链滴

Spring Boot 通用配置属性 - 核心属性

作者：chenlei65368
原文链接：https://ld246.com/article/1572505125845
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)

可以在application.properties文件内部application.yml，文件内部或命令行开关中指定各种属性。附录提供了常见的Spring Boot属性列表以及对使用它们的基础类的引用。

1.核心属性

键	默认值	描述
debug	false	启用调试日志。
info.*		要添加到信息端点的任意性。
logging.config		日志记录配置件的位置。例如，用于logback的classpath：logback.xml。
logging.exception-conversion-word	%wEx	录异常时使用的转换字。
logging.file.clean-history-on-start	false	否在启动时清除存档日志文件。仅默认登录设置支持。
logging.file.max-history	7	存日志文件的最大保留天数。仅默认登录设置支持。
logging.file.max-size	10MB	最日志文件大小。仅默认登录设置支持。
logging.file.name		日志文件名例如，“myapp.log”）。名称可以是确切位置，也可以是相对于当前目录的位置。
logging.file.path		日志文件的置。例如，/ var / log。
logging.file.total-size-cap	0B	保留的日志备份的总大小。仅默认登录设置支持。
logging.group.*		日志组可快速速地同时更改多个记录器。例如，logging.group.db = org.hibernate，org.springframework.jdbc。
logging.level.*		日志级别严重映射。例如，logging.level.org.springframework = DEBUG。
logging.pattern.console	%clr(%d{{LOG_DATEFORMAT_PATTEN:-yyyy-MM-dd HH:mm:ss.SSS}}){faint} %clr({LOG_LEVEL_PATTERN:-%5p}) %clr({PID:- }){magenta} %clr(---){faint} %clr([%15.15t]){faint} %clr(%-40.40logger{39}){cyan} %clr(:){faint} %m%n{LG_EXCEPTION_CONVERSION_WORD:-%wEx}	Appender模式，于输出到控制台。仅默认的Logback设置受支持。
logging.pattern.dateformat	yyyy-MM-dd HH:mm:ss.SSS	录日期格式的附加模式。仅默认的Logback设置受支持。
logging.pattern.file	%d{{LOG_DATEFORMAT_PATTERN:-yyyy-M-dd HH:mm:ss.SSS}} {LOG_LEVEL_PATTERN:-%5p} {PID:- } --- [%t] %-40.40logger{39} : %m%n LOG_EXCEPTION_CONVERSION_WORD:-%wEx}	用于输出到文的附加模式。仅默认的Logback设置受支持。
logging.pattern.level	%5p	日级别的附加模式。仅默认的Logback设置受支持。
logging.pattern.rolling-file-name	${LOG_FILE}.%d{yyyy-MM-d}.%i.gz	过渡日志文件名的模式。仅默认的Logback设置受支持。

logging.register-shutdown-hook false
始化日志系统时，注册一个关闭挂钩。

spring.aop.auto true 添加@En
bleAspectJAutoProxy。

spring.aop.proxy-target-class true
基于标准Java接口的代理（false）相反，是否要创建基于子类的代理（CGLIB）（true）。

spring.application.admin.enabled false
否为应用程序启用管理功能。

spring.application.admin.jmx-name org.springframework.boot
type=Admin,name=SpringApplication 应用程序管理员MBean的
MX名称。

spring.application.name 应用
称。

spring.autoconfigure.exclude
排除的自动配置类。

spring.banner.charset UTF-8
语文件编码。

spring.banner.image.bitdepth 4
于ANSI颜色的位深度。支持的值为4（16色）或8（256色）。

spring.banner.image.height
语图像的高度（以字符为单位）（默认基于图像高度）。

spring.banner.image.invert false
否应针对黑暗的终端主题反转图像。

spring.banner.image.location classpath:banner.gif
语图片文件的位置（也可以使用jpg或png）。

spring.banner.image.margin 2
手图像边距（以字符为单位）。

spring.banner.image.pixelmode TEXT
染图像时使用的像素模式。

spring.banner.image.width 76
幅图像的宽度（以字符为单位）。

spring.banner.location classpath:banner.txt
语文字资源位置。

spring.beaninfo.ignore true
否跳过对BeanInfo类的搜索。

spring.config.additional-location
了默认值以外，还使用了配置文件位置。

spring.config.location 替换默
设置的配置文件位置。

spring.config.name application
置文件名。

spring.info.build.encoding UTF-8
件编码。

spring.info.build.location classpath:META-INF/build-info.prop

rties 生成的build-info.properties文件的位置。

spring.info.git.encoding UTF-8
件编码。

spring.info.git.location classpa
h:git.properties

spring.jmx.default-domain JM
域名。

spring.jmx.enabled false 将管
bean公开到JMX域。

spring.jmx.server mbeanServer
BeanServer Bean名称。

spring.jmx.unique-names false
否应确保唯一的运行时对象名称。

spring.main.allow-bean-definition-overriding false
否允许通过使用与现有定义同名的定义来覆盖Bean定义。

spring.main.banner-mode console
用程序运行时用于显示横幅的模式。

spring.main.lazy-initialization false
否应延迟执行初始化。

spring.main.register-shutdown-hook true
用程序是否应该注册一个关闭挂钩。

spring.main.sources 包含在App
icationContext中的源（类名，包名或XML资源位置）。

spring.main.web-application-type
记以显式请求特定类型的Web应用程序。如果未设置，则基于类路径自动检测。

spring.mandatory-file-encoding
用程序必须使用的预期字符编码。

spring.messages.always-use-message-format false
否始终应用MessageFormat规则，甚至分析不带参数的消息。

spring.messages.basename messages
名的逗号分隔列表（本质上是完全合格的类路径位置），每个都遵循ResourceBundle约定，并轻松
持基于斜杠的位置。如果它不包含包限定符（例如“org.mypackage”），它将从类路径根目录中
析。

spring.messages.cache-duration
载的资源束文件缓存持续时间。如果未设置，则捆绑包将永久缓存。如果未指定持续时间后缀，则将
用秒。

spring.messages.encoding UTF-8
息编码。

spring.messages.fallback-to-system-locale true
果未找到特定语言环境的文件，是否退回到系统语言环境。如果关闭此选项，则唯一的后备选项将是
认文件（例如，基本名称“messages”的“messages.properties”）。

spring.messages.use-code-as-default-message false
否使用消息代码作为默认消息，而不是引发“NoSuchMessageException”。仅在开发期间推荐。

spring.output.ansi.enabled detect
置ANSI输出。

spring.pid.fail-on-write-error
果使用ApplicationPidFileWriter失败，但无法写入PID文件。

spring.pid.file 要写入的PID文
的位置（如果使用了ApplicationPidFileWriter）。

spring.profiles 用逗号分隔的
要文件表达式列表，至少要匹配一个概要文件表达式才能包含该文档。

spring.profiles.active 以逗号
隔的活动配置文件列表。可以被命令行开关覆盖。

spring.profiles.include 无条件
活指定的逗号分隔的配置文件列表（如果使用YAML，则激活配置文件列表）。

spring.quartz.auto-startup true
始化后是否自动启动调度程序。

spring.quartz.jdbc.comment-prefix #, --
QL初始化脚本中单行注释的前缀。

spring.quartz.jdbc.initialize-schema embedded
据库模式初始化模式。

spring.quartz.jdbc.schema clas
path:org/quartz/impl/jdbcjobstore/tables_@@platform@@.sql

spring.quartz.job-store-type memory
uartz作业存储类型。

spring.quartz.overwrite-existing-jobs false
置的作业是否应覆盖现有的作业定义。

spring.quartz.properties.* Qua
tz Scheduler的其他属性。

spring.quartz.scheduler-name quartzScheduler
度程序的名称。

spring.quartz.startup-delay 0s
始化完成后启动调度程序的延迟时间。如果在整个应用程序启动之前不运行任何作业，则设置此属性
有意义。

spring.quartz.wait-for-jobs-to-complete-on-shutdown false
否等待正在关闭的正在运行的作业。

spring.reactor.debug-agent.enabled true
在反应器工具时是否应启用反应器调试代理。

spring.task.execution.pool.allow-core-thread-timeout true
否允许核心线程超时。这样可以动态增加和缩小池。

spring.task.execution.pool.core-size 8
心线程数。

spring.task.execution.pool.keep-alive 60s
程在终止之前可能保持空闲的时间限制。

spring.task.execution.pool.max-size
许的最大线程数。如果任务填满队列，则池可以扩展到该大小以容纳负载。忽略队列是否无界。

spring.task.execution.pool.queue-capacity
列容量。无限制的容量不会增加池，因此会忽略“最大大小”属性。

spring.task.execution.shutdown.await-termination false

行程序是否应该等待计划的任务在关闭时完成。

spring.task.execution.shutdown.await-termination-period
行者应该等待剩余任务完成的最长时间。

spring.task.execution.thread-name-prefix task-
于新创建线程名称的前缀。

spring.task.scheduling.pool.size 1
许的最大线程数。

spring.task.scheduling.shutdown.await-termination false
行程序是否应该等待计划的任务在关闭时完成。

spring.task.scheduling.shutdown.await-termination-period
行者应该等待剩余任务完成的最长时间。

spring.task.scheduling.thread-name-prefix scheduling-
于新创建线程名称的前缀。

trace false 启用跟踪日志。