

JavaDocGuidelines

HttpComponents JavaDoc guidelines

Interface description must consist of

- Purpose
- Description (*optional*)
- Statement whether this interface represents an entity or a process defined by a standard specification such as RFC document, when applicable (*optional*)

Class description must consist of

- Purpose
- Description (*optional*)
- Statement whether this class implements an entity or a process defined by a standard specification such as RFC document, when applicable (*optional*)
- Statement whether instances of this class are mutable (*optional*). If not explicitly stated, instances of this class are assumed to be mutable
- Statement whether this class is multi-threading safe (*optional*). If not explicitly stated, the class is assumed to be **NOT** threading-safe

Method description must consist of

- Purpose
- Expected effect
- Description (*optional*)
- Statement whether this method is modal, that is, if the object must be in a specific state (pre-conditions) or mode in order to execute correctly (*optional*). If not explicitly stated, the method is assumed to be non-modal.
- Required parameters. It must be explicitly stated whether null is permitted as a parameter value. Per default parameters are assumed to require a non-null value and to cause a

```
IllegalArgumentException
```

if null is given.

- Exceptions that can be thrown in the course of method's execution and their possible cause (post-conditions).
- private methods can have minimal or no JavaDocs if they are short and trivial enough.
- methods that implement a signature from an interface or base class that is sufficiently documented there can omit JavaDocs in favor of comments like:

```
// non-javadoc, see interface XXX
```

```
// non-javadoc, see class XYZ
```

The non-javadoc comment with a pointer to the documentation is required for people reading the source code. The JavaDoc tool will automatically copy the description from the interface or base class into the generated documentation.