

JDBC



作成者:周东

作成日:2019.08.29

Neusoft

目录

- ◆ JDBC技术基础
- ◆ Driver
- ◆ DriverManager
- ◆ Statement
- ◆ ResultSet
- ◆ PreparedStatement
- ◆ 事务

JDBC技术基础

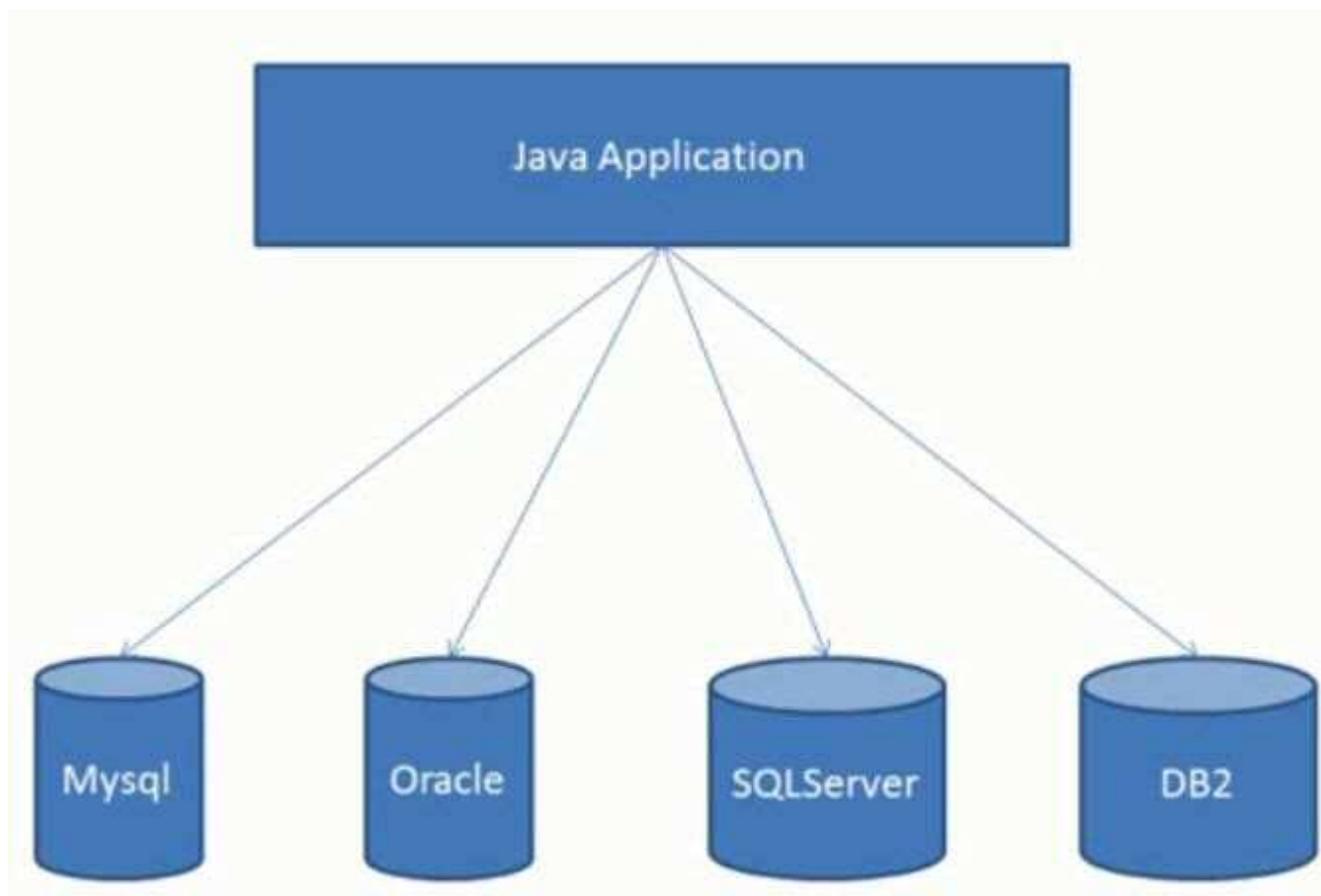
JDBC (Java DataBase Connectivity) 是一个独立于特定数据库管理系统、通用的SQL数据库存取和操作的公共接口 (一组API), 定义了用来访问数据库的标准Java类库, 使用这个类库可以以一种标准的方法, 方便地访问数据库资源。

JDBC为访问不同的数据库提供了一种统一的途径, 为开发者屏蔽了一些细节问题。

JDBC的目标是使Java程序员使用JDBC可以连接任何提供了JDBC驱动程序的数据系统, 这样就使得程序员无需对特定的数据库系统的特点有过多的了解, 从而大大简化和加快了开发过程。

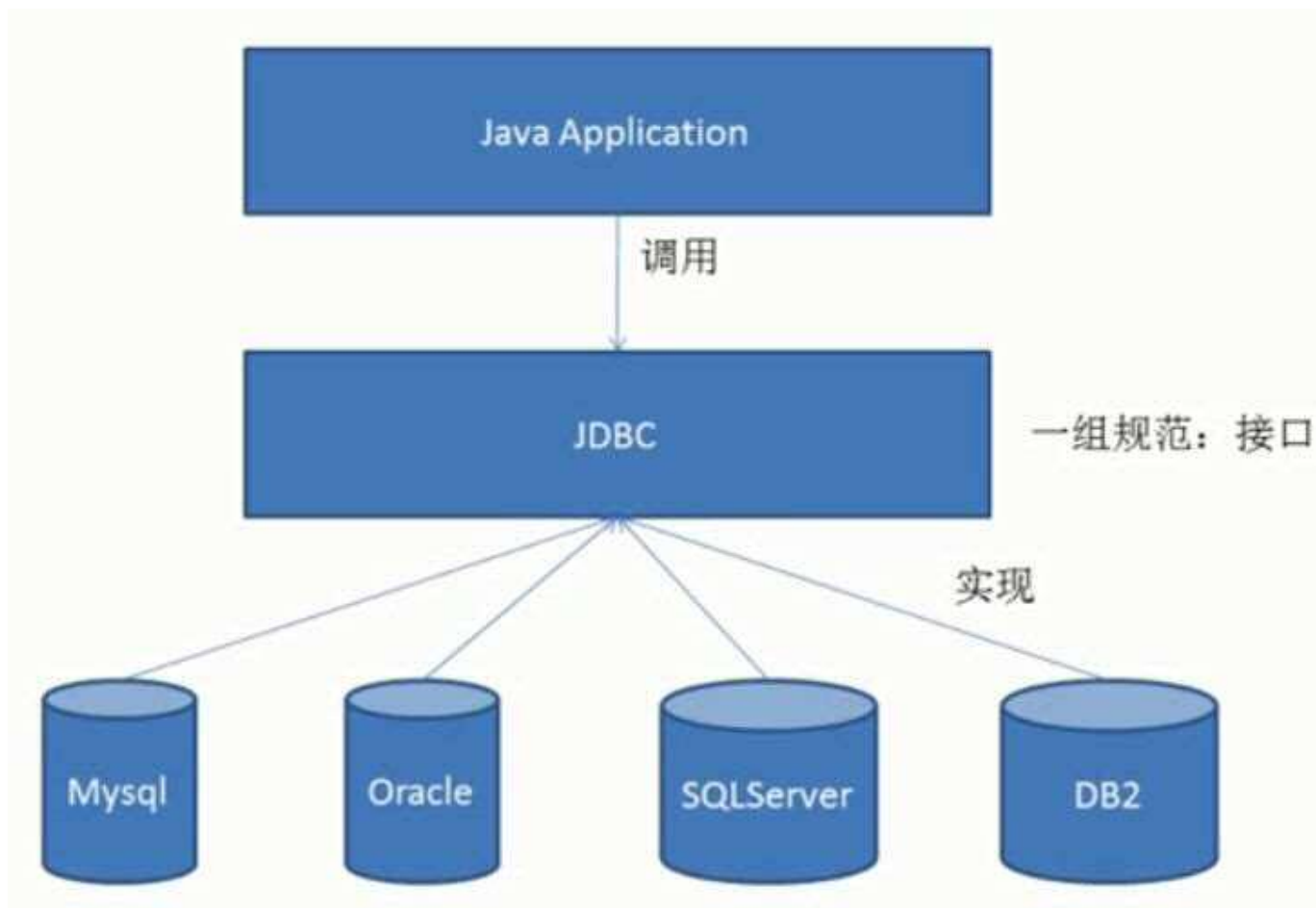
JDBC技术基础

无JDBC



JDBC技术基础

有JDBC



JDBC技术基础

JDBC接口 (API) 包括两个层次:

- ①面向应用的API:Java API, 抽象接口, 供应用程序开发人员使用。(连接数据库, 执行SQL语句, 获得结果)
- ②面向数据库的API:Java Driver API, 供开发商开发数据库驱动程序用。

JDBC技术基础

JDBC驱动程序:各个数据库厂商根据JDBC的规范制作的JDBC实现类的类库。

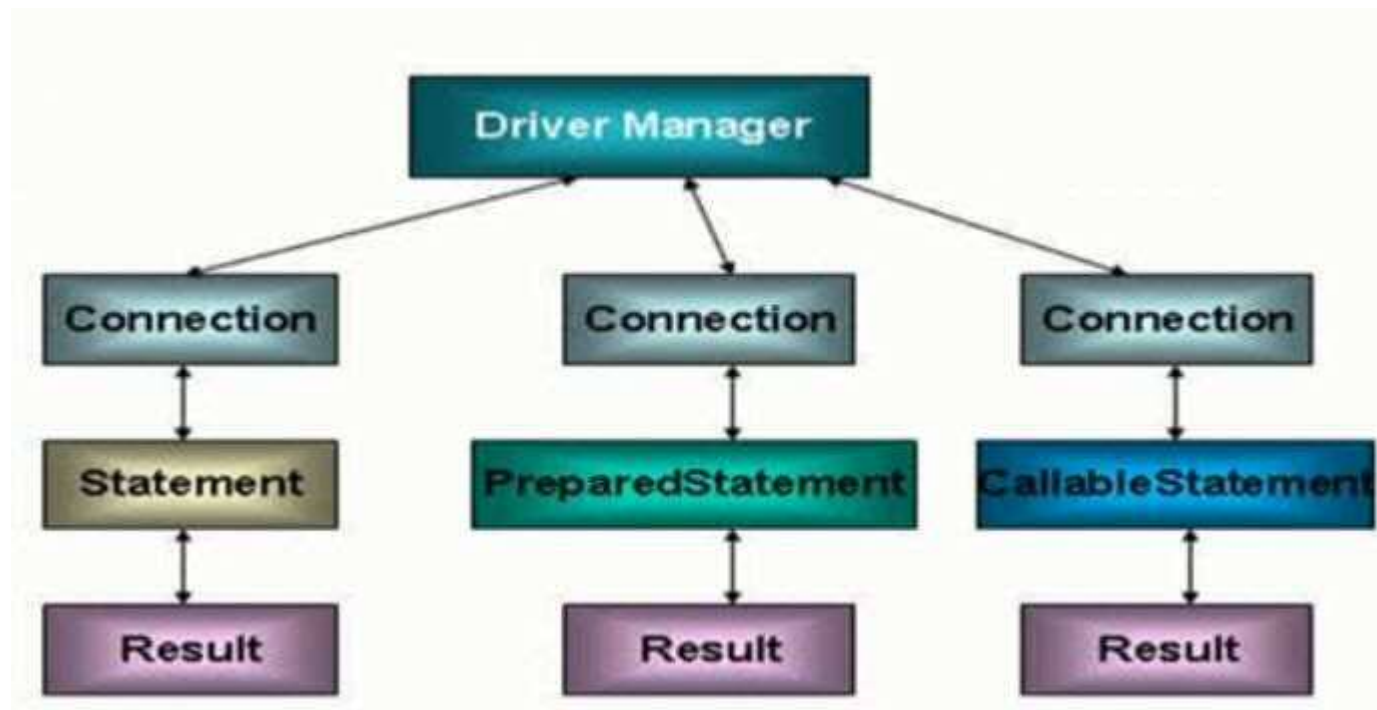
JDBC驱动程序总共有四种类型:

- ①JDBC-ODBC桥。
- ②部分本地API部分Java的驱动程序。
- ③JDBC网络纯Java驱动程序。
- ④本地协议的纯Java驱动程序。

JDBC技术基础

JDBC API的层次关系

JDBC API是一系列的接口，它使得应用程序能够进行数据库联接，执行SQL语句，并且得到返回结果。



Driver

Driver接口

Java.sql.Driver接口是所有JDBC驱动程序需要实现的接口。这个接口是提供给数据库厂商使用的, 不同数据库厂商提供不同的实现。

DriverManager

DriverManager类

在程序中不需要直接去访问实现了Driver接口的类, 而是由驱动程序管理器类(`java.sql.DriverManager`)去调用这些Driver实现。

- ①可以通过重载的`getConnection`方法获取数据库连接, 较为方便。
- ②可以同时管理多个驱动程序。若注册了多个数据库连接, 则调用`getConnection()`方法时传入的参数不同, 即返回不同的数据库连接。

Statement

插入、更新、删除

①获取语句执行平台

```
Statement st = conn.createStatement();
```

②书写SQL

```
String sql = "";
```

③执行SQL

```
st.executeUpdate(sql);
```

ResultSet

查询

①书写SQL

```
String sql = "";
```

②调用Statement对象的executeQuery(sql)方法可以得到结果集

```
ResultSet rs = statement.executeQuery(sql);
```

③ResultSet返回的实际上就是一张数据表

④ResultSet可以调用next()方法检测下一行是否有效

```
if (rs.next()) {  
}
```

```
While (rs.next()) {  
}
```

⑤当指针对位到一行时可以调用getXxxx(index)或getXxxx(columnName)获取每一列的值

```
int id = rs.getInt(1);
```

```
String name = rs.getString("name");
```

PreparedStatement

预编译SQL语句

①创建PreparedStatement

```
PreparedStatement ps = connection. prepareStatement(sql);
```

②调用PreparedStatement的setXxxx(index, value)方法实现参数值设定

③执行SQL

```
preparedStatement.executeUpdate();
```

④防止SQL注入

事务

①事务自动提交

```
connection.setAutoCommit(false);
```

②事务回滚

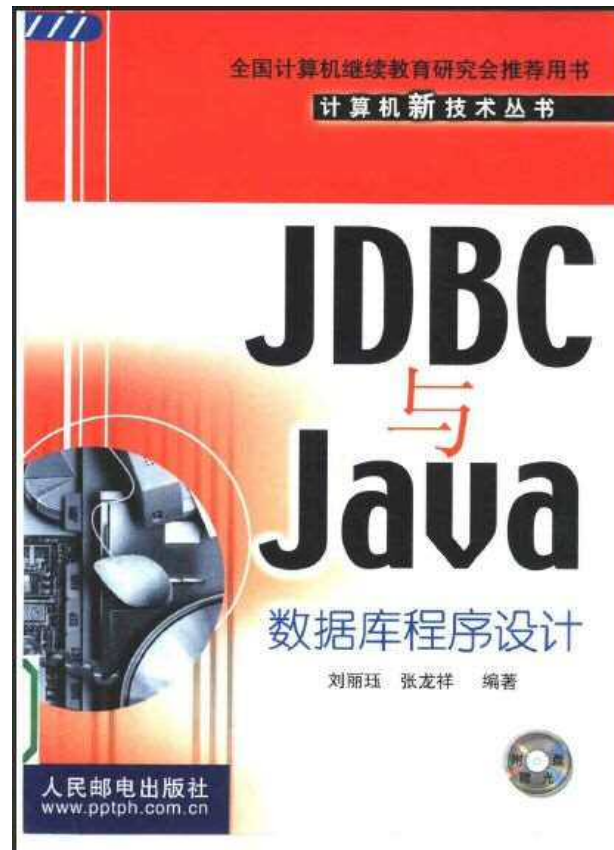
```
connection.rollback();
```

③事务提交

```
connection.commit();
```

参考资料

JDBC与Java数据库程序设计



Neusoft

Beyond Technology