

2006年通过JDBC连接oracle数据库的几个技巧 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/143/2021\\_2022\\_2006\\_E5\\_B9\\_B4\\_E9\\_80\\_9A\\_c102\\_143179.htm](https://www.100test.com/kao_ti2020/143/2021_2022_2006_E5_B9_B4_E9_80_9A_c102_143179.htm) Java数据库连接（JDBC）API

是一系列能够让Java编程人员访问数据库的接口，各个开发商的接口并不完全相同。以下是一些技巧，这些技巧能够更好地发挥系统的性能和实现更多的功能。

- 1、在客户端软件开发中使用Thin驱动程序 在开发Java软件方面，Oracle的数据库提供了四种类型的驱动程序，二种用于应用软件、applets、servlets等客户端软件，另外二种用于数据库中的Java存储过程等服务器端软件。在客户机端软件的开发中，我们可以选择OCI驱动程序或Thin驱动程序。OCI驱动程序利用Java本地化接口（JNI），通过Oracle客户端软件与数据库进行通讯。Thin驱动程序是纯Java驱动程序，它直接与数据库进行通讯。为了获得最高的性能，Oracle建议在客户端软件的开发中使用OCI驱动程序，这似乎是正确的。但我建议使用Thin驱动程序，因为通过多次测试发现，在通常情况下，Thin驱动程序的性能都超过了OCI驱动程序。
- 2、关闭自动提交功能，提高系统性能 在第一次建立与数据库的连接时，在缺省情况下，连接是在自动提交模式下的。为了获得更好的性能，可以通过调用带布尔值false参数的Connection类的setAutoCommit()方法关闭自动提交功能，如下所示：  
`conn.setAutoCommit(false)`. 值得注意的是，一旦关闭了自动提交功能，我们就需要通过调用Connection类的commit()和rollback()方法来人工的方式对事务进行管理。
- 3、在动态SQL或有时间限制的命令中使用Statement对象 在执行SQL命

令时，我们有二种选择：可以使用PreparedStatement对象，也可以使用Statement对象。无论多少次地使用同一个SQL命令，PreparedStatement都只对它解析和编译一次。当使用Statement对象时，每次执行一个SQL命令时，都会对它进行解析和编译。这可能会使你认为，使用PreparedStatement对象比使用Statement对象的速度更快。然而，我进行的测试表明，在客户端软件中，情况并非如此。因此，在有时间限制的SQL操作中，除非成批地处理SQL命令，我们应当考虑使用Statement对象。此外，使用Statement对象也使得编写动态SQL命令更加简单，因为我们可以将字符串连接在一起，建立一个有效的SQL命令。因此，我认为，Statement对象可以使动态SQL命令的创建和执行变得更加简单。

#### 4、利用helper函数对动态SQL命令进行格式化

在创建使用Statement对象执行的动态SQL命令时，我们需要处理一些格式化方面的问题。例如，如果我们想创建一个将名字OReilly插入表中的SQL命令，则必须使用二个相连的“ ”号替换OReilly中的“ ”号。完成这些工作的最好的方法是创建一个完成替换操作的helper方法，然后在连接字符串中服用公式表达一个SQL命令时，使用创建的helper方法。与此类似的是，我们可以让helper方法接受一个Date型的值，然后让它输出基于Oracle的to\_date()函数的字符串表达式。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)