

## Interface Set<E>

### Method Summary

boolean	<a href="#">add</a> ( <a href="#">E</a> e) Adds the specified element to this set if it is not already present (optional operation).
boolean	<a href="#">addAll</a> ( <a href="#">Collection</a> <? extends <a href="#">E</a> > c) Adds all of the elements in the specified collection to this set if they're not already present (optional operation).
void	<a href="#">clear</a> () Removes all of the elements from this set (optional operation).
boolean	<a href="#">contains</a> ( <a href="#">Object</a> o) Returns <code>true</code> if this set contains the specified element.
boolean	<a href="#">containsAll</a> ( <a href="#">Collection</a> <?> c) Returns <code>true</code> if this set contains all of the elements of the specified collection.
boolean	<a href="#">equals</a> ( <a href="#">Object</a> o) Compares the specified object with this set for equality.
int	<a href="#">hashCode</a> () Returns the hash code value for this set.
boolean	<a href="#">isEmpty</a> () Returns <code>true</code> if this set contains no elements.
<a href="#">Iterator</a> < <a href="#">E</a> >	<a href="#">iterator</a> () Returns an iterator over the elements in this set.
boolean	<a href="#">remove</a> ( <a href="#">Object</a> o) Removes the specified element from this set if it is present (optional operation).
boolean	<a href="#">removeAll</a> ( <a href="#">Collection</a> <?> c) Removes from this set all of its elements that are contained in the specified collection (optional operation).
boolean	<a href="#">retainAll</a> ( <a href="#">Collection</a> <?> c) Retains only the elements in this set that are contained in the specified collection (optional operation).
int	<a href="#">size</a> () Returns the number of elements in this set (its cardinality).
<a href="#">Object</a> []	<a href="#">toArray</a> () Returns an array containing all of the elements in this set.
<T> T[]	<a href="#">toArray</a> (T[] a) Returns an array containing all of the elements in this set; the runtime type of the returned array is that of the specified array.

# Interface Map<K,V>

## Nested Class Summary

static interface	<a href="#">Map.Entry&lt;K,V&gt;</a> A map entry (key-value pair).
------------------	---

## Method Summary

void	<a href="#">clear</a> () Removes all of the mappings from this map (optional operation).
boolean	<a href="#">containsKey</a> ( <a href="#">Object</a> key) Returns true if this map contains a mapping for the specified key.
boolean	<a href="#">containsValue</a> ( <a href="#">Object</a> value) Returns true if this map maps one or more keys to the specified value.
<a href="#">Set&lt;Map.Entry&lt;K,V&gt;&gt;</a>	<a href="#">entrySet</a> () Returns a <a href="#">Set</a> view of the mappings contained in this map.
boolean	<a href="#">equals</a> ( <a href="#">Object</a> o) Compares the specified object with this map for equality.
<a href="#">V</a>	<a href="#">get</a> ( <a href="#">Object</a> key) Returns the value to which the specified key is mapped, or null if this map contains no mapping for the key.
int	<a href="#">hashCode</a> () Returns the hash code value for this map.
boolean	<a href="#">isEmpty</a> () Returns true if this map contains no key-value mappings.
<a href="#">Set&lt;K&gt;</a>	<a href="#">keySet</a> () Returns a <a href="#">Set</a> view of the keys contained in this map.
<a href="#">V</a>	<a href="#">put</a> ( <a href="#">K</a> key, <a href="#">V</a> value) Associates the specified value with the specified key in this map (optional operation).
void	<a href="#">putAll</a> ( <a href="#">Map</a> <? extends <a href="#">K</a> ,? extends <a href="#">V</a> > m) Copies all of the mappings from the specified map to this map (optional operation).
<a href="#">V</a>	<a href="#">remove</a> ( <a href="#">Object</a> key) Removes the mapping for a key from this map if it is present (optional operation).
int	<a href="#">size</a> () Returns the number of key-value mappings in this map.
<a href="#">Collection&lt;V&gt;</a>	<a href="#">values</a> () Returns a <a href="#">Collection</a> view of the values contained in this map.

# Class `HashMap<K, V>`

All Implemented Interfaces: [Serializable](#), [Cloneable](#), [Map<K, V>](#)

## Constructor Summary

[HashMap](#)()

Constructs an empty `HashMap` with the default initial capacity (16) and the default load factor (0.75).

[HashMap](#)(int initialCapacity)

Constructs an empty `HashMap` with the specified initial capacity and the default load factor (0.75).

[HashMap](#)(int initialCapacity, float loadFactor)

Constructs an empty `HashMap` with the specified initial capacity and load factor.

[HashMap](#)([Map](#)<? extends [K](#), ? extends [V](#)> m)

Constructs a new `HashMap` with the same mappings as the specified `Map`.

## Method Summary

void	<a href="#">clear</a> ()	Removes all of the mappings from this map.
<a href="#">Object</a>	<a href="#">clone</a> ()	Returns a shallow copy of this <code>HashMap</code> instance: the keys and values themselves are not cloned.
boolean	<a href="#">containsKey</a> ( <a href="#">Object</a> key)	Returns <code>true</code> if this map contains a mapping for the specified key.
boolean	<a href="#">containsValue</a> ( <a href="#">Object</a> value)	Returns <code>true</code> if this map maps one or more keys to the specified value.
<a href="#">Set</a> < <a href="#">Map.Entry</a> < <a href="#">K</a> , <a href="#">V</a> >>	<a href="#">entrySet</a> ()	Returns a <a href="#">Set</a> view of the mappings contained in this map.
<a href="#">V</a>	<a href="#">get</a> ( <a href="#">Object</a> key)	Returns the value to which the specified key is mapped, or <code>null</code> if this map contains no mapping for the key.
boolean	<a href="#">isEmpty</a> ()	Returns <code>true</code> if this map contains no key-value mappings.
<a href="#">Set</a> < <a href="#">K</a> >	<a href="#">keySet</a> ()	Returns a <a href="#">Set</a> view of the keys contained in this map.
<a href="#">V</a>	<a href="#">put</a> ( <a href="#">K</a> key, <a href="#">V</a> value)	Associates the specified value with the specified key in this map.
void	<a href="#">putAll</a> ( <a href="#">Map</a> <? extends <a href="#">K</a> , ? extends <a href="#">V</a> > m)	Copies all of the mappings from the specified map to this map.
<a href="#">V</a>	<a href="#">remove</a> ( <a href="#">Object</a> key)	Removes the mapping for the specified key from this map if present.
int	<a href="#">size</a> ()	Returns the number of key-value mappings in this map.
<a href="#">Collection</a> < <a href="#">V</a> >	<a href="#">values</a> ()	Returns a <a href="#">Collection</a> view of the values contained in this map.

## Interface Collection<E>

Method Summary	
boolean	<a href="#"><u>add</u></a> ( <a href="#"><u>E</u></a> e) Ensures that this collection contains the specified element (optional operation).
boolean	<a href="#"><u>addAll</u></a> ( <a href="#"><u>Collection</u></a> <? extends <a href="#"><u>E</u></a> > c) Adds all of the elements in the specified collection to this collection (optional operation).
void	<a href="#"><u>clear</u></a> () Removes all of the elements from this collection (optional operation).
boolean	<a href="#"><u>contains</u></a> ( <a href="#"><u>Object</u></a> o) Returns true if this collection contains the specified element.
boolean	<a href="#"><u>containsAll</u></a> ( <a href="#"><u>Collection</u></a> <?> c) Returns true if this collection contains all of the elements in the specified collection.
boolean	<a href="#"><u>equals</u></a> ( <a href="#"><u>Object</u></a> o) Compares the specified object with this collection for equality.
int	<a href="#"><u>hashCode</u></a> () Returns the hash code value for this collection.
boolean	<a href="#"><u>isEmpty</u></a> () Returns true if this collection contains no elements.
<a href="#"><u>Iterator</u></a> < <a href="#"><u>E</u></a> >	<a href="#"><u>iterator</u></a> () Returns an iterator over the elements in this collection.
boolean	<a href="#"><u>remove</u></a> ( <a href="#"><u>Object</u></a> o) Removes a single instance of the specified element from this collection, if it is present (optional operation).
boolean	<a href="#"><u>removeAll</u></a> ( <a href="#"><u>Collection</u></a> <?> c) Removes all of this collection's elements that are also contained in the specified collection (optional operation).
boolean	<a href="#"><u>retainAll</u></a> ( <a href="#"><u>Collection</u></a> <?> c) Retains only the elements in this collection that are contained in the specified collection (optional operation).
int	<a href="#"><u>size</u></a> () Returns the number of elements in this collection.
<a href="#"><u>Object</u></a> []	<a href="#"><u>toArray</u></a> () Returns an array containing all of the elements in this collection.
<T> T[]	<a href="#"><u>toArray</u></a> (T[] a) Returns an array containing all of the elements in this collection; the runtime type of the returned array is that of the specified array.