参数配置详见config/config.properties

经测试，如下环境下正确运行。如有其它环境不能运行，请加群反馈我们~

jdk: 1.6、1.7

tomcat: 6&7

mysql 5.x

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 取值 | 默认值 | 参数说明 |
| 定时器配置 | | | |
| timer.enable | true/false | true | 是否开启定时器。定时器开启会定时发送邮件 |
| 数据库配置（需配置：driverClassName,url,username,password） | | | |
| driverClassName |  | com.mysql.jdbc.Driver  (mysql驱动) | 数据库驱动类的名字 |
| url |  | jdbc:mysql://localhost:3306/cynthia?useUnicode=true&characterEncoding=utf8 | 连接数据库的url，不同数据库不一样。例如：  mysql: jdbc:mysql://10.20.153.104:3306/druid2  oracle: jdbc:oracle:thin:@10.20.149.85:1521:ocnauto |
| username |  | user | 连接数据库的用户名 |
| password |  | passwd | 连接数据库的密码 |
| initialSize |  | 100 | 初始化时建立物理连接的个数。初始化发生在显示调用init方法，或者第一次getConnection时 |
| maxActive |  | 5000 | 最大连接池数量 |
| maxWait |  | 60000 | 获取连接时最大等待时间，单位毫秒。配置了maxWait之后，  缺省启用公平锁，并发效率会有所下降，  如果需要可以通过配置useUnfairLock属性为true使用非公平锁。 |
| timeBetweenEvictionRunsMillis |  | 60000 | 有两个含义：  1) Destroy线程会检测连接的间隔时间  2) testWhileIdle的判断依据，详细看testWhileIdle属性的说明 |
| minEvictableIdleTimeMillis |  | 300000 |  |
| validationQuery |  | SELECT 1 | 用来检测连接是否有效的sql，要求是一个查询语句。  如果validationQuery为null，testOnBorrow、testOnReturn、  testWhileIdle都不会其作用。 |
| testWhileIdle |  | true | 可以保证连接池内部定时检测连接的可用性，不可用的连接会被抛弃或者重建，最大情况的保证从连接池中得到的Connection对象是可用的 |
| testOnBorrow |  | false | 申请连接时执行validationQuery检测连接是否有效，  做了这个配置会降低性能。 |
| testOnReturn |  | false | 归还连接时执行validationQuery检测连接是否有效，  做了这个配置会降低性能 |
| poolPreparedStatements |  | false | 是否缓存preparedStatement，也就是PSCache。  PSCache对支持游标的数据库性能提升巨大，比如说oracle。  在mysql5.5以下的版本中没有PSCache功能，建议关闭掉。  5.5及以上版本有PSCache，建议开启。 |
| 邮件服务器配置（需配置：mail.protocal,mail.user,mail.pass,mail.smtp.host） | | | |
| mail.enable | true/false | true | 是否开启发送邮件功能，默认为true,打开。 |
| mail.protocal |  | smtp | 邮件服务器使用协议 |
| mail.smtp.auth |  | true | 是否验证邮箱服务器正确性，必须为true |
| mail.user |  |  | 邮件服务器用户名 |
| mail.pass |  |  | 邮件服务器密码 |
| mail.smtp.host |  | smtp.163.com | 邮件服务器主机域名 |
| 部署配置（需配置:deploy.host,deploy.port） | | | |
| deploy.host |  | localhost | 项目部署服务器IP |
| deploy.port |  | 8080 | 项目部署服务器端口 |