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

## **javax.lang.model.element**

Interface Element

**All Known Subinterfaces:** [ExecutableElement](http://docs.google.com/javax/lang/model/element/ExecutableElement.html), [PackageElement](http://docs.google.com/javax/lang/model/element/PackageElement.html), [TypeElement](http://docs.google.com/javax/lang/model/element/TypeElement.html), [TypeParameterElement](http://docs.google.com/javax/lang/model/element/TypeParameterElement.html), [VariableElement](http://docs.google.com/javax/lang/model/element/VariableElement.html)

public interface **Element**

Represents a program element such as a package, class, or method. Each element represents a static, language-level construct (and not, for example, a runtime construct of the virtual machine).

Elements should be compared using the [equals(Object)](http://docs.google.com/javax/lang/model/element/Element.html#equals(java.lang.Object)) method. There is no guarantee that any particular element will always be represented by the same object.

To implement operations based on the class of an Element object, either use a [visitor](http://docs.google.com/javax/lang/model/element/ElementVisitor.html) or use the result of the [getKind()](http://docs.google.com/javax/lang/model/element/Element.html#getKind()) method. Using instanceof is *not* necessarily a reliable idiom for determining the effective class of an object in this modeling hierarchy since an implementation may choose to have a single object implement multiple Element subinterfaces.

**Since:** 1.6 **See Also:**[Elements](http://docs.google.com/javax/lang/model/util/Elements.html), [TypeMirror](http://docs.google.com/javax/lang/model/type/TypeMirror.html)

| **Method Summary** | |
| --- | --- |
| | <R,P> R | | --- | | [**accept**](http://docs.google.com/javax/lang/model/element/Element.html#accept(javax.lang.model.element.ElementVisitor,%20P))([ElementVisitor](http://docs.google.com/javax/lang/model/element/ElementVisitor.html)<R,P> v, P p)            Applies a visitor to this element. |
| [TypeMirror](http://docs.google.com/javax/lang/model/type/TypeMirror.html) | [**asType**](http://docs.google.com/javax/lang/model/element/Element.html#asType())()            Returns the type defined by this element. |
| boolean | [**equals**](http://docs.google.com/javax/lang/model/element/Element.html#equals(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) obj)            Returns true if the argument represents the same element as this, or false otherwise. |
| | <A extends [Annotation](http://docs.google.com/java/lang/annotation/Annotation.html)>  A | | --- | | [**getAnnotation**](http://docs.google.com/javax/lang/model/element/Element.html#getAnnotation(java.lang.Class))([Class](http://docs.google.com/java/lang/Class.html)<A> annotationType)            Returns this element's annotation for the specified type if such an annotation is present, else null. |
| [List](http://docs.google.com/java/util/List.html)<? extends [AnnotationMirror](http://docs.google.com/javax/lang/model/element/AnnotationMirror.html)> | [**getAnnotationMirrors**](http://docs.google.com/javax/lang/model/element/Element.html#getAnnotationMirrors())()            Returns the annotations that are directly present on this element. |
| [List](http://docs.google.com/java/util/List.html)<? extends [Element](http://docs.google.com/javax/lang/model/element/Element.html)> | [**getEnclosedElements**](http://docs.google.com/javax/lang/model/element/Element.html#getEnclosedElements())()            Returns the elements that are, loosely speaking, directly enclosed by this element. |
| [Element](http://docs.google.com/javax/lang/model/element/Element.html) | [**getEnclosingElement**](http://docs.google.com/javax/lang/model/element/Element.html#getEnclosingElement())()            Returns the innermost element within which this element is, loosely speaking, enclosed. |
| [ElementKind](http://docs.google.com/javax/lang/model/element/ElementKind.html) | [**getKind**](http://docs.google.com/javax/lang/model/element/Element.html#getKind())()            Returns the kind of this element. |
| [Set](http://docs.google.com/java/util/Set.html)<[Modifier](http://docs.google.com/javax/lang/model/element/Modifier.html)> | [**getModifiers**](http://docs.google.com/javax/lang/model/element/Element.html#getModifiers())()            Returns the modifiers of this element, excluding annotations. |
| [Name](http://docs.google.com/javax/lang/model/element/Name.html) | [**getSimpleName**](http://docs.google.com/javax/lang/model/element/Element.html#getSimpleName())()            Returns the simple (unqualified) name of this element. |
| int | [**hashCode**](http://docs.google.com/javax/lang/model/element/Element.html#hashCode())()            Obeys the general contract of [Object.hashCode](http://docs.google.com/java/lang/Object.html#hashCode()). |

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

### asType

[TypeMirror](http://docs.google.com/javax/lang/model/type/TypeMirror.html) **asType**()

Returns the type defined by this element.

A generic element defines a family of types, not just one. If this is a generic element, a *prototypical* type is returned. This is the element's invocation on the type variables corresponding to its own formal type parameters. For example, for the generic class element C<N extends Number>, the parameterized type C<N> is returned. The [Types](http://docs.google.com/javax/lang/model/util/Types.html) utility interface has more general methods for obtaining the full range of types defined by an element.

**Returns:**the type defined by this element**See Also:**[Types](http://docs.google.com/javax/lang/model/util/Types.html)

### getKind

[ElementKind](http://docs.google.com/javax/lang/model/element/ElementKind.html) **getKind**()

Returns the kind of this element.

**Returns:**the kind of this element

### getAnnotationMirrors

[List](http://docs.google.com/java/util/List.html)<? extends [AnnotationMirror](http://docs.google.com/javax/lang/model/element/AnnotationMirror.html)> **getAnnotationMirrors**()

Returns the annotations that are directly present on this element.

To get inherited annotations as well, use [getAllAnnotationMirrors](http://docs.google.com/javax/lang/model/util/Elements.html#getAllAnnotationMirrors(javax.lang.model.element.Element)).

**Returns:**the annotations directly present on this element; an empty list if there are none**See Also:**[ElementFilter](http://docs.google.com/javax/lang/model/util/ElementFilter.html)

### getAnnotation

<A extends [Annotation](http://docs.google.com/java/lang/annotation/Annotation.html)> A **getAnnotation**([Class](http://docs.google.com/java/lang/Class.html)<A> annotationType)

Returns this element's annotation for the specified type if such an annotation is present, else null. The annotation may be either inherited or directly present on this element.

The annotation returned by this method could contain an element whose value is of type Class. This value cannot be returned directly: information necessary to locate and load a class (such as the class loader to use) is not available, and the class might not be loadable at all. Attempting to read a Class object by invoking the relevant method on the returned annotation will result in a [MirroredTypeException](http://docs.google.com/javax/lang/model/type/MirroredTypeException.html), from which the corresponding [TypeMirror](http://docs.google.com/javax/lang/model/type/TypeMirror.html) may be extracted. Similarly, attempting to read a Class[]-valued element will result in a [MirroredTypesException](http://docs.google.com/javax/lang/model/type/MirroredTypesException.html).

*Note:* This method is unlike others in this and related interfaces. It operates on runtime reflective information — representations of annotation types currently loaded into the VM — rather than on the representations defined by and used throughout these interfaces. Consequently, calling methods on the returned annotation object can throw many of the exceptions that can be thrown when calling methods on an annotation object returned by core reflection. This method is intended for callers that are written to operate on a known, fixed set of annotation types.

**Type Parameters:**A - the annotation type**Parameters:**annotationType - the Class object corresponding to the annotation type **Returns:**this element's annotation for the specified annotation type if present on this element, else null**See Also:**[getAnnotationMirrors()](http://docs.google.com/javax/lang/model/element/Element.html#getAnnotationMirrors()), [AnnotatedElement.getAnnotation(java.lang.Class)](http://docs.google.com/java/lang/reflect/AnnotatedElement.html#getAnnotation(java.lang.Class)), [EnumConstantNotPresentException](http://docs.google.com/java/lang/EnumConstantNotPresentException.html), [AnnotationTypeMismatchException](http://docs.google.com/java/lang/annotation/AnnotationTypeMismatchException.html), [IncompleteAnnotationException](http://docs.google.com/java/lang/annotation/IncompleteAnnotationException.html), [MirroredTypeException](http://docs.google.com/javax/lang/model/type/MirroredTypeException.html), [MirroredTypesException](http://docs.google.com/javax/lang/model/type/MirroredTypesException.html)

### getModifiers

[Set](http://docs.google.com/java/util/Set.html)<[Modifier](http://docs.google.com/javax/lang/model/element/Modifier.html)> **getModifiers**()

Returns the modifiers of this element, excluding annotations. Implicit modifiers, such as the public and static modifiers of interface members, are included.

**Returns:**the modifiers of this element, or an empty set if there are none

### getSimpleName

[Name](http://docs.google.com/javax/lang/model/element/Name.html) **getSimpleName**()

Returns the simple (unqualified) name of this element. The name of a generic type does not include any reference to its formal type parameters. For example, the simple name of the type element java.util.Set<E> is "Set". If this element represents an unnamed package, an empty name is returned. If it represents a constructor, the name "<init>" is returned. If it represents a static initializer, the name "<clinit>" is returned. If it represents an anonymous class or instance initializer, an empty name is returned.

**Returns:**the simple name of this element

### getEnclosingElement

[Element](http://docs.google.com/javax/lang/model/element/Element.html) **getEnclosingElement**()

Returns the innermost element within which this element is, loosely speaking, enclosed.

* If this element is one whose declaration is lexically enclosed immediately within the declaration of another element, that other element is returned.
* If this is a top-level type, its package is returned.
* If this is a package, null is returned.
* If this is a type parameter, null is returned.

**Returns:**the enclosing element, or null if there is none**See Also:**[Elements.getPackageOf(javax.lang.model.element.Element)](http://docs.google.com/javax/lang/model/util/Elements.html#getPackageOf(javax.lang.model.element.Element))

### getEnclosedElements

[List](http://docs.google.com/java/util/List.html)<? extends [Element](http://docs.google.com/javax/lang/model/element/Element.html)> **getEnclosedElements**()

Returns the elements that are, loosely speaking, directly enclosed by this element. A class or interface is considered to enclose the fields, methods, constructors, and member types that it directly declares. This includes any (implicit) default constructor and the implicit values and valueOf methods of an enum type. A package encloses the top-level classes and interfaces within it, but is not considered to enclose subpackages. Other kinds of elements are not currently considered to enclose any elements; however, that may change as this API or the programming language evolves.

Note that elements of certain kinds can be isolated using methods in [ElementFilter](http://docs.google.com/javax/lang/model/util/ElementFilter.html).

**Returns:**the enclosed elements, or an empty list if none**See Also:**[Elements.getAllMembers(javax.lang.model.element.TypeElement)](http://docs.google.com/javax/lang/model/util/Elements.html#getAllMembers(javax.lang.model.element.TypeElement))**See**  [**The Java Language Specification, Third Edition**](http://java.sun.com/docs/books/jls/)**:** 8.8.9 Default Constructor, 8.9 Enums

### equals

boolean **equals**([Object](http://docs.google.com/java/lang/Object.html) obj)

Returns true if the argument represents the same element as this, or false otherwise.

Note that the identity of an element involves implicit state not directly accessible from the element's methods, including state about the presence of unrelated types. Element objects created by different implementations of these interfaces should *not* be expected to be equal even if "the same" element is being modeled; this is analogous to the inequality of Class objects for the same class file loaded through different class loaders.

**Overrides:**[equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)) in class [Object](http://docs.google.com/java/lang/Object.html) **Parameters:**obj - the object to be compared with this element **Returns:**true if the specified object represents the same element as this**See Also:**[Object.hashCode()](http://docs.google.com/java/lang/Object.html#hashCode()), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### hashCode

int **hashCode**()

Obeys the general contract of [Object.hashCode](http://docs.google.com/java/lang/Object.html#hashCode()).

**Overrides:**[hashCode](http://docs.google.com/java/lang/Object.html#hashCode()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**a hash code value for this object.**See Also:**[equals(java.lang.Object)](http://docs.google.com/javax/lang/model/element/Element.html#equals(java.lang.Object))

### accept

<R,P> R **accept**([ElementVisitor](http://docs.google.com/javax/lang/model/element/ElementVisitor.html)<R,P> v,  
 P p)

Applies a visitor to this element.

**Type Parameters:**R - the return type of the visitor's methodsP - the type of the additional parameter to the visitor's methods**Parameters:**v - the visitor operating on this elementp - additional parameter to the visitor **Returns:**a visitor-specified result

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Element.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/lang/model/element/AnnotationValueVisitor.html)   [**NEXT CLASS**](http://docs.google.com/javax/lang/model/element/ElementKind.html) | [**FRAMES**](http://docs.google.com/index.html?javax/lang/model/element/Element.html)    [**NO FRAMES**](http://docs.google.com/Element.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).