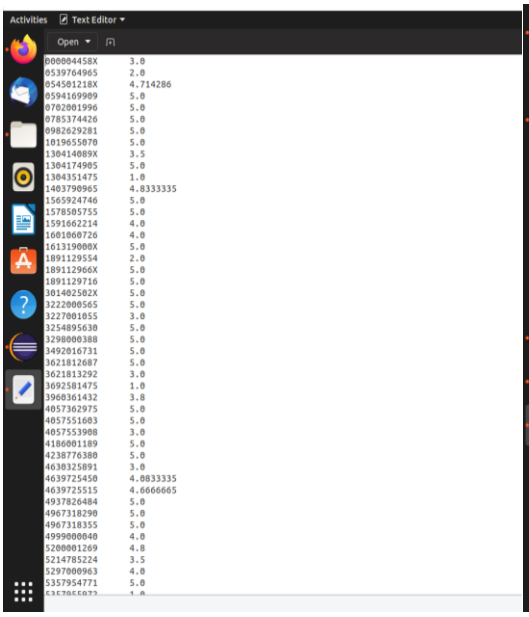
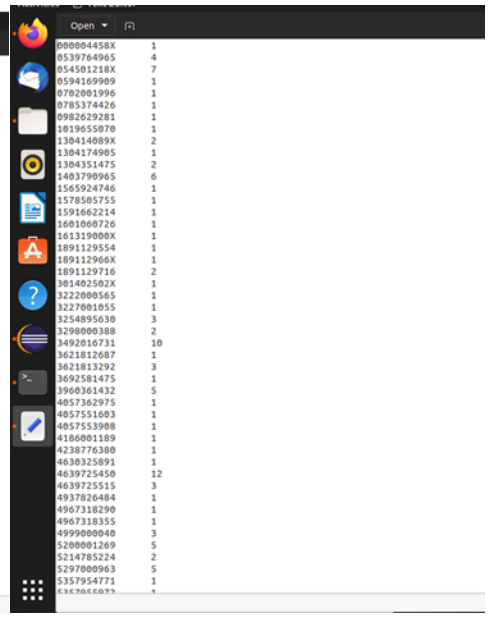
Inner Join

I have used the output from previous map reduce job and then I joined them. I have Average of reviews for each product and I have Number of reviews per product. Using inner join I am going to join them.

INPUT:

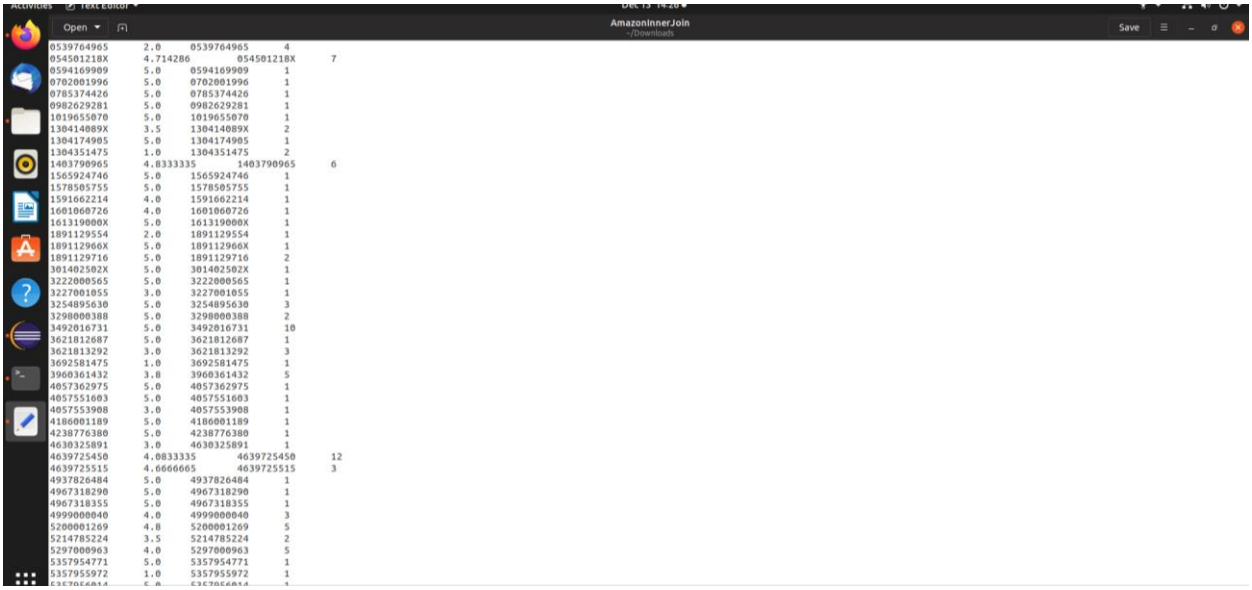


```
000004458X      3.0
0539764965      2.0
054501218X      4.714286
0594169909      5.0
0702001996      5.0
0785374426      5.0
0982629281      5.0
1019655070      5.0
130414089X      3.5
1304174905      5.0
1304351475      1.0
1403790965      4.8333335
1565924746      5.0
1578505755      5.0
1591662214      4.0
1601060726      4.0
161319000X      5.0
1891129554      2.0
189112966X      5.0
1891129716      5.0
301402502X      5.0
3222000565      5.0
3227001055      3.0
3254895630      5.0
3298000388      5.0
3492016731      5.0
3621812687      5.0
3621813292      3.0
3692581475      1.0
3960361432      3.8
4057362975      5.0
4057551603      5.0
4057553908      3.0
4186001189      5.0
4238776380      5.0
4630325891      3.0
4639725450      4.0833335
4639725515      4.6666665
4937826484      5.0
4967318290      5.0
4967318355      5.0
4999000040      4.0
5200001269      4.8
5214785224      3.5
5297000963      4.0
5357954771      5.0
```



```
000004458X      1
0539764965      4
054501218X      7
0594169909      1
0702001996      1
0785374426      1
0982629281      1
1019655070      1
130414089X      2
1304174905      1
1304351475      2
1403790965      6
1565924746      1
1578505755      1
1591662214      1
1601060726      1
161319000X      1
1891129554      1
189112966X      1
1891129716      2
301402502X      1
3222000565      1
3227001055      1
3254895630      3
3298000388      2
3492016731      10
3621812687      1
3621813292      3
3692581475      1
3960361432      5
4057362975      1
4057551603      1
4057553908      1
4186001189      1
4238776380      1
4630325891      1
4639725450      12
4639725515      3
4937826484      1
4967318290      1
4967318355      1
4999000040      3
5200001269      5
5214785224      2
5297000963      5
5357954771      1
```

OUTPUT: ./hadoop jar /home/sayali/Desktop/InnerJoin.jar sayali.AmazonReviews4.InnerJoin /AverageReviews/part-r-00000 /AmazonReviewsAggregate/part-r-00000 /InnerJoin



```
0539764965      2.0     0539764965      4
054501218X      4.714286        054501218X      7
0594169909      5.0     0594169909      1
0702001996      5.0     0702001996      1
0785374426      5.0     0785374426      1
0982629281      5.0     0982629281      1
1019655070      5.0     1019655070      1
130414089X      3.5     130414089X      2
1304174905      5.0     1304174905      1
1304351475      1.0     1304351475      2
1403790965      4.8333335       1403790965      6
1565924746      5.0     1565924746      1
1578505755      5.0     1578505755      1
1591662214      4.0     1591662214      1
1601060726      4.0     1601060726      1
161319000X      5.0     161319000X      1
1891129554      2.0     1891129554      1
189112966X      5.0     189112966X      1
1891129716      5.0     1891129716      2
301402502X      5.0     301402502X      1
3222000565      5.0     3222000565      1
3227001055      3.0     3227001055      1
3254895630      5.0     3254895630      3
3298000388      5.0     3298000388      2
3492016731      5.0     3492016731      10
3621812687      5.0     3621812687      1
3621813292      3.0     3621813292      3
3692581475      1.0     3692581475      1
3960361432      3.8     3960361432      5
4057362975      5.0     4057362975      1
4057551603      5.0     4057551603      1
4057553908      3.0     4057553908      1
4186001189      5.0     4186001189      1
4238776380      5.0     4238776380      1
4630325891      3.0     4630325891      1
4639725450      4.0833335       4639725450      12
4639725515      4.6666665       4639725515      3
4937826484      5.0     4937826484      1
4967318290      5.0     4967318290      1
4967318355      5.0     4967318355      1
4999000040      4.0     4999000040      3
5200001269      4.8     5200001269      5
5214785224      3.5     5214785224      2
5297000963      4.0     5297000963      5
5357954771      5.0     5357954771      1
5357955972      1.0     5357955972      1
```

Map Reduce Code:

```
package sayali.AmazonReviews4;
import java.io.IOException;
import java.util.ArrayList;
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.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.MultipleInputs;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;
```

Mapper 1:

```
public class AmazonAverageRatingsMapper1 extends Mapper<LongWritable, Text, Text, Text> {

	private Text outkey = new Text();
	private Text outvalue = new Text();
public void map(LongWritable key, Text value, Mapper.Context context) throws IOException,
InterruptedException {

		if (key.get() == 0) {
			return; }
		// Select product Id from first input
		String[] separatedInput = value.toString().split("\\t");
		String productID = separatedInput[0].trim();

		if (productID == null) {
			return;
		}
		outkey.set(productID);
		// Flag this record for the reducer and then output
		outvalue.set("*" + value.toString());
		context.write(outkey, outvalue);
	}
}
```

Mapper 2:

```
public class AmazonNoOfRatingsPerProductMapper2 extends Mapper<LongWritable, Text, Text, Text> {
	private Text outkey = new Text();
	private Text outvalue = new Text();
```

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

		if (key.get() == 0) {
			return;
		}
		//Select Product ID from Second input
		String[] separatedInput = value.toString().split("\\t");
		String productId = separatedInput[0];
		if (productId == null) {
			return;
		}
		outkey.set(productId);
		// Flag this record for the reducer and then output
		outvalue.set("#" + value.toString());
		context.write(outkey, outvalue);
	}
}
```

Reducer :

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

	private static final Text EMPTY_TEXT = new Text("");
	private Text tmp = new Text();
	private ArrayList<Text> listA = new ArrayList<Text>();
	private ArrayList<Text> listB = new ArrayList<Text>();
	private String joinType = null;
	public void setup(Reducer.Context context) {
	// Get the type of join from our configuration
		joinType = context.getConfiguration().get("join.type");
	}
public void reduce(Text key, Iterable<Text> values, Context context) throws IOException,
InterruptedException {
		// Clearing the lists
		listA.clear();
		listB.clear();

		while (values.iterator().hasNext()) {
			tmp = values.iterator().next();
			if (Character.toString((char) tmp.charAt(0)).equals("*")) {
			listA.add(new Text(tmp.toString().substring(1)));
			}
			if (Character.toString((char) tmp.charAt(0)).equals("#")) {
			listB.add(new Text(tmp.toString().substring(1)));
			}
		}
		executeJoinLogic(context);
```

```
	}
	private void executeJoinLogic(Context context) throws IOException, InterruptedException {
		if (joinType.equalsIgnoreCase("inner")) {
			if (!listA.isEmpty() && !listB.isEmpty()) {
				for (Text A : listA) {
					for (Text B : listB) {
						context.write(A, B);
					}
				}
			}
		}
	}
}
```

Driver:

```
public class InnerJoin {
public static void main(String[] args) throws IOException, InterruptedException,
ClassNotFoundException {

		Configuration conf = new Configuration();
		Job job = Job.getInstance(conf, "Inner Join");
		job.setJarByClass(InnerJoin.class);

		MultipleInputs.addInputPath(job, new Path(args[0]), TextInputFormat.class,
AmazonAverageRatingsMapper1.class);
		MultipleInputs.addInputPath(job, new Path(args[1]), TextInputFormat.class,
				AmazonNoOfRatingsPerProductMapper2.class);
		job.getConfiguration().set("join.type", "inner");
		// job.setNumReduceTasks(0);
		job.setReducerClass(InnerJoinReducer.class);

		job.setOutputFormatClass(TextOutputFormat.class);
		TextOutputFormat.setOutputPath(job, new Path(args[2]));

		job.setOutputKeyClass(Text.class);
		job.setOutputValueClass(Text.class);

		System.exit(job.waitForCompletion(true) ? 0 : 2);
	}
}
```