| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DefaultPersistenceDelegate.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/beans/Customizer.html)   [**NEXT CLASS**](http://docs.google.com/java/beans/DesignMode.html) | [**FRAMES**](http://docs.google.com/index.html?java/beans/DefaultPersistenceDelegate.html)    [**NO FRAMES**](http://docs.google.com/DefaultPersistenceDelegate.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#17dp8vu) |

## **java.beans**

Class DefaultPersistenceDelegate

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.beans.PersistenceDelegate](http://docs.google.com/java/beans/PersistenceDelegate.html)  
 **java.beans.DefaultPersistenceDelegate**

public class **DefaultPersistenceDelegate**extends [PersistenceDelegate](http://docs.google.com/java/beans/PersistenceDelegate.html)

The DefaultPersistenceDelegate is a concrete implementation of the abstract PersistenceDelegate class and is the delegate used by default for classes about which no information is available. The DefaultPersistenceDelegate provides, version resilient, public API-based persistence for classes that follow the JavaBeans conventions without any class specific configuration.

The key assumptions are that the class has a nullary constructor and that its state is accurately represented by matching pairs of "setter" and "getter" methods in the order they are returned by the Introspector. In addition to providing code-free persistence for JavaBeans, the DefaultPersistenceDelegate provides a convenient means to effect persistent storage for classes that have a constructor that, while not nullary, simply requires some property values as arguments.

**Since:** 1.4 **See Also:**[DefaultPersistenceDelegate(String[])](http://docs.google.com/java/beans/DefaultPersistenceDelegate.html#DefaultPersistenceDelegate(java.lang.String%5B%5D)), [Introspector](http://docs.google.com/java/beans/Introspector.html)

| **Constructor Summary** | |
| --- | --- |
| [**DefaultPersistenceDelegate**](http://docs.google.com/java/beans/DefaultPersistenceDelegate.html#DefaultPersistenceDelegate())()            Creates a persistence delegate for a class with a nullary constructor. |
| [**DefaultPersistenceDelegate**](http://docs.google.com/java/beans/DefaultPersistenceDelegate.html#DefaultPersistenceDelegate(java.lang.String%5B%5D))([String](http://docs.google.com/java/lang/String.html)[] constructorPropertyNames)            Creates a default persistence delegate for a class with a constructor whose arguments are the values of the property names as specified by constructorPropertyNames. |

| **Method Summary** | |
| --- | --- |
| protected  void | [**initialize**](http://docs.google.com/java/beans/DefaultPersistenceDelegate.html#initialize(java.lang.Class,%20java.lang.Object,%20java.lang.Object,%20java.beans.Encoder))([Class](http://docs.google.com/java/lang/Class.html)<?> type, [Object](http://docs.google.com/java/lang/Object.html) oldInstance, [Object](http://docs.google.com/java/lang/Object.html) newInstance, [Encoder](http://docs.google.com/java/beans/Encoder.html) out)            This default implementation of the initialize method assumes all state held in objects of this type is exposed via the matching pairs of "setter" and "getter" methods in the order they are returned by the Introspector. |
| protected  [Expression](http://docs.google.com/java/beans/Expression.html) | [**instantiate**](http://docs.google.com/java/beans/DefaultPersistenceDelegate.html#instantiate(java.lang.Object,%20java.beans.Encoder))([Object](http://docs.google.com/java/lang/Object.html) oldInstance, [Encoder](http://docs.google.com/java/beans/Encoder.html) out)            This default implementation of the instantiate method returns an expression containing the predefined method name "new" which denotes a call to a constructor with the arguments as specified in the DefaultPersistenceDelegate's constructor. |
| protected  boolean | [**mutatesTo**](http://docs.google.com/java/beans/DefaultPersistenceDelegate.html#mutatesTo(java.lang.Object,%20java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) oldInstance, [Object](http://docs.google.com/java/lang/Object.html) newInstance)            If the number of arguments in the specified constructor is non-zero and the class of oldInstance explicitly declares an "equals" method this method returns the value of oldInstance.equals(newInstance). |

| **Methods inherited from class java.beans.**[**PersistenceDelegate**](http://docs.google.com/java/beans/PersistenceDelegate.html) |
| --- |
| [writeObject](http://docs.google.com/java/beans/PersistenceDelegate.html#writeObject(java.lang.Object,%20java.beans.Encoder)) |

| **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)) |

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

### DefaultPersistenceDelegate

public **DefaultPersistenceDelegate**()

Creates a persistence delegate for a class with a nullary constructor.

**See Also:**[DefaultPersistenceDelegate(java.lang.String[])](http://docs.google.com/java/beans/DefaultPersistenceDelegate.html#DefaultPersistenceDelegate(java.lang.String%5B%5D))

### DefaultPersistenceDelegate

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

Creates a default persistence delegate for a class with a constructor whose arguments are the values of the property names as specified by constructorPropertyNames. The constructor arguments are created by evaluating the property names in the order they are supplied. To use this class to specify a single preferred constructor for use in the serialization of a particular type, we state the names of the properties that make up the constructor's arguments. For example, the Font class which does not define a nullary constructor can be handled with the following persistence delegate:

new DefaultPersistenceDelegate(new String[]{"name", "style", "size"});

**Parameters:**constructorPropertyNames - The property names for the arguments of this constructor.**See Also:**[instantiate(java.lang.Object, java.beans.Encoder)](http://docs.google.com/java/beans/DefaultPersistenceDelegate.html#instantiate(java.lang.Object,%20java.beans.Encoder))

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

### mutatesTo

protected boolean **mutatesTo**([Object](http://docs.google.com/java/lang/Object.html) oldInstance,  
 [Object](http://docs.google.com/java/lang/Object.html) newInstance)

If the number of arguments in the specified constructor is non-zero and the class of oldInstance explicitly declares an "equals" method this method returns the value of oldInstance.equals(newInstance). Otherwise, this method uses the superclass's definition which returns true if the classes of the two instances are equal.

**Overrides:**[mutatesTo](http://docs.google.com/java/beans/PersistenceDelegate.html#mutatesTo(java.lang.Object,%20java.lang.Object)) in class [PersistenceDelegate](http://docs.google.com/java/beans/PersistenceDelegate.html) **Parameters:**oldInstance - The instance to be copied.newInstance - The instance that is to be modified. **Returns:**True if an equivalent copy of newInstance may be created by applying a series of mutations to oldInstance.**See Also:**[DefaultPersistenceDelegate(String[])](http://docs.google.com/java/beans/DefaultPersistenceDelegate.html#DefaultPersistenceDelegate(java.lang.String%5B%5D))

### instantiate

protected [Expression](http://docs.google.com/java/beans/Expression.html) **instantiate**([Object](http://docs.google.com/java/lang/Object.html) oldInstance,  
 [Encoder](http://docs.google.com/java/beans/Encoder.html) out)

This default implementation of the instantiate method returns an expression containing the predefined method name "new" which denotes a call to a constructor with the arguments as specified in the DefaultPersistenceDelegate's constructor.

**Specified by:**[instantiate](http://docs.google.com/java/beans/PersistenceDelegate.html#instantiate(java.lang.Object,%20java.beans.Encoder)) in class [PersistenceDelegate](http://docs.google.com/java/beans/PersistenceDelegate.html) **Parameters:**oldInstance - The instance to be instantiated.out - The code output stream. **Returns:**An expression whose value is oldInstance.**See Also:**[DefaultPersistenceDelegate(String[])](http://docs.google.com/java/beans/DefaultPersistenceDelegate.html#DefaultPersistenceDelegate(java.lang.String%5B%5D))

### initialize

protected void **initialize**([Class](http://docs.google.com/java/lang/Class.html)<?> type,  
 [Object](http://docs.google.com/java/lang/Object.html) oldInstance,  
 [Object](http://docs.google.com/java/lang/Object.html) newInstance,  
 [Encoder](http://docs.google.com/java/beans/Encoder.html) out)

This default implementation of the initialize method assumes all state held in objects of this type is exposed via the matching pairs of "setter" and "getter" methods in the order they are returned by the Introspector. If a property descriptor defines a "transient" attribute with a value equal to Boolean.TRUE the property is ignored by this default implementation. Note that this use of the word "transient" is quite independent of the field modifier that is used by the ObjectOutputStream.

For each non-transient property, an expression is created in which the nullary "getter" method is applied to the oldInstance. The value of this expression is the value of the property in the instance that is being serialized. If the value of this expression in the cloned environment mutatesTo the target value, the new value is initialized to make it equivalent to the old value. In this case, because the property value has not changed there is no need to call the corresponding "setter" method and no statement is emitted. If not however, the expression for this value is replaced with another expression (normally a constructor) and the corresponding "setter" method is called to install the new property value in the object. This scheme removes default information from the output produced by streams using this delegate.

In passing these statements to the output stream, where they will be executed, side effects are made to the newInstance. In most cases this allows the problem of properties whose values depend on each other to actually help the serialization process by making the number of statements that need to be written to the output smaller. In general, the problem of handling interdependent properties is reduced to that of finding an order for the properties in a class such that no property value depends on the value of a subsequent property.

**Overrides:**[initialize](http://docs.google.com/java/beans/PersistenceDelegate.html#initialize(java.lang.Class,%20java.lang.Object,%20java.lang.Object,%20java.beans.Encoder)) in class [PersistenceDelegate](http://docs.google.com/java/beans/PersistenceDelegate.html) **Parameters:**oldInstance - The instance to be copied.newInstance - The instance that is to be modified.out - The stream to which any initialization statements should be written.**See Also:**[Introspector.getBeanInfo(java.lang.Class)](http://docs.google.com/java/beans/Introspector.html#getBeanInfo(java.lang.Class)), [PropertyDescriptor](http://docs.google.com/java/beans/PropertyDescriptor.html)

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DefaultPersistenceDelegate.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/beans/Customizer.html)   [**NEXT CLASS**](http://docs.google.com/java/beans/DesignMode.html) | [**FRAMES**](http://docs.google.com/index.html?java/beans/DefaultPersistenceDelegate.html)    [**NO FRAMES**](http://docs.google.com/DefaultPersistenceDelegate.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#17dp8vu) |

[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).