

# Code Style Guidelines

In order to improve readability ([Why readability matters](#)) here are some code style guidelines for the JMeter project.

## Java

We now enforce many of these rules, and more using checkstyle.

### Indentation/White space

- 4 spaces for indentation
  - No tabs 
- "Soft" maximum line length of ~80 characters
  - Editors and monitors used by most programmers with good eyes can easily handle 120 characters. However:
    - lines longer than this can be harder to read and frustrating work with
    - not all programmers have good eyes and/or wide high resolution monitors
    - in typography line length is recommended to be [45-75](#) with the optimum being 66 or [45-90](#)
  - Here is an example from `UrlConfigGui.java`
    - which do you find easier to read?

```
element.setProperty(HTTPSamplerBase.BROWSER_COMPATIBLE_MULTIPART,
useBrowserCompatibleMultipartMode.isSelected(),HTTPSamplerBase.
BROWSER_COMPATIBLE_MULTIPART_MODE_DEFAULT);
```

or

```
element.setProperty(
    HTTPSamplerBase.BROWSER_COMPATIBLE_MULTIPART,
    useBrowserCompatibleMultipartMode.isSelected(),
    HTTPSamplerBase.BROWSER_COMPATIBLE_MULTIPART_MODE_DEFAULT);
```

- "Hard" maximum line length of 120 characters
  - Except for imports or other places where breaking the line wouldn't aid readability

### Spacing

- Spacing between elements on a line e.g.
  - `if (x | y) {`
  - `while (true) {`
  - `methodCall(arg1, arg2) {`
  - `String s = "con" + "cat";`
- Braces are always used with `if, else, for, do` and `while` statements, even if the body is empty or only a single statement

### JavaDoc

- Parameter descriptions are aligned and **not** on new lines.
- No invalid tags
- No tags without a description
- Blank line after main description but not after any tags

### Other

- Import order
  - `java, javax, org, net, com, others`
- Import spacing
  - line break between each group
- No wildcard `(.* )` imports
- Method line length (soft limit of 50)
- Class line length (soft limit of 500)

### Java 8 Specific

- Use of `Optional`
  - Return types where null is possible (which aren't performance critical)
- Default/static methods on Interfaces
- Utilise lambdas where possible (max ~5 lines)

- If the lambda is >3 line then consider making this a separate method

## See Also

- <https://google.github.io/styleguide/javaguide.html>
- <http://javaranch.com/style.jsp>