JavaScript is disabled on your browser.

[Skip navigation links](#1fob9te)

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* Class
* [Use](http://docs.google.com/class-use/ArrayConverter.html)
* [Tree](http://docs.google.com/package-tree.html)
* [Deprecated](http://docs.google.com/deprecated-list.html)
* [Index](http://docs.google.com/index-all.html)
* [Help](http://docs.google.com/help-doc.html)
* [Prev Class](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html)
* [Next Class](http://docs.google.com/org/apache/commons/beanutils/converters/BigDecimalConverter.html)
* [Frames](http://docs.google.com/index.html?org/apache/commons/beanutils/converters/ArrayConverter.html)
* [No Frames](http://docs.google.com/ArrayConverter.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* Field |
* [Constr](#3znysh7) |
* [Method](#2et92p0)
* Detail:
* Field |
* [Constr](#1t3h5sf) |
* [Method](#17dp8vu)

org.apache.commons.beanutils.converters

## Class ArrayConverter

* [java.lang.Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true)
  + [org.apache.commons.beanutils.converters.AbstractConverter](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html)
    - org.apache.commons.beanutils.converters.ArrayConverter
* All Implemented Interfaces: [Converter](http://docs.google.com/org/apache/commons/beanutils/Converter.html)  
    
  public class ArrayConverter  
  extends [AbstractConverter](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html)
* Generic [Converter](http://docs.google.com/org/apache/commons/beanutils/Converter.html) implementation that handles conversion to and from **array** objects.  
  Can be configured to either return a *default value* or throw a ConversionException if a conversion error occurs.  
  The main features of this implementation are:
  + **Element Conversion** - delegates to a [Converter](http://docs.google.com/org/apache/commons/beanutils/Converter.html), appropriate for the type, to convert individual elements of the array. This leverages the power of existing converters without having to replicate their functionality for converting to the element type and removes the need to create a specifc array type converters.
  + **Arrays or Collections** - can convert from either arrays or Collections to an array, limited only by the capability of the delegate [Converter](http://docs.google.com/org/apache/commons/beanutils/Converter.html).
  + **Delimited Lists** - can Convert **to** and **from** a delimited list in String format.
  + **Conversion to String** - converts an array to a String in one of two ways: as a *delimited list* or by converting the first element in the array to a String - this is controlled by the [setOnlyFirstToString(boolean)](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#setOnlyFirstToString-boolean-) parameter.
  + **Multi Dimensional Arrays** - it is possible to convert a String to a multi-dimensional arrays, by embedding [ArrayConverter](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html) within each other - see example below.
  + **Default Value**
    - ***No Default*** - use the [ArrayConverter(Class, Converter)](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#ArrayConverter-java.lang.Class-org.apache.commons.beanutils.Converter-) constructor to create a converter which throws a [ConversionException](http://docs.google.com/org/apache/commons/beanutils/ConversionException.html) if the value is missing or invalid.
    - ***Default values*** - use the [ArrayConverter(Class, Converter, int)](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#ArrayConverter-java.lang.Class-org.apache.commons.beanutils.Converter-int-) constructor to create a converter which returns a *default value*. The *defaultSize* parameter controls the *default value* in the following way:
      * *defaultSize < 0* - default is null
      * *defaultSize = 0* - default is an array of length zero
      * *defaultSize > 0* - default is an array with a length specified by defaultSize (N.B. elements in the array will be null)

### Parsing Delimited ListsThis implementation can convert a delimited list in String format into an array of the appropriate type. By default, it uses a comma as the delimiter but the following methods can be used to configure parsing:

* + setDelimiter(char) - allows the character used as the delimiter to be configured [default is a comma].
  + setAllowedChars(char[]) - adds additional characters (to the default alphabetic/numeric) to those considered to be valid token characters.

### Multi Dimensional ArraysIt is possible to convert a String to mulit-dimensional arrays by using [ArrayConverter](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html) as the element [Converter](http://docs.google.com/org/apache/commons/beanutils/Converter.html) within another [ArrayConverter](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html). For example, the following code demonstrates how to construct a [Converter](http://docs.google.com/org/apache/commons/beanutils/Converter.html) to convert a delimited String into a two dimensional integer array: // Construct an Integer Converter IntegerConverter integerConverter = new IntegerConverter(); // Construct an array Converter for an integer array (i.e. int[]) using // an IntegerConverter as the element converter. // N.B. Uses the default comma (i.e. ",") as the delimiter between individual numbers ArrayConverter arrayConverter = new ArrayConverter(int[].class, integerConverter); // Construct a "Matrix" Converter which converts arrays of integer arrays using // the pre-ceeding ArrayConverter as the element Converter. // N.B. Uses a semi-colon (i.e. ";") as the delimiter to separate the different sets of numbers. // Also the delimiter used by the first ArrayConverter needs to be added to the // "allowed characters" for this one. ArrayConverter matrixConverter = new ArrayConverter(int[][].class, arrayConverter); matrixConverter.setDelimiter(';'); matrixConverter.setAllowedChars(new char[] {','}); // Do the Conversion String matrixString = "11,12,13 ; 21,22,23 ; 31,32,33 ; 41,42,43"; int[][] result = (int[][])matrixConverter.convert(int[][].class, matrixString); Since: 1.8.0 Version: $Id$

### Constructor SummaryConstructors

|  |
| --- |
| * + Constructor and Description |
| * + [ArrayConverter](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#ArrayConverter-java.lang.Class-org.apache.commons.beanutils.Converter-)([Class](https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html?is-external=true)<?> defaultType, [Converter](http://docs.google.com/org/apache/commons/beanutils/Converter.html) elementConverter) Construct an **array** Converter with the specified **component** Converter that throws a ConversionException if an error occurs. |
| * + [ArrayConverter](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#ArrayConverter-java.lang.Class-org.apache.commons.beanutils.Converter-int-)([Class](https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html?is-external=true)<?> defaultType, [Converter](http://docs.google.com/org/apache/commons/beanutils/Converter.html) elementConverter, int defaultSize) Construct an **array** Converter with the specified **component** Converter that returns a default array of the specified size (or null) if an error occurs. |

### Method SummaryAll Methods Instance Methods Concrete Methods

|  |  |
| --- | --- |
| * + Modifier and Type | * + Method and Description |
| * + protected [Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true) | * + [convertArray](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#convertArray-java.lang.Object-)([Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true) value) Returns the value unchanged. |
| * + protected [Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<?> | * + [convertToCollection](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#convertToCollection-java.lang.Class-java.lang.Object-)([Class](https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html?is-external=true)<?> type, [Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true) value) Converts non-array values to a Collection prior to being converted either to an array or a String. |
| * + protected [String](https://docs.oracle.com/javase/7/docs/api/java/lang/String.html?is-external=true) | * + [convertToString](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#convertToString-java.lang.Object-)([Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true) value) Handles conversion to a String. |
| * + protected <T> T | * + [convertToType](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#convertToType-java.lang.Class-java.lang.Object-)([Class](https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html?is-external=true)<T> type, [Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true) value) Handles conversion to an array of the specified type. |
| * + protected [Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true) | * + [getDefault](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#getDefault-java.lang.Class-)([Class](https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html?is-external=true)<?> type) Return the default value for conversions to the specified type. |
| * + protected [Class](https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html?is-external=true)<?> | * + [getDefaultType](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#getDefaultType--)() Return the default type this Converter handles. |
| * + void | * + [setAllowedChars](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#setAllowedChars-char:A-)(char[] allowedChars) Set the allowed characters to be used for parsing a delimited String. |
| * + void | * + [setDelimiter](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#setDelimiter-char-)(char delimiter) Set the delimiter to be used for parsing a delimited String. |
| * + void | * + [setOnlyFirstToString](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#setOnlyFirstToString-boolean-)(boolean onlyFirstToString) Indicates whether converting to a String should create a delimited list or just convert the first value. |
| * + [String](https://docs.oracle.com/javase/7/docs/api/java/lang/String.html?is-external=true) | * + [toString](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#toString--)() Provide a String representation of this array converter. |

### Methods inherited from class org.apache.commons.beanutils.converters.[**AbstractConverter**](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html)[conversionException](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html#conversionException-java.lang.Class-java.lang.Object-), [convert](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html#convert-java.lang.Class-java.lang.Object-), [handleError](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html#handleError-java.lang.Class-java.lang.Object-java.lang.Throwable-), [handleMissing](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html#handleMissing-java.lang.Class-), [isUseDefault](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html#isUseDefault--), [setDefaultValue](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html#setDefaultValue-java.lang.Object-)

### Methods inherited from class java.lang.[**Object**](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true)[clone](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#clone--), [equals](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#equals-java.lang.Object-), [finalize](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#finalize--), [getClass](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#getClass--), [hashCode](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#hashCode--), [notify](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#notify--), [notifyAll](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#notifyAll--), [wait](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#wait--), [wait](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#wait-long-), [wait](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#wait-long-int-)

### Constructor Detail

#### ArrayConverter public ArrayConverter([Class](https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html?is-external=true)<?> defaultType, [Converter](http://docs.google.com/org/apache/commons/beanutils/Converter.html) elementConverter) Construct an **array** Converter with the specified **component** Converter that throws a ConversionException if an error occurs.Parameters: defaultType - The default array type this Converter handles elementConverter - Converter used to convert individual array elements.

#### ArrayConverter public ArrayConverter([Class](https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html?is-external=true)<?> defaultType, [Converter](http://docs.google.com/org/apache/commons/beanutils/Converter.html) elementConverter, int defaultSize) Construct an **array** Converter with the specified **component** Converter that returns a default array of the specified size (or null) if an error occurs.Parameters: defaultType - The default array type this Converter handles elementConverter - Converter used to convert individual array elements. defaultSize - Specifies the size of the default array value or if less than zero indicates that a null default value should be used.

### Method Detail

#### setDelimiter public void setDelimiter(char delimiter) Set the delimiter to be used for parsing a delimited String.Parameters: delimiter - The delimiter [default ',']

#### setAllowedChars public void setAllowedChars(char[] allowedChars) Set the allowed characters to be used for parsing a delimited String.Parameters: allowedChars - Characters which are to be considered as part of the tokens when parsing a delimited String [default is '.' and '-']

#### setOnlyFirstToString public void setOnlyFirstToString(boolean onlyFirstToString) Indicates whether converting to a String should create a delimited list or just convert the first value.Parameters: onlyFirstToString - true converts only the first value in the array to a String, false converts all values in the array into a delimited list (default is true

#### getDefaultType protected [Class](https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html?is-external=true)<?> getDefaultType() Return the default type this Converter handles.Specified by: [getDefaultType](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html#getDefaultType--) in class [AbstractConverter](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html) Returns: The default type this Converter handles.

#### convertToString protected [String](https://docs.oracle.com/javase/7/docs/api/java/lang/String.html?is-external=true) convertToString([Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true) value) throws [Throwable](https://docs.oracle.com/javase/7/docs/api/java/lang/Throwable.html?is-external=true) Handles conversion to a String.Overrides: [convertToString](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html#convertToString-java.lang.Object-) in class [AbstractConverter](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html) Parameters: value - The value to be converted. Returns: the converted String value. Throws: [Throwable](https://docs.oracle.com/javase/7/docs/api/java/lang/Throwable.html?is-external=true) - if an error occurs converting to a String

#### convertToType protected <T> T convertToType([Class](https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html?is-external=true)<T> type, [Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true) value) throws [Throwable](https://docs.oracle.com/javase/7/docs/api/java/lang/Throwable.html?is-external=true) Handles conversion to an array of the specified type.Specified by: [convertToType](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html#convertToType-java.lang.Class-java.lang.Object-) in class [AbstractConverter](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html) Type Parameters: T - Target type of the conversion. Parameters: type - The type to which this value should be converted. value - The input value to be converted. Returns: The converted value. Throws: [Throwable](https://docs.oracle.com/javase/7/docs/api/java/lang/Throwable.html?is-external=true) - if an error occurs converting to the specified type

#### convertArray protected [Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true) convertArray([Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true) value) Returns the value unchanged.Overrides: [convertArray](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html#convertArray-java.lang.Object-) in class [AbstractConverter](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html) Parameters: value - The value to convert Returns: The value unchanged

#### convertToCollection protected [Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<?> convertToCollection([Class](https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html?is-external=true)<?> type, [Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true) value)

* + - Converts non-array values to a Collection prior to being converted either to an array or a String.  
      * [Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true) values are returned unchanged
      * [Number](https://docs.oracle.com/javase/7/docs/api/java/lang/Number.html?is-external=true), [Boolean](https://docs.oracle.com/javase/7/docs/api/java/lang/Boolean.html?is-external=true) and [Date](https://docs.oracle.com/javase/7/docs/api/java/util/Date.html?is-external=true) values returned as a the only element in a List.
      * All other types are converted to a String and parsed as a delimited list.

**N.B.** The method is called by both the [convertToType(Class, Object)](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#convertToType-java.lang.Class-java.lang.Object-) and [convertToString(Object)](http://docs.google.com/org/apache/commons/beanutils/converters/ArrayConverter.html#convertToString-java.lang.Object-) methods for *non-array* types.Parameters: type - The type to convert the value to value - value to be converted Returns: Collection elements.

#### getDefault protected [Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true) getDefault([Class](https://docs.oracle.com/javase/7/docs/api/java/lang/Class.html?is-external=true)<?> type) Return the default value for conversions to the specified type.Overrides: [getDefault](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html#getDefault-java.lang.Class-) in class [AbstractConverter](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html) Parameters: type - Data type to which this value should be converted. Returns: The default value for the specified type.

#### toString public [String](https://docs.oracle.com/javase/7/docs/api/java/lang/String.html?is-external=true) toString() Provide a String representation of this array converter.Overrides: [toString](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html#toString--) in class [AbstractConverter](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html) Returns: A String representation of this array converter

[Skip navigation links](#1ci93xb)

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* Class
* [Use](http://docs.google.com/class-use/ArrayConverter.html)
* [Tree](http://docs.google.com/package-tree.html)
* [Deprecated](http://docs.google.com/deprecated-list.html)
* [Index](http://docs.google.com/index-all.html)
* [Help](http://docs.google.com/help-doc.html)
* [Prev Class](http://docs.google.com/org/apache/commons/beanutils/converters/AbstractConverter.html)
* [Next Class](http://docs.google.com/org/apache/commons/beanutils/converters/BigDecimalConverter.html)
* [Frames](http://docs.google.com/index.html?org/apache/commons/beanutils/converters/ArrayConverter.html)
* [No Frames](http://docs.google.com/ArrayConverter.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* Field |
* [Constr](#3znysh7) |
* [Method](#2et92p0)
* Detail:
* Field |
* [Constr](#1t3h5sf) |
* [Method](#17dp8vu)

Copyright © 2000–2019 [The Apache Software Foundation](https://www.apache.org/). All rights reserved.