



语法

public interface CharacterIterator extends Cloneable

描述

CharacterIterator是一个接口，它定义了一个用于对统一码字符进行向前向后迭代的协议。如果一个类想在一个文本范围内移动并且返回单个的统一码字符或它们的索引值，该类就应该实现此接口。

关于它的详细内容参见《The Java Class Libraries，Second Edition，Volume 1》。

版本1.2中所作的修改

为了修补版本1.1中的一个小错误，函数current()、first()、next()、previous()和setIndex()的语意都进行了修改。现在它们能接受由 getEndIndex()所返回的位置值。在这个类以及 StringCharacterIterator中的所有变化都由这个修改所导出。

例如，在下图中，getEndIndex()返回8，但是，last()返回最后一个字符“！”，它的索引是7。在版本1.1中，迭代器只允许指向实际的字符。所以，如果迭代范围为空，或者试图将迭代器指向位置8，迭代器将无处所指，只能得到一个异常。这就意味着如果开发人员不希望在空串上创建一个迭代器或者不希望把迭代器移到串尾，他们就必须要在所编的代码中对这种特殊情况进行处理。在版本1.2中，这样的位置对于迭代器来说是合法的，并且当迭代器指向这样的位置current()将返回DONE。因为这些修改影响了语意，所以这些小错误在版本1.1系列中是不能被修复的。

```
|T|h|e| |e|n|d|!|
0 1 2 3 4 5 6 7 8
|               |
               getEndIndex()
```

| 成员概述 |         |                            |
|------|---------|----------------------------|
| 移动方法 |         |                            |
| Δ    | first() | 把下标移到局部文本的开始，并且返回在该下标处的字符。 |
| Δ    | last()  | 把下标移到局部文本的结尾，并且返回在该下标处的字符。 |
| Δ    | next()  | 下标加1，并返回新下标处的字符。           |

(续)

## 成员概述

|            |                 |                                |
|------------|-----------------|--------------------------------|
| ▲          | previous()      | 下标减1，并返回新下标处的字符。               |
| ▲          | setIndex()      | 把下标移动到指定位置，并返回该处的字符。           |
| 获得下标和字符的方法 |                 |                                |
| ▲          | current()       | 获得当前下标所指向的字符。                  |
|            | getIndex()      | 获得当前下标。                        |
|            | getBeginIndex() | 获得局部文本开始处的下标。                  |
|            | getEndIndex()   | 获得局部文本结尾处的下标。                  |
| 常数         |                 |                                |
|            | DONE            | 当迭代器试图移出文本结尾时该常数将被返回。          |
| 对象方法       |                 |                                |
|            | clone()         | 创建一个该 CharacterIterator 对象的拷贝。 |

## 参见

StringCharacterIterator, AttributedCharacterIterator。

《The Java Class Libraries, Second Edition, Volume 1》中的 CharacterIterator。

## ▲ current()

|           |  |
|-----------|--|
| 目的        | 返回当前下标所指向的字符。  |
| 语法        | public abstract char current()   |
| 描述        | 该方法用于返回当前下标所指向的字符，或如果局部文本为空（也就是，getBeginIndex()和getEndIndex()返回相同的值），则返回StringCharacterIterator.DONE。它与getIndex()所返回的下标相同。该方法不移动当前的下标。 |
| 版本1.2中的改动 | 在版本1.1中，如果局部文本为空，该方法将抛出一个StringIndexOutOfBoundsException异常。在版本1.2中，如果局部文本为空，该方法将返回CharacterIterator.DONE。                              |
| 返回        | 返回当前下标所指向的字符，或者如果局部文本为空，返回StringCharacterIterator.DONE。  |
| 参见        | getIndex()。  |
| 示例        | 见StringCharacterIterator.next()。   |

## ▲ first()

|    |   |
|----|---|
| 目的 | 把下标移到局部文本的开始，并且返回在新下标处的字符。  |
| 语法 | public abstract char first()  |
| 描述 | 把下标移到局部文本的开始，并且返回在该下标处的字符。这个新的下标与getBeginIndex()所返回的下标相同。如果局部文本为空，该方法并不移动迭代器，只是返回CharacterIterator.DONE(局部文本为空，即getBeginIndex()和getEndIndex()返回相同的值)。 |

|           |   |
|-----------|---|
| 版本1.2中的改动 | 在版本 1.1 中，如果局部文本为空，该方法将抛出一个 <code>StringIndexOutOfBoundsException</code> 异常。在版本 1.2 中，如果局部文本为空，该方法将返回 <code>CharacterIterator.DONE</code> 。 |
| 返回        | 局部文本的第一个字符，如果局部文本为空，将返回 <code>CharacterIterator.DONE</code> 。   |
| 参见        | <code>getBeginIndex()</code> 。  |
| 示例        | 见 <code>StringCharacterIterator.next()</code> 。   |

## ▲ last()

|           |   |
|-----------|---|
| 目的        | 把下标移到局部文本的结尾，并且返回在新下标处的字符。  |
| 语法        | <code>public abstract char last()</code>  |
| 描述        | 把下标移到局部文本的结尾，并且返回在该下标处的字符。这个新的下标与 <code>getEndIndex()-1</code> 所返回的下标相同。或当局部文本为空时与 <code>getEndIndex()</code> 所返回的下标相同。对于空的局部文本是一个特殊情况，在这种情况下， <code>last()</code> 并不移动上一个局部文本的下标。如果下一个局部文本为空，该方法不移动该局部文本的索引下标，是返回 <code>CharacterIterator.DONE</code> (如果局部文本为空，即 <code>getBeginIndex()</code> 和 <code>getEndIndex()</code> 返回相同的值)。 |
| 版本1.2中的改动 | 在版本 1.1 中，如果局部文本为空，该方法将设置下标为 <code>getEndIndex()-1</code> ，并抛出一个 <code>StringIndexOutOfBoundsException</code> 异常。在版本 1.2 中，如果局部文本为空，该方法将返回 <code>CharacterIterator.DONE</code> 。  |
| 返回        | 局部文本的最后一个字符，如果局部文本为空，将返回 <code>CharacterIterator.DONE</code> 。  |
| 参见        | <code>getEndIndex()</code> 。  |
| 示例        | 见 <code>StringCharacterIterator.next()</code> 。   |

## ▲ next()

|           |  |
|-----------|--|
| 目的        | 下标加 1，并返回新下标所指向的字符。  |
| 语法        | <code>public abstract char next()</code>   |
| 描述        | 该方法对下标加 1，并返回新下标所指向的字符。当前下标索引通常被限制在从 <code>getBeginIndex()</code> 到 <code>getEndIndex()</code> 的范围内，包括这两个端点。如果迭代器试图指向局部文本末尾以外，当前下标将保持原来位置不变，并返回值 <code>CharacterIterator.DONE</code> 。如果局部文本为空，调用 <code>next()</code> 迭代器并不移动，只是返回 <code>CharacterIterator.DONE</code> (如果 <code>getBeginIndex()</code> 和 <code>getEndIndex()</code> 所返回的值相同)。 |
| 版本1.2中的改动 | 在版本 1.1 中，如果局部文本为空，该方法将抛出一个 <code>StringIndexOutOfBoundsException</code> 异常。在版本 1.2 中，如果局部文本为空，该方法将返回值 <code>CharacterIterator.DONE</code> 。   |
| 返回        | 新下标所指向的字符或 <code>CharacterIterator.DONE</code> 。   |
| 参见        | <code>CharacterIterator.DONE</code> 。  |

示例 见StringCharacterIterator.next()。

## ▲ previous()

|           |  |
|-----------|--|
| 目的        | 下标减1，并返回新下标所指向的字符。   |
| 语法        | public abstract char previous()  |
| 描述        | 该方法对下标减1，并返回新下标所指向的字符。当前下标通常被限制在从getBeginIndex()到getEndIndex()的范围内，包括这两个端点。如果迭代器试图指向局部文本开头以外，当前下标将保持原来位置不变，并返回值CharacterIterator.DONE。如果局部文本为空，调用previous()迭代器并不移动，只是返回CharacterIterator.DONE（如果getBeginIndex()和getEndIndex()所返回的值相同）。 |
| 版本1.2中的改动 | 在版本1.1中，如果局部文本为空，该方法将抛出一个StringIndexOutOfBoundsException异常。在版本1.2中，如果局部文本为空，该方法将返回值CharacterIterator.DONE。   |
| 返回        | 新下标所指向的字符或CharacterIterator.DONE。  |
| 参见        | CharacterIterator.DONE。  |
| 示例        | 见StringCharacterIterator.next()。   |

## ▲ setIndex()

|                          |   |
|--------------------------|---|
| 目的                       | 将下标指向指定位置，并返回新下标所指向的字符。   |
| 语法                       | public abstract char setIndex( int idx )  |
| 描述                       | 该方法将当前下标设置为idx，并返回新下标所指向的字符。目前，如果一个局部文本以外的下标值通过idx传进来，该方法将抛出一个异常。   |
| 版本1.2中的改动                | 在版本1.1中，调用iterator.setIndex( iterator.getEndIndex() )将抛出一个异常，这是存在于版本1.1中的一个小错误，如果局部文本为空，调用setIndex(iterator.getBeginIndex() )也将抛出一个异常。在版本1.2中，在这两种情况下，该方法将返回CharacterIterator.DONE。 |
| 参数                       |   |
| idx                      | 在文本中的一个位置。当getBeginIndex() idx getEndIndex()，idx为合法值。   |
| 返回                       | 指定位置的字符，如果指定位置与getEndIndex()相等，将返回CharacterIterator.DONE。   |
| 异常                       |   |
| IllegalArgumentException | 如果idx超出了合法范围。   |
| 示例                       | 见StringCharacterIterator.next()。  |