|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **说明** | **例子** | |
| %c | 列出logger名字空间的全称，如果加上{<层数>}表示列出从最内层算起的指定层数的名字空间 | log4j配置文件参数举例 | 输出显示媒介 |
| 假设当前logger名字空间是“a.b.c” | |
| %c | a.b.c |
| %c{2} | b.c |
| %20c | （若名字空间长度小于20，则左边用空格填充） |
| %-20c | （若名字空间长度小于20，则右边用空格填充） |
| %.30c | （若名字空间长度超过30，截去多余字符） |
| %20.30c | （若名字空间长度小于20，则左边用空格填充；若名字空间长度超过30，截去多余字符） |
| %-20.30c | （若名字空间长度小于20，则右边用空格填充；若名字空间长度超过30，截去多余字符） |
| %C | 列出调用logger的类的全名（包含包路径） | 假设当前类是“org.apache.xyz.SomeClass” | |
| %C | org.apache.xyz.SomeClass |
| %C{2} | xyz.SomeClass |
| %d | 显示日志记录时间，{<日期格式>}使用ISO8601定义的日期格式 | %d{yyyy/MM/dd HH:mm:ss,SSS} | 2005/10/12 22:23:30,117 |
| %d{ABSOLUTE} | 22:23:30,117 |
| %d{DATE} | 12 Oct 2005 22:23:30,117 |
| %d{ISO8601} | 2005-10-12 22:23:30,117 |
| %F | 显示调用logger的源文件名 | %F | MyClass.java |
| %l | 输出日志事件的发生位置，包括类目名、发生的线程，以及在代码中的行数 | %l | MyClass.main(MyClass.java:129) |
| %L | 显示调用logger的代码行 | %L | 129 |
| %m | 显示输出消息 | %m | This is a message for debug. |
| %M | 显示调用logger的方法名 | %M | main |
| %n | 当前平台下的换行符 | %n | Windows平台下表示rn UNIX平台下表示n |
| %p | 显示该条日志的优先级 | %p | INFO |
| %r | 显示从程序启动时到记录该条日志时已经经过的毫秒数 | %r | 1215 |
| %t | 输出产生该日志事件的线程名 | %t | MyClass |
| %x | 按NDC（Nested Diagnostic Context，线程堆栈）顺序输出日志 | 假设某程序调用顺序是MyApp调用com.foo.Bar | |
| %c %x - %m%n | MyApp - Call com.foo.Bar. com.foo.Bar - Log in Bar MyApp - Return to MyApp. |
| %X | 按MDC（Mapped Diagnostic Context，线程映射表）输出日志。通常用于多个客户端连接同一台服务器，方便服务器区分是那个客户端访问留下来的日志。 | %X{5} | （记录代号为5的客户端的日志） |
| %% | 显示一个百分号 | %% | % |