Generating documentation for Java programs

Martin Pola

When is documentation necessary?

- The code base becomes big
- To be able to look back at what you did several months or years ago
- Remembering non-trivial classes and relations between objects

When is documentation necessary?

- The code base becomes big
- To be able to look back at what you did several months or years ago
- Remembering non-trivial classes and relations between objects
- *When other people are supposed to interact with your code*

Javadoc

- Automates the process of documentation
- Creates consistent HTML documentation that can be viewed in any modern browser
- Using Javadoc, developers who are familar with Java will know how to read the documentation

Preparing for Javadoc

Add special comments with two asterisks (*) right before class, method, enum or property declarations.

```
/** Some description of what's being declared below */
```

Preparing for Javadoc

Add special comments with two asterisks (*) right before class, method, enum or property declarations.

```
/** Some description of what's being declared below */
```

It's possible to specify metadata in the documentation.

- `@author Martin Pola`
- `@return An array of previously saved book objects.`
- `@param title The title of the book.`

Preparing for Javadoc

Add special comments with two asterisks (*) right before class, method, enum or property declarations.

```java
/** Some description of what's being declared below */
```

It's possible to specify metadata in the documentation.

- `@author Martin Pola`
- `@return An array of previously saved book objects.`
- `@param title The title of the book.`

Navigate to the folder where the documentation should be saved and run the command:

```
$ javadoc C:\path\to\MyClass.java
```