Driver:

package squareNumbers;

import java.io.IOException;

import org.apache.hadoop.conf.Configured;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapred.FileInputFormat;

import org.apache.hadoop.mapred.FileOutputFormat;

import org.apache.hadoop.mapred.JobClient;

import org.apache.hadoop.mapred.JobConf;

import org.apache.hadoop.util.Tool;

import org.apache.hadoop.util.ToolRunner;

public class SQDriver extends Configured implements Tool {

    public int run(String args[]) throws IOException

    {

        if (args.length < 2)

        {

            System.out.println("Please give valid inputs");

            return -1;

        }

        JobConf conf = new JobConf(SQDriver.class);

        FileInputFormat.setInputPaths(conf, new Path(args[0]));

        FileOutputFormat.setOutputPath(conf, new Path(args[1]));

        conf.setMapperClass(SQMapper.class);

        conf.setReducerClass(SQReducer.class);

        conf.setMapOutputKeyClass(Text.class);

        conf.setMapOutputValueClass(IntWritable.class);

        conf.setOutputKeyClass(Text.class);

        conf.setOutputValueClass(IntWritable.class);

        JobClient.runJob(conf);

        return 0;

    }

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

    {

        int exitCode = ToolRunner.run(new SQDriver(), args);

        System.out.println(exitCode);

    }

}

Mapper:

package squareNumbers;

import java.io.IOException;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.LongWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapred.MapReduceBase;

import org.apache.hadoop.mapred.Mapper;

import org.apache.hadoop.mapred.OutputCollector;

import org.apache.hadoop.mapred.Reporter;

public class SQMapper extends MapReduceBase implements Mapper<LongWritable, Text, Text, IntWritable>{

    public void map(LongWritable key, Text value, OutputCollector<Text, IntWritable> output, Reporter rep) throws IOException {

       String line = value.toString();

       for(String word: line.split(" ")) {

           Integer num = Integer.parseInt(word);

           output.collect(new Text("sum"), new IntWritable(num\*num));

       }

    }

}

Reducer:

package squareNumbers;

import java.io.IOException;

import java.util.Iterator;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapred.MapReduceBase;

import org.apache.hadoop.mapred.OutputCollector;

import org.apache.hadoop.mapred.Reducer;

import org.apache.hadoop.mapred.Reporter;

public class SQReducer extends MapReduceBase implements Reducer<Text,IntWritable, Text, IntWritable> {

    public void reduce(Text key, Iterator<IntWritable> value, OutputCollector<Text, IntWritable> output, Reporter rep) throws IOException

    {

        int count = 0;

        while (value.hasNext())

        {

            IntWritable i = value.next();

            count += i.get();

        }

        output.collect(key, new IntWritable(count));

    }

}