| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AccessibleSelection.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/javax/accessibility/AccessibleRole.html)   [**NEXT CLASS**](http://docs.google.com/javax/accessibility/AccessibleState.html) | [**FRAMES**](http://docs.google.com/index.html?javax/accessibility/AccessibleSelection.html)    [**NO FRAMES**](http://docs.google.com/AccessibleSelection.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **javax.accessibility**

Interface AccessibleSelection

**All Known Implementing Classes:** [CheckboxMenuItem.AccessibleAWTCheckboxMenuItem](http://docs.google.com/java/awt/CheckboxMenuItem.AccessibleAWTCheckboxMenuItem.html), [JComboBox.AccessibleJComboBox](http://docs.google.com/javax/swing/JComboBox.AccessibleJComboBox.html), [JList.AccessibleJList](http://docs.google.com/javax/swing/JList.AccessibleJList.html), [JMenu.AccessibleJMenu](http://docs.google.com/javax/swing/JMenu.AccessibleJMenu.html), [JMenuBar.AccessibleJMenuBar](http://docs.google.com/javax/swing/JMenuBar.AccessibleJMenuBar.html), [JTabbedPane.AccessibleJTabbedPane](http://docs.google.com/javax/swing/JTabbedPane.AccessibleJTabbedPane.html), [JTable.AccessibleJTable](http://docs.google.com/javax/swing/JTable.AccessibleJTable.html), [JTree.AccessibleJTree](http://docs.google.com/javax/swing/JTree.AccessibleJTree.html), [JTree.AccessibleJTree.AccessibleJTreeNode](http://docs.google.com/javax/swing/JTree.AccessibleJTree.AccessibleJTreeNode.html), [List.AccessibleAWTList](http://docs.google.com/java/awt/List.AccessibleAWTList.html), [Menu.AccessibleAWTMenu](http://docs.google.com/java/awt/Menu.AccessibleAWTMenu.html), [MenuBar.AccessibleAWTMenuBar](http://docs.google.com/java/awt/MenuBar.AccessibleAWTMenuBar.html), [MenuComponent.AccessibleAWTMenuComponent](http://docs.google.com/java/awt/MenuComponent.AccessibleAWTMenuComponent.html), [MenuItem.AccessibleAWTMenuItem](http://docs.google.com/java/awt/MenuItem.AccessibleAWTMenuItem.html), [PopupMenu.AccessibleAWTPopupMenu](http://docs.google.com/java/awt/PopupMenu.AccessibleAWTPopupMenu.html)

public interface **AccessibleSelection**

This AccessibleSelection interface provides the standard mechanism for an assistive technology to determine what the current selected children are, as well as modify the selection set. Any object that has children that can be selected should support the AccessibleSelection interface. Applications can determine if an object supports the AccessibleSelection interface by first obtaining its AccessibleContext (see [Accessible](http://docs.google.com/javax/accessibility/Accessible.html)) and then calling the [AccessibleContext.getAccessibleSelection()](http://docs.google.com/javax/accessibility/AccessibleContext.html#getAccessibleSelection()) method. If the return value is not null, the object supports this interface.

**See Also:**[Accessible](http://docs.google.com/javax/accessibility/Accessible.html), [Accessible.getAccessibleContext()](http://docs.google.com/javax/accessibility/Accessible.html#getAccessibleContext()), [AccessibleContext](http://docs.google.com/javax/accessibility/AccessibleContext.html), [AccessibleContext.getAccessibleSelection()](http://docs.google.com/javax/accessibility/AccessibleContext.html#getAccessibleSelection())

| **Method Summary** | |
| --- | --- |
| void | [**addAccessibleSelection**](http://docs.google.com/javax/accessibility/AccessibleSelection.html#addAccessibleSelection(int))(int i)            Adds the specified Accessible child of the object to the object's selection. |
| void | [**clearAccessibleSelection**](http://docs.google.com/javax/accessibility/AccessibleSelection.html#clearAccessibleSelection())()            Clears the selection in the object, so that no children in the object are selected. |
| [Accessible](http://docs.google.com/javax/accessibility/Accessible.html) | [**getAccessibleSelection**](http://docs.google.com/javax/accessibility/AccessibleSelection.html#getAccessibleSelection(int))(int i)            Returns an Accessible representing the specified selected child of the object. |
| int | [**getAccessibleSelectionCount**](http://docs.google.com/javax/accessibility/AccessibleSelection.html#getAccessibleSelectionCount())()            Returns the number of Accessible children currently selected. |
| boolean | [**isAccessibleChildSelected**](http://docs.google.com/javax/accessibility/AccessibleSelection.html#isAccessibleChildSelected(int))(int i)            Determines if the current child of this object is selected. |
| void | [**removeAccessibleSelection**](http://docs.google.com/javax/accessibility/AccessibleSelection.html#removeAccessibleSelection(int))(int i)            Removes the specified child of the object from the object's selection. |
| void | [**selectAllAccessibleSelection**](http://docs.google.com/javax/accessibility/AccessibleSelection.html#selectAllAccessibleSelection())()            Causes every child of the object to be selected if the object supports multiple selections. |

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

### getAccessibleSelectionCount

int **getAccessibleSelectionCount**()

Returns the number of Accessible children currently selected. If no children are selected, the return value will be 0.

**Returns:**the number of items currently selected.

### getAccessibleSelection

[Accessible](http://docs.google.com/javax/accessibility/Accessible.html) **getAccessibleSelection**(int i)

Returns an Accessible representing the specified selected child of the object. If there isn't a selection, or there are fewer children selected than the integer passed in, the return value will be null.

Note that the index represents the i-th selected child, which is different from the i-th child.

**Parameters:**i - the zero-based index of selected children **Returns:**the i-th selected child**See Also:**[getAccessibleSelectionCount()](http://docs.google.com/javax/accessibility/AccessibleSelection.html#getAccessibleSelectionCount())

### isAccessibleChildSelected

boolean **isAccessibleChildSelected**(int i)

Determines if the current child of this object is selected.

**Parameters:**i - the zero-based index of the child in this Accessible object. **Returns:**true if the current child of this object is selected; else false.**See Also:**[AccessibleContext.getAccessibleChild(int)](http://docs.google.com/javax/accessibility/AccessibleContext.html#getAccessibleChild(int))

### addAccessibleSelection

void **addAccessibleSelection**(int i)

Adds the specified Accessible child of the object to the object's selection. If the object supports multiple selections, the specified child is added to any existing selection, otherwise it replaces any existing selection in the object. If the specified child is already selected, this method has no effect.

**Parameters:**i - the zero-based index of the child**See Also:**[AccessibleContext.getAccessibleChild(int)](http://docs.google.com/javax/accessibility/AccessibleContext.html#getAccessibleChild(int))

### removeAccessibleSelection

void **removeAccessibleSelection**(int i)

Removes the specified child of the object from the object's selection. If the specified item isn't currently selected, this method has no effect.

**Parameters:**i - the zero-based index of the child**See Also:**[AccessibleContext.getAccessibleChild(int)](http://docs.google.com/javax/accessibility/AccessibleContext.html#getAccessibleChild(int))

### clearAccessibleSelection

void **clearAccessibleSelection**()

Clears the selection in the object, so that no children in the object are selected.

### selectAllAccessibleSelection

void **selectAllAccessibleSelection**()

Causes every child of the object to be selected if the object supports multiple selections.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AccessibleSelection.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/javax/accessibility/AccessibleRole.html)   [**NEXT CLASS**](http://docs.google.com/javax/accessibility/AccessibleState.html) | [**FRAMES**](http://docs.google.com/index.html?javax/accessibility/AccessibleSelection.html)    [**NO FRAMES**](http://docs.google.com/AccessibleSelection.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[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).