## Analysis 06 – Find Distinct Genres in the dataset

|  |  |
| --- | --- |
| **public** **class** GenreMapper **extends** Mapper<LongWritable, Text, Text, NullWritable>{  **public** **void** map(LongWritable key, Text value, Context context) **throws** IOException, InterruptedException {  String input = value.toString();  //Split data on comma and ignore comma within double quotes  String[] movieData = input.split(",(?=(?:[^\"]\*\"[^\"]\*\")\*[^\"]\*$)", -1);  **try** {  String genreList = movieData[10];  **if**(genreList.equals("listed\_in"))  **return**;  String[] genres = genreList.replace("\"", "").split(",");  **for**(String genre : genres) {  context.write(**new** Text(genre), NullWritable.*get*());  }  }  **catch**(Exception ex) {    }  }  } | |
| **public** **class** GenreReducer **extends** Reducer<Text, NullWritable, Text, NullWritable> {  **protected** **void** reduce(Text key, Iterable<NullWritable> values, Context context)  **throws** IOException, InterruptedException {    context.write(key, NullWritable.*get*());  }  } | |
| **public** **class** Driver {  **public** **static** **void** main(String[] args) **throws** IOException, ClassNotFoundException, InterruptedException {  Configuration conf = **new** Configuration();  FileSystem fs = FileSystem.*get*(conf);  **if** (fs.exists(**new** Path(args[1]))) {  fs.delete(**new** Path(args[1]), **true**);  }  Job job = Job.*getInstance*(conf);  job.setMapperClass(GenreMapper.**class**);  job.setReducerClass(GenreReducer.**class**);  job.setCombinerClass(GenreReducer.**class**);    job.setJarByClass(Driver.**class**);  TextInputFormat.*addInputPath*(job, **new** Path(args[0]));  TextOutputFormat.*setOutputPath*(job, **new** Path(args[1]));  job.setMapOutputKeyClass(Text.**class**);  job.setMapOutputValueClass(NullWritable.**class**);  job.setOutputKeyClass(Text.**class**);  job.setOutputValueClass(NullWritable.**class**);  System.*exit*(job.waitForCompletion(**true**) ? 0 : 1);  }  } | |
|  |
|  |