新建项目 log4jSample

新建package,新建main函数

添加引用

<dependency>

<groupId>org.apache.logging.log4j</groupId>

<artifactId>log4j-core</artifactId>

<version>2.5</version>

</dependency>

<dependency>

<groupId>org.apache.logging.log4j</groupId>

<artifactId>log4j-api</artifactId>

<version>2.5</version>

</dependency>

在resource目录下 新建 log4j2.xml

<?xml version="1.0" encoding="UTF-8"?>

<configuration status="error">

<!-- 先定义所有的appender -->

<appenders>

<!-- 这个输出控制台的配置 -->

<Console name="Console" target="SYSTEM\_OUT">

<!-- 控制台只输出level及以上级别的信息（onMatch），其他的直接拒绝（onMismatch） -->

<ThresholdFilter level="trace" onMatch="ACCEPT" onMismatch="DENY"/>

<!-- 这个都知道是输出日志的格式 -->

<PatternLayout pattern="%d{HH:mm:ss.SSS} %-5level %t %class{36} %L %M - %msg%xEx%n"/>

</Console>

<!-- 文件会打印出所有信息，这个log每次运行程序会自动清空，由append属性决定，这个也挺有用的，适合临时测试用 -->

<!-- append为TRUE表示消息增加到指定文件中，false表示消息覆盖指定的文件内容，默认值是true -->

<File name="log" fileName="logs/test.log" append="false">

<PatternLayout pattern="%d{HH:mm:ss.SSS} %-5level %t %class{36} %L %M - %msg%xEx%n"/>

</File>

<!-- 添加过滤器ThresholdFilter,可以有选择的输出某个级别以上的类别 onMatch="ACCEPT" onMismatch="DENY"意思是匹配就接受,否则直接拒绝 -->

<File name="ERROR" fileName="logs/error.log">

<ThresholdFilter level="error" onMatch="ACCEPT" onMismatch="DENY"/>

<PatternLayout pattern="%d{yyyy.MM.dd 'at' HH:mm:ss z} %-5level %t %class{36} %L %M - %msg%xEx%n"/>

</File>

<!-- 这个会打印出所有的信息，每次大小超过size，则这size大小的日志会自动存入按年份-月份建立的文件夹下面并进行压缩，作为存档 -->

<RollingFile name="RollingFile" fileName="logs/web.log"

filePattern="logs/$${date:yyyy-MM}/web-%d{MM-dd-yyyy}-%i.log.gz">

<PatternLayout pattern="%d{yyyy-MM-dd 'at' HH:mm:ss z} %-5level %t %class{36} %L %M - %msg%xEx%n"/>

<SizeBasedTriggeringPolicy size="2MB"/>

</RollingFile>

</appenders>

<!-- 然后定义logger，只有定义了logger并引入的appender，appender才会生效 -->

<loggers>

<!-- 建立一个默认的root的logger -->

<root level="trace">

<appender-ref ref="RollingFile"/>

<appender-ref ref="Console"/>

<appender-ref ref="ERROR" />

<appender-ref ref="log"/>

</root>

</loggers>

</configuration>

在mainclass 中 添加

private static Logger logger = LogManager.getLogger(MainClass.class.getName());

main方法中添加

logger.trace("aa");