JavaScript is disabled on your browser.

[Skip navigation links](#1fob9te)

* [Package](http://docs.google.com/searchTree/package-summary.html)
* Class
* [Use](http://docs.google.com/class-use/EmptyTree.html)
* [Tree](http://docs.google.com/package-tree.html)
* [Deprecated](http://docs.google.com/deprecated-list.html)
* [Index](http://docs.google.com/index-files/index-1.html)
* [Help](http://docs.google.com/help-doc.html)
* Prev Class
* [Next Class](http://docs.google.com/searchTree/NonEmptyTree.html)
* [Frames](http://docs.google.com/index.html?searchTree/EmptyTree.html)
* [No Frames](http://docs.google.com/EmptyTree.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* Field |
* Constr |
* [Method](#3znysh7)
* Detail:
* Field |
* Constr |
* [Method](#tyjcwt)

searchTree

## Class EmptyTree<K extends java.lang.Comparable<K>,V>

* java.lang.Object
  + searchTree.EmptyTree<K,V>
* All Implemented Interfaces: [Tree](http://docs.google.com/searchTree/Tree.html)<K,V>  
    
  public class EmptyTree<K extends java.lang.Comparable<K>,V>  
  extends java.lang.Object  
  implements [Tree](http://docs.google.com/searchTree/Tree.html)<K,V>  
  This class is used to represent the empty search tree: a search tree that contains no entries. This class is a singleton class: since all empty search trees are the same, there is no need for multiple instances of this class. Instead, a single instance of the class is created and made available through the static field SINGLETON. The constructor is private, preventing other code from mistakenly creating additional instances of the class.

### Method SummaryAll Methods Static Methods Instance Methods Concrete Methods

|  |  |
| --- | --- |
| * + Modifier and Type | * + Method and Description |
| * + void | * + [addKeysToCollection](http://docs.google.com/searchTree/EmptyTree.html#addKeysToCollection-java.util.Collection-)(java.util.Collection<[K](http://docs.google.com/searchTree/EmptyTree.html)> c) Add all keys bound in this tree to the collection c. |
| * + [Tree](http://docs.google.com/searchTree/Tree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html),[V](http://docs.google.com/searchTree/EmptyTree.html)> | * + [delete](http://docs.google.com/searchTree/EmptyTree.html#delete-K-)([K](http://docs.google.com/searchTree/EmptyTree.html) key) Delete any binding the key has in this tree. |
| * + static <K extends java.lang.Comparable<K>,V> [EmptyTree](http://docs.google.com/searchTree/EmptyTree.html)<K,V> | * + [getInstance](http://docs.google.com/searchTree/EmptyTree.html#getInstance--)() |
| * + [NonEmptyTree](http://docs.google.com/searchTree/NonEmptyTree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html),[V](http://docs.google.com/searchTree/EmptyTree.html)> | * + [insert](http://docs.google.com/searchTree/EmptyTree.html#insert-K-V-)([K](http://docs.google.com/searchTree/EmptyTree.html) key, [V](http://docs.google.com/searchTree/EmptyTree.html) value) Insert/update the Tree with a new key:value pair. |
| * + [K](http://docs.google.com/searchTree/EmptyTree.html) | * + [max](http://docs.google.com/searchTree/EmptyTree.html#max--)() Return the maximum key in the subtree |
| * + [K](http://docs.google.com/searchTree/EmptyTree.html) | * + [min](http://docs.google.com/searchTree/EmptyTree.html#min--)() Return the minimum key in the subtree |
| * + [V](http://docs.google.com/searchTree/EmptyTree.html) | * + [search](http://docs.google.com/searchTree/EmptyTree.html#search-K-)([K](http://docs.google.com/searchTree/EmptyTree.html) key) Find the value that this key is bound to in this tree. |
| * + int | * + [size](http://docs.google.com/searchTree/EmptyTree.html#size--)() Return number of keys that are bound in this tree. |
| * + [Tree](http://docs.google.com/searchTree/Tree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html),[V](http://docs.google.com/searchTree/EmptyTree.html)> | * + [subTree](http://docs.google.com/searchTree/EmptyTree.html#subTree-K-K-)([K](http://docs.google.com/searchTree/EmptyTree.html) fromKey, [K](http://docs.google.com/searchTree/EmptyTree.html) toKey) Returns a Tree containing all entries between fromKey and toKey, inclusive |

### Methods inherited from class java.lang.Objectequals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Method Detail

#### getInstance public static <K extends java.lang.Comparable<K>,V> [EmptyTree](http://docs.google.com/searchTree/EmptyTree.html)<K,V> getInstance()

#### search public [V](http://docs.google.com/searchTree/EmptyTree.html) search([K](http://docs.google.com/searchTree/EmptyTree.html) key) Description copied from interface: [Tree](http://docs.google.com/searchTree/Tree.html#search-K-) Find the value that this key is bound to in this tree.Specified by: [search](http://docs.google.com/searchTree/Tree.html#search-K-) in interface [Tree](http://docs.google.com/searchTree/Tree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html) extends java.lang.Comparable<[K](http://docs.google.com/searchTree/EmptyTree.html)>,[V](http://docs.google.com/searchTree/EmptyTree.html)> Parameters: key - -- Key to search for Returns: -- value associated with the key by this Tree, or null if the key does not have an association in this tree.

#### insert public [NonEmptyTree](http://docs.google.com/searchTree/NonEmptyTree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html),[V](http://docs.google.com/searchTree/EmptyTree.html)> insert([K](http://docs.google.com/searchTree/EmptyTree.html) key, [V](http://docs.google.com/searchTree/EmptyTree.html) value) Description copied from interface: [Tree](http://docs.google.com/searchTree/Tree.html#insert-K-V-) Insert/update the Tree with a new key:value pair. If the key already exists in the tree, update the value associated with it. If the key doesn't exist, insert the new key value pair. This method returns a reference to an Tree that represents the updated value. In many, but not all cases, the method may just return a reference to this. This method is annotated @CheckReturnValue because you have to pay attention to the return value; if you simply invoke insert on a Tree and ignore the return value, your code is almost certainly wrong.Specified by: [insert](http://docs.google.com/searchTree/Tree.html#insert-K-V-) in interface [Tree](http://docs.google.com/searchTree/Tree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html) extends java.lang.Comparable<[K](http://docs.google.com/searchTree/EmptyTree.html)>,[V](http://docs.google.com/searchTree/EmptyTree.html)> Parameters: key - -- Key value - -- Value that the key maps to Returns: -- updated tree

#### delete public [Tree](http://docs.google.com/searchTree/Tree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html),[V](http://docs.google.com/searchTree/EmptyTree.html)> delete([K](http://docs.google.com/searchTree/EmptyTree.html) key) Description copied from interface: [Tree](http://docs.google.com/searchTree/Tree.html#delete-K-) Delete any binding the key has in this tree. If the key isn't bound, this is a no-op This method returns a reference to an Tree that represents the updated value. In many, but not all cases, the method may just return a reference to this. This method is annotated @CheckReturnValue because you have to pay attention to the return value; if you simply invoke delete on a Tree and ignore the return value, your code is almost certainly wrong.Specified by: [delete](http://docs.google.com/searchTree/Tree.html#delete-K-) in interface [Tree](http://docs.google.com/searchTree/Tree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html) extends java.lang.Comparable<[K](http://docs.google.com/searchTree/EmptyTree.html)>,[V](http://docs.google.com/searchTree/EmptyTree.html)> Parameters: key - -- Key Returns: -- updated tree

#### max public [K](http://docs.google.com/searchTree/EmptyTree.html) max() throws searchTree.TreeIsEmptyException Description copied from interface: [Tree](http://docs.google.com/searchTree/Tree.html#max--) Return the maximum key in the subtreeSpecified by: [max](http://docs.google.com/searchTree/Tree.html#max--) in interface [Tree](http://docs.google.com/searchTree/Tree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html) extends java.lang.Comparable<[K](http://docs.google.com/searchTree/EmptyTree.html)>,[V](http://docs.google.com/searchTree/EmptyTree.html)> Returns: maximum key Throws: TreeIsEmptyException - if the tree is empty

#### min public [K](http://docs.google.com/searchTree/EmptyTree.html) min() throws searchTree.TreeIsEmptyException Description copied from interface: [Tree](http://docs.google.com/searchTree/Tree.html#min--) Return the minimum key in the subtreeSpecified by: [min](http://docs.google.com/searchTree/Tree.html#min--) in interface [Tree](http://docs.google.com/searchTree/Tree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html) extends java.lang.Comparable<[K](http://docs.google.com/searchTree/EmptyTree.html)>,[V](http://docs.google.com/searchTree/EmptyTree.html)> Returns: minimum key Throws: TreeIsEmptyException - if the tree is empty

#### size public int size() Description copied from interface: [Tree](http://docs.google.com/searchTree/Tree.html#size--) Return number of keys that are bound in this tree.Specified by: [size](http://docs.google.com/searchTree/Tree.html#size--) in interface [Tree](http://docs.google.com/searchTree/Tree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html) extends java.lang.Comparable<[K](http://docs.google.com/searchTree/EmptyTree.html)>,[V](http://docs.google.com/searchTree/EmptyTree.html)> Returns: number of keys that are bound in this tree.

#### addKeysToCollection public void addKeysToCollection(java.util.Collection<[K](http://docs.google.com/searchTree/EmptyTree.html)> c) Description copied from interface: [Tree](http://docs.google.com/searchTree/Tree.html#addKeysToCollection-java.util.Collection-) Add all keys bound in this tree to the collection c. The elements must be added in their sorted order.Specified by: [addKeysToCollection](http://docs.google.com/searchTree/Tree.html#addKeysToCollection-java.util.Collection-) in interface [Tree](http://docs.google.com/searchTree/Tree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html) extends java.lang.Comparable<[K](http://docs.google.com/searchTree/EmptyTree.html)>,[V](http://docs.google.com/searchTree/EmptyTree.html)> Parameters: c - - A list that acts as an accumulator for keys. Keys are inserted in the list in increasing order. You may not use any sorting method or Collections.sort to keep the list sorted.

#### subTree public [Tree](http://docs.google.com/searchTree/Tree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html),[V](http://docs.google.com/searchTree/EmptyTree.html)> subTree([K](http://docs.google.com/searchTree/EmptyTree.html) fromKey, [K](http://docs.google.com/searchTree/EmptyTree.html) toKey) Description copied from interface: [Tree](http://docs.google.com/searchTree/Tree.html#subTree-K-K-) Returns a Tree containing all entries between fromKey and toKey, inclusiveSpecified by: [subTree](http://docs.google.com/searchTree/Tree.html#subTree-K-K-) in interface [Tree](http://docs.google.com/searchTree/Tree.html)<[K](http://docs.google.com/searchTree/EmptyTree.html) extends java.lang.Comparable<[K](http://docs.google.com/searchTree/EmptyTree.html)>,[V](http://docs.google.com/searchTree/EmptyTree.html)> Parameters: fromKey - - Lower bound value for keys in subtree toKey - - Upper bound value for keys in subtree Returns: Tree containing all entries between fromKey and toKey, inclusive

[Skip navigation links](#4i7ojhp)

* [Package](http://docs.google.com/searchTree/package-summary.html)
* Class
* [Use](http://docs.google.com/class-use/EmptyTree.html)
* [Tree](http://docs.google.com/package-tree.html)
* [Deprecated](http://docs.google.com/deprecated-list.html)
* [Index](http://docs.google.com/index-files/index-1.html)
* [Help](http://docs.google.com/help-doc.html)
* Prev Class
* [Next Class](http://docs.google.com/searchTree/NonEmptyTree.html)
* [Frames](http://docs.google.com/index.html?searchTree/EmptyTree.html)
* [No Frames](http://docs.google.com/EmptyTree.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* Field |
* Constr |
* [Method](#3znysh7)
* Detail:
* Field |
* Constr |
* [Method](#tyjcwt)

[Web Accessibility](https://www.umd.edu/web-accessibility)