| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ElementScanner6.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/util/Elements.html)   [**NEXT CLASS**](http://docs.google.com/javax/lang/model/util/SimpleAnnotationValueVisitor6.html) | [**FRAMES**](http://docs.google.com/index.html?javax/lang/model/util/ElementScanner6.html)    [**NO FRAMES**](http://docs.google.com/ElementScanner6.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#4d34og8) | [CONSTR](#17dp8vu) | [METHOD](#35nkun2) |

## **javax.lang.model.util**

Class ElementScanner6<R,P>

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [javax.lang.model.util.AbstractElementVisitor6](http://docs.google.com/javax/lang/model/util/AbstractElementVisitor6.html)<R,P>  
 **javax.lang.model.util.ElementScanner6<R,P>**

**Type Parameters:**R - the return type of this visitor's methods. Use [Void](http://docs.google.com/java/lang/Void.html) for visitors that do not need to return results.P - the type of the additional parameter to this visitor's methods. Use Void for visitors that do not need an additional parameter. **All Implemented Interfaces:** [ElementVisitor](http://docs.google.com/javax/lang/model/element/ElementVisitor.html)<R,P>

[@SupportedSourceVersion](http://docs.google.com/javax/annotation/processing/SupportedSourceVersion.html)([value](http://docs.google.com/javax/annotation/processing/SupportedSourceVersion.html#value())=[RELEASE\_6](http://docs.google.com/javax/lang/model/SourceVersion.html#RELEASE_6))  
public class **ElementScanner6<R,P>**extends [AbstractElementVisitor6](http://docs.google.com/javax/lang/model/util/AbstractElementVisitor6.html)<R,P>

A scanning visitor of program elements with default behavior appropriate for the [RELEASE\_6](http://docs.google.com/javax/lang/model/SourceVersion.html#RELEASE_6) source version. The visit*XYZ* methods in this class scan their component elements by calling scan on their [enclosed elements](http://docs.google.com/javax/lang/model/element/Element.html#getEnclosedElements()), [parameters](http://docs.google.com/javax/lang/model/element/ExecutableElement.html#getParameters()), etc., as indicated in the individual method specifications. A subclass can control the order elements are visited by overriding the visit*XYZ* methods. Note that clients of a scanner may get the desired behavior be invoking v.scan(e, p) rather than v.visit(e, p) on the root objects of interest.

When a subclass overrides a visit*XYZ* method, the new method can cause the enclosed elements to be scanned in the default way by calling super.visit*XYZ*. In this fashion, the concrete visitor can control the ordering of traversal over the component elements with respect to the additional processing; for example, consistently calling super.visit*XYZ* at the start of the overridden methods will yield a preorder traversal, etc. If the component elements should be traversed in some other order, instead of calling super.visit*XYZ*, an overriding visit method should call scan with the elements in the desired order.

Methods in this class may be overridden subject to their general contract. Note that annotating methods in concrete subclasses with [@Override](http://docs.google.com/java/lang/Override.html) will help ensure that methods are overridden as intended.

**WARNING:** The ElementVisitor interface implemented by this class may have methods added to it in the future to accommodate new, currently unknown, language structures added to future versions of the Java™ programming language. Therefore, methods whose names begin with "visit" may be added to this class in the future; to avoid incompatibilities, classes which extend this class should not declare any instance methods with names beginning with "visit".

When such a new visit method is added, the default implementation in this class will be to call the [visitUnknown](http://docs.google.com/javax/lang/model/util/AbstractElementVisitor6.html#visitUnknown(javax.lang.model.element.Element,%20P)) method. A new element scanner visitor class will also be introduced to correspond to the new language level; this visitor will have different default behavior for the visit method in question. When the new visitor is introduced, all or portions of this visitor may be deprecated.

**Since:** 1.6

| **Field Summary** | |
| --- | --- |
| protected  [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) | [**DEFAULT\_VALUE**](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#DEFAULT_VALUE)            The specified default value. |

| **Constructor Summary** | |
| --- | --- |
| protected | [**ElementScanner6**](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#ElementScanner6())()            Constructor for concrete subclasses; uses null for the default value. |
| protected | [**ElementScanner6**](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#ElementScanner6(R))([R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) defaultValue)            Constructor for concrete subclasses; uses the argument for the default value. |

| **Method Summary** | |
| --- | --- |
| [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) | [**scan**](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#scan(javax.lang.model.element.Element))([Element](http://docs.google.com/javax/lang/model/element/Element.html) e)            Convenience method equivalent to v.scan(e, null). |
| [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) | [**scan**](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#scan(javax.lang.model.element.Element,%20P))([Element](http://docs.google.com/javax/lang/model/element/Element.html) e, [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)            Processes an element by calling e.accept(this, p); this method may be overridden by subclasses. |
| [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) | [**scan**](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#scan(java.lang.Iterable,%20P))([Iterable](http://docs.google.com/java/lang/Iterable.html)<? extends [Element](http://docs.google.com/javax/lang/model/element/Element.html)> iterable, [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)            Iterates over the given elements and calls [scan(Element, P)](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#scan(javax.lang.model.element.Element,%20P)) on each one. |
| [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) | [**visitExecutable**](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#visitExecutable(javax.lang.model.element.ExecutableElement,%20P))([ExecutableElement](http://docs.google.com/javax/lang/model/element/ExecutableElement.html) e, [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)            Visits an executable element. |
| [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) | [**visitPackage**](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#visitPackage(javax.lang.model.element.PackageElement,%20P))([PackageElement](http://docs.google.com/javax/lang/model/element/PackageElement.html) e, [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)            Visits a package element. |
| [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) | [**visitType**](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#visitType(javax.lang.model.element.TypeElement,%20P))([TypeElement](http://docs.google.com/javax/lang/model/element/TypeElement.html) e, [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)            Visits a type element. |
| [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) | [**visitTypeParameter**](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#visitTypeParameter(javax.lang.model.element.TypeParameterElement,%20P))([TypeParameterElement](http://docs.google.com/javax/lang/model/element/TypeParameterElement.html) e, [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)            Visits a type parameter element. |
| [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) | [**visitVariable**](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#visitVariable(javax.lang.model.element.VariableElement,%20P))([VariableElement](http://docs.google.com/javax/lang/model/element/VariableElement.html) e, [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)            Visits a variable element. |

| **Methods inherited from class javax.lang.model.util.**[**AbstractElementVisitor6**](http://docs.google.com/javax/lang/model/util/AbstractElementVisitor6.html) |
| --- |
| [visit](http://docs.google.com/javax/lang/model/util/AbstractElementVisitor6.html#visit(javax.lang.model.element.Element)), [visit](http://docs.google.com/javax/lang/model/util/AbstractElementVisitor6.html#visit(javax.lang.model.element.Element,%20P)), [visitUnknown](http://docs.google.com/javax/lang/model/util/AbstractElementVisitor6.html#visitUnknown(javax.lang.model.element.Element,%20P)) |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Field Detail** |
| --- |

### DEFAULT\_VALUE

protected final [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) **DEFAULT\_VALUE**

The specified default value.

| **Constructor Detail** |
| --- |

### ElementScanner6

protected **ElementScanner6**()

Constructor for concrete subclasses; uses null for the default value.

### ElementScanner6

protected **ElementScanner6**([R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) defaultValue)

Constructor for concrete subclasses; uses the argument for the default value.

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

### scan

public final [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) **scan**([Iterable](http://docs.google.com/java/lang/Iterable.html)<? extends [Element](http://docs.google.com/javax/lang/model/element/Element.html)> iterable,  
 [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)

Iterates over the given elements and calls [scan(Element, P)](http://docs.google.com/javax/lang/model/util/ElementScanner6.html#scan(javax.lang.model.element.Element,%20P)) on each one. Returns the result of the last call to scan or DEFAULT\_VALUE for an empty iterable.

**Parameters:**iterable - the elements to scanp - additional parameter **Returns:**the scan of the last element or DEFAULT\_VALUE if no elements

### scan

public [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) **scan**([Element](http://docs.google.com/javax/lang/model/element/Element.html) e,  
 [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)

Processes an element by calling e.accept(this, p); this method may be overridden by subclasses.

**Returns:**the result of visiting e.

### scan

public final [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) **scan**([Element](http://docs.google.com/javax/lang/model/element/Element.html) e)

Convenience method equivalent to v.scan(e, null).

**Returns:**the result of scanning e.

### visitPackage

public [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) **visitPackage**([PackageElement](http://docs.google.com/javax/lang/model/element/PackageElement.html) e,  
 [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)

Visits a package element. This implementation scans the enclosed elements.

**Parameters:**e - the element to visitp - a visitor-specified parameter **Returns:**the result of scanning

### visitType

public [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) **visitType**([TypeElement](http://docs.google.com/javax/lang/model/element/TypeElement.html) e,  
 [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)

Visits a type element. This implementation scans the enclosed elements.

**Parameters:**e - the element to visitp - a visitor-specified parameter **Returns:**the result of scanning

### visitVariable

public [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) **visitVariable**([VariableElement](http://docs.google.com/javax/lang/model/element/VariableElement.html) e,  
 [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)

Visits a variable element. This implementation scans the enclosed elements.

**Parameters:**e - the element to visitp - a visitor-specified parameter **Returns:**the result of scanning

### visitExecutable

public [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) **visitExecutable**([ExecutableElement](http://docs.google.com/javax/lang/model/element/ExecutableElement.html) e,  
 [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)

Visits an executable element. This implementation scans the parameters.

**Parameters:**e - the element to visitp - a visitor-specified parameter **Returns:**the result of scanning

### visitTypeParameter

public [R](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) **visitTypeParameter**([TypeParameterElement](http://docs.google.com/javax/lang/model/element/TypeParameterElement.html) e,  
 [P](http://docs.google.com/javax/lang/model/util/ElementScanner6.html) p)

Visits a type parameter element. This implementation scans the enclosed elements.

**Parameters:**e - the element to visitp - a visitor-specified parameter **Returns:**the result of scanning

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ElementScanner6.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/util/Elements.html)   [**NEXT CLASS**](http://docs.google.com/javax/lang/model/util/SimpleAnnotationValueVisitor6.html) | [**FRAMES**](http://docs.google.com/index.html?javax/lang/model/util/ElementScanner6.html)    [**NO FRAMES**](http://docs.google.com/ElementScanner6.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#4d34og8) | [CONSTR](#17dp8vu) | [METHOD](#35nkun2) |

[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).