

# springboot + logback + docker 实现多级 别日志打印

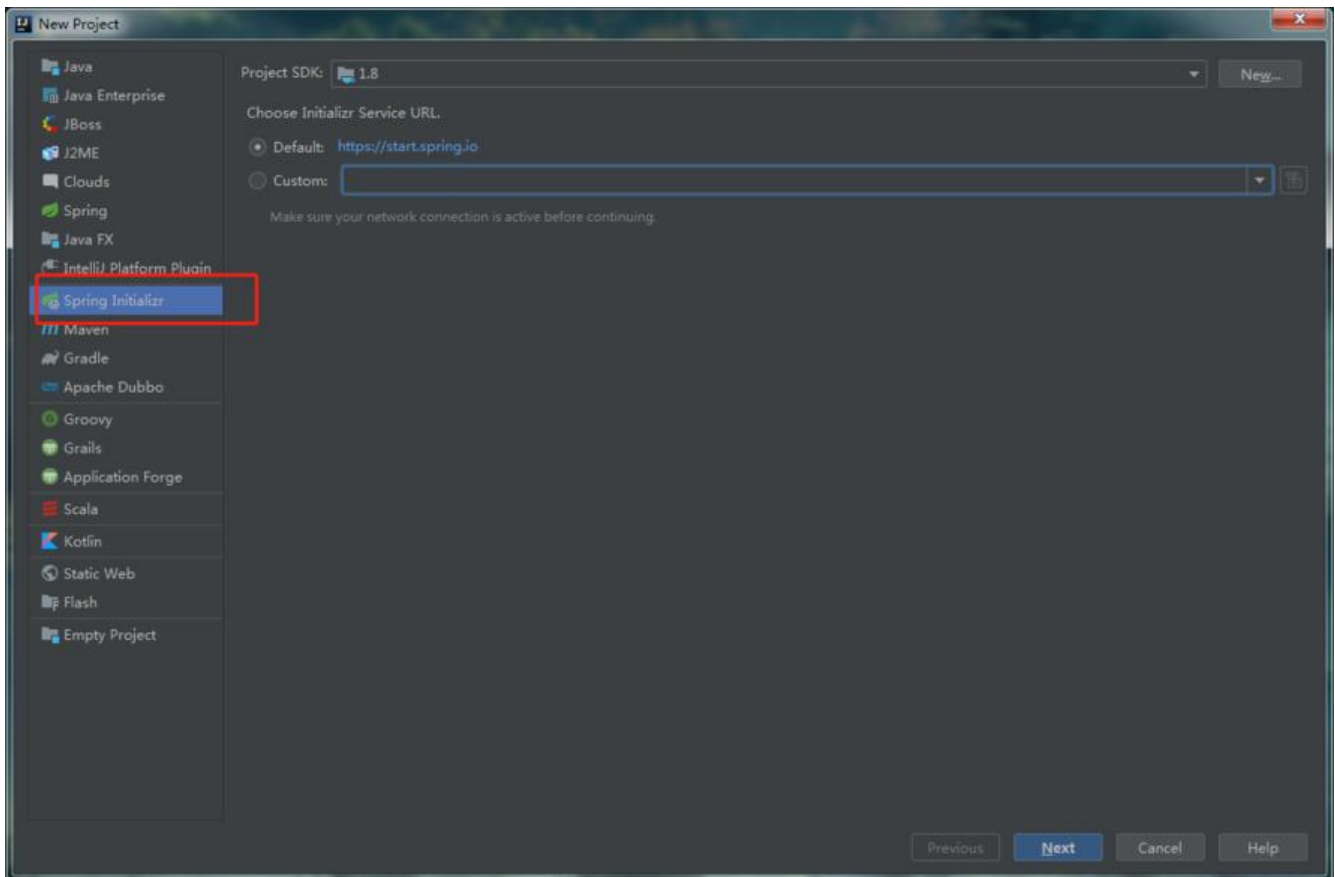
作者: [zzm110110](#)

原文链接: <https://ld246.com/article/1567994730855>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

# 1.创建springbot项目



- 输入自己项目名称，点击下一步下一步，走着去就行了

## 2.在项目中的resources文件夹中创建一个logback-spring.xml文件，输入如下内容

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration debug="false">
```

```
  <!--配置本地日志输出路径-->
```

```
  <property name="log_path" value="./target"/>
```

```
  <if condition='isDefined("catalina.home")'>
```

```
    <then>
```

```
      <!-- 配置远程日志输出路径 -->
```

```
      <property name="log_path" value="${catalina.home}"/>
```

```
    </then>
```

```
  </if>
```

```
  <!--配置项目名称-->
```

```
  <property name="entryName" value="你项目的名称"/>
```

```
  <!-- 控制台输出 -->
```

```
  <appender name="STDOUT" class="ch.qos.logback.core.ConsoleAppender">
```

```
    <encoder charset="UTF-8">
```

```
      <pattern>%date{yyyy-MM-dd HH:mm:ss} | %highlight(%-5level) | %boldYellow(%three
```

```

d) | %boldGreen(%logger) | %msg%n
    </pattern>
  </encoder>
  <filter class="ch.qos.logback.classic.filter.LevelFilter">
    <level>INFO</level>
    <onMatch>ACCEPT</onMatch>
    <onMismatch>DENY</onMismatch>
  </filter>
</appender>

<!-- 按照每天生成error日志文件 -->
<appender name="ERRORFILE" class="ch.qos.logback.core.rolling.RollingFileAppender">
  <Prudent>true</Prudent>
  <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">
    <!-- 日志文件输出的文件名 -->
    <FileNamePattern>${log_path}/logs/${entryName}/error/${entryName}_error-%d{yyyy
MM-dd}.log</FileNamePattern>
    <!-- 日志文件保留天数 -->
    <MaxHistory>5</MaxHistory>
  </rollingPolicy>
  <encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">
    <!-- 格式化输出: %d表示日期, %thread表示线程名, %-5level: 级别从左显示5个字符宽
%msg: 日志消息, %n是换行符 -->
    <pattern>%d{yyyy-MM-dd HH:mm:ss.SSS} [%thread] %-5level %logger{50} - %msg%n
    </pattern>
  </encoder>
  <!-- 只打印错误日志 -->
  <filter class="ch.qos.logback.classic.filter.LevelFilter">
    <level>ERROR</level>
    <onMatch>ACCEPT</onMatch>
    <onMismatch>DENY</onMismatch>
  </filter>
</appender>

<!-- 按照每天生成warn日志文件 -->
<appender name="WARNFILE" class="ch.qos.logback.core.rolling.RollingFileAppender">
  <Prudent>true</Prudent>
  <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">
    <!-- 日志文件输出的文件名 -->
    <FileNamePattern>${log_path}/logs/${entryName}/warn/${entryName}_warn-%d{yyyy
MM-dd}.log</FileNamePattern>
    <!-- 日志文件保留天数 -->
    <MaxHistory>5</MaxHistory>
  </rollingPolicy>
  <encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">
    <!-- 格式化输出: %d表示日期, %thread表示线程名, %-5level: 级别从左显示5个字符宽
%msg: 日志消息, %n是换行符 -->
    <pattern>%d{yyyy-MM-dd HH:mm:ss.SSS} [%thread] %-5level %logger{50} - %msg%n
    </pattern>
  </encoder>
  <!-- 只打印警告日志 -->
  <filter class="ch.qos.logback.classic.filter.LevelFilter">
    <level>WARN</level>
    <onMatch>ACCEPT</onMatch>

```

```

        <onMismatch>DENY</onMismatch>
    </filter>
</appender>

<!-- 按照每天生成INFO日志文件 -->
<appender name="INFOFILE" class="ch.qos.logback.core.rolling.RollingFileAppender">
    <Prudent>true</Prudent>
    <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">
        <!-- 日志文件输出的文件名 -->
        <FileNamePattern>${log_path}/logs/${entryName}/info/${entryName}_info-%d{yyyy-
M-dd}.log</FileNamePattern>
        <!-- 日志文件保留天数 -->
        <MaxHistory>5</MaxHistory>
    </rollingPolicy>
    <encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">
        <!-- 格式化输出: %d表示日期, %thread表示线程名, %-5level: 级别从左显示5个字符宽
%msg: 日志消息, %n是换行符 -->
        <pattern>%d{yyyy-MM-dd HH:mm:ss.SSS} [%thread] %-5level %logger{50} - %msg%n
        </pattern>
    </encoder>
    <!-- 只打印INFO日志 -->
    <filter class="ch.qos.logback.classic.filter.LevelFilter">
        <level>INFO</level>
        <onMatch>ACCEPT</onMatch>
        <onMismatch>DENY</onMismatch>
    </filter>
</appender>

<!-- 按照每天生成DEBUG日志文件 -->
<appender name="DEBUGFILE" class="ch.qos.logback.core.rolling.RollingFileAppender">
    <Prudent>true</Prudent>
    <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">
        <!-- 日志文件输出的文件名 -->
        <FileNamePattern>${log_path}/logs/${entryName}/debug/${entryName}_debug-%d{
yy-MM-dd}.log</FileNamePattern>
        <!-- 日志文件保留天数 -->
        <MaxHistory>5</MaxHistory>
    </rollingPolicy>
    <encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">
        <!-- 格式化输出: %d表示日期, %thread表示线程名, %-5level: 级别从左显示5个字符宽
%msg: 日志消息, %n是换行符 -->
        <pattern>%d{yyyy-MM-dd HH:mm:ss.SSS} [%thread] %-5level %logger{50} - %msg%n
        </pattern>
    </encoder>
    <!-- 只打印DEBUG日志 -->
    <filter class="ch.qos.logback.classic.filter.LevelFilter">
        <level>DEBUG</level>
        <onMatch>ACCEPT</onMatch>
        <onMismatch>DENY</onMismatch>
    </filter>
</appender>

<!-- 日志输出级别 -->
<root level="DEBUG">

```

```
<appender-ref ref="STDOUT"/>
<appender-ref ref="ERRORFILE"/>
<appender-ref ref="WARNFILE"/>
<appender-ref ref="INFOFILE"/>
<appender-ref ref="DEBUGFILE"/>
</root>
```

</configuration>

logback-spring.xml这里我就不解释了，必进注释都怎么明显了；

这里会打印4中级别的日志，分别为：ERROR、WARN、INFO、DEBUG

这些日志是打印在tomcat中的log文件下面的，路径：/usr/local/tomcat/logs/你项目名称/，文件下面会有4个文件夹，分别为：error/warn/info/debug

日志格式：项目名称\_error-2019-09-09.log、项目名称\_info-2019-09-09.log、项目名称\_warn-2019-09-09.log、项目名称\_debug-2019-09-09.log

还有一点，就是控制台打印哪里在做项目调试的时候最好去掉，因为他会把错误日志打印到error文件面，控制台会没有显示；

## tomcat进行在宿主机中完成日志文件的挂载

启动tomcat镜像时，把下面这句话加入启动命令中，即可完成日志的挂载  
-v /usr/local/docker/tomcat/logs:/usr/local/tomcat/logs/项目名称/

**以后利用ELK做日志监控也比较方便了（这是后续，目前在学习中.....）**