[(63) How to implement Logging in Springboot using Logback | Rolling file | Max History delete log file - YouTube](https://www.youtube.com/watch?v=3KCUG256aCc&t=221s)

LogBack este un framework pentru loguri.

Pasi:

1. Adaugm in pom.xml:

<dependency>  
 <groupId>ch.qos.logback</groupId>  
 <artifactId>logback-core</artifactId>  
 <version>1.4.4</version>  
</dependency>  
  
 <dependency>  
 <groupId>ch.qos.logback</groupId>  
 <artifactId>logback-classic</artifactId>  
 <version>1.4.4</version>  
 </dependency>  
 <dependency>  
 <groupId>org.slf4j</groupId>  
 <artifactId>slf4j-api</artifactId>  
 <version>2.0.3</version>  
 <scope>compile</scope>  
 </dependency>

1. Cream un logback.xml in resources si adaugam:

<configuration>  
 <appender name = "STDOUT" class = "ch.qos.logback.core.ConsoleAppender">  
 <encoder>  
 <pattern>[%d{yyyy-MM-dd HH:mm:ss}] - %msg%n</pattern>  
 </encoder>  
 </appender>  
 <appender name="SIZE\_TIME\_OUTPUT\_FILE" class="ch.qos.logback.core.rolling.RollingFileAppender">  
 <file>C:/Users/user/Desktop/Logs/NumeLog.log</file>  
 <encoder>  
 <pattern>[%d{yyyy-MM-dd HH:mm:ss}] - %msg%n</pattern>  
 </encoder>  
 <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">  
 <fileNamePattern>C:/Users/user/Desktop/Logs/NumeLog.%d{yyyy-MM-dd}.%i.log</fileNamePattern>  
 <timeBasedFileNamingAndTriggeringPolicy class="ch.qos.logback.core.rolling.SizeAndTimeBasedFNATP">  
 <!--Max Size of file to start Archive -->  
 <maxFileSize>10KB</maxFileSize>  
 </timeBasedFileNamingAndTriggeringPolicy>  
  
 <!-- Days till log history to keep -->  
 <maxHistory>3</maxHistory>  
 </rollingPolicy>  
  
 </appender>  
  
 <root level = "DEBUG">  
 <appender-ref ref = "SIZE\_TIME\_OUTPUT\_FILE"/>  
 <appender-ref ref = "STDOUT"/>  
 </root>  
</configuration>

<file>C:/Users/user/Desktop/Logs/NumeLog.log</file>

NumeLog.log deja e numele logului care va fi afisat, ultimul adica, restul loguilor afisate vor fi cu numele NumeLog.%d{yyyy-MM-dd}.%i.log

C:/Users/user/Desktop/Logs/ e locati logurilor

1. Atentie la importuri!!!:

import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;

private final Logger logger = LoggerFactory.*getLogger*(ControllerConfig.class);

unde ControllerConfig.class e clasa unde se afla variabila.

Pentru a face ca sa fie generat cate un nou file log la fiecare rulare, facem asa:

<configuration>  
 <timestamp key="time" datePattern="yyyyMMdd'T'HHmmss"/>  
 <appender name = "STDOUT" class = "ch.qos.logback.core.ConsoleAppender">  
 <encoder>  
 <pattern>[%d{yyyy-MM-dd HH:mm:ss}] - %msg%n</pattern>  
 </encoder>  
 </appender>  
 <appender name="SIZE\_TIME\_OUTPUT\_FILE" class="ch.qos.logback.core.rolling.RollingFileAppender">  
 <file>S:/Studii/Endava/Project/src/main/resources/Logs/Log.${time}.log</file>  
 <append>true</append>  
 <encoder>  
 <pattern>[%d{yyyy-MM-dd HH:mm:ss}] - %msg%n</pattern>  
 </encoder>  
 <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">  
 <fileNamePattern>S:/Studii/Endava/Project/src/main/resources/Logs/Log.%d{yyyy-MM-dd}.%i.log</fileNamePattern>  
 <timeBasedFileNamingAndTriggeringPolicy class="ch.qos.logback.core.rolling.SizeAndTimeBasedFNATP">  
 <!--Max Size of file to start Archive -->  
 <maxFileSize>10KB</maxFileSize>  
 </timeBasedFileNamingAndTriggeringPolicy>  
  
 <!-- Days till log history to keep -->  
 <maxHistory>3</maxHistory>  
 </rollingPolicy>  
  
 </appender>  
  
 <root level = "DEBUG">  
 <appender-ref ref = "SIZE\_TIME\_OUTPUT\_FILE"/>  
 <appender-ref ref = "STDOUT"/>  
 </root>  
</configuration>