https://blog.csdn.net/majian\_1987/article/details/18598589

**解决Caused by: java.sql.SQLRecoverableException: IO Error: Connection reset相关问题**

2014年01月21日 10:38:46 [大饼卷馒头蘸米饭](https://me.csdn.net/majian_1987) 阅读数：86941

 版权声明：本文为博主原创文章，未经博主允许不得转载。 https://blog.csdn.net/majian\_1987/article/details/18598589

[**java.sql.SQLException: Io 异常: Connection reset**](http://www.cnblogs.com/younes/archive/2012/06/01/2529483.html)

当数据库连接池中的连接被创建而长时间不使用的情况下，该连接会自动回收并失效，但客户端并不知道，在进行数据库操作时仍然使用的是无效的数据库连接，这样，就导致客户端程序报“ java.sql.SQLException: Io 异常: Connection reset” 或“java.sql.SQLException 关闭的连接”异常。

在配置数据源后面加上

<property name="validationQuery" value="select \* from dual"/>

配置后，客户端在使用一个无效的连接时会先对该连接进行测试，如果发现该连接已经无效，则重新从连接池获取有效数据库连接来使用。

在tomcat的context.xml里面设置数据源时候可参考：

 <Resource auth="Container"  
  driverClassName="oracle.jdbc.OracleDriver"  
  type="javax.sql.DataSource"   
  url="jdbc:oracle:thin:@11.11.11.45:1521:orcl"  
  name="jdbc/login"  
  username="login"  
  password="login"  
  maxActive="15"  
  maxIdle="10"  
  maxWait="-1"  
  minIdle="2"  
  removeAbandonedTimeout="5"  
  testOnBorrow="true"  
  testWhileIdle="true"  
  testOnReturn="true"  
  removeAbandoned="true"  
  logAbandoned="true"  
  validationQuery="select 1 from dual"  
 />   
   
 <Resource auth="Container"  
  driverClassName="oracle.jdbc.OracleDriver"  
  type="javax.sql.DataSource"   
  url="jdbc:oracle:thin:@11.11.11.44:1521:orcl"  
  name="jdbc/intraweb"  
  username="intraweb"  
  password="intraweb"  
  maxActive="15"  
  maxIdle="10"  
  maxWait="-1"  
  minIdle="2"  
  removeAbandonedTimeout="5"  
  testOnBorrow="true"  
  testWhileIdle="true"  
  testOnReturn="true"  
  removeAbandoned="true"  
  logAbandoned="true"  
  validationQuery="select 1 from dual"  
 />

-------------------------------------------------------------

以上是从网上搜到一篇文章。网上跟这个问题相关的解决思路是Connection Reset的原因有可能有以下几种原因：

1.配置的数据连接池的连接数不够用；

2.数据库的连接池中的连接，长时间不用，数据库主动断开连接，而客户端不知道，在用的时候仍然拿到的是无效的连接；

分别对应以上的两种猜想的原因，进行处理：

1.配置连接池的最大、最小、空闲连接数等；

2.配置对连接池里的连接进行有效性检查，如，配置有效性连接检查sql语句，配置是否进行有效性检查等；

以上只是思路，具体配置内容要根据不同的数据源进行不同的配置，解决思路可以参照上面。至于是否还有别的方面原因，暂时未知。

我的项目中用到的是c3p0数据源，当时就按c3p0的配置方式添加了有效性检查的相关配置，关于c3p0的详细配置可以参照：

http://blog.csdn.net/majian\_1987/article/details/18598857