# 连接池

<http://blog.csdn.net/wenwen091100304/article/details/48035003>

<https://www.cnblogs.com/wym789/p/6374440.html>

数据库连接池（Connection pooling）是程序启动时建立足够的数据库连接，并将这些连接组成一个连接池，由**程序动态地对池中的连接进行申请，使用，释放**。**复用连接。通过建立一个数据库连接池以及一套连接使用管理策略，使得一个数据库连接可以得到高效、安全的复用，避免了数据库连接频繁建立、关闭的开销。**

创建数据库连接是一个很耗时的操作，也容易对数据库造成安全隐患。所以，在程序初始化的时候，集中创建多个数据库连接，并把他们集中管理，供程序使用，可以保证较快的数据库读写速度，还更加安全可靠。

对于一个简单的数据库应用，由于对于数据库的访问不是很频繁。这时可以简单地在需要访问数据库时，就新创建一个连接，用完后就关闭它，这样做也不会带来什么明显的性能上的开销。但是对于一个复杂的数据库应用，情况就完全不同了。**频繁的建立、关闭连接，会极大的减低系统的性能**，因为对于连接的使用成了系统性能的瓶颈。

## 优势

1. **资源重用**

由于数据库连接得到重用，避免了频繁创建、释放连接引起的大量性能开销。在减少系统消耗的基础上，另一方面也增进了系统运行环境的平稳性（减少内存碎片以及数据库临时进程/线程的数量）。

1. **更快的系统响应速度**

数据库连接池在初始化过程中，往往已经创建了若干数据库连接置于池中备用。此时连接的初始化工作均已完成。对于业务请求处理而言，直接利用现有可用连接，避免了数据库连接初始化和释放过程的时间开销，从而缩减了系统整体响应时间。

1. **新的资源分配手段**

对于多应用共享同一数据库的系统而言，可在应用层通过数据库连接的配置，实现数据库连接池技术，几年前也许还是个新鲜话题，对于目前的业务系统而言，如果设计中还没有考虑到连接池的应用，那么…….快在设计文档中加上这部分的内容吧。某一应用最大可用数据库连接数的限制，避免某一应用独占所有数据库资源。

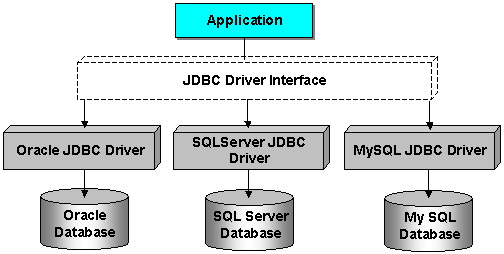
1. **统一的连接管理，避免数据库连接泄漏**

在较为完备的数据库连接池实现中，可根据预先的连接占用超时设定，强制收回被占用连接。从而避免了常规数据库连接操作中可能出现的资源泄漏。

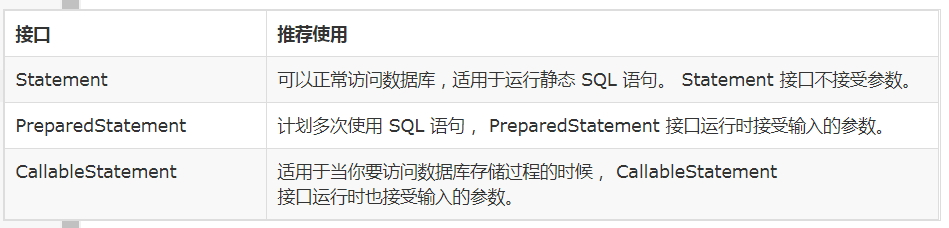
## JDBC

JDBC是一个规范，遵循JDBC接口规范，各个数据库厂家各自实现自己的驱动程序(Driver)。

JDBC提供了完备的数据库操作方法接口。但考虑到规范的适用性，JDBC只提供了最直接的数据库操作规范。



在完成数据操作后，还一定要关闭所有涉及到的数据库资源。



### Statement

使用Connection类的createStatement方法创建一个Statement对象。

创建之后可使用Statement的execute，executeUpdate（insert、update和delete…）和executeQuery（会返回一个结果集ResultSet）方法执行**静态sql语句**。

查询结束之后调用Statement的close方法关闭Statement。

### PreparedStatement

PreparedStatement 接口扩展了 Statement 接口，它让你用一个常用的 Statement 对象增加几个高级功能。

用？占据参数的位置，执行SQL语句时需要传递实际参数以代替？。

使用结束时也应该使用close方法关闭连接。

### CallableStatement

正如一个 Connection 对象可以创建 Statement 对象和 PreparedStatement 对象，它也可以创建被用来执行调用数据库存储过程的 CallableStatement 对象。

## JDBC连接池

在标准JDBC对应用的接口中，并没有提供资源的管理方法。所以，JDBC为第三方应用服务器（Application Server）提供了一个由数据库厂家实现的管理标准接口：连接缓冲(connection pooling)。引入了连接池( Connection Pool )的概念 ，也就是以缓冲池的机制管理数据库的资源。

JDBC最常用的资源有三类:

**Connection**: 数据库连接。

**Statement**: 会话声明。

**ResultSet**: 结果集游标。

数据库连接池的实现及原理

对Connection的管理，就是对数据库资源的管理。在关闭Connection前，需要关闭所有相关的Statement和ResultSet。因此，连接池(Connection Pool)所起到的作用，不仅仅简单地管理Connection，还涉及到 Statement和ResultSet。

## 数据库连接池的任务

1. 连接池的第第一个任务是限制每个应用或系统可以拥有的最大资源。也就是**确定连接池的大小**(PoolSize)。
2. 在连接池的大小(PoolSize)范围内，**最大限度地使用资源，缩短数据库访问的使用周期**。
3. 对资源的申请、释放、回收、共享和同步，这些管理是复杂精密的。所以，数据库连接池的另一个功能就是，封装这些操作，**为应用提供简单的，甚至是不改变应用风格的调用接口。**

## C3P0

C3p0连接池特性：

编码简单易用、复用数据库连接、管理数据库连接。

