

# JavaMadeSoEasy.com (JMSE)

[Home](#)   [Core Java Tutorials](#)   [Interview Questions](#)   [Interview Programs](#)   [Custom](#)   [Quiz](#)   [Database](#)   [More](#)   [About](#)

- [Algorithm](#)
- [Apache Tomcat Server](#)
- [Collection Framework](#)
- [Collection programs](#)
- [Collections implementation](#)
- [Core Java](#)
- [Core Java Differences](#)
- [Core Java Tutorials](#)
- [Custom implementation of Data Structures](#)
- [Data Structure](#)
- [Date tutorials](#)
- [eclipse](#)
- [Exceptions](#)
- [File IO/File handling](#)
- [Garbage collection in java](#)



## Collection framework Tutorial in java in detail with diagrams and programs

*You are here :* [Home](#) / [Core Java Tutorials](#) / [Collection framework Tutorial in java](#)



*Basic Collection tutorial in core java >*

[Collection tutorial in java](#)



### Search This Blog

- [Generics](#)
- [Hashing](#)
- [Interview questions](#)
- [iText Pdf tutorial](#)
- [Java 7](#)
- [Java 8](#)
- [Java 9](#)
- [Java QUIZ](#)
- [JavaScript](#)
- [JDBC](#)
- [JUNIT](#)
- [JVM](#)
- [Level1 programs \(beginners\)](#)
- [Level2 programs \(intermediate\)](#)
- [Level3 programs \(advanced\)](#)
- [Linked List](#)
- [Linux](#)
- [Matrix programs](#)
- [MongoDB](#)
- [MongoDB Java](#)
- [MsSql](#)
- [MultiThreading](#)
- [MySql](#)
- [Notepad++](#)
- [Oracle](#)

List tutorial in java >

[java.util.ArrayList tutorial in java](#)

0	1	2	3	4	5	6	7	8	9
null	null	null	null	null	null	null	null	null	null

[ArrayList custom implementation in java](#)

java.util.[LinkedList tutorial in java](#)

Set tutorial in java >

[java.util.HashSet tutorial in java](#)



[Set Custom implementation in java](#)



[LinkedHashSet Custom implementation in java](#)

[EnumSet tutorial in java with program](#)

Map tutorial in java >

[HashMap tutorial in java](#)

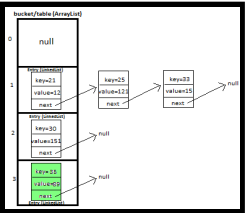


Subscribe-Our exclusive posts  
for FREE. Stay on top! Join

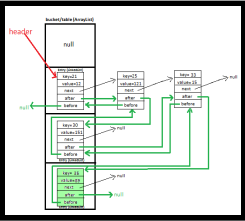
- [OutOfMemoryErr or](#)
- [Overriding equals and hashCode](#)
- [PostgreSQL](#)
- [Producer Consumer problem/pattern](#)
- [Pyramid generation](#)
- [RegEx](#)
- [Serialization](#)
- [Thread Concurrency](#)
- [Threads](#)
- [TUTORIALS](#)
- [Windows](#)

Join our Groups>

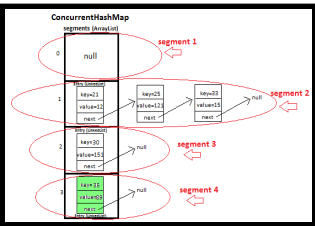
- [FACEBOOK](#)
- [LINKED IN](#)



[HashMap Custom implementation in java](#)



[LinkedHashMap Custom implementation in java](#)



[ConcurrentHashMap tutorial in java - with Segments formation in detail with diagram](#)

[IdentityHashMap tutorial in java with program](#)

[WeakHashMap tutorial in java](#)

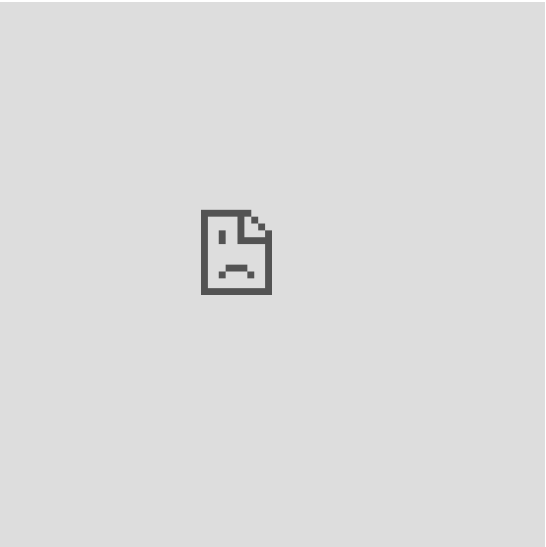
[EnumMap tutorial in java with program](#)

*Basic Collection tutorial in core java - All properties in tabular form >*

2693+ subscribers

Email address..

Submit



All Labels

[Algorithm](#)(34) [Apache Tomcat Server](#)(4) [Arrays](#) [Java](#)(30) [Autoboxing](#)(5) [Basic java programs for beginners](#)(15) [Binary Trees](#)(6) [Collection Framework](#)(68) [Collection programs](#)(105) [Collections implementation](#)(16) [Comparable and Comparator program](#)(22) [Core Java](#)(1032) [core java Basics](#)(38) [Core java Conversions](#)(21) [Core Java Differences](#)(11) [Core Java Tutorials](#)(12) [CRUD MongoDB java](#)(5) [CRUD operations MongoDB](#)(3) [Custom implementation of Data Structures](#)(11) [Data Structure](#)(26) [Database](#)(1) [Database Tutorials](#)(3) [Date tutorials](#)(11) [DeSerialization](#)(19) [eclipse](#)(27) [Exceptions](#)(71)

[Collection tutorial in java - java.util.List, java.util.Set and java.util.Map all properties in tabular form](#)

*Collection framework java tutorial : Collection hierarchy in core java >*

[List hierarchy tutorial in java - Detailed - java.util.ArrayList, java.util.LinkedList, java.util.vector, java.util.concurrent.CopyOnWriteArrayList classes](#)

[Set hierarchy tutorial in java - Detailed - java.util.HashSet, java.util.concurrent.CopyOnWriteArraySet, java.util.LinkedHashSet, java.util.TreeSet, java.util.concurrent.ConcurrentSkipListSet, java.util.EnumSet classes](#)

[Map hierarchy tutorial in java - Detailed - java.util.HashMap, java.util.Hashtable, java.util.concurrent.ConcurrentHashMap, java.util.LinkedHashMap, java.util.TreeMap, java.util.concurrent.ConcurrentSkipListMap, java.util.IdentityHashMap, java.util.WeakHashMap, java.util.EnumMap classes](#)

*ConcurrentModificationException , Fail-fast and Fail-safe >*

[ConcurrentModificationException, Fail-fast and Fail-safe in detail in java](#)

*Collection framework java tutorial : Important Collection - Similarity and Differences in core java >*

[Difference between List, Set and Map in java](#)

[File IO/File handling](#)(71) [Garbage collection in java](#)(43) [Generics](#)(5) [Hashing](#)(8) [Hibernate](#)(1) [Interview questions](#)(14) [iText Pdf tutorial](#)(45) [Java 4](#)(1) [java 5](#)(3) [Java 7](#)(32) [Java 8](#)(80) [Java 9](#)(12) [Java keywords](#)(11) [Java Mcq\(Multiple choice questions\)](#)(5) [Java Programs](#)(8) [Java QUIZ](#)(7) [JavaScript](#)(2) [JDBC](#)(74) [JUNIT](#)(7) [JVM](#)(22) [Level1 programs \(beginners\)](#)(34) [Level2 programs \(intermediate\)](#)(23) [Level3 programs \(advanced\)](#)(13) [Linux](#)(6) [Matrix programs](#)(10) [MongoDB](#)(87) [MongoDB Java](#)(20) [MsSql](#)(1) [MultiThreading](#)(97) [MySql](#)(7) [Notepad++](#)(6) [Oracle](#)(18) [OutOfMemoryError](#)(13) [Overriding equals and hashCode](#)(11) [Pattern generating](#)(14) [PostgreSQL](#)(1) [Producer Consumer problem/pattern](#)(10) [Pyramid generation](#)(14) [Recursion](#)(10) [RegEx](#)(7) [Serialization](#)(20) [Serialization top interview questions and answers in java](#)(18) [Thread Concurrency](#)(35) [Threads](#)(73) [TUTORIALS](#)(22) [Windows](#)(8)



## *List Differences tutorial in java >*

[ArrayList vs LinkedList - Similarity and Differences in java](#)

[ArrayList vs Vector - Similarity and Differences in java](#)

[List vs Set - Similarity and Differences in java](#)

## *Set Differences tutorial in java >*

[HashSet vs LinkedHashSet vs TreeSet - Similarity and Differences in java](#)

## *Map Differences tutorial in java >*

[Similarity and Differences between HashMap and Hashtable in java](#)

[Similarity and Differences between HashMap and ConcurrentHashMap - in java](#)

[HashMap vs Hashtable vs LinkedHashMap vs TreeMap - Differences in java](#)

[Similarity and Differences between HashMap vs IdentityHashMap - with program in java](#)

## *Differences between few more important Collection framework classes in java >*

## Popular Posts of JavaMadeSoEasy

- [HashMap Custom implementation in java - How HashMap works internally with diagrams and full program](#)
- [Collection Quiz in Java - MCQ - Multiple choice questions](#)
- [CORE JAVA - Top 120 most interesting and important interview questions and answers in core java](#)
- [THREADS - Top 80 interview questions and answers in java for freshers and experienced\(detailed explanation with programs\) Set-1 > Q1- Q60](#)
- [Core Java Tutorial in detail with diagram and programs - BEST EXPLANATION EVER](#)
- [COLLECTION - Top 100 interview questions and answers in java for fresher and experienced in detail - Set-1 > Q1- Q50](#)

[Collection vs Collections - Differences between in java](#)

[Comparable vs Comparator - differences between in java](#)

[Similarity and Differences between Iterator vs ListIterator - in java](#)

[Differences and similarities between Iterator vs Enumeration - in java](#)

*Convert Array to list, list to array,  
Array to set, Set to array in java >*

[How to convert ArrayList to HashSet and Set to Array in Java examples](#)

[convert Array to Set and Set to array in Java](#)

[How to convert Array to List and ArrayList to array in Java](#)

*Map to List conversion >*

[How to convert Map to List in java](#)

*Synchronize ArrayList - collection framework tutorial in java  
tutorial >*

[How to synchronize arraylist in java to make it completely thread safe in java](#)

- [Find sum of both diagonals in matrix - program in java](#)
- [LinkedList Custom implementation in java - How LinkedList works internally with diagrams and full program](#)
- [Thread/multi threading Quiz in Java - MCQ - Multiple choice questions](#)
- [EXCEPTIONS - Top 60 interview questions and answers in java for fresher and experienced - detailed explanation with diagrams Set-1 > Q1- Q25](#)



## *What is default capacity of ArrayList and Vector, and how they are resized in collection framework tutorial in core java*

>

[what is the default initial capacity of ARRAYLIST, how it is resized and size is increased in java](#)

[what is the default initial capacity of VECTOR, how it is resized and size is increased in java](#)

## *Collection framework java tutorial : Important Similarity and Differences Collection classes in concurrent and non-concurrent packages in java >*

[ArrayList vs CopyOnWriteArrayList - Similarity and Differences with program in java](#)

[HashSet vs CopyOnWriteArraySet - Similarity and Differences with program in java](#)

[TreeSet vs ConcurrentSkipListSet - Similarity and Differences with program in java](#)

[TreeMap vs ConcurrentSkipListMap - Similarity and Differences with program in java](#)

## *Collection framework java tutorial : Sorting Collection by implementing comparator and Comparable, using TreeMap*

### JavaMadeSoeasy Archive

- ▶ [2018](#) (17)
- ▶ [2017](#) (211)
- ▶ [2016](#) (240)
- ▶ [2015](#) (722)
  - ▶ [December](#) (61)
  - ▶ [November](#) (96)
  - ▶ [October](#) (14)
  - ▶ [September](#) (13)
  - ▶ [August](#) (73)
  - ▶ [July](#) (54)
  - ▶ [June](#) (29)
  - ▶ [May](#) (53)
  - ▶ [April](#) (109)

## *and TreeSet, Collections.sort and Arrays.sort tutorial in java*

>

[Comparable vs Comparator - differences and sorting list by implementing Comparable and Comparator in classes and inner classes in java](#)

[Arrays.sort to sort arrays by implementing Comparator and how Comparator of superclass can be used by subclasses in java](#)

[Sort Set by using TreeSet and by implementing Comparator and Comparable interface in java](#)

[Sort Map by key in Ascending and descending order by implementing Comparator interface and overriding its compare method and using TreeMap in java](#)

[Sort Map by value in Ascending and descending order by implementing Comparator interface and overriding its compare method in java](#)

## *Top Collection Interviews question and answers >*

**[COLLECTION - Top 50 interview questions and answers in core java for fresher and experienced](#)**

- ▶ [March](#) (93)
- ▶ [February](#) (99)
- ▶ [January](#) (28)



## *Consequence and advantage of using ArrayList and Vector tutorial in multithreading environment in java >*

[Consequence of using ArrayList in multithreading environment in java](#)

[Advantage of using Vector in multithreading environment in java](#)

## *>>>> Collection framework Programs and examples in core java >>>>*

### *java.util.ArrayList Programs in java >*

- >[ArrayList tutorial - add, add element at specific index methods program in java](#)
- >[ArrayList tutorial - remove, get, contains and set methods program in java](#)
- >[ArrayList tutorial - iterate using iterator, listIterator, Enumeration and enhanced for loop in java](#)
- >[ArrayList tutorial - fail-safe or fail-fast iteration using iterator, listIterator, Enumeration and enhanced for loop example in java](#)
- >[ArrayList tutorial - isEmpty, size and clear methods program in java](#)
- >[ArrayList tutorial - synchronizing using Collections.synchronizedList example in java](#)
- >[ArrayList tutorial - making list unmodifiable using Collections.unmodifiableList example in java](#)

## ***LinkedList Programs in java >***

- >[LinkedList - add, add element at specific index methods program in java](#)
- >[LinkedList - iterate using iterator, listIterator, Enumeration and enhanced for loop in java](#)
- >[LinkedList - fail-safe or fail-fast iteration using iterator, listIterator, Enumeration and enhanced for loop in java](#)
- >[LinkedList - isEmpty, size and clear methods program in java](#)
- >[LinkedList - remove, get, contains and set methods program in java](#)
- >[LinkedList - making list unmodifiable using Collections.unmodifiableList example in java](#)
- >[LinkedList - synchronizing using Collections.synchronizedList example in java](#)

## ***java.util.Vector Programs in java >***

- >[Vector tutorial - add, add element at specific index methods program in java](#)
- >[Vector tutorial - remove, get, contains and set methods program in java](#)
- >[Vector tutorial - iterate using iterator, listIterator, Enumeration and enhanced for loop example in java](#)
- >[Vector tutorial - fail-safe or fail-fast iteration using iterator, listIterator, Enumeration and enhanced for loop example in java](#)

## ***java.util.concurrent.CopyOnWriteArrayList Programs in java >***

- >[CopyOnWriteArrayList tutorial - add, add element at specific index methods program in java](#)
- >[CopyOnWriteArrayList tutorial - remove, get, contains and set methods program in java](#)
- >[CopyOnWriteArrayList tutorial - iterate using iterator, listIterator, Enumeration and enhanced for loop example in java](#)
- >[CopyOnWriteArrayList tutorial - fail-safe or fail-fast iteration using iterator, listIterator, Enumeration and enhanced for loop example in java](#)
- >[CopyOnWriteArrayList tutorial - isEmpty, size and clear methods program in java](#)

## *java.util.HashSet Programs in java >*

- >[HashSet tutorial - add, contains, remove, size methods program in java](#)
- >[HashSet tutorial - isEmpty, size and clear methods program in java](#)
- >[HashSet tutorial - iterate using iterator, Enumeration and enhanced for loop program in java](#)
- >[HashSet tutorial - fail-safe or fail-fast iteration using iterator, Enumeration and enhanced for loop program in java](#)
- >[HashSet tutorial - synchronizing using Collections.synchronizedSet program in java](#)
- >[HashSet tutorial - making set unmodifiable using Collections.unmodifiableSet in java](#)

## *java.util.concurrent.CopyOnWriteArraySet Programs in java >*

- >[CopyOnWriteArraySet tutorial - add, contains, remove, size methods program in java](#)
- >[CopyOnWriteArraySet tutorial - isEmpty, size and clear methods program in java](#)
- >[CopyOnWriteArraySet tutorial - iterate using iterator, Enumeration and enhanced for loop program in java](#)
- >[CopyOnWriteArraySet tutorial - fail-safe or fail-fast iteration using iterator, Enumeration and enhanced for loop program in java](#)

## *java.util.LinkedHashSet Programs in java >*

- >[LinkedHashSet tutorial - adding elements order is maintained in java](#)
- >[LinkedHashSet - add, contains, remove, size methods program in java](#)
- >[LinkedHashSet - isEmpty, size and clear methods program](#)
- >[LinkedHashSet - iterate using iterator, Enumeration and enhanced for loop program in java](#)
- >[LinkedHashSet - fail-fast iteration using iterator, Enumeration and enhanced for loop example program in java](#)
- >[LinkedHashSet - making set unmodifiable using Collections.unmodifiableSet example](#)
- >[LinkedHashSet - synchronizing using Collections.synchronizedSet program in java](#)

## *java.util.TreeSet Programs in java >*

- >[TreeSet tutorial - iterate using iterator, listIterator, Enumeration and enhanced for loop example in java](#)

>[TreeSet - fail-safe or fail-fast iteration using iterator, listIterator, Enumeration and enhanced for loop example in java](#)

## ***java.util.concurrent.ConcurrentSkipListSet Programs in java >***

>[ConcurrentSkipListSet tutorial - adding elements, maintains natural order of elements example in java](#)

>[ConcurrentSkipListSet tutorial - iterate using iterator, listIterator, Enumeration and enhanced for loop example in java](#)

>[ConcurrentSkipListSet tutorial - fail-safe or fail-fast iteration using iterator, listIterator, Enumeration and enhanced for loop example in java](#)

## ***java.util.HashMap Programs in java >***

>[HashMap tutorial - put, get, containsKey, containsValue, remove, size methods example in java](#)

>[HashMap tutorial - Iterate on keys by obtaining keySet, Iterate on values by obtaining values, Iterate on entry by obtaining entrySet example in java](#)

>[HashMap tutorial - Iterator on keySet, values and entrySet is fail-safe or fail-fast example in java?](#)

>[HashMap tutorial - synchronizing map using Collections.synchronizedMap example in java](#)

>[HashMap tutorial - making map unmodifiable using Collections.unmodifiableMap example in java](#)

## ***Hashtable Programs in java >***

>[Hashtable - put, get, containsKey, containsValue, remove, size methods example in java](#)

>[Hashtable - Iterate on keys by obtaining keySet, Iterate on values by obtaining values, Iterate on entry by obtaining entrySet example in java](#)

>[Hashtable - Iterator on keySet, values and entrySet is fail-fast in java](#)

>[Hashtable - making map unmodifiable using Collections.unmodifiableMap example in java](#)

## ***java.util.concurrent.ConcurrentSkipListMap Programs in java***

>

>[ConcurrentSkipListMap tutorial - Iterate on keys by obtaining keySet, Iterate on values by obtaining values, Iterate on entry by obtaining entrySet in java](#)

- >[ConcurrentSkipListMap tutorial - Iterator on keySet, values and entrySet is fail-safe or fail-fast example in java?](#)
- >[ConcurrentSkipListMap program to use put, get, containsKey, containsValue, remove, size methods example in java](#)
- >[ConcurrentSkipListMap example to make map unmodifiable in java](#)

## *java.util.TreeMap Programs in java >*

- >[TreeMap program to use put, get, containsKey, containsValue, remove, size example methods in java](#)
- >[TreeMap program to Iterate on keys by obtaining keySet, Iterate on values example in java](#)
- >[TreeMap program to find Iterator on keySet, values and entrySet is fail-fast example in java](#)
- >[TreeMap program to make it unmodifiable example in java](#)

## *LinkedHashMap Programs in java >*

- >[LinkedHashMap - put, get, containsKey, containsValue, remove, size methods example in java](#)
- >[LinkedHashMap - Iterator on keySet, values and entrySet is fail-safe or fail-fast example in java](#)
- >[LinkedHashMap - Iterate on keys by obtaining keySet, Iterate on values by obtaining values, Iterate on entry by obtaining entrySet example in java](#)
- >[LinkedHashMap - making it unmodifiable example in java](#)

## *java.util.concurrent.ConcurrentHashMap Programs in java >*

[Program to use ConcurrentHashMap's putIfAbsent method in java](#)

[Program to create method that provides functionality similar to putIfAbsent method of ConcurrentHashMap and to be used with HashMap in java](#)

[ConcurrentHashMap program to use put, get, containsKey, containsValue, remove, size methods in java](#)

[ConcurrentHashMap program to Iterate on keys and values in java](#)

[ConcurrentHashMap program to show Iterator is fail-safe in java](#)

[ConcurrentHashMap program to make it unmodifiable in java](#)

## *Programs to sort Collection framework classes in java >>*

*Collection framework java tutorial : Different approaches/Programs to Sort List in java >*

[Program to sort Employee list on basis of name in ascending order by implementing Comparable interface and overriding its compareTo method in java](#)

[Program to sort Employee list on basis of name in descending order by implementing Comparable interface and overriding its compareTo method in java](#)

[Program to sort Employee list on basis of name in ascending order by implementing Comparator interface and overriding its compare method in java](#)

[Program to sort Employee list on basis of name in descending order by implementing Comparator interface and overriding its compare method in java](#)

[Program to sort Employee list on basis of name and id in ascending order by implementing Comparator interface and overriding its compare method in java](#)

[Program to sort Employee list on basis of name and id in ascending order by implementing](#)

[Comparator interface in inner and static inner class and overriding its compare method in java](#)

## *Collection framework java tutorial : Different approaches/Programs to Sort Arrays in java >*

[Program to sort Integer array using Arrays.sort \(by default Arrays.sort will sort array in ascending order\) in java](#)

[Program to sort Integer array by using Arrays.sort \(we will define Comparator to sort elements in descending order\) in java](#)

[Program to sort Employee array using Arrays.sort in ascending order on basis of name and id by implementing Comparator interface in java](#)

[Program to understand Comparator of superclass can be used by subclasses in java](#)

## *Collection framework java tutorial : Different approaches/Programs to Sort Set in java >*

[Program to Sort Set by using TreeSet\(by default elements are sorted in ascending order\), where elements are Integer type in java](#)

[Program to Sort Set by using TreeSet\(we will define Comparator to sort elements in descending order\), where elements are Integer type in java.](#)

[Program to Sort Set in Ascending order by using TreeSet and implementing Comparable interface, where elements are customObject/Employee type in java.](#)



[Program to Sort Set in Ascending order by using TreeSet and implementing Comparator interface, where elements are customObject/Employee type in java.](#)

[TreeMap's addAll method to sort java Collection framework classes in java](#)

[TreeMap's constructor to sort java Collection framework classes in java](#)

## *Collection framework java tutorial : Different approaches/Programs to Sort Map by key in java >*

[Program to Sort Map by key in Ascending order by implementing Comparator interface and overriding its compare method in java](#)

[Program to Sort Map by key in Descending order by implementing Comparator interface and overriding its compare method in java](#)

-

[Program to Sort Map by key in Ascending order by using TreeMap, where key is Integer type in java.](#)

[Program to Sort Map by key in Ascending order by using TreeMap and implementing Comparable interface, where key is customObject/Employee type in java](#)

[Program to Sort Map by key in descending order by using TreeMap and implementing Comparator interface, where key is Integer type in java](#)

[TreeMap's putAll method to sort map on basis of key in java](#)

[TreeMap's constructor to sort map on basis of key in java](#)

## *Collection framework java tutorial : Different approaches/Programs to Sort Map by value in java >*

[Program to Sort Map by value in Ascending order by implementing Comparator interface and overriding its compare method in java](#)

[Program to Sort Map by value in Descending order by implementing Comparator interface and overriding its compare method in java](#)

### Summary -

In this core java tutorial we learned about exception handling in java in detail.

*Having any doubt? or you liked the tutorial! Please comment in below section.*

*Please express your love by liking [\*\*JavaMadeSoEasy.com\*\*](https://www.javamadesoeasy.com) ([\*\*JMSE\*\*](#)) on [\*\*facebook\*\*](#), following on [\*\*google+\*\*](#) or [\*\*Twitter\*\*](#).*

**[RELATED LINKS>](#)**

[Thread Tutorial in java](#)

[Thread Concurrency Tutorial in java](#)

[Exceptions Tutorial in java in detail with diagrams and programs](#)

**Labels:** [Collection Framework](#) [Core Java](#) [Core Java Tutorials](#) [TUTORIALS](#)

**Must read for you :**

0 Comments

www.javamadesoeasy.com

1


Login

Recommend

1

Share

Sort by Best



Start the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name

Be the first to comment.

ALSO ON WWW.JAVAMADESOEASY.COM

JavaMadeSoEasy.com (JMSE): Copyright © 2015-2016 AnkitMittal. Contents of ...

1 comment • 2 years ago

Reena raj — This is excellent information. It is amazing and wonderful to visit your site.....

JavaMadeSoEasy.com (JMSE): Exception in thread java.lang.OutOfMemoryError: Java ...

1 comment • a year ago

Rahul Dhawan — What all can be taken as prevention measures for OutOfMemoryException?

JavaMadeSoEasy: JDBC- Insert/Store/save IMAGE in database by using ...

1 comment • 3 years ago

Kingshore Naidu — Exception in thread "main" java.lang.AbstractMethodError: Method ...

JavaMadeSoEasy.com (JMSE): How to pass VM argument to tomcat in eclipse

1 comment • 2 years ago

Anonymous — Very well explained.Too much helpful for freshers as well as for experienced java lovers. Keep it up

Subscribe

Add Disqus to your siteAdd DisqusAdd

Disqus' Privacy PolicyPrivacy PolicyPrivacy

**[Newer Post](#)**

**[Home](#)**

**[Older Post](#)**