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 \*/  
017package org.apache.commons.collections4;  
018  
019/\*\*  
020 \* The "read" subset of the {@link java.util.Map} interface.  
021 \*  
022 \* @param <K> the type of the keys in this map  
023 \* @param <V> the type of the values in this map  
024 \*  
025 \* @since 4.0  
026 \* @see Put  
027 \*/  
028public interface IterableGet<K, V> extends Get<K, V> {  
029 /\*\*  
030 \* Obtains a <code>MapIterator</code> over the map.  
031 \* <p>  
032 \* A map iterator is an efficient way of iterating over maps.  
033 \* There is no need to access the entry set or use Map Entry objects.  
034 \* <pre>  
035 \* IterableMap<String,Integer> map = new HashedMap<String,Integer>();  
036 \* MapIterator<String,Integer> it = map.mapIterator();  
037 \* while (it.hasNext()) {  
038 \* String key = it.next();  
039 \* Integer value = it.getValue();  
040 \* it.setValue(value + 1);  
041 \* }  
042 \* </pre>  
043 \*  
044 \* @return a map iterator  
045 \*/  
046 MapIterator<K, V> mapIterator();  
047  
048}