| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/NavigableMap.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) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/util/MissingResourceException.html)   [**NEXT CLASS**](http://docs.google.com/java/util/NavigableSet.html) | [**FRAMES**](http://docs.google.com/index.html?java/util/NavigableMap.html)    [**NO FRAMES**](http://docs.google.com/NavigableMap.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#tyjcwt) | DETAIL: FIELD | CONSTR | [METHOD](#4d34og8) |

## **java.util**

Interface NavigableMap<K,V>

**Type Parameters:**K - the type of keys maintained by this mapV - the type of mapped values **All Superinterfaces:** [Map](http://docs.google.com/java/util/Map.html)<K,V>, [SortedMap](http://docs.google.com/java/util/SortedMap.html)<K,V> **All Known Subinterfaces:** [ConcurrentNavigableMap](http://docs.google.com/java/util/concurrent/ConcurrentNavigableMap.html)<K,V> **All Known Implementing Classes:** [ConcurrentSkipListMap](http://docs.google.com/java/util/concurrent/ConcurrentSkipListMap.html), [TreeMap](http://docs.google.com/java/util/TreeMap.html)

public interface **NavigableMap<K,V>**extends [SortedMap](http://docs.google.com/java/util/SortedMap.html)<K,V>

A [SortedMap](http://docs.google.com/java/util/SortedMap.html) extended with navigation methods returning the closest matches for given search targets. Methods lowerEntry, floorEntry, ceilingEntry, and higherEntry return Map.Entry objects associated with keys respectively less than, less than or equal, greater than or equal, and greater than a given key, returning null if there is no such key. Similarly, methods lowerKey, floorKey, ceilingKey, and higherKey return only the associated keys. All of these methods are designed for locating, not traversing entries.

A NavigableMap may be accessed and traversed in either ascending or descending key order. The descendingMap method returns a view of the map with the senses of all relational and directional methods inverted. The performance of ascending operations and views is likely to be faster than that of descending ones. Methods subMap, headMap, and tailMap differ from the like-named SortedMap methods in accepting additional arguments describing whether lower and upper bounds are inclusive versus exclusive. Submaps of any NavigableMap must implement the NavigableMap interface.

This interface additionally defines methods firstEntry, pollFirstEntry, lastEntry, and pollLastEntry that return and/or remove the least and greatest mappings, if any exist, else returning null.

Implementations of entry-returning methods are expected to return Map.Entry pairs representing snapshots of mappings at the time they were produced, and thus generally do *not* support the optional Entry.setValue method. Note however that it is possible to change mappings in the associated map using method put.

Methods [subMap(K, K)](http://docs.google.com/java/util/NavigableMap.html#subMap(K,%20K)), [headMap(K)](http://docs.google.com/java/util/NavigableMap.html#headMap(K)), and [tailMap(K)](http://docs.google.com/java/util/NavigableMap.html#tailMap(K)) are specified to return SortedMap to allow existing implementations of SortedMap to be compatibly retrofitted to implement NavigableMap, but extensions and implementations of this interface are encouraged to override these methods to return NavigableMap. Similarly, [SortedMap.keySet()](http://docs.google.com/java/util/SortedMap.html#keySet()) can be overriden to return NavigableSet.

This interface is a member of the  [Java Collections Framework](http://docs.google.com/technotes/guides/collections/index.html).

**Since:** 1.6

| **Nested Class Summary** | |
| --- | --- |

| **Nested classes/interfaces inherited from interface java.util.**[**Map**](http://docs.google.com/java/util/Map.html) |
| --- |
| [Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/Map.Entry.html),[V](http://docs.google.com/java/util/Map.Entry.html)> |

| **Method Summary** | |
| --- | --- |
| [Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**ceilingEntry**](http://docs.google.com/java/util/NavigableMap.html#ceilingEntry(K))([K](http://docs.google.com/java/util/NavigableMap.html) key)            Returns a key-value mapping associated with the least key greater than or equal to the given key, or null if there is no such key. |
| [K](http://docs.google.com/java/util/NavigableMap.html) | [**ceilingKey**](http://docs.google.com/java/util/NavigableMap.html#ceilingKey(K))([K](http://docs.google.com/java/util/NavigableMap.html) key)            Returns the least key greater than or equal to the given key, or null if there is no such key. |
| [NavigableSet](http://docs.google.com/java/util/NavigableSet.html)<[K](http://docs.google.com/java/util/NavigableMap.html)> | [**descendingKeySet**](http://docs.google.com/java/util/NavigableMap.html#descendingKeySet())()            Returns a reverse order [NavigableSet](http://docs.google.com/java/util/NavigableSet.html) view of the keys contained in this map. |
| [NavigableMap](http://docs.google.com/java/util/NavigableMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**descendingMap**](http://docs.google.com/java/util/NavigableMap.html#descendingMap())()            Returns a reverse order view of the mappings contained in this map. |
| [Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**firstEntry**](http://docs.google.com/java/util/NavigableMap.html#firstEntry())()            Returns a key-value mapping associated with the least key in this map, or null if the map is empty. |
| [Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**floorEntry**](http://docs.google.com/java/util/NavigableMap.html#floorEntry(K))([K](http://docs.google.com/java/util/NavigableMap.html) key)            Returns a key-value mapping associated with the greatest key less than or equal to the given key, or null if there is no such key. |
| [K](http://docs.google.com/java/util/NavigableMap.html) | [**floorKey**](http://docs.google.com/java/util/NavigableMap.html#floorKey(K))([K](http://docs.google.com/java/util/NavigableMap.html) key)            Returns the greatest key less than or equal to the given key, or null if there is no such key. |
| [SortedMap](http://docs.google.com/java/util/SortedMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**headMap**](http://docs.google.com/java/util/NavigableMap.html#headMap(K))([K](http://docs.google.com/java/util/NavigableMap.html) toKey)            Returns a view of the portion of this map whose keys are strictly less than toKey. |
| [NavigableMap](http://docs.google.com/java/util/NavigableMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**headMap**](http://docs.google.com/java/util/NavigableMap.html#headMap(K,%20boolean))([K](http://docs.google.com/java/util/NavigableMap.html) toKey, boolean inclusive)            Returns a view of the portion of this map whose keys are less than (or equal to, if inclusive is true) toKey. |
| [Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**higherEntry**](http://docs.google.com/java/util/NavigableMap.html#higherEntry(K))([K](http://docs.google.com/java/util/NavigableMap.html) key)            Returns a key-value mapping associated with the least key strictly greater than the given key, or null if there is no such key. |
| [K](http://docs.google.com/java/util/NavigableMap.html) | [**higherKey**](http://docs.google.com/java/util/NavigableMap.html#higherKey(K))([K](http://docs.google.com/java/util/NavigableMap.html) key)            Returns the least key strictly greater than the given key, or null if there is no such key. |
| [Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**lastEntry**](http://docs.google.com/java/util/NavigableMap.html#lastEntry())()            Returns a key-value mapping associated with the greatest key in this map, or null if the map is empty. |
| [Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**lowerEntry**](http://docs.google.com/java/util/NavigableMap.html#lowerEntry(K))([K](http://docs.google.com/java/util/NavigableMap.html) key)            Returns a key-value mapping associated with the greatest key strictly less than the given key, or null if there is no such key. |
| [K](http://docs.google.com/java/util/NavigableMap.html) | [**lowerKey**](http://docs.google.com/java/util/NavigableMap.html#lowerKey(K))([K](http://docs.google.com/java/util/NavigableMap.html) key)            Returns the greatest key strictly less than the given key, or null if there is no such key. |
| [NavigableSet](http://docs.google.com/java/util/NavigableSet.html)<[K](http://docs.google.com/java/util/NavigableMap.html)> | [**navigableKeySet**](http://docs.google.com/java/util/NavigableMap.html#navigableKeySet())()            Returns a [NavigableSet](http://docs.google.com/java/util/NavigableSet.html) view of the keys contained in this map. |
| [Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**pollFirstEntry**](http://docs.google.com/java/util/NavigableMap.html#pollFirstEntry())()            Removes and returns a key-value mapping associated with the least key in this map, or null if the map is empty. |
| [Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**pollLastEntry**](http://docs.google.com/java/util/NavigableMap.html#pollLastEntry())()            Removes and returns a key-value mapping associated with the greatest key in this map, or null if the map is empty. |
| [NavigableMap](http://docs.google.com/java/util/NavigableMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**subMap**](http://docs.google.com/java/util/NavigableMap.html#subMap(K,%20boolean,%20K,%20boolean))([K](http://docs.google.com/java/util/NavigableMap.html) fromKey, boolean fromInclusive, [K](http://docs.google.com/java/util/NavigableMap.html) toKey, boolean toInclusive)            Returns a view of the portion of this map whose keys range from fromKey to toKey. |
| [SortedMap](http://docs.google.com/java/util/SortedMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**subMap**](http://docs.google.com/java/util/NavigableMap.html#subMap(K,%20K))([K](http://docs.google.com/java/util/NavigableMap.html) fromKey, [K](http://docs.google.com/java/util/NavigableMap.html) toKey)            Returns a view of the portion of this map whose keys range from fromKey, inclusive, to toKey, exclusive. |
| [SortedMap](http://docs.google.com/java/util/SortedMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**tailMap**](http://docs.google.com/java/util/NavigableMap.html#tailMap(K))([K](http://docs.google.com/java/util/NavigableMap.html) fromKey)            Returns a view of the portion of this map whose keys are greater than or equal to fromKey. |
| [NavigableMap](http://docs.google.com/java/util/NavigableMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> | [**tailMap**](http://docs.google.com/java/util/NavigableMap.html#tailMap(K,%20boolean))([K](http://docs.google.com/java/util/NavigableMap.html) fromKey, boolean inclusive)            Returns a view of the portion of this map whose keys are greater than (or equal to, if inclusive is true) fromKey. |

| **Methods inherited from interface java.util.**[**SortedMap**](http://docs.google.com/java/util/SortedMap.html) |
| --- |
| [comparator](http://docs.google.com/java/util/SortedMap.html#comparator()), [entrySet](http://docs.google.com/java/util/SortedMap.html#entrySet()), [firstKey](http://docs.google.com/java/util/SortedMap.html#firstKey()), [keySet](http://docs.google.com/java/util/SortedMap.html#keySet()), [lastKey](http://docs.google.com/java/util/SortedMap.html#lastKey()), [values](http://docs.google.com/java/util/SortedMap.html#values()) |

| **Methods inherited from interface java.util.**[**Map**](http://docs.google.com/java/util/Map.html) |
| --- |
| [clear](http://docs.google.com/java/util/Map.html#clear()), [containsKey](http://docs.google.com/java/util/Map.html#containsKey(java.lang.Object)), [containsValue](http://docs.google.com/java/util/Map.html#containsValue(java.lang.Object)), [equals](http://docs.google.com/java/util/Map.html#equals(java.lang.Object)), [get](http://docs.google.com/java/util/Map.html#get(java.lang.Object)), [hashCode](http://docs.google.com/java/util/Map.html#hashCode()), [isEmpty](http://docs.google.com/java/util/Map.html#isEmpty()), [put](http://docs.google.com/java/util/Map.html#put(K,%20V)), [putAll](http://docs.google.com/java/util/Map.html#putAll(java.util.Map)), [remove](http://docs.google.com/java/util/Map.html#remove(java.lang.Object)), [size](http://docs.google.com/java/util/Map.html#size()) |

| **Method Detail** |
| --- |

### lowerEntry

[Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **lowerEntry**([K](http://docs.google.com/java/util/NavigableMap.html) key)

Returns a key-value mapping associated with the greatest key strictly less than the given key, or null if there is no such key.

**Parameters:**key - the key **Returns:**an entry with the greatest key less than key, or null if there is no such key **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if the specified key cannot be compared with the keys currently in the map [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the specified key is null and this map does not permit null keys

### lowerKey

[K](http://docs.google.com/java/util/NavigableMap.html) **lowerKey**([K](http://docs.google.com/java/util/NavigableMap.html) key)

Returns the greatest key strictly less than the given key, or null if there is no such key.

**Parameters:**key - the key **Returns:**the greatest key less than key, or null if there is no such key **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if the specified key cannot be compared with the keys currently in the map [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the specified key is null and this map does not permit null keys

### floorEntry

[Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **floorEntry**([K](http://docs.google.com/java/util/NavigableMap.html) key)

Returns a key-value mapping associated with the greatest key less than or equal to the given key, or null if there is no such key.

**Parameters:**key - the key **Returns:**an entry with the greatest key less than or equal to key, or null if there is no such key **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if the specified key cannot be compared with the keys currently in the map [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the specified key is null and this map does not permit null keys

### floorKey

[K](http://docs.google.com/java/util/NavigableMap.html) **floorKey**([K](http://docs.google.com/java/util/NavigableMap.html) key)

Returns the greatest key less than or equal to the given key, or null if there is no such key.

**Parameters:**key - the key **Returns:**the greatest key less than or equal to key, or null if there is no such key **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if the specified key cannot be compared with the keys currently in the map [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the specified key is null and this map does not permit null keys

### ceilingEntry

[Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **ceilingEntry**([K](http://docs.google.com/java/util/NavigableMap.html) key)

Returns a key-value mapping associated with the least key greater than or equal to the given key, or null if there is no such key.

**Parameters:**key - the key **Returns:**an entry with the least key greater than or equal to key, or null if there is no such key **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if the specified key cannot be compared with the keys currently in the map [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the specified key is null and this map does not permit null keys

### ceilingKey

[K](http://docs.google.com/java/util/NavigableMap.html) **ceilingKey**([K](http://docs.google.com/java/util/NavigableMap.html) key)

Returns the least key greater than or equal to the given key, or null if there is no such key.

**Parameters:**key - the key **Returns:**the least key greater than or equal to key, or null if there is no such key **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if the specified key cannot be compared with the keys currently in the map [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the specified key is null and this map does not permit null keys

### higherEntry

[Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **higherEntry**([K](http://docs.google.com/java/util/NavigableMap.html) key)

Returns a key-value mapping associated with the least key strictly greater than the given key, or null if there is no such key.

**Parameters:**key - the key **Returns:**an entry with the least key greater than key, or null if there is no such key **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if the specified key cannot be compared with the keys currently in the map [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the specified key is null and this map does not permit null keys

### higherKey

[K](http://docs.google.com/java/util/NavigableMap.html) **higherKey**([K](http://docs.google.com/java/util/NavigableMap.html) key)

Returns the least key strictly greater than the given key, or null if there is no such key.

**Parameters:**key - the key **Returns:**the least key greater than key, or null if there is no such key **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if the specified key cannot be compared with the keys currently in the map [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if the specified key is null and this map does not permit null keys

### firstEntry

[Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **firstEntry**()

Returns a key-value mapping associated with the least key in this map, or null if the map is empty.

**Returns:**an entry with the least key, or null if this map is empty

### lastEntry

[Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **lastEntry**()

Returns a key-value mapping associated with the greatest key in this map, or null if the map is empty.

**Returns:**an entry with the greatest key, or null if this map is empty

### pollFirstEntry

[Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **pollFirstEntry**()

Removes and returns a key-value mapping associated with the least key in this map, or null if the map is empty.

**Returns:**the removed first entry of this map, or null if this map is empty

### pollLastEntry

[Map.Entry](http://docs.google.com/java/util/Map.Entry.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **pollLastEntry**()

Removes and returns a key-value mapping associated with the greatest key in this map, or null if the map is empty.

**Returns:**the removed last entry of this map, or null if this map is empty

### descendingMap

[NavigableMap](http://docs.google.com/java/util/NavigableMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **descendingMap**()

Returns a reverse order view of the mappings contained in this map. The descending map is backed by this map, so changes to the map are reflected in the descending map, and vice-versa. If either map is modified while an iteration over a collection view of either map is in progress (except through the iterator's own remove operation), the results of the iteration are undefined.

The returned map has an ordering equivalent to [Collections.reverseOrder](http://docs.google.com/java/util/Collections.html#reverseOrder(java.util.Comparator))(comparator()). The expression m.descendingMap().descendingMap() returns a view of m essentially equivalent to m.

**Returns:**a reverse order view of this map

### navigableKeySet

[NavigableSet](http://docs.google.com/java/util/NavigableSet.html)<[K](http://docs.google.com/java/util/NavigableMap.html)> **navigableKeySet**()

Returns a [NavigableSet](http://docs.google.com/java/util/NavigableSet.html) view of the keys contained in this map. The set's iterator returns the keys in ascending order. The set is backed by the map, so changes to the map are reflected in the set, and vice-versa. If the map is modified while an iteration over the set is in progress (except through the iterator's own remove operation), the results of the iteration are undefined. The set supports element removal, which removes the corresponding mapping from the map, via the Iterator.remove, Set.remove, removeAll, retainAll, and clear operations. It does not support the add or addAll operations.

**Returns:**a navigable set view of the keys in this map

### descendingKeySet

[NavigableSet](http://docs.google.com/java/util/NavigableSet.html)<[K](http://docs.google.com/java/util/NavigableMap.html)> **descendingKeySet**()

Returns a reverse order [NavigableSet](http://docs.google.com/java/util/NavigableSet.html) view of the keys contained in this map. The set's iterator returns the keys in descending order. The set is backed by the map, so changes to the map are reflected in the set, and vice-versa. If the map is modified while an iteration over the set is in progress (except through the iterator's own remove operation), the results of the iteration are undefined. The set supports element removal, which removes the corresponding mapping from the map, via the Iterator.remove, Set.remove, removeAll, retainAll, and clear operations. It does not support the add or addAll operations.

**Returns:**a reverse order navigable set view of the keys in this map

### subMap

[NavigableMap](http://docs.google.com/java/util/NavigableMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **subMap**([K](http://docs.google.com/java/util/NavigableMap.html) fromKey,  
 boolean fromInclusive,  
 [K](http://docs.google.com/java/util/NavigableMap.html) toKey,  
 boolean toInclusive)

Returns a view of the portion of this map whose keys range from fromKey to toKey. If fromKey and toKey are equal, the returned map is empty unless fromExclusive and toExclusive are both true. The returned map is backed by this map, so changes in the returned map are reflected in this map, and vice-versa. The returned map supports all optional map operations that this map supports.

The returned map will throw an IllegalArgumentException on an attempt to insert a key outside of its range, or to construct a submap either of whose endpoints lie outside its range.

**Parameters:**fromKey - low endpoint of the keys in the returned mapfromInclusive - true if the low endpoint is to be included in the returned viewtoKey - high endpoint of the keys in the returned maptoInclusive - true if the high endpoint is to be included in the returned view **Returns:**a view of the portion of this map whose keys range from fromKey to toKey **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if fromKey and toKey cannot be compared to one another using this map's comparator (or, if the map has no comparator, using natural ordering). Implementations may, but are not required to, throw this exception if fromKey or toKey cannot be compared to keys currently in the map. [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if fromKey or toKey is null and this map does not permit null keys [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if fromKey is greater than toKey; or if this map itself has a restricted range, and fromKey or toKey lies outside the bounds of the range

### headMap

[NavigableMap](http://docs.google.com/java/util/NavigableMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **headMap**([K](http://docs.google.com/java/util/NavigableMap.html) toKey,  
 boolean inclusive)

Returns a view of the portion of this map whose keys are less than (or equal to, if inclusive is true) toKey. The returned map is backed by this map, so changes in the returned map are reflected in this map, and vice-versa. The returned map supports all optional map operations that this map supports.

The returned map will throw an IllegalArgumentException on an attempt to insert a key outside its range.

**Parameters:**toKey - high endpoint of the keys in the returned mapinclusive - true if the high endpoint is to be included in the returned view **Returns:**a view of the portion of this map whose keys are less than (or equal to, if inclusive is true) toKey **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if toKey is not compatible with this map's comparator (or, if the map has no comparator, if toKey does not implement [Comparable](http://docs.google.com/java/lang/Comparable.html)). Implementations may, but are not required to, throw this exception if toKey cannot be compared to keys currently in the map. [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if toKey is null and this map does not permit null keys [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if this map itself has a restricted range, and toKey lies outside the bounds of the range

### tailMap

[NavigableMap](http://docs.google.com/java/util/NavigableMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **tailMap**([K](http://docs.google.com/java/util/NavigableMap.html) fromKey,  
 boolean inclusive)

Returns a view of the portion of this map whose keys are greater than (or equal to, if inclusive is true) fromKey. The returned map is backed by this map, so changes in the returned map are reflected in this map, and vice-versa. The returned map supports all optional map operations that this map supports.

The returned map will throw an IllegalArgumentException on an attempt to insert a key outside its range.

**Parameters:**fromKey - low endpoint of the keys in the returned mapinclusive - true if the low endpoint is to be included in the returned view **Returns:**a view of the portion of this map whose keys are greater than (or equal to, if inclusive is true) fromKey **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if fromKey is not compatible with this map's comparator (or, if the map has no comparator, if fromKey does not implement [Comparable](http://docs.google.com/java/lang/Comparable.html)). Implementations may, but are not required to, throw this exception if fromKey cannot be compared to keys currently in the map. [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if fromKey is null and this map does not permit null keys [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if this map itself has a restricted range, and fromKey lies outside the bounds of the range

### subMap

[SortedMap](http://docs.google.com/java/util/SortedMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **subMap**([K](http://docs.google.com/java/util/NavigableMap.html) fromKey,  
 [K](http://docs.google.com/java/util/NavigableMap.html) toKey)

Returns a view of the portion of this map whose keys range from fromKey, inclusive, to toKey, exclusive. (If fromKey and toKey are equal, the returned map is empty.) The returned map is backed by this map, so changes in the returned map are reflected in this map, and vice-versa. The returned map supports all optional map operations that this map supports.

The returned map will throw an IllegalArgumentException on an attempt to insert a key outside its range.

Equivalent to subMap(fromKey, true, toKey, false).

**Specified by:**[subMap](http://docs.google.com/java/util/SortedMap.html#subMap(K,%20K)) in interface [SortedMap](http://docs.google.com/java/util/SortedMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **Parameters:**fromKey - low endpoint (inclusive) of the keys in the returned maptoKey - high endpoint (exclusive) of the keys in the returned map **Returns:**a view of the portion of this map whose keys range from fromKey, inclusive, to toKey, exclusive **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if fromKey and toKey cannot be compared to one another using this map's comparator (or, if the map has no comparator, using natural ordering). Implementations may, but are not required to, throw this exception if fromKey or toKey cannot be compared to keys currently in the map. [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if fromKey or toKey is null and this map does not permit null keys [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if fromKey is greater than toKey; or if this map itself has a restricted range, and fromKey or toKey lies outside the bounds of the range

### headMap

[SortedMap](http://docs.google.com/java/util/SortedMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **headMap**([K](http://docs.google.com/java/util/NavigableMap.html) toKey)

Returns a view of the portion of this map whose keys are strictly less than toKey. The returned map is backed by this map, so changes in the returned map are reflected in this map, and vice-versa. The returned map supports all optional map operations that this map supports.

The returned map will throw an IllegalArgumentException on an attempt to insert a key outside its range.

Equivalent to headMap(toKey, false).

**Specified by:**[headMap](http://docs.google.com/java/util/SortedMap.html#headMap(K)) in interface [SortedMap](http://docs.google.com/java/util/SortedMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **Parameters:**toKey - high endpoint (exclusive) of the keys in the returned map **Returns:**a view of the portion of this map whose keys are strictly less than toKey **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if toKey is not compatible with this map's comparator (or, if the map has no comparator, if toKey does not implement [Comparable](http://docs.google.com/java/lang/Comparable.html)). Implementations may, but are not required to, throw this exception if toKey cannot be compared to keys currently in the map. [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if toKey is null and this map does not permit null keys [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if this map itself has a restricted range, and toKey lies outside the bounds of the range

### tailMap

[SortedMap](http://docs.google.com/java/util/SortedMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **tailMap**([K](http://docs.google.com/java/util/NavigableMap.html) fromKey)

Returns a view of the portion of this map whose keys are greater than or equal to fromKey. The returned map is backed by this map, so changes in the returned map are reflected in this map, and vice-versa. The returned map supports all optional map operations that this map supports.

The returned map will throw an IllegalArgumentException on an attempt to insert a key outside its range.

Equivalent to tailMap(fromKey, true).

**Specified by:**[tailMap](http://docs.google.com/java/util/SortedMap.html#tailMap(K)) in interface [SortedMap](http://docs.google.com/java/util/SortedMap.html)<[K](http://docs.google.com/java/util/NavigableMap.html),[V](http://docs.google.com/java/util/NavigableMap.html)> **Parameters:**fromKey - low endpoint (inclusive) of the keys in the returned map **Returns:**a view of the portion of this map whose keys are greater than or equal to fromKey **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if fromKey is not compatible with this map's comparator (or, if the map has no comparator, if fromKey does not implement [Comparable](http://docs.google.com/java/lang/Comparable.html)). Implementations may, but are not required to, throw this exception if fromKey cannot be compared to keys currently in the map. [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if fromKey is null and this map does not permit null keys [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if this map itself has a restricted range, and fromKey lies outside the bounds of the range

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/NavigableMap.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) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/util/MissingResourceException.html)   [**NEXT CLASS**](http://docs.google.com/java/util/NavigableSet.html) | [**FRAMES**](http://docs.google.com/index.html?java/util/NavigableMap.html)    [**NO FRAMES**](http://docs.google.com/NavigableMap.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#tyjcwt) | DETAIL: FIELD | CONSTR | [METHOD](#4d34og8) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).