Springboot 配置使用log4j2

<dependency>  
<groupId>org.springframework.boot</groupId>  
<artifactId>spring-boot-starter-web</artifactId>  
</dependency>  
<dependency>  
<groupId>org.springframework.boot</groupId>  
<artifactId>spring-boot-starter</artifactId>  
<exclusions>  
<exclusion>  
<groupId>org.springframework.boot</groupId>  
<artifactId>spring-boot-starter-logging</artifactId>  
</exclusion>  
</exclusions>  
</dependency>  
  
<dependency>  
<groupId>org.springframework.boot</groupId>  
<artifactId>spring-boot-starter-log4j2</artifactId>  
</dependency>

这个依赖是官方给的。

日志配置文件默认是xml形式的。如果需要使用yml文件配置，需要添加依赖

<dependency>  
<groupId>com.fasterxml.jackson.dataformat</groupId>  
<artifactId>jackson-dataformat-yaml</artifactId>  
<version>2.9.9</version>  
</dependency>

这个也是springboot官方给的。

最后就是日志配置文件

Appenders:  
Console:  
name: CONSOLE  
target: SYSTEM\_OUT  
PatternLayout:  
pattern: "%d{yyyy-MM-dd HH:mm:ss,SSS}:%4p %t (%F:%L) - %m%n"  
RollingFile:  
 - name: ROLLING\_FILE  
ignoreExceptions: false  
fileName: /logs/kafka\_influxdb.log  
filePattern: "/logs/kafka\_influxdb -%d{yyyy-MM-dd}-%i.log.gz"  
PatternLayout:  
pattern: "%d{yyyy-MM-dd HH:mm:ss,SSS}:%4p %t (%F:%L) - %m%n"  
Policies:  
SizeBasedTriggeringPolicy:  
size: "256 MB"  
DefaultRolloverStrategy:  
max: 1000  
Loggers:  
Root:  
level: info  
AppenderRef:  
 - ref: CONSOLE  
Logger:  
 - name: com.java.kafkainfluxdb  
additivity: false  
level: trace  
AppenderRef:  
 - ref: CONSOLE  
 - ref: ROLLING\_FILE

说明一下，

如果是数组，变量需要配置多个值，前面用横线—。

有百分号的值须要用引号括起来。

日志文件默认都是log4j2.\*

可以在springboot的配置文件指定日志配置文件，可以不遵循这个。比如日志配置文件为log4j2-xxx.yml，就可以如下配置。

logging:

config:

classpath: log4j2-xxx.yml

如果要使用slf4j的api

<dependency>  
<groupId>org.slf4j</groupId>  
<artifactId>slf4j-api</artifactId>  
<version>1.7.26</version>  
</dependency>

上面说的是在springboot环境中配置。在其他环境配置应该有些差异。

官网有一些中文文章和文档可以参考学习

<https://logging.apache.org/log4j/2.x/articles.html>

现在还不知道日志配置文件怎么配置。慢慢学习

　%m 输出代码中指定的消息   
    %p 输出优先级，即DEBUG，INFO，WARN，ERROR，FATAL   
    %r 输出自应用启动到输出该log信息耗费的毫秒数   
    %c 输出所属的类目，通常就是所在类的全名   
    %t 输出产生该日志事件的线程名   
    %n 输出一个回车换行符，Windows平台为“rn”，Unix平台为“n”   
    %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{yyyy MMM dd HH:mm:ss,SSS}，输出类似：2002年10月18日 ：10：28，921  
    %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。

    %x Used to output the NDC (nested diagnostic context) associated with the thread that generated the logging event  
    %X Used to output the MDC (mapped diagnostic context) associated with the thread that generated the logging event for specified key