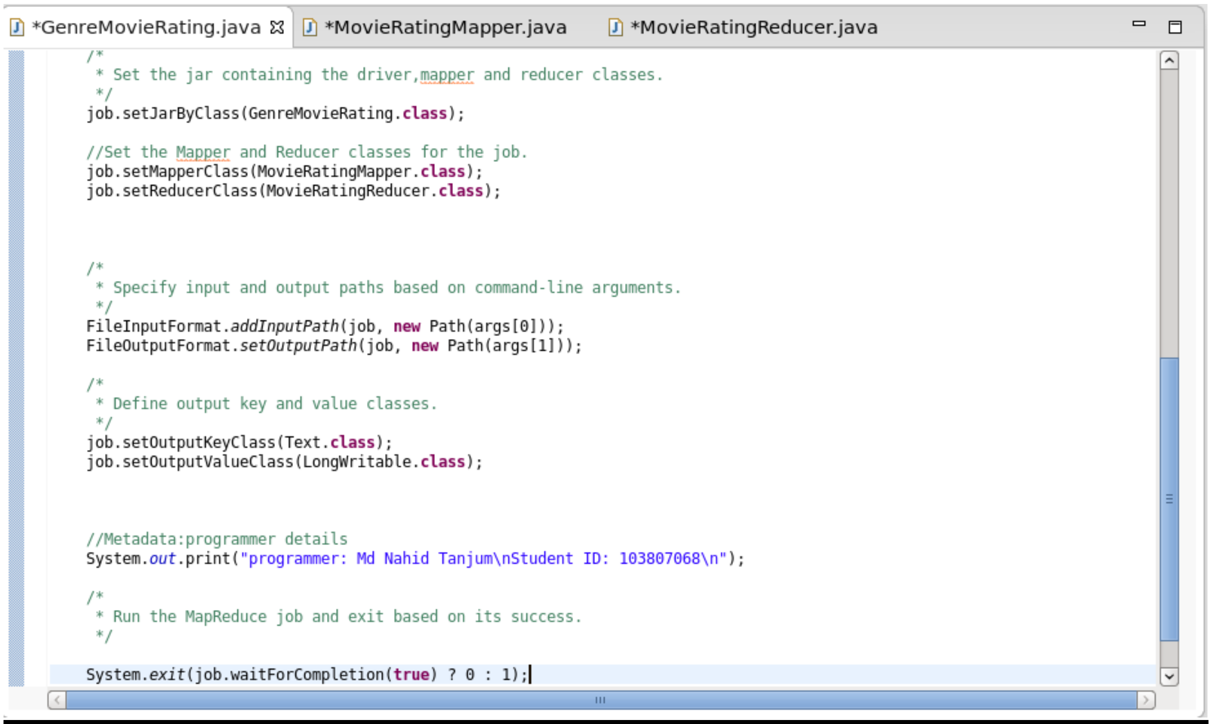
Semester 2, 2023

Name: MD NAHID TANJUM
Student ID: 10807068
Date: August 29, 2023

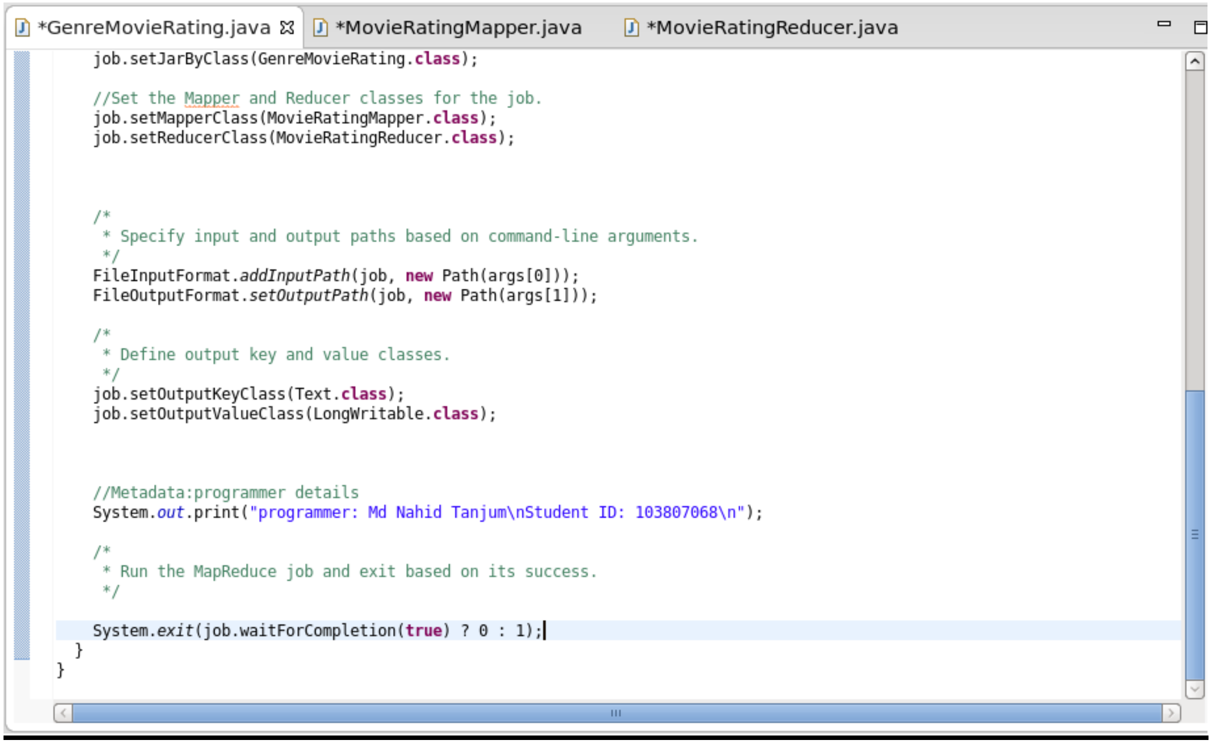
Driver Code



```java
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;



/*
 * Driver class for the GenreMovieRating MapReduce job.
 * It sets up the job configuration,input and output paths,
 * specifics the Mapper and Reducer classes, and executes the job.

 */
public class GenreMovieRating {

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

    //Ensure correct usage by expecting exactly 2 command-line arguments:input and output paths.
    if (args.length != 2) {
      System.err.println("Usage: GenreMovieRating <input path> <output path>");
      System.exit(-1);
    }

    /*
     * Set up the job Configuration and name
     */
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "Genre Movie Rating");
```

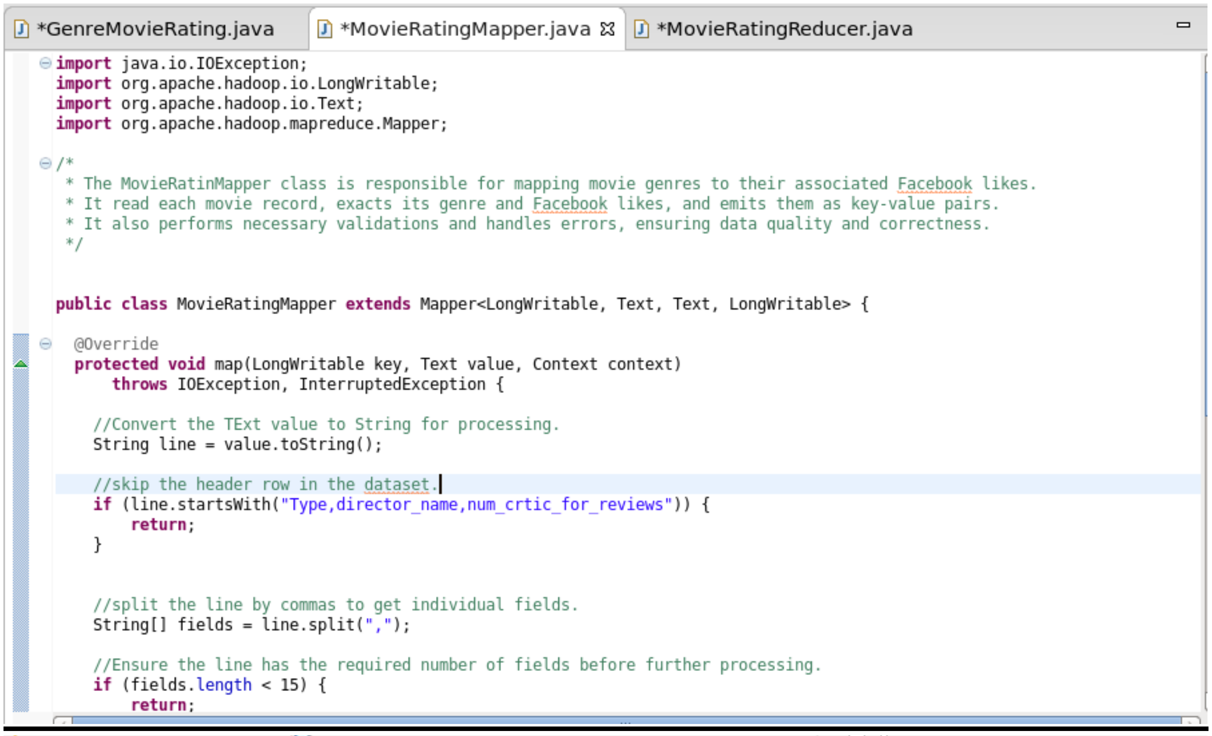


```
    /*
     * Set the jar containing the driver,mapper and reducer classes.
     */
    job.setJarByClass(GenreMovieRating.class);

    //Set the Mapper and Reducer classes for the job.
    job.setMapperClass(MovieRatingMapper.class);
    job.setReducerClass(MovieRatingReducer.class);



    /*
     * Specify input and output paths based on command-line arguments.
     */
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));

    /*
     * Define output key and value classes.
     */
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(LongWritable.class);



    //Metadata:programmer details
    System.out.print("programmer: Md Nahid Tanjum\nStudent ID: 103807068\n");

    /*
     * Run the MapReduce job and exit based on its success.
     */

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

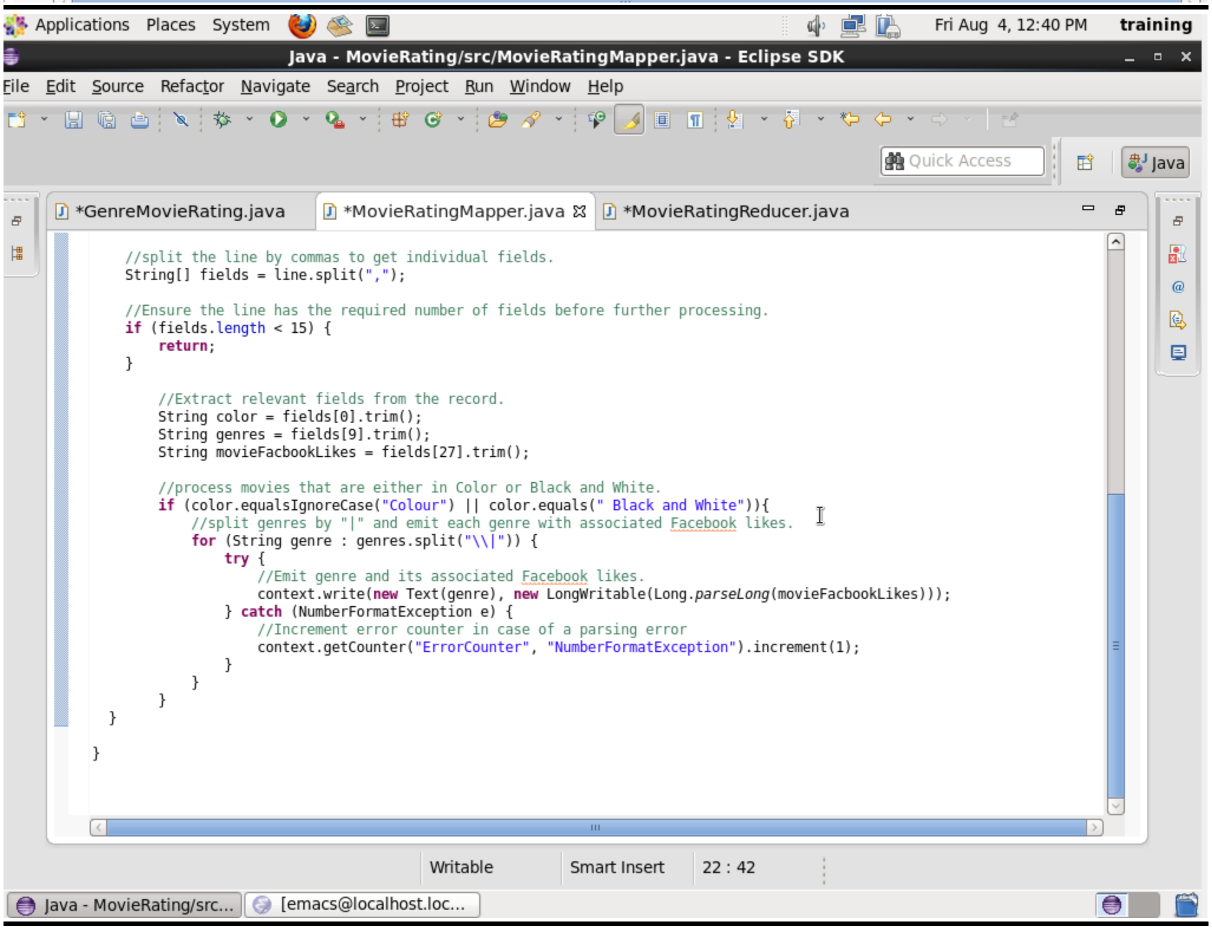


```
    job.setJarByClass(GenreMovieRating.class);

    //Set the Mapper and Reducer classes for the job.
    job.setMapperClass(MovieRatingMapper.class);
    job.setReducerClass(MovieRatingReducer.class);



    /*
     * Specify input and output paths based on command-line arguments.
     */
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));

    /*
     * Define output key and value classes.
     */
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(LongWritable.class);



    //Metadata:programmer details
    System.out.print("programmer: Md Nahid Tanjum\nStudent ID: 103807068\n");

    /*
     * Run the MapReduce job and exit based on its success.
     */

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

Mapper Code

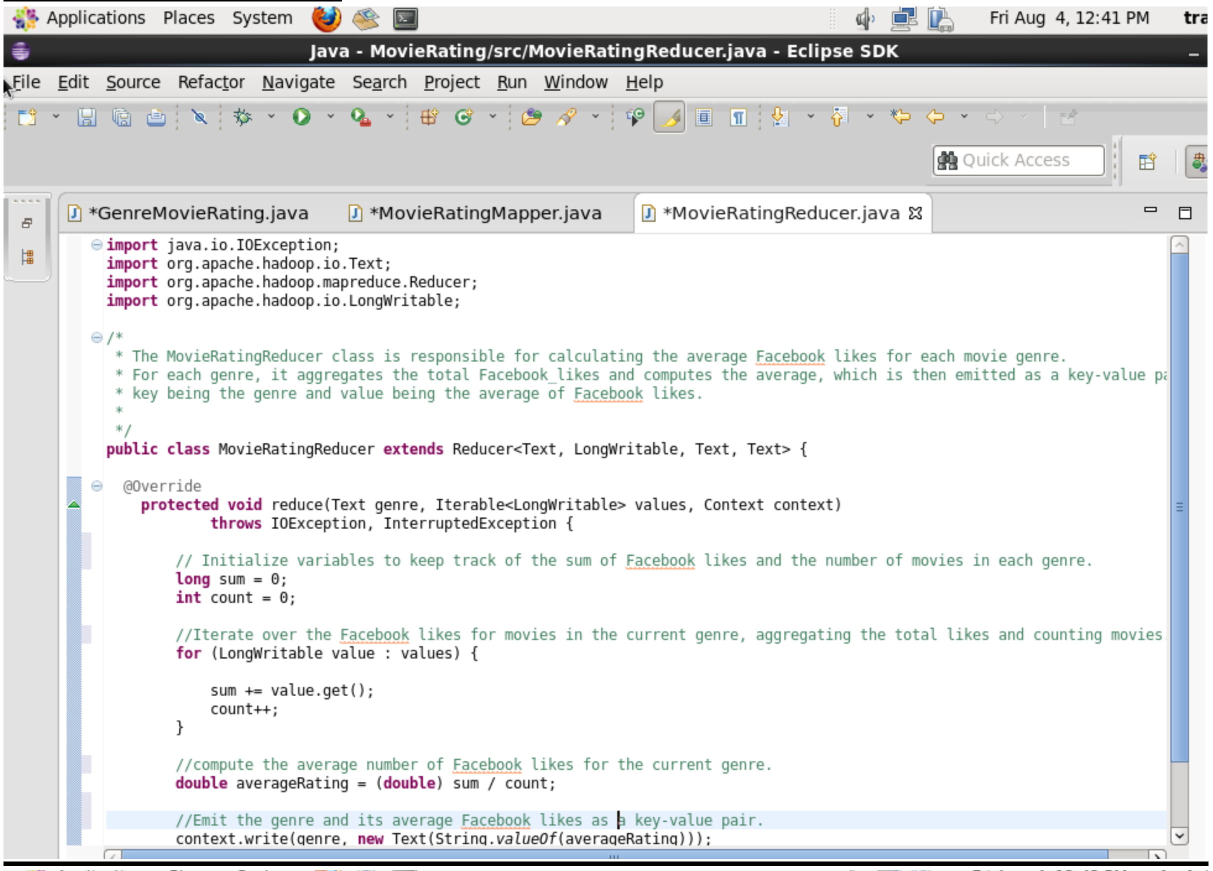


```java
import java.io.IOException;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

/*
 * The MovieRatinMapper class is responsible for mapping movie genres to their associated Facebook likes.
 * It read each movie record, exacts its genre and Facebook likes, and emits them as key-value pairs.
 * It also performs necessary validations and handles errors, ensuring data quality and correctness.
 */


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

    @Override
    protected void map(LongWritable key, Text value, Context context)
        throws IOException, InterruptedException {

      //Convert the TExt value to String for processing.
      String line = value.toString();

      //skip the header row in the dataset.
      if (line.startsWith("Type,director_name,num_crtic_for_reviews")) {
          return;
      }


      //split the line by commas to get individual fields.
      String[] fields = line.split(",");

      //Ensure the line has the required number of fields before further processing.
      if (fields.length < 15) {
          return;
```

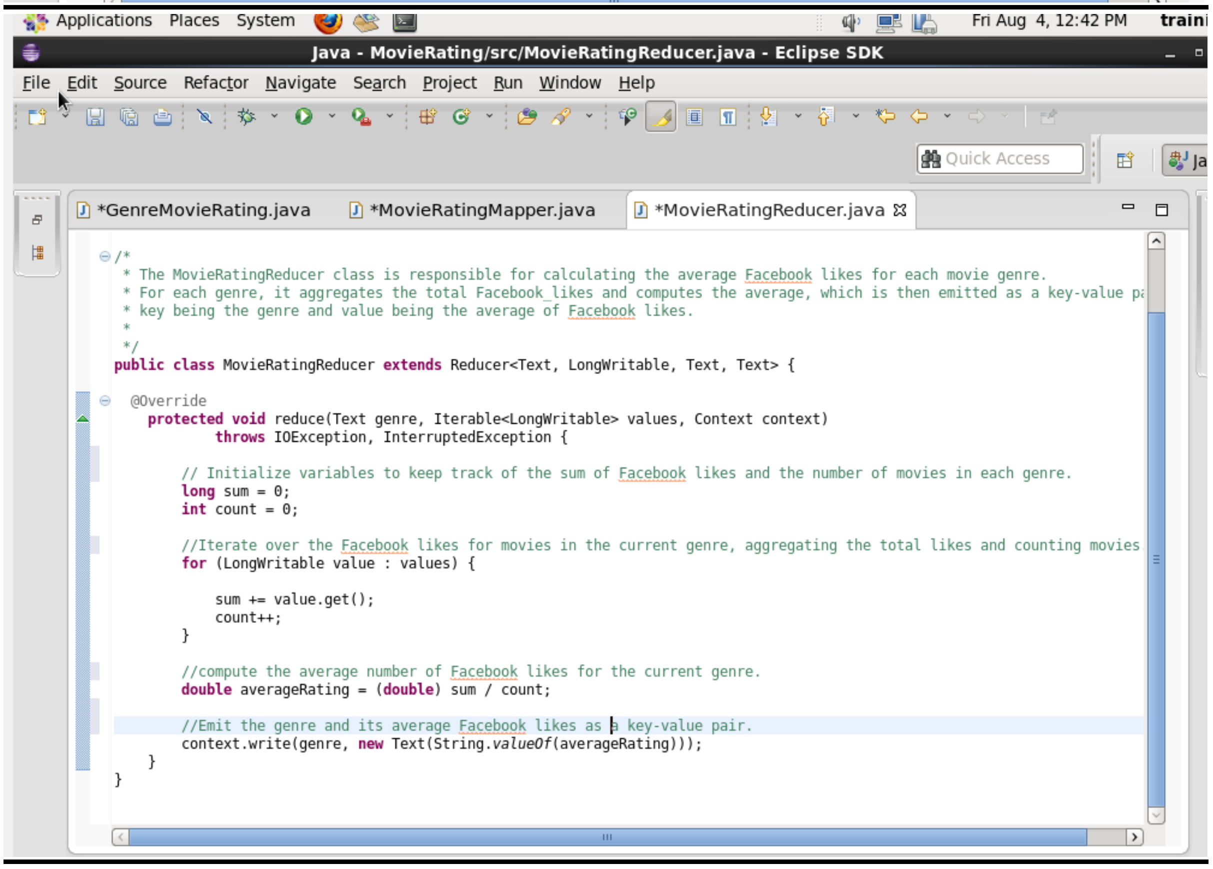


```java
      //split the line by commas to get individual fields.
      String[] fields = line.split(",");

      //Ensure the line has the required number of fields before further processing.
      if (fields.length < 15) {
          return;
      }

          //Extract relevant fields from the record.
          String color = fields[0].trim();
          String genres = fields[9].trim();
          String movieFacbookLikes = fields[27].trim();

          //process movies that are either in Color or Black and White.
          if (color.equalsIgnoreCase("Colour") || color.equals(" Black and White")){
              //split genres by "|" and emit each genre with associated Facebook likes.
              for (String genre : genres.split("\\|")) {
                  try {
                      //Emit genre and its associated Facebook likes.
                      context.write(new Text(genre), new LongWritable(Long.parseLong(movieFacbookLikes)));
                  } catch (NumberFormatException e) {
                      //Increment error counter in case of a parsing error
                      context.getCounter("ErrorCounter", "NumberFormatException").increment(1);
                  }
              }
          }
    }

  }
```

Reducer Code

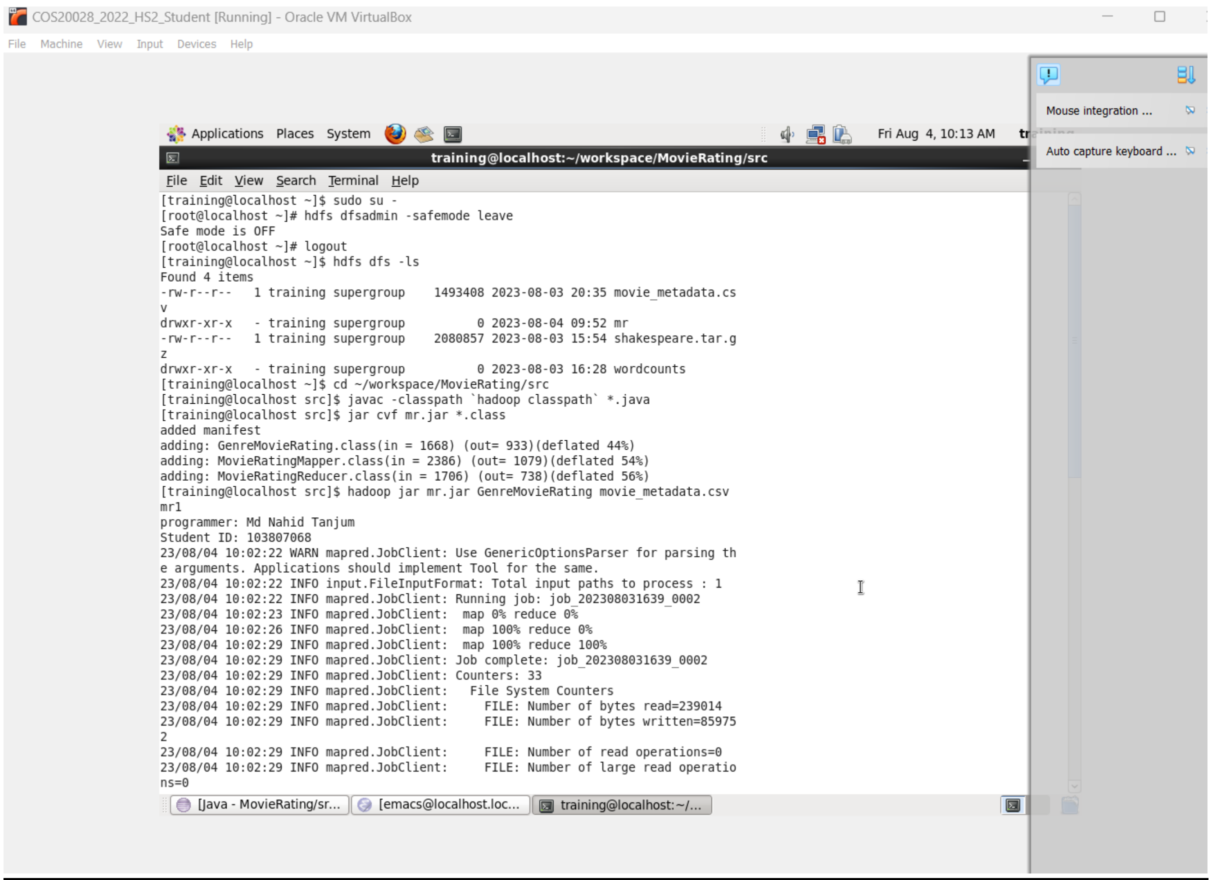

Applications Places System
Fri Aug 4, 12:41 PM
Java - MovieRating/src/MovieRatingReducer.java - Eclipse SDK
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
*GenreMovieRating.java
*MovieRatingMapper.java
*MovieRatingReducer.java

```java
import java.io.IOException;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.io.LongWritable;

/*
 * The MovieRatingReducer class is responsible for calculating the average Facebook likes for each movie genre.
 * For each genre, it aggregates the total Facebook_likes and computes the average, which is then emitted as a key-value pa
 * key being the genre and value being the average of Facebook likes.
 *
 */
public class MovieRatingReducer extends Reducer<Text, LongWritable, Text, Text> {

    @Override
    protected void reduce(Text genre, Iterable<LongWritable> values, Context context)
            throws IOException, InterruptedException {

        // Initialize variables to keep track of the sum of Facebook likes and the number of movies in each genre.
        long sum = 0;
        int count = 0;

        //Iterate over the Facebook likes for movies in the current genre, aggregating the total likes and counting movies
        for (LongWritable value : values) {

            sum += value.get();
            count++;
        }

        //compute the average number of Facebook likes for the current genre.
        double averageRating = (double) sum / count;

        //Emit the genre and its average Facebook likes as a key-value pair.
        context.write(genre, new Text(String.valueOf(averageRating)));
```

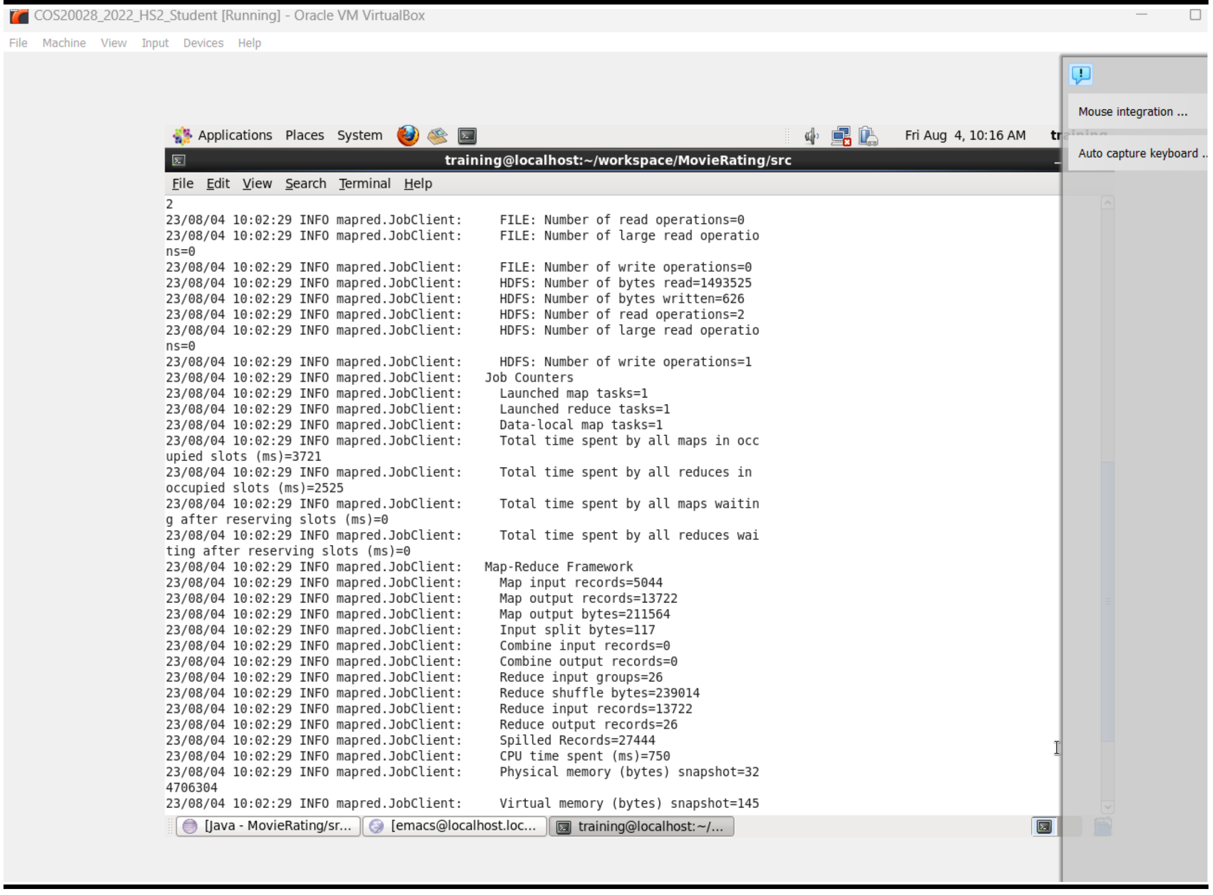

Applications Places System
Fri Aug 4, 12:42 PM
Java - MovieRating/src/MovieRatingReducer.java - Eclipse SDK
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
*GenreMovieRating.java
*MovieRatingMapper.java
*MovieRatingReducer.java

```java
/*
 * The MovieRatingReducer class is responsible for calculating the average Facebook likes for each movie genre.
 * For each genre, it aggregates the total Facebook_likes and computes the average, which is then emitted as a key-value pa
 * key being the genre and value being the average of Facebook likes.
 *
 */
public class MovieRatingReducer extends Reducer<Text, LongWritable, Text, Text> {

    @Override
    protected void reduce(Text genre, Iterable<LongWritable> values, Context context)
            throws IOException, InterruptedException {

        // Initialize variables to keep track of the sum of Facebook likes and the number of movies in each genre.
        long sum = 0;
        int count = 0;

        //Iterate over the Facebook likes for movies in the current genre, aggregating the total likes and counting movies
        for (LongWritable value : values) {

            sum += value.get();
            count++;
        }

        //compute the average number of Facebook likes for the current genre.
        double averageRating = (double) sum / count;

        //Emit the genre and its average Facebook likes as a key-value pair.
        context.write(genre, new Text(String.valueOf(averageRating)));
    }
}
```

Commands and the Execution Screenshot

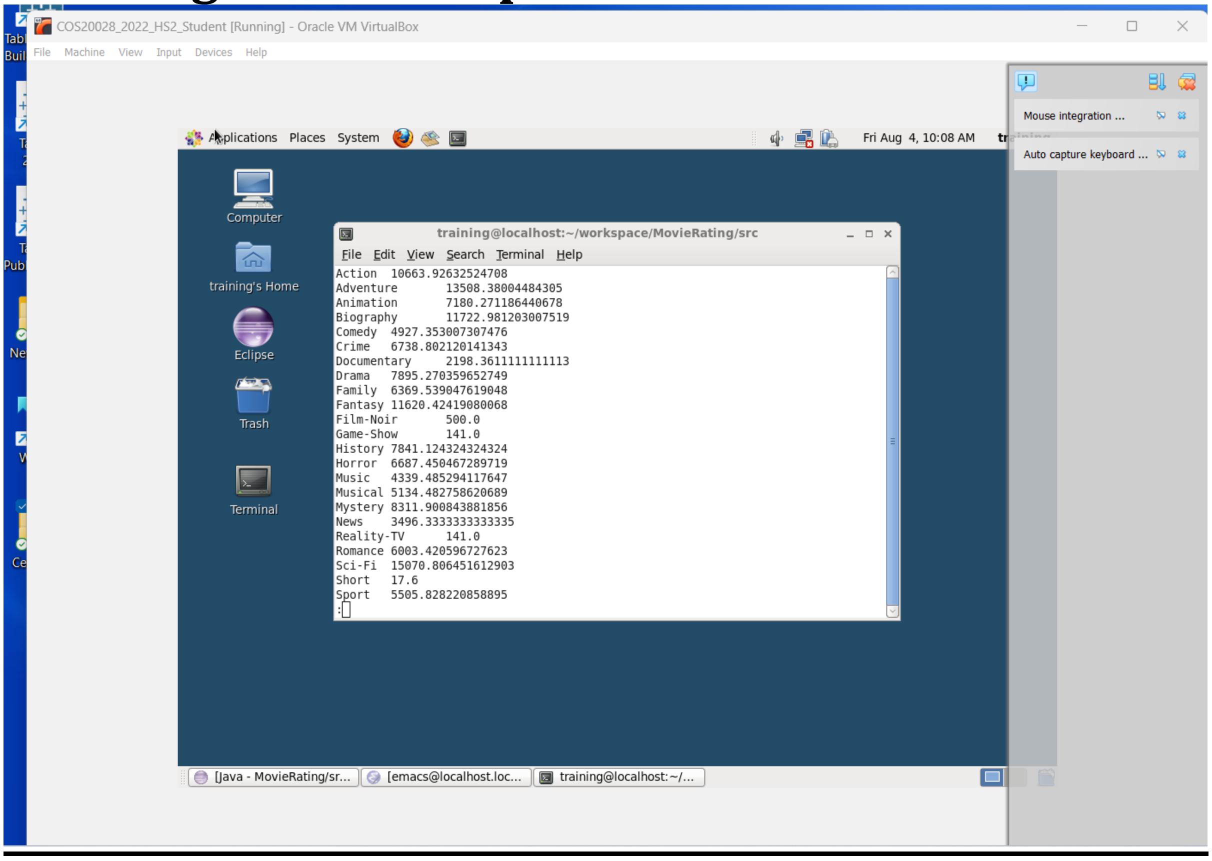

COS20028_2022_HS2_Student [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Mouse integration ...
Auto capture keyboard ...
Applications Places System
Fri Aug 4, 10:13 AM
training@localhost:~/workspace/MovieRating/src
File Edit View Search Terminal Help
[Java - MovieRating/sr...
[emacs@localhost.loc...
training@localhost:~/...

```
[training@localhost ~]$ sudo su -
[root@localhost ~]# hdfs dfsadmin -safemode leave
Safe mode is OFF
[root@localhost ~]# logout
[training@localhost ~]$ hdfs dfs -ls
Found 4 items
-rw-r--r--   1 training supergroup    1493408 2023-08-03 20:35 movie_metadata.cs
v
drwxr-xr-x   - training supergroup          0 2023-08-04 09:52 mr
-rw-r--r--   1 training supergroup    2080857 2023-08-03 15:54 shakespeare.tar.g
z
drwxr-xr-x   - training supergroup          0 2023-08-03 16:28 wordcounts
[training@localhost ~]$ cd ~/workspace/MovieRating/src
[training@localhost src]$ javac -classpath `hadoop classpath` *.java
[training@localhost src]$ jar cvf mr.jar *.class
added manifest
adding: GenreMovieRating.class(in = 1668) (out= 933)(deflated 44%)
adding: MovieRatingMapper.class(in = 2386) (out= 1079)(deflated 54%)
adding: MovieRatingReducer.class(in = 1706) (out= 738)(deflated 56%)
[training@localhost src]$ hadoop jar mr.jar GenreMovieRating movie_metadata.csv
mr1
programmer: Md Nahid Tanjum
Student ID: 103807068
23/08/04 10:02:22 WARN mapred.JobClient: Use GenericOptionsParser for parsing th
e arguments. Applications should implement Tool for the same.
23/08/04 10:02:22 INFO input.FileInputFormat: Total input paths to process : 1
23/08/04 10:02:22 INFO mapred.JobClient: Running job: job_202308031639_0002
23/08/04 10:02:23 INFO mapred.JobClient:  map 0% reduce 0%
23/08/04 10:02:26 INFO mapred.JobClient:  map 100% reduce 0%
23/08/04 10:02:29 INFO mapred.JobClient:  map 100% reduce 100%
23/08/04 10:02:29 INFO mapred.JobClient: Job complete: job_202308031639_0002
23/08/04 10:02:29 INFO mapred.JobClient: Counters: 33
23/08/04 10:02:29 INFO mapred.JobClient:   File System Counters
23/08/04 10:02:29 INFO mapred.JobClient:     FILE: Number of bytes read=239014
23/08/04 10:02:29 INFO mapred.JobClient:     FILE: Number of bytes written=85975
2
23/08/04 10:02:29 INFO mapred.JobClient:     FILE: Number of read operations=0
23/08/04 10:02:29 INFO mapred.JobClient:     FILE: Number of large read operatio
ns=0
```


COS20028_2022_HS2_Student [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Mouse integration ...
Auto capture keyboard ...
Applications Places System
Fri Aug 4, 10:16 AM
training@localhost:~/workspace/MovieRating/src
File Edit View Search Terminal Help
[Java - MovieRating/sr...
[emacs@localhost.loc...
training@localhost:~/...

```
2
23/08/04 10:02:29 INFO mapred.JobClient:     FILE: Number of read operations=0
23/08/04 10:02:29 INFO mapred.JobClient:     FILE: Number of large read operatio
ns=0
23/08/04 10:02:29 INFO mapred.JobClient:     FILE: Number of write operations=0
23/08/04 10:02:29 INFO mapred.JobClient:     HDFS: Number of bytes read=1493525
23/08/04 10:02:29 INFO mapred.JobClient:     HDFS: Number of bytes written=626
23/08/04 10:02:29 INFO mapred.JobClient:     HDFS: Number of read operations=2
23/08/04 10:02:29 INFO mapred.JobClient:     HDFS: Number of large read operatio
ns=0
23/08/04 10:02:29 INFO mapred.JobClient:     HDFS: Number of write operations=1
23/08/04 10:02:29 INFO mapred.JobClient:   Job Counters
23/08/04 10:02:29 INFO mapred.JobClient:     Launched map tasks=1
23/08/04 10:02:29 INFO mapred.JobClient:     Launched reduce tasks=1
23/08/04 10:02:29 INFO mapred.JobClient:     Data-local map tasks=1
23/08/04 10:02:29 INFO mapred.JobClient:     Total time spent by all maps in occ
upied slots (ms)=3721
23/08/04 10:02:29 INFO mapred.JobClient:     Total time spent by all reduces in
occupied slots (ms)=2525
23/08/04 10:02:29 INFO mapred.JobClient:     Total time spent by all maps waitin
g after reserving slots (ms)=0
23/08/04 10:02:29 INFO mapred.JobClient:     Total time spent by all reduces wai
ting after reserving slots (ms)=0
23/08/04 10:02:29 INFO mapred.JobClient:   Map-Reduce Framework
23/08/04 10:02:29 INFO mapred.JobClient:     Map input records=5044
23/08/04 10:02:29 INFO mapred.JobClient:     Map output records=13722
23/08/04 10:02:29 INFO mapred.JobClient:     Map output bytes=211564
23/08/04 10:02:29 INFO mapred.JobClient:     Input split bytes=117
23/08/04 10:02:29 INFO mapred.JobClient:     Combine input records=0
23/08/04 10:02:29 INFO mapred.JobClient:     Combine output records=0
23/08/04 10:02:29 INFO mapred.JobClient:     Reduce input groups=26
23/08/04 10:02:29 INFO mapred.JobClient:     Reduce shuffle bytes=239014
23/08/04 10:02:29 INFO mapred.JobClient:     Reduce input records=13722
23/08/04 10:02:29 INFO mapred.JobClient:     Reduce output records=26
23/08/04 10:02:29 INFO mapred.JobClient:     Spilled Records=27444
23/08/04 10:02:29 INFO mapred.JobClient:     CPU time spent (ms)=750
23/08/04 10:02:29 INFO mapred.JobClient:     Physical memory (bytes) snapshot=32
4706304
23/08/04 10:02:29 INFO mapred.JobClient:     Virtual memory (bytes) snapshot=145
```

First Page of the Output File Screenshot


COS20028_2022_HS2_Student [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places System
Fri Aug 4, 10:08 AM
Mouse integration ...
Auto capture keyboard ...
Computer
training's Home
Eclipse
Trash
Terminal
training@localhost:~/workspace/MovieRating/src
File Edit View Search Terminal Help
[Java - MovieRating/sr...
[emacs@localhost.loc...
training@localhost:~/...

```
Action  10663.92632524708
Adventure       13508.38004484305
Animation       7180.271186440678
Biography       11722.981203007519
Comedy  4927.353007307476
Crime   6738.802120141343
Documentary     2198.3611111111113
Drama   7895.270359652749
Family  6369.539047619048
Fantasy 11620.42419080068
Film-Noir       500.0
Game-Show       141.0
History 7841.124324324324
Horror  6687.450467289719
Music   4339.485294117647
Musical 5134.482758620689
Mystery 8311.900843881856
News    3496.3333333333335
Reality-TV      141.0
Romance 6003.420596727623
Sci-Fi  15070.806451612903
Short   17.6
Sport   5505.828220858895
:
```