***Kaan YAVUZ***

Log4J HW

***CODE:***

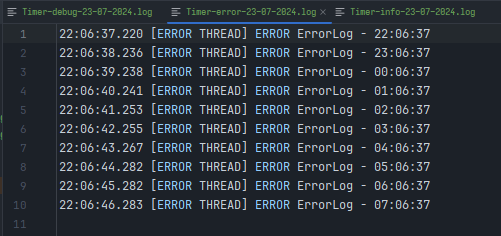
***myTimerLoggings.java***

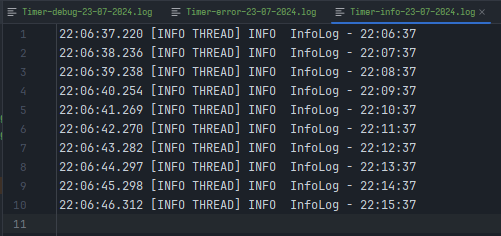
package org.example;  
  
import org.apache.logging.log4j.LogManager;  
import org.apache.logging.log4j.Logger;  
import java.time.LocalDateTime;  
import java.time.format.DateTimeFormatter;  
  
public class myTimerLoggings {  
 private static final Logger debugLog = LogManager.getLogger("DebugLog");  
 private static final Logger errorLog = LogManager.getLogger("ErrorLog");  
 private static final Logger infoLog = LogManager.getLogger("InfoLog");  
 private static final DateTimeFormatter formatter = DateTimeFormatter.ofPattern("HH:mm:ss");  
 private static final int MAX\_COUNT = 10;  
  
 public static void main(String[] args) throws InterruptedException {  
  
 Thread debugThread = new Thread(new DebugLogTask(),"DEBUG THREAD");  
 Thread infoThread = new Thread(new InfoLogTask(), "INFO THREAD");  
 Thread errorThread = new Thread(new ErrorLogTask(), "ERROR THREAD");  
 debugThread.start();  
 infoThread.start();  
 errorThread.start();  
  
 debugThread.join();  
 infoThread.join();  
 errorThread.join();  
 System.out.println("FINISHED...");  
 }  
 static class DebugLogTask implements Runnable {  
 private int count = 0;  
 @Override  
 public void run() {  
 LocalDateTime time = LocalDateTime.now();  
 while (count < MAX\_COUNT) {  
 String formattedTime = time.format(formatter);  
 debugLog.debug(formattedTime);  
 time = time.plusSeconds(1);  
 count++;  
 try {  
 Thread.sleep(1000);  
 } catch (InterruptedException e) {  
 debugLog.error("Debug thread interrupted", e);  
 Thread.currentThread().interrupt();  
 }  
 }  
 }  
 }  
 static class InfoLogTask implements Runnable {  
 private int count = 0;  
 @Override  
 public void run() {  
 LocalDateTime time = LocalDateTime.now();  
 while (count < MAX\_COUNT) {  
 String formattedTime = time.format(formatter);  
 infoLog.info(formattedTime);  
 time = time.plusMinutes(1);  
 count++;  
 try {  
 Thread.sleep(1000);  
 } catch (InterruptedException e) {  
 infoLog.error("Info thread interrupted", e);  
 Thread.currentThread().interrupt();  
 }  
 }  
 }  
 }  
 static class ErrorLogTask implements Runnable {  
 private int count = 0;  
 @Override  
 public void run() {  
 LocalDateTime time = LocalDateTime.now();  
 while (count < MAX\_COUNT) {  
 String formattedTime = time.format(formatter);  
 errorLog.error(formattedTime);  
 time = time.plusHours(1);  
 count++;  
 try {  
 Thread.sleep(1000);  
 } catch (InterruptedException e) {  
 errorLog.error("Error thread interrupted", e);  
 Thread.currentThread().interrupt();  
 }  
 }  
 }  
 }  
}

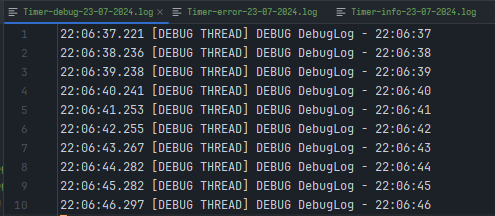
***log4j2.xml***

<?xml version="1.0" encoding="UTF-8"?>  
<Configuration status="WARN" monitorInterval="30">  
  
 <Properties>  
 <Property name="LOG\_PATTERN">%d{HH:mm:ss.SSS} [%t] %-5level %logger{36} - %msg%n</Property>  
 <Property name="APP\_LOG\_ROOT">C:\Users\kaan\_\i2i\log4j</Property>  
 </Properties>  
  
 <Appenders>  
  
 <Console name="Console" target="SYSTEM\_OUT" follow="true">  
 <PatternLayout pattern="${LOG\_PATTERN}"/>  
 </Console>  
  
 <RollingFile name="DebugLog"  
 fileName="${APP\_LOG\_ROOT}/logs/Timer-debug-${date:dd-MM-yyyy}.log"  
 filePattern="${APP\_LOG\_ROOT}/debug-log-%d{yyyy-MM-dd}-%i.log">  
 <PatternLayout pattern="${LOG\_PATTERN}"/>  
 <Policies>  
 <SizeBasedTriggeringPolicy size="1MB" />  
 </Policies>  
 <DefaultRolloverStrategy max="10"/>  
 </RollingFile>  
  
 <RollingFile name="ErrorLog"  
 fileName="${APP\_LOG\_ROOT}/logs/Timer-error-${date:dd-MM-yyyy}.log"  
 filePattern="${APP\_LOG\_ROOT}/error-log-%d{yyyy-MM-dd}-%i.log">  
 <PatternLayout pattern="${LOG\_PATTERN}"/>  
 <Policies>  
 <SizeBasedTriggeringPolicy size="1MB" />  
 </Policies>  
 <DefaultRolloverStrategy max="10"/>  
 </RollingFile>  
  
 <RollingFile name="InfoLog"  
 fileName="${APP\_LOG\_ROOT}/logs/Timer-info-${date:dd-MM-yyyy}.log"  
 filePattern="${APP\_LOG\_ROOT}/info-log-%d{yyyy-MM-dd}-%i.log">  
 <PatternLayout pattern="${LOG\_PATTERN}"/>  
 <Policies>  
 <SizeBasedTriggeringPolicy size="1MB" />  
 </Policies>  
 <DefaultRolloverStrategy max="10"/>  
 </RollingFile>  
  
 </Appenders>  
 <Loggers>  
 <Logger name="DebugLog" level="debug" additivity="false">  
 <AppenderRef ref="DebugLog"/>  
 <AppenderRef ref="Console"/>  
 </Logger>  
  
 <Logger name="ErrorLog" level="error" additivity="false">  
 <AppenderRef ref="ErrorLog"/>  
 <AppenderRef ref="Console"/>  
 </Logger>  
  
 <Logger name="InfoLog" level="info" additivity="false">  
 <AppenderRef ref="InfoLog"/>  
 <AppenderRef ref="Console"/>  
 </Logger>  
  
 <Root level="all">  
 <AppenderRef ref="Console"/>  
 </Root>  
 </Loggers>  
</Configuration>

***Screenshots:***

******





metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu