

Example Documentation Generated by the L^AT_EX Java 2 doclet

C2 Technologies Inc.

January 3, 2000

Contents

Chapter 1

Package com.c2_tech.doclets

<i>Package Contents</i>	<i>Page</i>
Classes	
Package	??
TableInfo	??
TexDoclet	??

1.1 Classes

1.1.1 Class Package

This class is used to manage the contents of a Java package. It accepts ClassDoc objects and examines them and groups them according to whether they are classes, interfaces, exceptions or errors. The accumulated Vectors can then be processed to get to all of the elements of the package that fall into each category.

Declaration

```
public class Package  
extends java.lang.Object
```

Constructors

- *Package*

```
public Package( java.lang.String pkg )
```

- **Usage**

- * Construct a new object corresponding to the passed package name.

- **Parameters**

- * `pkg` - the package name to use

Methods

- *addElement*

```
public void addElement( com.sun.javadoc.ClassDoc cd )
```

- **Usage**

- * Adds a ClassDoc element to this package.

- **Parameters**

- * `cd` - the object to add to this package

1.1.2 Class TableInfo

This class provides support for converting HTML tables into LaTeX tables. Some of the things **NOT** implemented include the following:

- valign attributes are not procesed, but align= is.
- rowspan attributes are not processed, but colspan= is.
- the argument to border= in the table tag is not used to control line size

Here is an example table.

Column 1 Heading	Column two heading	Column three heading
data	Span two columns	
more data	right	left
A nested table example		
Column 1 Heading	Column two heading	Column three heading
data	Span two columns	
more data	right	left

Declaration

```
public class TableInfo
extends java.lang.Object
```

Constructors

- *TableInfo*

```
public TableInfo( java.util.Properties p, java.lang.StringBuffer ret,
java.lang.String table, int off )

– Usage
* Constructs a new table object and starts processing of the table by scanning the
<table> passed to count columns.

– Parameters
* p - properties found on the <table> tag
* ret - the result buffer that will contain the output
* table - the input string that has the entire table definition in it.
* off - the offset into <table> where scanning should start
```

Methods

- *endCol*

```
public void endCol( java.lang.StringBuffer ret )

– Usage
* Ends the current column.

– Parameters
* ret - the output buffer to put LaTeX into
```

- *endRow*

```
public void endRow( java.lang.StringBuffer ret )

– Usage
* Ends the current row.

– Parameters
* ret - the output buffer to put LaTeX into
```

- *endTable*

```
public void endTable( java.lang.StringBuffer ret )
```

- **Usage**
 - * Ends the table, closing the last row as needed
- **Parameters**
 - * `ret` - the output buffer to put LaTeX into

- *startCol*

```
public void startCol( java.lang.StringBuffer ret, java.util.Properties p )
```

 - **Usage**
 - * Starts a new column, possibly closing the current column if needed
 - **Parameters**
 - * `ret` - the output buffer to put LaTeX into
 - * `p` - the properties from the <td> tag

- *startHeadCol*

```
public void startHeadCol( java.lang.StringBuffer ret, java.util.Properties p )
```

 - **Usage**
 - * Starts a new Heading column, possibly closing the current column if needed. A Heading column has a Bold Face font directive around it.
 - **Parameters**
 - * `ret` - the output buffer to put LaTeX into
 - * `p` - the properties from the <th> tag

- *startRow*

```
public void startRow( java.lang.StringBuffer ret, java.util.Properties p )
```

 - **Usage**
 - * Starts a new row, possibly closing the current row if needed
 - **Parameters**
 - * `ret` - the output buffer to put LaTeX into
 - * `p` - the properties from the <tr> tag

1.1.3 Class TexDoclet

This class provides a Java 2, javadoc Doclet which generates a LaTeX2e document out of the java classes that it is used on. This is convenient for creating printable documentation complete with cross reference information.

Supported HTML tags within comments include the following

<dl> with the associated <dt><dd></dl>tags
<p> but not align=center...yet

 but not clear=xxx
<table> including all the associated <td><th><tr></td></th></tr>
 ordered lists
 unordered lists

```
<font> font coloring  
<pre> preformatted text  
<code> fixed point fonts  
<i> italicized fonts  
<b> bold fonts
```

Declaration

```
public class TexDoclet  
extends com.sun.javadoc.Doclet
```

Fields

- public static PrintWriter os
 - Writer for writing to output file

Constructors

- *TexDoclet*
public TexDoclet()

Methods

- *optionLength*
public static int optionLength(java.lang.String option)
 - **Usage**
* Returns how many arguments would be consumed if **option** is a recognized option.
 - **Parameters**
* **option** - the option to check
- *start*
public static boolean start(com.sun.javadoc.RootDoc root)
 - **Usage**
* Called by the framework to format the entire document
 - **Parameters**
* **root** - the root of the starting document
- *validOptions*
public static boolean validOptions(java.lang.String [][] args,
com.sun.javadoc.DocErrorReporter err)
 - **Usage**
* Checks the passed options and their arguments for validity.
 - **Parameters**
* **args** - the arguments to check
* **err** - the interface to use for reporting errors

Methods inherited from class com.sun.javadoc.Doclet

- *optionLength*
public static int optionLength(java.lang.String)
- *start*
public static boolean start(com.sun.javadoc.RootDoc)
- *validOptions*
public static boolean validOptions(java.lang.String [][] , com.sun.javadoc.DocErrorReporter)