在OracleJDBC访问中加入Spring特性（2） PDF转换可能丢失图片或格式，建议阅读原文 https://www.100test.com/kao_ti2020/143/2021_2022__E5_9C_A8OracleJ_c102_143574.htm 值得注意的是与直接使用 JDBC 相比，利用 Spring 框架可以少得多的代码实现同样的功能。如代码清单 3 所示，您不需要编写和维护管理资源（连接、语句、结果集）的代码。甚至代码清单 3 中的少量的异常处理代码也不是绝对必需的，因为 DataAccessException 是一个非强制异常。因为 Number 类型用来返回奖金，因此不需要显式调用 ResultSet 的 wasNull 方法。实际上，您甚至在代码清单 3 中的任何地方都找不到 ResultSet 语法！代码清单 3 还说明了由 Spring 框架的 JDBC 支持所提供和使用的基础类之一 JdbcTemplate。我们将使用一个数据源来完成这个由 Spring 提供的类的实例化，然后在模板类上使用提供的 SQL 字符串调用它的被覆盖的 queryForList 方法之一。queryForList 方法将返回一个包含 HashMap 的 ArrayList，其中该 ArrayList 中的每一个元素都是一个返回的数据行，一个特定数组阵列元素中的每一个 Map 条目都是该行中的一个列值。 JdbcTemplate 提供了许多被覆盖的 queryForList 方法，它们可以用来查询潜在的多行数据。这个非常有用的类还提供了诸如 queryForInt（返回单个整数）、queryForLong（返回单个 long 型整数）、query、0update 之类的方法。要分辨这些不同的被覆盖的方法，最容易的方式是阅读与 Spring 框架一起提供的基于 Javadoc 的 API 文档中的“ 方法详情 ”部分。这些方法的不同点在于使用的语句的类型（例如 Statement 或 PreparedStatement）和支持的特性。JdbcTemplate 还提供了一

些方法，与上面使用的方法相比，它们需要更多的 JDBC 知识，但它们提供了更好的灵活性。这些更灵活但需要更多 JDBC 知识的方法将在本文稍后进行说明。 JDBC 异常处理 返回到代码清单 1，注意 java.sql.SQLException 是唯一一个显式捕获的异常。SQLException 中捕获了与数据库和 SQL 相关的各种异常情况。描述 SQLException 类的 Javadoc 注释介绍了可以从 SQLException 的实例中获得的基本信息。这些信息包括错误描述字符串 [getMessage()]、某个标准化 SQLState 异常 String [getSQLState()] 和供应商特有的整型错误码 [getErrorCode()]。在代码清单 1 中实现的简单的异常处理中使用了所有这三种信息。 SQLException 是一种强制异常（直接扩展 java.lang.Exception）。Java 的强制异常曾经引起很大争议，现在 Java 社区似乎正在取得共识：只有当在应用程序能够处理异常时才应使用强制异常。如应用程序代码不能以有意义的方式处理异常，则不应当强制处理该异常。因为 SQLException 是强制异常，所以应用程序代码必须处理它，或者捕获它并对其进行一些处理或显式地将其抛出给调用代码。 SQLException 的最后一点细微差别在于它是使用 SQL 的关系数据源所特有的异常。这使得不适合将它包含在真正可移植的数据访问对象 (DAO) 中，后者应当独立于数据信息库类型和访问语言。 Spring 框架对 SQLException 的处理是其在支持更容易的 JDBC 开发和维护方面最有用的特性之一。Spring 框架提供了完成 SQLException 抽象化的 JDBC 支持，并提供了一个对 DAO 友好的非检查异常层次结构。 处理供应商特有的错误码 如上所述，标准的 SQLException 提供了一个标准化的信息段 (SQLState) 和一个供应商特有的信息段

(ErrorCode)。正如大多数的数据库和它们的 JDBC 驱动程序实现一样，Oracle 数据库和 JDBC 驱动程序通过供应商特有的错误码所提供的关于问题的详细信息要比通过 SQLException 的与供应商无关的 SQLState 组件所提供的信息多得多。Oracle 数据库及其 JDBC 驱动程序通过 Error Code 提供的更丰富得多的详细信息的一个明显的例子是 SQLState 代码 42000（通常这指示语法错误或访问问题）。对于 Oracle JDBC 驱动程序的大量不同的 Oracle 错误码，SQLException 都将返回相同的 SQLState 值 (42000)。对应 SQLState 的 42000 值的一些 Oracle 错误码包括 900（“ 无效 SQL 语句 ”）、903（“ 无效表名 ”）、904（“ 无效标识符 ”）、911（“ 无效字符 ”）和 936（“ 缺少表达式 ”）。此外很明显任何来源于 Oracle JDBC 驱动程序的错误（与来源于 Oracle 数据库的错误相反）都没有任何对应的 SQLState。正如这些例子所显示那样，Oracle 特有的错误码所提供的关于错误情况的详细信息要比与供应商无关的 SQLState 所提供的信息更丰富得多。 有时候区分来源于 Oracle 数据库的错误和来源于 Oracle JDBC 驱动程序的错误非常重要。例如，903（“ 无效的表名 ”）错误码对应 Oracle 数据库错误码 ORA-00903。相反，17003（“ 无效的列索引 ”）错误码对应 Oracle JDBC 驱动程序错误码 ORA-17003。这两种类型的错误码（数据库和 JDBC 驱动程序）都是 Oracle 特有的（如 ORA- 前缀所指示的那样）。因为没有为来源于 Oracle JDBC 驱动程序的错误提供 SQLState 错误码，因此必须使用 Oracle 特有的错误码来区分由 JDBC 驱动程序导致的错误。 在下面的表 1 中列出了访问 Oracle 数据库的 JDBC 中的一些最常见的错误。“ 示例代码和/或注释 ” 列

中的示例显示了导致这种错误的 SQL 语句类型或提供了关于表中的该行上显示的特定错误的其他注释。 表 1 错误标记 Oracle 错误 SQLState 示例代码和/或注释 基于语句：SELECT ename FROM emp 变种的 SQL 相关错误“ 唯一性约束 ” 1 2300 例如主键违规“ 资源忙且指定 NOWAIT 获取资源 ” 54 61000 只有在指定了 NOWAIT 时才出现“ 无效的 SQL 语句 ” 900 42000 ename FROM emp“ 无效的表名 ” 903 42000 SELECT ename FROM“ 无效的标识符 ” 904 42000 SELECT empname FROM emp“ 无效的字符 ” 911 42000 SELECT ename FROM emp.“ 缺少列 ” 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com