# 数据源操作原理

## JDBC操作原理

使用JDBC操作数据库的步骤如下：

1. 加载数据库驱动程序；
2. 通过DriverManager类的getConnection( )方法取得数据库连接对象；
3. 编写SQL语句，通过Connection（数据库连接对象）实例化PreparedStatment对象，

进行数据库操作

1. 数据库属于资源操作，操作完成要关闭数据库以释放资源。

由于每一个用户进行数据库操作时都需要经过3个相同的步骤，即1、2、4步，但是每 个用户对于数据库的操作却是不同的，有的是更新数据库，有的是查询数据库，所以在 进行数据库操作时，如果可以将重复的1、2、4步骤去掉，只保留3步骤的话，那么性 能肯定会有所提高，而这就是数据源产生的原因。

## 数据源操作的核心原理

数据源操作的核心原理是：在一个数据库连接池（Connection Pool）中保存多个数据库 的连接对象，以后再进行数据库操作时，直接从连接池中取出一个数据库连接对象， 当数据库操作完成后，再将此连接对象放回到数据库连接池中，等待其他用户继续使用。

但也会存在以下问题：

1. 最小连接数：如果一个程序正在使用时没有一个用户连接，则数据库最小应该维 持的数据库连接数。
2. 最大连接数：在一个程序中一个数据库最多可以打开的数据连接数。
3. 最大等待时间：当一个数据库连接池中已经没有更多的数据库连接对象提供给用户 使用时，其他用户等待的最大时间，如果在等待时间内连接放回，

则可以继续使用；如果超过了最大等待时间，则用户无法取得数据 库连接对象。

在Web开发中可以通过Tomcat实现数据库连接池的功能。

## 其他数据库连接池

如果不使用Tomcat实现数据库连接池，也可以从网上搜索各数据库连接池的组件进行 程序功能的实现。现在流行的数据库连接池技术有：Druid、C3P0、DBCP等。

# 使用Tomcat数据库连接池

1）在Web容器中，数据库的连接池是通过数据源（javax.sql.DataSource）访问的，即 通过javax.sql.Source类在数据库连接池中取得一个Connection对象。

2）但是要想得到DataSource对象还需使用JNDI进行查找，JNDI（Java Naming and Directory Interface Java命名及目录接口）是JavaEE提供的一个服务，其服务的主要功能就是通过一个名称的“key”查找到对应的一个对象“value”。

Connection Pool

客户

DataSource

......

Database

DataSource

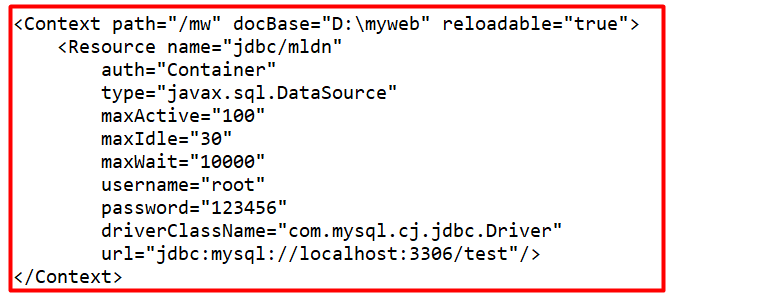
Connection

JNDI Tree

Connection

## 配置数据库连接池

在配置Tomcat连接池之前，需要先将mysql驱动包复制到Tomcat安装目录下的lib目 录中，然后Tomcat安装目录中的conf/server.xml文件中配置数据库连接池，如下：



在配置虚拟目录的<Context>节点中增加了一个<Resource>节点，用来表示配置的连接池 选项，其中：

name属性指定的此数据源的名称，一般采用jdbc/XXX的命名形式；

auth属性表示的是连接数据库的方法，可以有以下两种选择：

·Container：容器将代表应用程序登录到资源管理器，一般使用此方法；

·Application：应用程序必须程序化地登录到资源管理器。

type属性表示数据源的类型；

maxActive属性表示可以打开的最大连接数；

maxIdle属性表示维持的最小连接数；

maxWait属性表示用户等待的最大时间；

username属性表示数据库用户名；

password属性表示数据库密码；

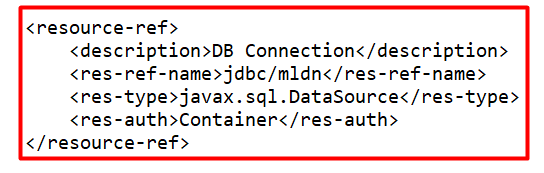
driverClassName属性表示数据库驱动程序；

url属性表示数据库的地址。

配置好数据源后，还需要在项目中的web.xml文件中进行引用。

## 引用数据源

在conf/server.xml文件中配置好数据池之后，还需要在Web项目中配置web.xml文件， 在此文件中指明要使用的数据源名称。



## 使用连接池

数据源的操作时使用JNDI方式进行查找的，所以如果使用数据源取得数据库连接，则 必须按照如下步骤进行：

1. 初始化名称查找上下文：Context ctx = new InitialContext( ) ;
2. 通过名称查找DataSource对象：DataSource ds = (DataSource) ctx.lookup( JNDI名称 ) ;
3. 通过DataSource取得一个数据库连接：Connection conn = ds.getConnection( ) ;

**示例一：datasource.jsp**



虽然在conf/server.xml文件中配置的DataSource名称是“jdbc/mldn”但是真正使 用时却在前面加上了一个“java:comp/env”的前缀，这实际上是JavaEE规定的一 个环境命名上下文（Environment Naming Context，简称ENC），主要是为了解决JNDI 查找时的冲突问题；

如果成功取得数据库连接对象，则会在页面进行打印；如果没有取得，则输出null。

**示例二：改良DAO设计模式中的数据库连接类DatabaseConnection.java**



## 小结

在Java开发中，都会使用数据库连接池进行数据库的连接操作，这样可以提升操作的 性能。通过数据库连接池可以提升数据库的操作性能，可以避免类加载、数据库连接、 数据库关闭等重复操作。

Tomcat数据源在使用时需要使用JNDI进行查找，而且查找指定“java:comp/env/”的环 境属，但实际上Tomcat数据源在Web开发中的应用很少见，实际开发中大多使用的是 Druid、C3P0等数据源。