



JDBC入门





课程介绍

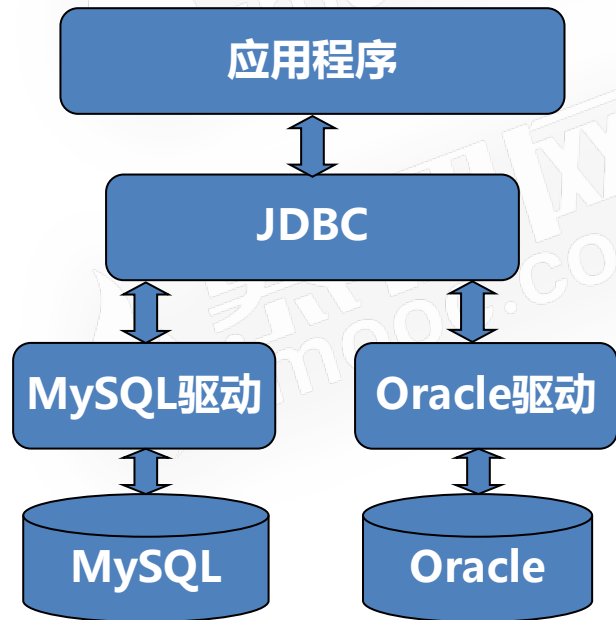
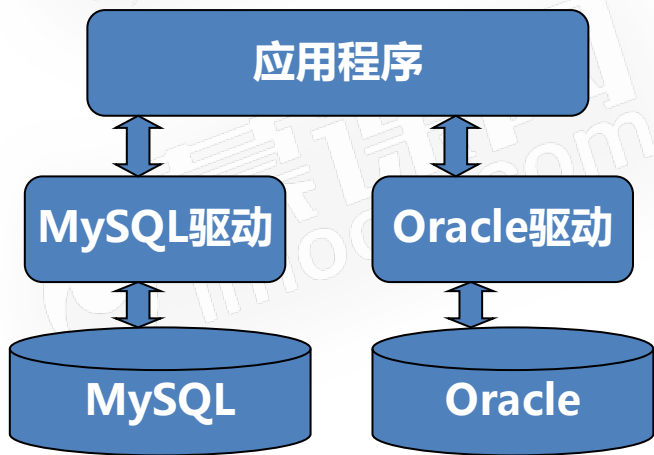


课程安排

- JDBC的概述
- JDBC的入门
- JDBC的API详解
- JDBC的资源释放
- JDBC的CRUD操作
- JDBC工具类的抽取
- JDBC中的PreparedStatement

JDBC的概述

JDBC的全称为：Java DataBase Connectivity（java数据库连接）。



JDBC的入门

- **搭建开发环境**
- **编写程序，在程序中加载数据库驱动**
- **建立连接**
- **创建用于向数据库发送SQL的Statement对象**
- **从代表结果集的ResultSet中取出数据**
- **断开与数据库的连接，并释放相关资源**

JDBC的API

- **DriverManager**
- **Connection**
- **Statement**
- **ResultSet**

JDBC的资源释放

- **Jdbc程序运行完后，切记要释放程序在运行过程中，创建的那些与数据库进行交互的对象，这些对象通常是ResultSet, Statement和Connection对象。**

JDBC的资源释放

- 特别是Connection对象，它是非常稀有资源，用完后必须马上释放，如果Connection不能及时、正确的关闭，极易导致系统宕机。Connection的使用原则是尽量晚创建，尽量早的释放。
- 为确保资源释放代码能运行，资源释放代码也一定要放在finally语句中。

JDBC的CRUD操作

- 向数据库中保存记录
- 修改数据库中的记录
- 删除数据库中的记录
- 查询数据库中的记录

JDBC的工具类的抽取

- 为了简化JDBC的开发，可以将一些重复的代码进行提取。

JDBC的SQL注入漏洞

SQL注入漏洞的演示

SQL注入漏洞的解决

- **PreparedStatement是Statement的子接口，它的实例对象可以通过调用Connection.prepareStatement(sql)方法获得，相对于Statement对象而言：**
 - **PreperedStatement可以避免SQL注入的问题。**
 - **Statement会使数据库频繁编译SQL，可能造成数据库缓冲区溢出。**
PreparedStatement 可对SQL进行预编译，从而提高数据库的执行效率。
 - **并且PreperedStatement对于sql中的参数，允许使用占位符的形式进行替换，简化sql语句的编写。**

PreparedStatement的使用

- 保存数据
- 修改数据
- 删除数据
- 查询数据

数据库连接池

连接池

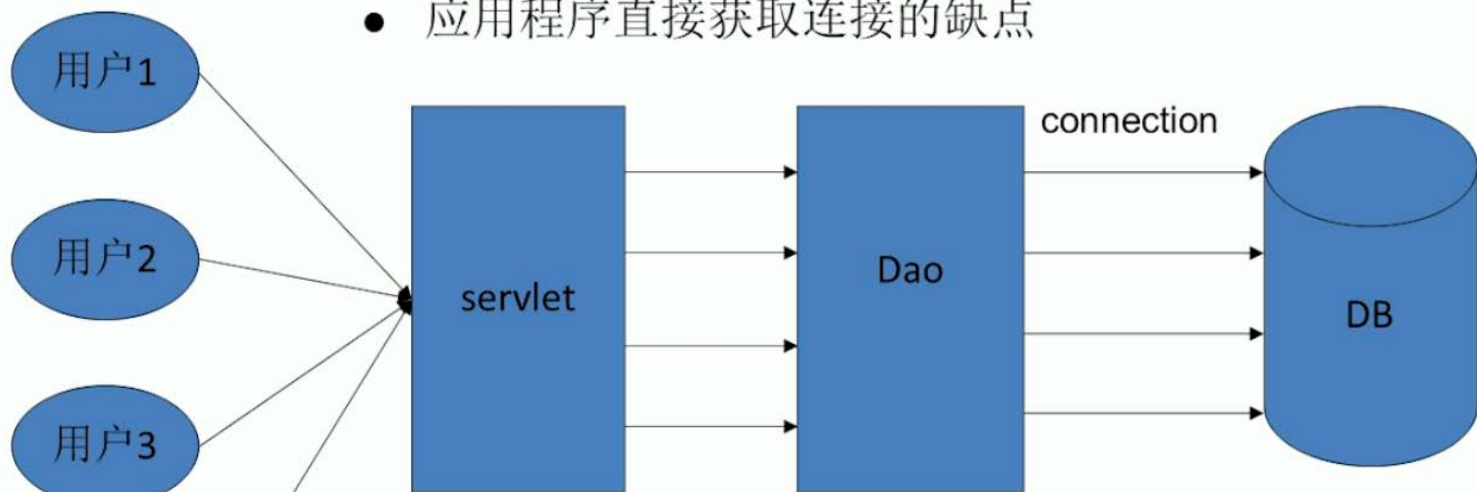


编辑

连接池是创建和管理一个连接的缓冲池的技术，这些连接准备好被任何需要它们的[线程](#)使用。

数据库连接池

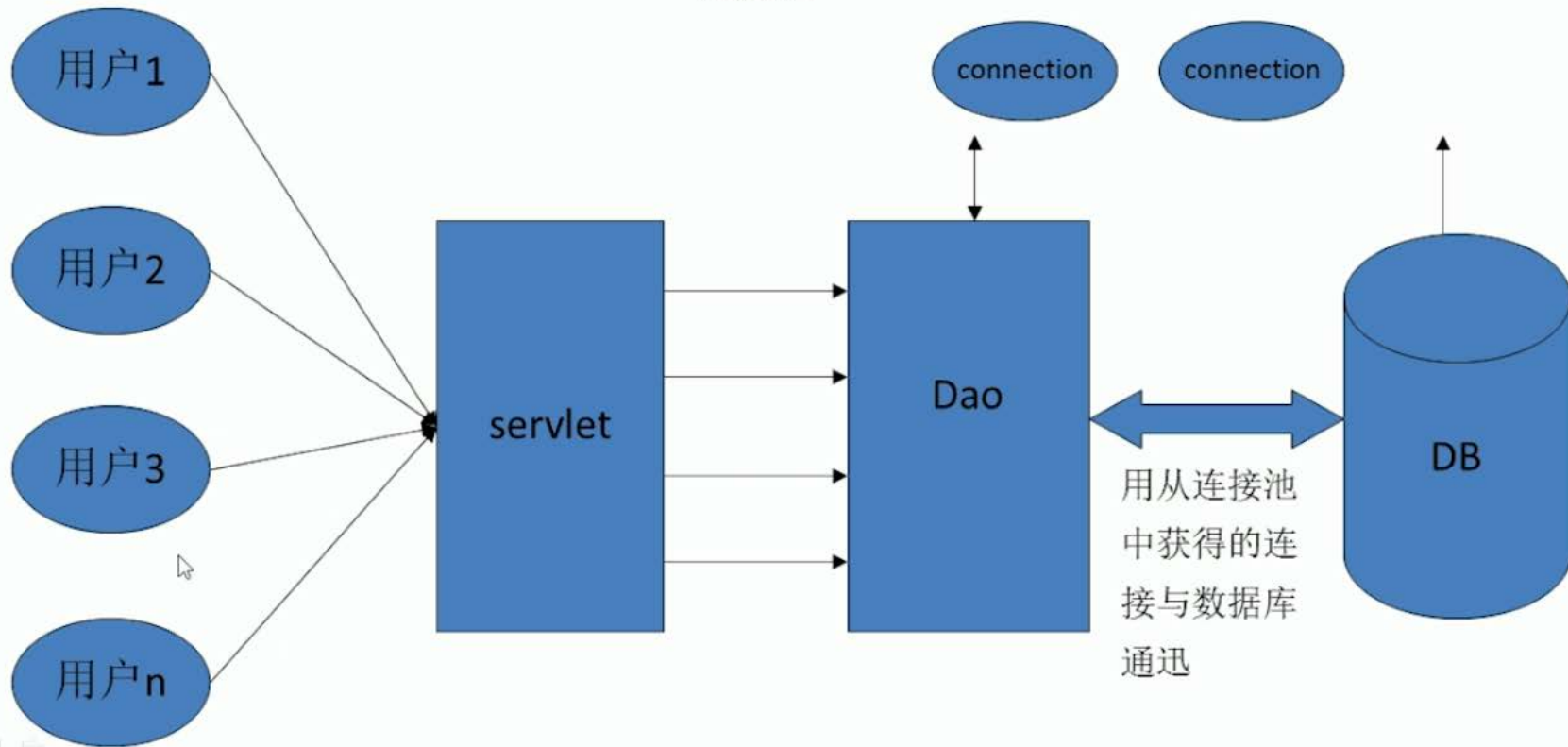
- 应用程序直接获取连接的缺点



- 缺点：用户每次请求都需要向数据库获得连接，而数据库创建连接通常需要消耗相对较大的资源，创建时间也较长。假设网站一天10万访问量，数据库服务器就需要创建10万次连接，极大的浪费数据库的资源，并且极易造成数据库服务器内存溢出。

数据库连接池

连接池



数据库连接池

- C3P0连接池的使用



课程总结

