001/\*  
002 \* Licensed to the Apache Software Foundation (ASF) under one or more  
003 \* contributor license agreements. See the NOTICE file distributed with  
004 \* this work for additional information regarding copyright ownership.  
005 \* The ASF licenses this file to You under the Apache License, Version 2.0  
006 \* (the "License"); you may not use this file except in compliance with  
007 \* the License. You may obtain a copy of the License at  
008 \*  
009 \* http://www.apache.org/licenses/LICENSE-2.0  
010 \*  
011 \* Unless required by applicable law or agreed to in writing, software  
012 \* distributed under the License is distributed on an "AS IS" BASIS,  
013 \* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
014 \* See the License for the specific language governing permissions and  
015 \* limitations under the License.  
016 \*/  
017  
018  
019package org.apache.commons.beanutils;  
020  
021  
022/\*\*  
023 \* <p>General purpose data type converter that can be registered and used  
024 \* within the BeanUtils package to manage the conversion of objects from  
025 \* one type to another.</p>  
026 \*  
027 \* <p>Converter subclasses bundled with the BeanUtils library are required  
028 \* to be thread-safe, as users of the library may call conversion methods  
029 \* from more than one thread simultaneously.</p>  
030 \*  
031 \* <p>Custom converter subclasses created by users of the library can be  
032 \* non-thread-safe if the application using them is single-threaded. However  
033 \* it is recommended that they be written in a thread-safe manner anyway.</p>  
034 \*  
035 \* @version $Id$  
036 \* @since 1.3  
037 \*/  
038  
039public interface Converter {  
040  
041  
042 /\*\*  
043 \* Convert the specified input object into an output object of the  
044 \* specified type.  
045 \*  
046 \* @param <T> the desired result type  
047 \* @param type Data type to which this value should be converted  
048 \* @param value The input value to be converted  
049 \* @return The converted value  
050 \*  
051 \* @throws ConversionException if conversion cannot be performed  
052 \* successfully  
053 \*/  
054 public <T> T convert(Class<T> type, Object value);  
055  
056  
057}