

OCAJP 7 or OCAJP 8? Which Java Certification will you take?

- ☐ OCAJP7
☐ OCAJP8

Vote [Show results](#)

Votes so far: 1117
Days left to vote: 300

How to find highest repeating word from a text File in Java - Word Count Problem

How to find the word and their count from a text file is another frequently asked coding question from Java interviews. The logic to solve this problem is similar to what we have seen in [how to find duplicate words in a String](#). In the first step you need to build a word Map by reading contents of a text File. This Map should contain word as a key and their count as value. Once you have this Map ready, you can simply sort the Map based upon values. If you don't know how to sort a Map on values, see this [tutorial](#) first. It will teach you by sorting HashMap on values. Now getting key and value in sorted should be easy, but remember `HashMap` doesn't maintain order, so you need to use a List to keep the entry in sorted order. Once you got this list, you can simply **loop over the list** and print each key and value from the entry. This way, you can also create a table of words and their count in decreasing order. This problem is sometimes also asked as to print all word and their count in tabular format.



How to find highest repeated word from a file

Here is the Java program to find the duplicate word which has occurred maximum number of times in a file. You can also print frequency of word from highest to lowest because you have the Map, which contains word and their count in sorted order. All you need to do is **iterate over each entry of Map** and print the keys and values.

Categories

- [core java](#) (297)
- [programming](#) (184)
- [core java interview question answer](#) (86)
- [Java collection tutorial](#) (71)
- [interview questions](#) (56)
- [coding](#) (51)
- [java](#) (50)
- [Coding Problems](#) (40)
- [Java programming Tutorial](#) (32)
- [error and exception](#) (30)
- [data structure and algorithm](#) (23)
- [homework](#) (21)
- [Java 8](#) (19)
- [collection tutorial example](#) (19)
- [books](#) (16)
- [J2EE](#) (14)
- [String](#) (14)

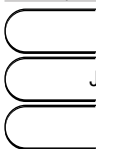
Most important part of this solution is sorting all entries. Since `Map.Entry` doesn't implement the `Comparable` interface, we need to write our own custom `Comparator` to sort the entries. If you look at my implementation, I am comparing entries on their values because that's what we want. Many programmer says that why not use the `LinkedHashMap` class? but remember, the `LinkedHashMap` class keeps the keys in sorted order, not the values. So you need this special `Comparator` to compare values and store them in List.

Here is one approach to solve this problem using map-reduce technique:

Interview Q

- [core java](#)
- [program](#)
- [SQL inte](#)
- [data stru](#)
- [coding in](#)
- [java coll](#)
- [java des](#)
- [questio](#)
- [thread in](#)
- [hibernat](#)
- [jdbc inte](#)
- [array int](#)
- [j2ee inte](#)
- [spring in](#)

Ads by Go



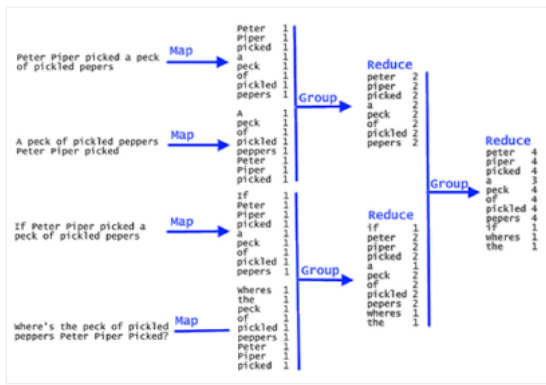
Books and

- [6 Websi](#)
- [Online](#)
- [Top 10](#)
- [5 Free J](#)
- [Develop](#)
- [Best Bo](#)
- [Prograr](#)
- [Top 3 Fr](#)
- [develop](#)
- [5 Books](#)
- [Skill](#)
- [10 Book](#)
- [Read](#)
- [10 Com](#)
- [Prograr](#)
- [10 Free](#)
- [5 Books](#)

Follow by E

Email address

- date and time (14)
- thread (14)
- sql (13)
- SQL interview Question (12)
- Java 5 tutorial (11)
- array (11)
- java concurrency tutorial (11)
- java io tutorial (11)
- JSP (10)
- Java Multithreading Tutorial (10)
- Servlet (10)
- thread interview questions (10)
- database (8)
- object oriented programming (8)
- JDBC (6)
- Eclipse (5)
- basics (5)
- troubleshooting (5)
- JEE Interview Questions (4)
- JavaScript (4)
- MySQL tutorial example (4)
- design pattern (4)
- java design pattern (4)
- JSP Interview Question (3)
- Java Enum (3)
- Linux (3)
- OCAJP (3)
- OOPS (3)
- Struts (3)
- Web Service (3)
- debugging (3)
- enum (3)
- Hibernate interview Question (2)



Java Program to Print word and their count from File

```
import java.io.BufferedReader;
import java.io.DataInputStream;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Map.Entry;
import java.util.Set;
import java.util.StringTokenizer;
import java.util.regex.Pattern;

/**
 * Java program to find count of repeated words in a file.
 *
 * @author
 */
public class Problem {

    public static void main(String args[]) {
        Map<String, Integer> wordMap = buildWordMap("C:/temp/words.txt");
        List<Entry<String, Integer>> list = sortByValueInDecreasingOrder(wordMap);
        System.out.println("List of repeated word from file and their count");
        for (Map.Entry<String, Integer> entry : list) {
            if (entry.getValue() > 1) {
                System.out.println(entry.getKey() + " => " + entry.getValue());
            }
        }
    }

    public static Map<String, Integer> buildWordMap(String fileName) {
        // Using diamond operator for clean code
        Map<String, Integer> wordMap = new HashMap<>();
        // Using try-with-resource statement for automatic resource management
        try (FileInputStream fis = new FileInputStream(fileName);
            DataInputStream dis = new DataInputStream(fis);
            BufferedReader br = new BufferedReader(new InputStreamReader(dis))) {
            // words are separated by whitespace
            Pattern pattern = Pattern.compile("\\s+");
            String line = null;
            while ((line = br.readLine()) != null) {
                // do this if case sensitivity is not required i.e. Java = java
                line = line.toLowerCase();
                String[] words = pattern.split(line);
            }
        }
    }
}
```

Search This

Recommen

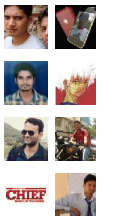
- The Bes days
- 10 Java Question
- Top 10 / for Java
- How to t
- 10 Book Read
- 5 Great

Ads by Go

Java

Followers

Followers (



Follow

Blog Archiv

- 2017 (1
- 2016 (1
- ▼ 2015 (1
 - Dec
 - Nov
 - ▼ Oct
 - 2 Wa
 - in

```

        for (String word : words) {
            if (wordMap.containsKey(word)) {
                wordMap.put(word, (wordMap.get(word) + 1));
            } else {
                wordMap.put(word, 1);
            }
        }
    }
}
} catch (IOException ioex) {
    ioex.printStackTrace();
}
return wordMap;
}

public static List<Entry<String, Integer>> sortByValueInDecreasingOrder(Map<String, Integer> wordMap) {
    Set<Entry<String, Integer>> entries = wordMap.entrySet();
    List<Entry<String, Integer>> list = new ArrayList<>(entries);
    Collections.sort(list, new Comparator<Map.Entry<String, Integer>>() {
        @Override
        public int compare(Map.Entry<String, Integer> o1, Map.Entry<String, Integer> o2) {
            return (o2.getValue()).compareTo(o1.getValue());
        }
    });
    return list;
}
}

```

Output:

List of repeated word from file and their count
 its => 2
 of => 2
 programming => 2
 java => 2
 language => 2

Things to note

If you writing code on interviews make sure they are production quality code, which means you must handle as many errors as possible, you must write unit tests, you must comment the code and you do proper resource management. Here are couple of more points to remember:

1) Close files and streams once you are through with it, see this [tutorial](#) learn right way to close the stream. If you are in Java 7, just use **try-with-resource** statement.

2) Since the size of the file is not specified, the interviewer may grill you on cases like What happens if the file is large? With a large file, your program will run out of memory and throw **java.lang.OutOfMemory: Java Heap space**. One solution for this is to do this task in chunk e.g. first read 20% content, find maximum repeated word on that, then read next 20% content and find repeated maximum by taking the previous maximum in consideration. This way, you don't need to store all words in memory and you can process any arbitrary length file.

3) Always use Generics for type-safety.

That's all on **how to find repeated word from a file and print their count**. You can apply the same technique to find duplicate words in a String. Since now you have a sorted list of words and their count, you can also find the maximum, minimum or repeated words which has counted more than the specific number.

Further Reading

If you are preparing for programming job interview then you must prepare for all-important topic e.g. data structure, string, array etc. One book which can help you on this task is the **Cracking the Coding Interview** book. It contains 150 Programming Questions and Solutions, which is good enough to clear most of the coding interviews.

How to
andTop 1
queHow to
JavaJava
ME

How to

How to
JavaTop 5
DoHow to
valva
InteError:
'C:'Avoid
Co
onHow to
wo

► Sept

► Aug

► July

► June

► May

► April

► Marc

► Febr

► Janu

► 2014 (6

► 2013 (4

► 2012 (1



fr



Related Java Tutorials

Powered by Google



Bank Exams Shortcut Tricks PDF - Prepare...

Ad talentsprint.com

How to Sort HashMap in Java based on...

java67.com

Top 30 OOPS Concept Interview Questions...

java67.com

Why Abstract class is Important in Java...

java67.com

How to Print Pyramid Pattern in Java?...

java67.com

How to Remove Duplicates from...

java67.com

3 ways to count words in Java String

java67.com

How to reverse String in Java with or...

java67.com

Posted by **Javin Paul**



+2 Recommend this on Google

Labels: **Coding Problems**

8 comments:

Anonymous October 2, 2015 at 5:22 AM

Nice smile emoticon On java 8 it has done with 4 code line :

```
String data = String.join("", Files.readAllLines(Paths.get("D:\\lab.txt")));
```

```
Arrays.asList(data.split("\\s+")).stream().collect(Collectors.groupingBy(str->str)).forEach((text,list)->{
    System.out.println(String.format("Word: %s - Occur: %d",text,list.size()));});
```

Reply

Replies



Roland Spatzenegger October 4, 2015 at 10:16 PM

Nice solution showing the power of streams, but uses temporarily more memory.



Roland Spatzenegger October 4, 2015 at 10:27 PM

How does the second point solve your memory problem. Out of memory can occur, if the line is too long or the word map gets too big. For a too long line you need a custom read method. For the big map you need some external storage solution. If you only store the maximum repeated words (like the top 10, did I understand this correctly?) , you may get wrong results.

Reply



Roland Spatzenegger October 4, 2015 at 10:08 PM

For an interview, I would also expect that the source follows the "single level of abstraction" principle (the buildWordMap function).

[Reply](#)



test May 13, 2016 at 9:12 PM

How to print the line number in which maximum occurrences occurred , i wanted to show top 5 occurrences of the word with the line number

[Reply](#)



test May 13, 2016 at 9:24 PM

How to print the occurrences of the word bases on the line number , i want to print 5 most occurrences of the word.

[Reply](#)

Anonymous December 8, 2016 at 9:08 AM

Let's see this chain of object creation, file name -> FileInputStream -> DataInputStream -> InputStreamReader -> BufferedReader, then finally you can stream content of this file, no wonder Java is such resource hog

[Reply](#)

Anonymous January 27, 2017 at 9:43 PM

Why sort the list at all? The original question asks to "find highest repeating word", you can just navigate the map and find the highest value. This way you are still at $O(n)$ rather than $O(n \log n)$ once you resort to sorting.

[Reply](#)

Enter your comment...

Comment as: Select profile... ▼

[Publish](#)

[Preview](#)

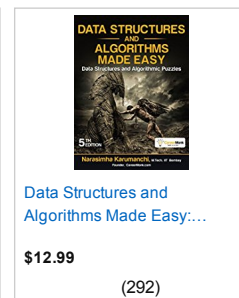
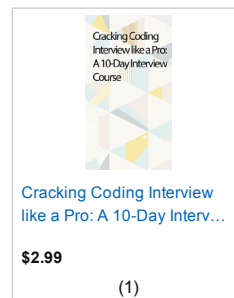
[Newer Post](#)


[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)


Related Books for Further Reading






Cracking the Coding Interview
The Interview Preparation Guide
by Gayle McDowell

\$5.49
Cracking the Coding Interview
(2)programmin...




ELEMENTS OF PROGRAMMING INTERVIEWS
ADAM KATZ, GUY MONOD, LEE HART PRAKASH

\$24.26
Elements of Programming Interviews
(248)ders' ...



Programming Interviews Exposed
Secrets to Landing Your Next Job
John Karypis, Ed. Elmasri, Rick Vander

\$17.52
Programming Interviews
(96) to L...



Java Programming Interviews Exposed
Secrets to Landing Your Next Job
John Karypis, Ed. Elmasri, Rick Vander

\$30.86
Java Programming
(16)ed

All

Search Amazon

Go

Ads by Amazon

Powered by [Blogger](#).