Log4j配置文件使用说明

在项目路径下建一个configure(此名字可以自定义)文件夹,新建log4j.properties文件为Log4j的配置使用文件;

# 1：配置根(rootLogger)：

## ①配置根Logger，其语法为： log4j.rootLogger = [level],appenderName,appenderName2,...

#rootLogger

log4j.rootLogger=info,error,Console, FileInfo

其中:

---info:表示优先级;后面的appName可以自定义;eg: error,Console, FileInfo

分别代表:输出的目的地, Console:控制台;

---error和FileInfo在此是为了记录所有模块里面的所有对应的信息的;名字可以自定义;

注意:一般在log4j.rootLogger后面定义的优先级只有第一个,表示输出的最低级别的;

# 2:配置子根(logger)

**log4j.logger.**com.ber.modelA = info,modelA

log4j.logger.com.ber.modelB = info,modelB

log4j.logger.com.ber.modelC = info,modelC

其中: **log4j.logger后面跟的是包名:** eg:com.ber.modelA;表示com.ber.modelA包下面的所有的类;此处表示项目中的模块A.

注意:此处也可以自定义子根,再继续细分.

同里(其他模块)也可以根据需要自定义所在项目中的所有模块的信息,对应的包路径在此处配置即可.

# 3:配置定义的对应模块的信息:

以ModelA为列来举列说明:

####################modelA####################

log4j.appender.modelA=org.apache.log4j.MyLog4jRollingFileAppender

此处表示:

log4j.appender.modelA.file=./home/yyyy/MM/dd/modelA/info.log

此处表示:把模块A日记存储的路径;

#### 下面两行表示:日志打印 10M 6个循环文件

log4j.appender.modelA.MaxFileSize=10240KB

此处表示:可以根据具体的需要设置文件的大小;单位为:KB,M,G等;

log4j.appender.modelA.MaxBackupIndex=6

此处表示:可以根据具体的需要设置需要备份的数目.此处为最多可以备份6份文件;

log4j.appender.modelA.DatePattern='.'yyyy-MM-dd-HH

此处表示:可以根据需要设置格式:

log4j.appender.modelA.layout=org.apache.log4j.PatternLayout

此处表示:可以根据需要设置输出的具体格式.

log4j.appender.modelA.layout.ConversionPattern=[%-5p] [%d{yyyy/MM/dd HH\:mm\:ss\:SSS}] method\:%l%n%m%n

同理其它模块也可以配置实现.

4:设置输出到控制台的信息# 打印到控制台

#################Console####################

log4j.appender.Console=org.apache.log4j.ConsoleAppender

log4j.appender.Console.layout=org.apache.log4j.PatternLayout

log4j.appender.Console.layout.ConversionPattern=[%-5p] [%d{yyyy/MM/dd HH\:mm\:ss\:SSS}] method\:%l%n%m%n

5:设置用于保存项目中的所有的error的信息;

#################error####################

log4j.logger.error=error

log4j.appender.error=org.apache.log4j.MyLog4jRollingFileAppender

log4j.appender.error.Threshold = ERROR

此处Threshold:表示输出的最低级别为ERROR的信息;可以根据需要设置优先级;

log4j.appender.error.layout=org.apache.log4j.PatternLayout

设置输出的格式;

log4j.appender.error.layout.ConversionPattern=[%-5p] [%d{yyyy/MM/dd HH\:mm\:ss\:SSS}] method\:%l%n%m%n

log4j.appender.error.datePattern='.'yyyy-MM-dd-HH

og4j.appender.error.append=true

此处append等于true表示追加,否则等于false表示覆盖之前的记录信息;

log4j.appender.error.MaxFileSize=1KB

log4j.appender.error.MaxBackupIndex=6

log4j.appender.error.File=./home/yyyy/MM/dd/error/error.log

# 6:主要用到一个自定义的类

MyLog4jRollingFileAppender.java

根据日期记录对应的信息.

# 7:其它相关配置信息:

----------🡪如果需要将不同的日志打印到不同的地方，则需要定义不同的Appender，然后定义每一个 Appender的日志级别、打印形式、输出位置！

注意:以下为具体的Log4j提供的Appender;

----------🡪Log4j提供的appender有以下几种：   
#1)org.apache.log4j.ConsoleAppender(输出到控制台)   
#2)org.apache.log4j.FileAppender(输出到文件)   
#3)org.apache.log4j.DailyRollingFileAppender(每天产生一个日志文件)   
#4)org.apache.log4j.RollingFileAppender(文件大小到达指定尺寸的时候产生一个新的文件)   
#5)org.apache.log4j.WriterAppender(将日志信息以流格式发送到任意指定的地方)   
1)ConsoleAppender选项属性   
-Threshold = DEBUG:指定日志消息的输出最低层次   
-ImmediateFlush = TRUE:默认值是true,所有的消息都会被立即输出

-Target = System.err:默认值System.out,输出到控制台(err为红色,out为黑色)

2)FileAppender选项属性   
 -Threshold = INFO:指定日志消息的输出最低层次   
 -ImmediateFlush = TRUE:默认值是true,所有的消息都会被立即输出   
 -File = C:\log4j.log:指定消息输出到C:\log4j.log文件   
 -Append = FALSE:默认值true,将消息追加到指定文件中，false指将消息覆盖指定的文件内容   
 -Encoding = UTF-8:可以指定文件编码格式   
3)DailyRollingFileAppender选项属性   
 -Threshold = WARN:指定日志消息的输出最低层次   
 -ImmediateFlush = TRUE:默认值是true,所有的消息都会被立即输出   
-File = C:\log4j.log:指定消息输出到C:\log4j.log文件   
-Append = FALSE:默认值true,将消息追加到指定文件中，false指将消息覆盖指定的文件内容   
 -DatePattern='.'yyyy-ww:每周滚动一次文件,即每周产生一个新的文件。还可以按用以下参数:

'.'yyyy-MM:每月   
 '.'yyyy-ww:每周   
 '.'yyyy-MM-dd:每天   
 '.'yyyy-MM-dd-a:每天两次

'.'yyyy-MM-dd-HH:每小时   
 '.'yyyy-MM-dd-HH-mm:每分钟   
-Encoding = UTF-8:可以指定文件编码格式   
4)RollingFileAppender选项属性   
-Threshold = ERROR:指定日志消息的输出最低层次   
-ImmediateFlush = TRUE:默认值是true,所有的消息都会被立即输出   
 -File = C:/log4j.log:指定消息输出到C:/log4j.log文件   
-Append = FALSE:默认值true,将消息追加到指定文件中，false指将消息覆盖指定的文件内容   
 -MaxFileSize = 100KB:后缀可以是KB,MB,GB.在日志文件到达该大小时,将会自动滚动.如:log4j.log.1   
 -MaxBackupIndex = 2:指定可以产生的滚动文件的最大数   
 -Encoding = UTF-8:可以指定文件编码格式

## 8:日志信息格式中几个符号所代表的含义：

　　 －X号: X信息输出时左对齐；

%p: 输出日志信息优先级，即DEBUG，INFO，WARN，ERROR，FATAL,

%d: 输出日志时间点的日期或时间，默认格式为ISO8601，也可以在其后指定格式，比如：%d{yyy MMM dd HH:mm:ss,SSS}，输出类似：2002年10月18日 22：10：28，921

%r: 输出自应用启动到输出该log信息耗费的毫秒数

%c: 输出日志信息所属的类目，通常就是所在类的全名

%t: 输出产生该日志事件的线程名

%l: 输出日志事件的发生位置，相当于%C.%M(%F:%L)的组合,包括类目名、发生的线程，以及在代码中的行数。举例：Testlog4.main(TestLog4.java:10)

%x: 输出和当前线程相关联的NDC(嵌套诊断环境),尤其用到像java servlets这样的多客户多线程的应用中。

%%: 输出一个"%"字符

%F: 输出日志消息产生时所在的文件名称

%L: 输出代码中的行号

%m: 输出代码中指定的消息,产生的日志具体信息

%n: 输出一个回车换行符，Windows平台为"\r\n"，Unix平台为"\n"输出日志信息换行

# 9:Log4j对日记文件的截取提供了两种实现方式:

日期(DailyRollingFileAppender)和文件是否超出指定的大小值,判断是否需要回滚文件(RollingFileAppender)实现;