

Spring Boot: logback 사용법

pom.xml

- logback-classic 의존성 추가

```
48  <!-- LJM: 추가한 의존성 -->
49  <dependency>
50      <groupId>org.apache.tomcat.embed</groupId>
51      <artifactId>tomcat-embed-jasper</artifactId>
52  </dependency>
53  <dependency>
54      <groupId>javax.servlet</groupId>
55      <artifactId>jstl</artifactId>
56  </dependency>
57
58  <dependency>
59      <groupId>ch.qos.logback</groupId>
60      <artifactId>logback-classic</artifactId>
61  </dependency>
62
63  </dependencies>
```

src/main/resources/logback.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <appender name="HTML" class="ch.qos.logback.core.FileAppender">
    <file>log.html</file>
    <encoder class="ch.qos.logback.core.encoder.LayoutWrappingEncoder">
      <layout class="ch.qos.logback.classic.html.HTMLLayout"/>
    </encoder>
  </appender>
  <appender name="console" class="ch.qos.logback.core.ConsoleAppender">
    <encoder>
      <pattern>%-5p, %d [%t-thread %c:%M]:%m%n</pattern>
    </encoder>
  </appender>
  <root level="INFO">
    <appender-ref ref="HTML"/>
    <appender-ref ref="console"/>
  </root>
</configuration>
```

Spring Boot: log4j 2 사용법

pom.xml (1/2)

```
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
    <!-- 기본적으로 적용되는 logback 제외 -->
    <exclusions>
      <exclusion>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-logging</artifactId>
      </exclusion>
    </exclusions>
  </dependency>
```

pom.xml (2/2)

```
<!-- dependency>  
  <groupId>ch.qos.logback</groupId>  
  <artifactId>logback-classic</artifactId>  
</dependency -->
```

```
<!-- Log4j 2 dependencies -->  
<!-- Ref. URL: https://www.baeldung.com/spring-boot-logback-log4j2 -->
```

```
<dependency>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-starter-log4j2</artifactId>  
</dependency>  
<dependency>  
  <groupId>org.apache.logging.log4j</groupId>  
  <artifactId>log4j-spring-boot</artifactId>  
</dependency>
```

```
</dependencies>
```

src/main/resources/log4j2.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<Configuration status="INFO">
  <Appenders>
    <!-- Console Appender -->
    <Console name="ConsoleAppender" target="SYSTEM_OUT">
      <PatternLayout pattern="%d{yyyy-MM-dd HH:mm:ss} [%t] %-5level %logger{36} - %msg%n" />
    </Console>

    <!-- File Appender with HtmlLayout -->
    <!-- ISO 8601: 날짜와 시간의 표기에 관한 국제 표준 규격 -->
    <File name="FileAppender" fileName="logs/app.html">
      <HtmlLayout datePattern="ISO8601" timezone="GMT+0"/>
    </File>
  </Appenders>
```

```
<Loggers>
  <!-- Root Logger -->
  <Root level="info">
    <AppenderRef ref="FileAppender" />
    <AppenderRef ref="ConsoleAppender" />

  </Root>
</Loggers>
</Configuration>
```

application.properties

- 로깅 환경 설정 파일 지정
 - logging.config=classpath:logback.xml 또는
 - logging.config=classpath:log4j2.xml (log4j2.json, log4j2.yaml, log4j2.properties 등도 가능)

log4j2.properties 예

Log4j2 내부 로깅 레벨

status = warn

콘솔 Appender 설정

appender.console.type = Console

appender.console.name = LogToConsole

appender.console.layout.type = PatternLayout

appender.console.layout.pattern = [%-5level] %d{yyyy-MM-dd HH:mm:ss.SSS} [%t] %c{1} - %msg%n

롤링 파일 Appender 설정

appender.rolling.type = RollingFile

appender.rolling.name = LogToRollingFile

appender.rolling.fileName = logs/app.log

appender.rolling.filePattern = logs/\${date:yyyy-MM}/app-%d{MM-dd-yyyy}-%i.log.gz

appender.rolling.layout.type = PatternLayout

appender.rolling.layout.pattern = %d %p %C{1.} [%t] %m%n

appender.rolling.policies.type = Policies

```
appender.rolling.policies.time.type = TimeBasedTriggeringPolicy
appender.rolling.policies.size.type = SizeBasedTriggeringPolicy
appender.rolling.policies.size.size = 10MB
appender.rolling.strategy.type = DefaultRolloverStrategy
appender.rolling.strategy.max = 10
```

특정 패키지 로거 설정

```
logger.app.name = com.example
logger.app.level = debug
logger.app.additivity = false
logger.app.appenderRef.rolling.ref = LogToRollingFile
logger.app.appenderRef.console.ref = LogToConsole
```

루트 로거 설정

```
rootLogger.level = info
rootLogger.appenderRef.stdout.ref = LogToConsole
```