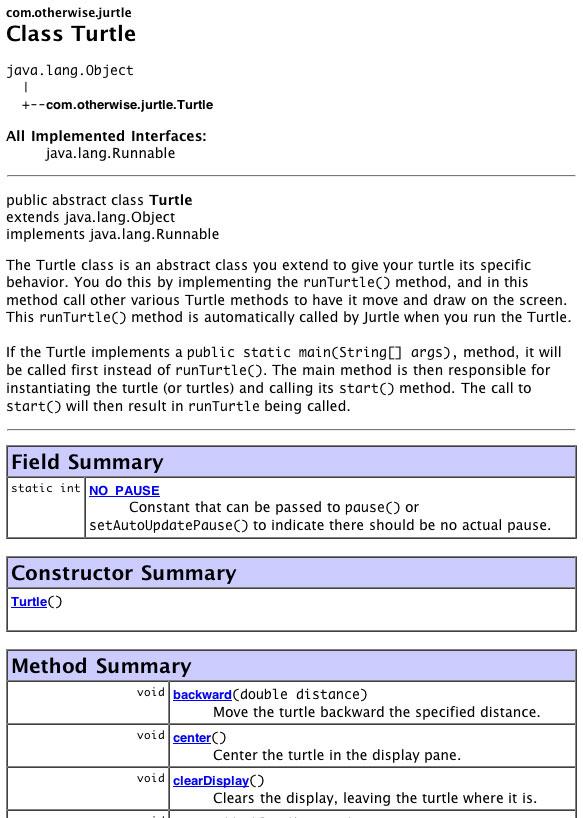
**How To Read A JavaDoc**

**Class Inheritance**

This section shows what package the class is in and where it is in the inheritance hierarchy.  As we can see, Turtle is in the com.otherwise.jurtle package and it inherits directly from Object.

**Class Description**

The Class Description comes from the javadoc comment you put at the top of the class's file.  It usually gives a brief description of what the class does and how to use it.

**Field Summary**

This section lists all the static and instance variables (also called fields) the class declares.  The left-hand column shows the field's data type and whether it is a static variable or an instance variable.  The right-hand column shows the variable's name.  The Turtle class declares only a single static integer called NO\_PAUSE.

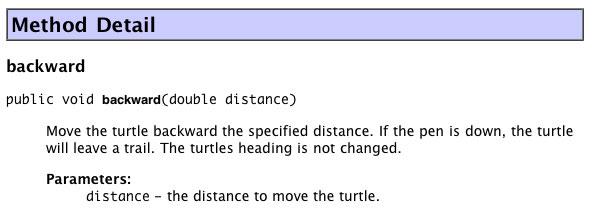
**Constructor Summary**

The next section is the Constructor Summary where the class's constructors are listed.  Recall that constructors are special methods used to create new instances of a class.  There is only one constructor for the Turtle class, and it takes no arguments.

**Method Summary**

The Constructor Summary is followed by the Method Summary section.  All the methods defined in the class are listed with a brief, one-line summary.  The left-hand column shows the method's return type; i.e., the data type of the value returned by the method (or void, if there is none).  The right-hand column shows the method name and the arguments it takes.    
  
The javadoc for the Turtle class only lists public methods.  When generating javadoc, it is possible to specify that private or protected methods also be listed.

**Method Detail**

By clicking on a method name in the Method Summary section, you will be taken to a location later in the same page where the detailed method description is given.  
  
  
  
The detailed description shows the method's name followed by a line showing its declaration in the source file.  Below that is the full description of the method as found in the javadoc comment associated with that method.  Finally, there is a description of the parameters (arguments) the method takes.  Again, the descriptions are taken from the source code's javadoc comments.

**Documenting your code**

When writing a code library that others will be using, it is important to do a thorough job documenting the API.  Most of the code you will be writing in Jurtle will be stand-alone and not for use by other programmers.  Nevertheless, it is still important to comment your code, and a good way to do this is with javadoc comments.  The turtles in the Examples folder show how you might use javadoc comments to document your code.