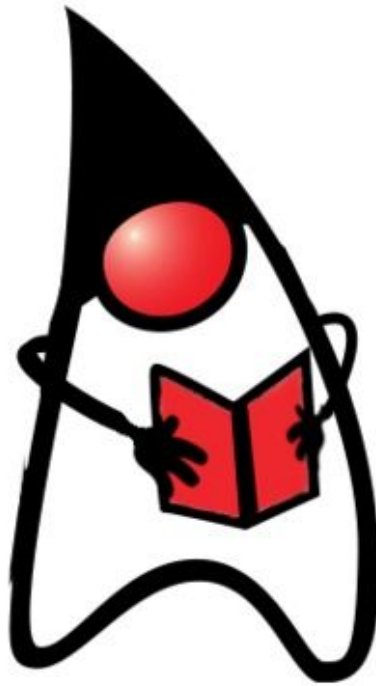


Week 11 - Javadoc

Louis Botha, louis.botha@tuni.fi



[Read more](#)

Commenting in Java

In Java there is 3 ways in which to comment source code:

```
// This is a single line comment
```

```
/*
```

```
 * This is a regular multi-line comment
```

```
*/
```

```
/**
 * This is a Javadoc
 */
```

Now that you are starting to code your own classes, it is time to add some documentation to the classes.

What is Javadoc?

Javadoc is a tool which comes with JDK and it is used for generating Java code documentation in HTML format from Java source code, which requires documentation in a predefined format.

Javadoc comments may be placed above any class, method, or field which we want to document. These comments are commonly made up of two sections:

1. The description of what we're commenting on
2. The standalone block tags (marked with the "@" symbol) which describe specific meta-data

```
/**
 * The class Math contains methods for performing basic .. (short explanation)
 *
 * Unlike some of the numeric methods of class StrictMath, all implementations
 * of the equivalent functions of class Math are not defined to return
 * the bit-for-bit same results... (long explanation)
 *
 * @author Tina Smith
 */
public class Math {
    /**
     * Returns the absolute value of an int value.
     */
}
```

```

    *
    * If the argument is not negative, the
    * argument is returned. If the argument is negative, the n
    * egation of the argument
    * is returned.
    *
    * @param a the argument whose absolute value is to be dete
    * rmined
    * @return the absolute value of the argument.
    */
    public static int abs(int a) {
        if (a < 0) {
            return a * -1;
        } else {
            return a;
        }
    }
}

```

To generate the documentation, try this on the command line:

```
javadoc -d doc Math.java
```

By default the author is not added to the generated documentation.

```
javadoc -author -d doc Math.java
```

This generates the documentation and place it in a directory called `doc`. It generates a lot of file so you want to nearly always make it generate into a specific directory.

Now browse to the directory and open the index.html file with an Internet browser.

PACKAGE
CLASS
TREE
INDEX
HELP

SUMMARY: NESTED | FIELD | CONSTR | METHOD
DETAIL: FIELD | CONSTR | METHOD
SEARCH:

Class Math

java.lang.Object[Ⓜ]
Math

```
public class Math
extends ObjectⓂ
```

The class Math contains methods for performing basic .. (short explanation) Unlike some of the numeric methods of class StrictMath, all implementations of the equivalent functions of class Math are not defined to return the bit-for-bit same results... (long explanation)

Constructor Summary

Constructors

Constructor	Description
Math()	

Method Summary

All Methods
Static Methods
Concrete Methods

Modifier and Type	Method	Description
static int	abs(int a)	Returns the absolute value of an int value.

Methods inherited from class java.lang.Object[Ⓜ]

[clone](#)[Ⓜ], [equals](#)[Ⓜ], [finalize](#)[Ⓜ], [getClass](#)[Ⓜ], [hashCode](#)[Ⓜ], [notify](#)[Ⓜ], [notifyAll](#)[Ⓜ], [toString](#)[Ⓜ], [wait](#)[Ⓜ], [wait](#)[Ⓜ], [wait](#)[Ⓜ]

Constructor Details

Math

```
public Math()
```

Method Details

abs

```
public static int abs(int a)
```

Returns the absolute value of an int value. If the argument is not negative, the argument is returned. If the argument is negative, the negation of the argument is returned.

Parameters:
a - the argument whose absolute value is to be determined

Returns:
the absolute value of the argument.

What to document?

Most important is the first short and long explanation of the class. Explain what this class knows and what it does. A author, version an may be added.

@author	Adds the author of a class.	@author name-text
@version	Adds a "Version" subheading with the specified version-text to the generated	@version version-text

	docs when the -version option is used.	
--	--	--

Above each of the methods in the class, explain what the method does, what parameters the method expects and what the method returns. If the method can throw an exception then it would be good to add that.

@param	Adds a parameter with the specified parameter-name followed by the specified description to the "Parameters" section.	@param parameter-name description
@return	Adds a "Returns" section with the description text.	@return description
@exception	Adds a Throws subheading to the generated documentation, with the classname and description text.	@exception class-name description