MAPPER:

```
package bdp.tweets;

import java.io.IOException;
import java.time.Instant;
import java.time.ZoneId;
import java.time.ZonedDateTime;
import java.util.ArrayList;
import java.util.List;
import java.util.regex.Matcher;
import java.util.regex.Pattern;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

public class HashtagsCountMapper extends Mapper<Object, Text, Text, IntWritable>
 {
   private IntWritable one = new IntWritable(1);
    private Text data = new Text();
    String[] fields = new String[4];
   public void map(Object key, Text line, Context context) throws IOException,
InterruptedException
   {
         final int number;
     //Fields contains line as follows.
     //   0          1                    2             3
     //epoch_time;tweetId;tweet(including #hashtags);device

if(line.toString().split(";").length == 4)
{
fields = line.toString().split(";");
}
number = fields[2].length();
int a = 0;
if(number <= 140)
{
        try
        {
                Instant t = Instant.ofEpochMilli(Long.valueOf(fields[0]).longValue());
                ZonedDateTime d = ZonedDateTime.ofInstant(t, ZoneId.of("-3"));
                a = d.getHour();
        }
        catch(NumberFormatException ex)
        {
```

```
                 ex.printStackTrace();
            }
            if (a == 22)
            {
  //Regex to match the hashtags from the line
            Pattern tags = Pattern.compile("#(\\S+)");
            Matcher mat = tags.matcher(fields[2]);
            List<String> strs = new ArrayList<String>();
            while (mat.find()) {
              //adding hashtag to the array of hashtags
              strs.add(mat.group(1));
            }
            for (int i = 0; i < strs.size(); i++) {
    //generating K-V
                        data.set(strs.get(i));
                        context.write(data, one);
}}}}}
```

In addition to the previous task, the result of an our extracted. If the hours is equal to 22, the string is parsed via regex. All the hashtags are placed in array and after that they are passed as a K-V pairs to a reducer (hashtag and 1). After that reducer summarise the hashtags amount and returns the K-V pairs of each unique hashtag and the amount of it in all tweets. The code for reducer is the same as in previous task.

Hashtag	Amount
Rio2016	1253022
Olympics	73100
rio2016	68192
Futebol	39889
Gold	35701
BRA	34256
USA	34108
CerimoniaDeAbertura	33361
OpeningCeremony	32013
Atletismo	28552