| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/XmlRootElement.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/xml/bind/annotation/XmlRegistry.html)   [**NEXT CLASS**](http://docs.google.com/javax/xml/bind/annotation/XmlSchema.html) | [**FRAMES**](http://docs.google.com/index.html?javax/xml/bind/annotation/XmlRootElement.html)    [**NO FRAMES**](http://docs.google.com/XmlRootElement.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: REQUIRED | [OPTIONAL](#3znysh7) | DETAIL: [ELEMENT](#2s8eyo1) |

## **javax.xml.bind.annotation**

Annotation Type XmlRootElement

[@Retention](http://docs.google.com/java/lang/annotation/Retention.html)([value](http://docs.google.com/java/lang/annotation/Retention.html#value())=[RUNTIME](http://docs.google.com/java/lang/annotation/RetentionPolicy.html#RUNTIME))  
[@Target](http://docs.google.com/java/lang/annotation/Target.html)([value](http://docs.google.com/java/lang/annotation/Target.html#value())=[TYPE](http://docs.google.com/java/lang/annotation/ElementType.html#TYPE))  
public @interface **XmlRootElement**

Maps a class or an enum type to an XML element.

**Usage**

The @XmlRootElement annotation can be used with the following program elements:

* a top level class
* an enum type

See "Package Specification" in javax.xml.bind.package javadoc for additional common information.

When a top level class or an enum type is annotated with the @XmlRootElement annotation, then its value is represented as XML element in an XML document.

This annotation can be used with the following annotations: [XmlType](http://docs.google.com/javax/xml/bind/annotation/XmlType.html), [XmlEnum](http://docs.google.com/javax/xml/bind/annotation/XmlEnum.html), [XmlAccessorType](http://docs.google.com/javax/xml/bind/annotation/XmlAccessorType.html), [XmlAccessorOrder](http://docs.google.com/javax/xml/bind/annotation/XmlAccessorOrder.html).

**Example 1:**  Associate an element with XML Schema type

// Example: Code fragment  
 @XmlRootElement  
 class Point {  
 int x;  
 int y;  
 Point(int \_x,int \_y) {x=\_x;y=\_y;}  
 }

//Example: Code fragment corresponding to XML output  
 marshal( new Point(3,5), System.out);

<point>  
 <x> 3 </x>  
 <y> 5 </y>  
 </point>

The annotation causes an global element declaration to be produced in the schema. The global element declaration is associated with the XML schema type to which the class is mapped.

<xs:element name="point" type="point"></xs:element>  
 <xs:complexType name="point">  
 <xs:sequence>  
 <xs:element name="x" type="xs:int"></xs:element>  
 <xs:element name="y" type="xs:int"></xs:element>  
 </xs:sequence>  
 </xs:complexType>

**Example 2: Orthogonality to type inheritance**

An element declaration annotated on a type is not inherited by its derived types. The following example shows this.

// Example: Code fragment  
 @XmlRootElement  
 class Point3D extends Point {  
 int z;  
 Point3D(int \_x,int \_y,int \_z) {super(\_x,\_y);z=\_z;}  
 }  
  
 //Example: Code fragment corresponding to XML output \*   
 marshal( new Point3D(3,5,0), System.out );  
  
 <!-- Example: XML output -->  
 <!-- The element name is point3D not point -->  
 <point3D>  
 <x>3</x>  
 <y>5</y>  
 <z>0</z>  
 </point3D>  
  
 <!-- Example: XML schema definition -->  
 <xs:element name="point3D" type="point3D"/>  
 <xs:complexType name="point3D">  
 <xs:complexContent>  
 <xs:extension base="point">  
 <xs:sequence>  
 <xs:element name="z" type="xs:int"/>  
 </xs:sequence>  
 </xs:extension>  
 </xs:complexContent>  
 </xs:complexType>

**Example 3:**  Associate a global element with XML Schema type to which the class is mapped.

//Example: Code fragment  
 @XmlRootElement(name="PriceElement")  
 public class USPrice {  
 @XmlElement  
 public java.math.BigDecimal price;  
 }  
  
 <!-- Example: XML schema definition -->  
 <xs:element name="PriceElement" type="USPrice"/>  
 <xs:complexType name="USPrice">  
 <xs:sequence>  
 <xs:element name="price" type="xs:decimal"/>  
 </sequence>  
 </xs:complexType>

**Since:** JAXB2.0

| **Optional Element Summary** | |
| --- | --- |
| [String](http://docs.google.com/java/lang/String.html) | [**name**](http://docs.google.com/javax/xml/bind/annotation/XmlRootElement.html#name())            local name of the XML element. |
| [String](http://docs.google.com/java/lang/String.html) | [**namespace**](http://docs.google.com/javax/xml/bind/annotation/XmlRootElement.html#namespace())            namespace name of the XML element. |

### namespace

public abstract [String](http://docs.google.com/java/lang/String.html) **namespace**

namespace name of the XML element.

If the value is "##default", then the XML namespace name is derived from the package of the class ( [XmlSchema](http://docs.google.com/javax/xml/bind/annotation/XmlSchema.html) ). If the package is unnamed, then the XML namespace is the default empty namespace.

**Default:**"##default"

### name

public abstract [String](http://docs.google.com/java/lang/String.html) **name**

local name of the XML element.

If the value is "##default", then the name is derived from the class name.

**Default:**"##default"

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/XmlRootElement.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/xml/bind/annotation/XmlRegistry.html)   [**NEXT CLASS**](http://docs.google.com/javax/xml/bind/annotation/XmlSchema.html) | [**FRAMES**](http://docs.google.com/index.html?javax/xml/bind/annotation/XmlRootElement.html)    [**NO FRAMES**](http://docs.google.com/XmlRootElement.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: REQUIRED | [OPTIONAL](#3znysh7) | DETAIL: [ELEMENT](#2s8eyo1) |

[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).