Find and count Hashtags

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import java.util.ArrayList;

import java.util.Collections;

import java.util.Comparator;

import java.util.Iterator;

import java.util.LinkedHashMap;

import java.util.List;

import java.util.Map;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class TwitterTrends {

public static void main(String[] args) throws NumberFormatException, IOException {

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

int t = Integer.parseInt(br.readLine());

Pattern pattern = Pattern.compile("(#\\w+)");

Matcher m;

LinkedHashMap<String, Integer> counter = new LinkedHashMap<String, Integer>();

for (int i = 0; i < t; i++) {

String line = br.readLine();

m = pattern.matcher(line);

while (m.find()) {

if (null != counter.get(m.group())) {

counter.put(m.group(), (counter.get(m.group()) + 1));

} else {

counter.put(m.group(), new Integer(1));

}

}

}

List<Map.Entry<String, Integer>> entries = new ArrayList<Map.Entry<String, Integer>>(counter.entrySet());

Collections.sort(entries, new Comparator<Map.Entry<String, Integer>>() {

public int compare(Map.Entry<String, Integer> a, Map.Entry<String, Integer> b) {

if (b.getValue().equals(a.getValue()))

return a.getKey().compareTo(b.getKey());

else

return b.getValue().compareTo(a.getValue());

}

});

Map<String, Integer> sortedMap = new LinkedHashMap<String, Integer>();

for (Map.Entry<String, Integer> entry : entries) {

sortedMap.put(entry.getKey(), entry.getValue());

}

Iterator it = sortedMap.entrySet().iterator();

int counterTemp = 1;

while (it.hasNext()) {

Map.Entry pair = (Map.Entry) it.next();

System.out.println(pair.getKey());

it.remove();

if (counterTemp == 5)

break;

counterTemp++;

}

}

}