**전자정부프레임워크(egov) logback 설정**

1. 로그(Log)

로그는 기록을 남기는 것을 의미함.

구체적으로는 프로그램 개발이나 운영 시 발생하는 문제점을 추적 하거나 운영 상태를 모니터링 하는 정보를 기록하는 것임.

또한 분석을 통해 통계를 낼 수도 있기 때문에 기록을 남기는 것은 중요하다고 할 수 있음.

하지만 로그를 남기면 성능이 나빠진다는 단점이 있는데, 그 보다 로그를 통해 얻는 정보가 훨씬 많기 때문에 필요한 부분에 파일로써 로그를 남기는 것이 중요함

1. logback

스프링에서는 기본적으로 commons.logging 라이브러리 (Apache의 JCL, Jakarta Commons Logging)을 사용한다.

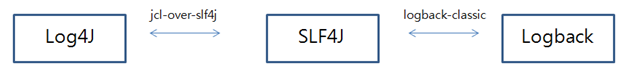
즉 스프링 개발을 할 때 스프링이 뿜어내는 메시지는 JCL에 의존하여 로그를 남기는 것이다.

실제로 spring-context 라이브러리를 설치할 때 프로젝트 폴더의 Maven Libarary를 확인해보면 commons.logging 라이브러리를 확인할 수 있다.



예전에는 스프링에서 로그를 남길 때 Log4J를 사용했었는데, 성능 및 기능상의 이유로 대체 logger들이 많아졌고, 현재 대부분은 SLF4J 인터페이스를 구현한 Logback을 사용한다.

Spring이 기존에 사용하던 로그 라이브러리 JCL 대신, 새로운 라이브러리 Logback을 사용하도록 하기 위해서는 SLF4J가 필요하다. >> 즉 SLF4J는 JCL과 Log4J의 징검다리 역할을 하는 셈이다.



1. logback환경 설정

pom.xml

|  |
| --- |
| <properties>           <org.springframework-version> 4.2.1.RELEASE </org.springframework-version>           <jcloverslf4j.version>1.7.6</jcloverslf4j.version>           <logback.version>1.1.1</logback.version>  </properties>  <dependencies>           <!-- Spring core -->           <dependency>                    <groupId>org.springframework</groupId>                    <artifactId>spring-context</artifactId>                    <version>${org.springframework-version}</version>                    <!-- JCL 제외 -->                    <exclusions>                             <exclusion>                                     <groupId>commons-logging</groupId>                                     <artifactId>commons-logging</artifactId>                             </exclusion>                    </exclusions>           </dependency>           <!-- Logback -->           <dependency>                    <groupId>org.slf4j</groupId>                    <artifactId>jcl-over-slf4j</artifactId>                    <version>${jcloverslf4j.version}</version>           </dependency>           <dependency>                    <groupId>ch.qos.logback</groupId>                    <artifactId>logback-classic</artifactId>                    <version>${logback.version}</version>           </dependency>  ... |

1. 전자정부프레임워크(egov)에서 logback라이브러리 설정

기존에 spring에서 logback을 사용하던 설정 그대로 egov에 적용해도 적용이 안될 때가 있다.

logback이 아닌 log4j 라이브러리를 참고하기 때문에 나타나는 현상이며 이럴 경우에는 log4j 관련 설정을 무시하도록 설정한다 >> pom.xml에 log4j 무시 exclusion 추가

pom.xml

|  |
| --- |
| <dependency>  <groupId>egovframework.rte</groupId>  <artifactId>egovframework.rte.fdl.logging</artifactId>  <version>${egovframework.rte.version}</version>  <exclusions>  <exclusion>  <groupId>org.apache.logging.log4j</groupId>  <artifactId>log4j-core</artifactId>  </exclusion>  <exclusion>  <groupId>org.apache.logging.log4j</groupId>  <artifactId>log4j-slf4j-impl</artifactId>  </exclusion>  <exclusion>  <groupId>org.slf4j</groupId>  <artifactId>jcl-over-slf4j</artifactId>  </exclusion>  <exclusion>  <groupId>org.slf4j</groupId>  <artifactId>log4j-over-slf4j</artifactId>  </exclusion>  </exclusions>  </dependency> |

1. Logback 적용

/src/main/resources/logback.xml

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <configuration>  <!-- 콘솔로 로그를 남김 -->  <appender name="consoleAppender" class="ch.qos.logback.core.ConsoleAppender">  <encoder>  <charset>UTF-8</charset>  <!-- 로그 메시지 패턴 -->  <Pattern>%d{HH:mm:ss.SSS} [%thread] %-5level %logger{5} - %msg%n</Pattern>  </encoder>  </appender>  <!-- 파일로 로그를 남김 -->  <appender name="fileAppender" class="ch.qos.logback.core.rolling.RollingFileAppender">  <file>c:\LogExample\logexample2.log</file>  <encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">  <Pattern>%d{HH:mm:ss.SSS} [%thread] %-5level %logger{5} - %msg%n</Pattern>  </encoder>  <!-- 로그를 남기는 파일의 용량이 50KB가 넘으면 이를 압축 파일로 만들고  새로 로그 파일로 만들라는 정책 -->  <triggeringPolicy  class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">  <MaxFileSize>10KB</MaxFileSize>  </triggeringPolicy>  <!-- 파일을 덮어쓰는 정책 -->  <rollingPolicy class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">  <FileNamePattern>C:\LogExample\logexample2.%i.log.zip</FileNamePattern>  <!-- MinIndex가 1이고, MaxIndex가 10이므로, 위의 파일 이름 패턴에 따라  아래의 로그 파일이 생길 것이다.  logexample2.1.log.zip logexample2.2.log.zip .... logexample2.10.log.zip  이 상태에서 또 10KB가 넘으면 logexample2.1.log.zip이 된다.  -->  <MinIndex>1</MinIndex>  <MaxIndex>10</MaxIndex>  </rollingPolicy>  </appender>  <!-- com.victolee.logExample 아래 패키지 로그들만 consoleAppender, fileAppender 방법으로  로그를 남긴다.  물론 <appender-ref ref="consoleAppender" />를 추가하여 콘솔로도 로그를 남길 수 있다.  -->  <logger name="com.victolee.logExample" level="info" additivity="false">  <appender-ref ref="fileAppender" />  </logger>  <!-- root는 글로벌 로거를 의미하며, 위의 logger에 해당하지 않으면 root 로거가 실행된다. -->  <root level="warn">  <appender-ref ref="consoleAppender" />  </root>  </configuration> |