log4j.appender.ConsoleAppender.layout.ConversionPattern=**[%-5p] %d %c - %m%n**

Where:

-          The percent sign (%) is a prefix for any conversion characters.

-          Conversion characters are: **p, d, c, m** and **n**. These characters are explained in the table below.

-          **-5p** is the format modifier for the conversion character p (priority), which left justifies the priority name with minimum 5 characters.

-          Other characters are literals: **[, ], -** and spaces

That pattern will format log messages something like this:

|  |  |
| --- | --- |
| 1  2  3 | [DEBUG] 2012-11-02 21:57:53,661 MyClass - this is a debug log message  [INFO ] 2012-11-02 21:57:53,662 MyClass - this is a information log message  [WARN ] 2012-11-02 21:57:53,662 MyClass - this is a warning log message |

The following table briefly summarizes the conversion characters defined by log4j:

|  |  |  |
| --- | --- | --- |
| **What to print** | **Conversion character** | **Performance** |
| Category name (or logger name) | **c** | Fast |
| Fully qualified class name | **C** | Slow |
| Date and time | **d**  **d{format}** | *Slow if using JDK’s formatters.*  *Fast if using log4j’s formatters.* |
| File name of Java class | **F** | **Extremely slow** |
| Location (class, method and line number) | **l** | **Extremely slow** |
| Line number only | **L** | **Extremely slow** |
| Log message | **m** | Fast |
| Method name | **M** | **Extremely slow** |
| Priority (level) | **p** | Fast |
| New line separator | **n** | Fast |
| Thread name | **t** | Fast |
| Time elapsed (milliseconds) | **r** | Fast |
| Thread’s nested diagnostic context | **x** | Fast |
| Thread’s mapped diagnostic context | **X** | Fast |
| Percent sign | **%%** | Fast |

