如何使用开源项目DBCP

1. 当对属性名的命名规范不同时，如有的用**“\_”**连接，有的用**“-”**连接等，这时候就**需要使用别名**。属性可以通过**别名**来查询，**sql语句**中通过**as语句**起别名。
2. 之前，通过手写了一个数据源即数据库连接池（实现了接口DataSource），由于写一个数据源的工程量巨大，在实际生活中，不会去手写，利用一些开源的项目加载进去些配置文件，即可创建出一个数据源。**在实际项目开发中，一般都是利用这种方法。**
3. 使用开源的DBCP：**需要三个jar包**，分别是：



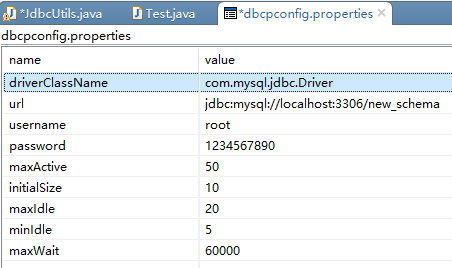
1. **commons-dbcp.jar包**：负责连接数据库。但是还需要另外两个jar包的支持。
2. **commons-collections.jar包**：主要是提供了连接Collection所用的集合。
3. **commons-pool.jar包**：存放数据库连接等。



1. **dbcpconfig.properties文件**：



MyEclipse中的**dbcpconfig.properties文件**：



1. **使用开源项目DBCP（数据库连接池）的步骤**：
2. 首先将三个jar包添加到路径，再创建一个配置文件dbcpconfig.properties，这样可以方便修改具体的数据库设置；
3. 创建一个Properties对象，然后将配置文件dbcpconfig.properties加载到properties对象中，用load方法。（当然不创建配置文件也可以，通过properties的set方法设置，但这样不利于维护和修改。如：properties.setProperty("driverClassName", "com.mysql.jdbc.Driver");）

**方法**：借助输入流InputStream。

1. 通过基本数据源工厂创建一个数据源DataSource dataSource = **BasicDataSourceFactory.createDataSource**(properties);，这个数据源本质就是一个存放很多连接的连接池。
2. 最后通过调用该数据源的getConnection方法即可返回一个连接对象。
3. 示例：

**public** **final** **class** JdbcUtils {

**private** **static** DataSource *dataSource* = **null**;

**static** {

**try** {

Class.*forName*("com.mysql.jdbc.Driver");

Properties properties = **new** Properties();

InputStream in = **new** FileInputStream(**new** File("dbcpconfig.properties"));

//InputStream in = JdbcUtils.class.getClassLoader().getResourceAsStream("dbcpconfig.properties");

**properties.load(in);**

***dataSource* = BasicDataSourceFactory.*createDataSource*(properties);**

} **catch** (Exception e) {

e.printStackTrace();

} }

**public** **static** Connection getConnection() **throws** SQLException {

Connection connection = *dataSource*.getConnection();

**return** connection;

}

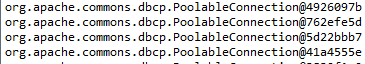
**public static DataSource getDataSource()** {

**return** *dataSource*;

}

还有其他的close（）方法。省略了。

}

注意：返回的数据库连接对象都是经过包装后的连接对象，如返回的是.