

教你使用Java中的RowSetJava认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/594/2021_2022__E6_95_99_E4_BD_A0_E4_BD_BF_E7_c104_594680.htm ResultSet是使用Jdbc编程的人入门和常用的操作数据库的类，自JDK 1.4开始，易于使用RowSet接口被引入。RowSet接口扩展了标准java.sql.ResultSet接口。RowSetMetaData接口扩展了java.sql.ResultSetMetaData接口。因此，熟悉JDBC API的开发人员必须学习少数几个新API才能使用rowset。此外，与JDBC ResultSet对象配套使用的第三方软件工具也可以方便地用于rowset。但是在JDK 1.4中，只有一个RowSet接口，使得RowSet的使用范围打了折扣。不过JDK 5.0定义了5个标准的JDBC RowSet接口，并且给出了相应的参考实现，因此可以很方便的使用RowSet接口所提供的功能。RowSet对象可以建立一个与数据源的连接并在其整个生命周期中维持该连接，在此情况下，该对象被称为连接的rowset。rowset还可以建立一个与数据源的连接，从其获取数据，然后关闭它。这种rowset被称为非连接rowset。非连接rowset可以在断开时更改其数据，然后将这些更改发送回原始数据源，不过它必须重新建立连接才能完成此操作。相比较java.sql.ResultSet而言，RowSet的离线操作能够有效的利用计算机越来越充足的内存，减轻数据库服务器的负担，由于数据操作都是在内存中进行然后批量提交到数据源，灵活性和性能都有了很大的提高。RowSet默认是一个可滚动，可更新，可序列化的结果集，而且它作为JavaBeans，可以方便地在网络间传输，用于两端的数据同步。

1、与ResultSet比较 (1) RowSet扩展

了ResultSet接口，因此可以像使用ResultSet一样使用RowSet。

(2) RowSet扩展了ResultSet接口，因此功能比ResultSet更多、更丰富。

(3) 默认情况下，所有RowSet对象都是可滚动的和可更新的。而ResultSet是只能向前滚动和只读的。

(4) RowSet可以是非链接的，而ResultSet是连接的。因此利用CachedRowSet接口可以离线操作数据。

(5) RowSet接口添加了对JavaBeans组件模型的JDBC API支持。rowset可用作可视化Bean开发环境中的JavaBeans组件。

(6) RowSet采用了新的连接数据库的方法。

(7) CachedRowSet是可以序列化的。

(8) RowSet和ResultSet都代表一行行的数据、属性以及相关操作方法。

(9) 自己认为，应该倾向于把RowSet看成是与数据库无关的东西，它只是一个代表一行行数据的对象，而ResultSet则是一个与数据库紧密联系的东西。

2、JDK 5.0的5个标准RowSet接口

在JDK 5.0中，5个标准RowSet接口包括CachedRowSet，WebRowSet，FilteredRowSet，JoinRowSet和JdbcRowSet。相应的参考实现是Sun公司给出的，位于com.sun.rowset包下，分别为CachedRowSetImpl，WebRowSetImpl，FilteredRowSetImpl，JoinRowSetImpl和JdbcRowSetImpl。这5个标准接口中JdbcRowSet是链接的rowset，而其他4个是非链接的rowset。

(1) CachedRowSet：最常用的一种RowSet。其他三种RowSet（WebRowSet，FilteredRowSet，JoinRowSet）都是直接或间接继承于它并进行了扩展。它提供了对数据库的离线操作，可以将数据读取到内存中进行增删改查，再同步到数据源。CachedRowSet是可滚动的、可更新的、可序列化，可作为JavaBeans在网络间传输。支持事件监听，分页等特性。CachedRowSet对象通常

包含取自结果集的多个行，但是也可包含任何取自表格式文件（如电子表格）的行。（2）WebRowSet：继承自CachedRowSet，并可以将WebRowSet写到XML文件中，也可以用符合规范的XML文件来填充WebRowSet。（3）FilteredRowSet：通过设置Predicate（在javax.sql.rowset包中），提供数据过滤的功能。可以根据不同的条件对RowSet中的数据进行筛选和过滤。（4）JoinRowSet：提供类似SQL JOIN的功能，将不同的RowSet中的数据组合起来。目前在Java 6中只支持内联（Inner Join）。（5）JdbcRowSet：对ResultSet的一个封装，使其能够作为JavaBeans被使用，是唯一一个保持数据库连接的RowSet。JdbcRowSet对象是连接的RowSet对象，也就是说，它必须使用启用JDBC技术的驱动程序（“JDBC驱动程序”）来持续维持它与数据源的连接。

3、填充RowSet 前面说过，应该倾向于把RowSet看成是与数据库无关而只代表一行行数据的对象，因此就涉及到数据从哪里来的问题。（1）从数据库直接获取数据 由于大部分情况下，与数据打交道也就是与数据库打交道，因此RowSet接口提供了通过JDBC直接从数据库获取数据的方法，以参考实现JdbcRowSetImpl为例，就是这样：
`RowSet rs = new JdbcRowSetImpl(). //也可以是CachedRowSetImpl, WebRowSetImpl, FilteredRowSetImpl, JoinRowSetImpl.
rs.setUrl("jdbc:mysql:///test"). rs.setUsername("root").
rs.setPassword(""). rs.setCommand("SELECT * FROM EMPLOYEES"). rs.execute().` 设置好相关属性，运行execute()方法后，EMPLOYEES表中的数据就被填充到rs对象中了。除了通过设置JDBC连接URL、用户名和密码外，RowSet也可以使

用数据源名称属性的值来查找已经在命名服务中注册的 DataSource 对象。完成检索后，可以使用 DataSource 对象创建到它所表示的数据源的连接，设置数据源名称可以使用 setDataSourceName() 方法。（2）用 ResultSet 填充 在有现成 ResultSet 的情况下，如果想将其作为 RowSet 使用；或者当 DBMS 不提供对滚动和更新的完全支持时，如果想使不可滚动和只读的 ResultSet 对象变得可滚动和可更新，可以创建一个使用该 ResultSet 对象的数据所填充的 CachedRowSet 对象。

```
ResultSet rs = stmt.executeQuery("SELECT * FROM  
EMPLOYEES"). CachedRowSet crs = new CachedRowSetImpl(). //  
也可以是 WebRowSetImpl, FilteredRowSetImpl
```

, JoinRowSetImpl, 因为他们均继承自 CachedRowSetImpl
crs.populate(rs). 运行 populate() 方法后，ResultSet 对象 rs 中的数据就被填充到 crs 对象中了。（3）用 XML 填充 如果您打算将 XML 作为数据交换格式在客户端和你的服务器之间传输数据并且向实现数据离线编辑、或者向使用 XML 格式的数据的话，可以使用 WebRowSet 接口来用 XML 填充数据。

```
WebRowSet wrs = new WebRowSetImpl(). wrs.readXml(new  
FileReader(new File("D:\\employees.xml"))). 运行 readXml() 方法  
后，employees.xml 文件的数据就被填充到 wrs 对象中了
```

。employees.xml 文件的格式参见附录。（4）用其他方法填充 如果形用其他方式填充，比如 csv、excel、text、http 等格式或方法填充数据，那么就需要自己编写代码实现 RowSet。4、操作 RowSet 中的数据及元数据 除了 ResultSet 提供的操作数据和元数据方法外，RowSet 接口没有提供太多额外的方法。1) 更新数据 rs.absolute(5). rs.updateInt(1, 10). rs.updateInt(2,

1000). rs.updateString(3, "John"). rs.updateRow(). (2) 插入数据 rs.moveToInsertRow(). rs.updateInt(1, 10). rs.updateInt(2, 1000). rs.updateString(3, "John"). rs.insertRow(). (3) 删除数据 rs.absolute(5). rs.deleteRow(). (4) 设置属性 rs.setCommand("select id, salary, name from employees where id = ?"). rs.setInt(1, 1). rs.execute(). (5) 元数据 RowSetMetaData rsmd = (RowSetMetaData)rs.getMetaData(). int count = rsmd.getColumnCount(). int type = rsmd.getColumnType(2).

5、事务与更新底层数据源

RowSet本身只代表具体数据，事务以及底层数据源的更新是与底层数据源密切相关的概念。对于JDBC数据源，相应的标准接口JdbcRowSet通过与数据库相关的方法来实现，如commit()，rollback()等。对于标准接口的非连接rowset，如CachedRowSet，则在RowSet中的数据改动后，通过运行acceptChanges()方法，在内部调用RowSet对象的writer将这些更改写入数据源，从而将CachedRowSet对象中的更改传播回底层数据源。

6、可序列化非连接RowSet

使用CachedRowSet对象的主要原因之一是要在应用程序的不同组件之间传递数据。因为CachedRowSet对象是可序列化的，所以可使用它（举例来说）将运行于服务器环境的企业JavaBeans组件执行查询的结果通过网络发送到运行于web浏览器的客户端。由于CachedRowSet对象是非连接的，所以和具有相同数据的ResultSet对象相比更为简洁。因此，它特别适于向瘦客户端（如PDA）发送数据，这种瘦客户端由于资源限制或安全考虑而不适于使用JDBC驱动程序。所以CachedRowSet对象可提供一种“获取各行”的方式而无需实现全部JDBC API。 ebRowSet继承自CachedRowSet，

除了拥有CachedRowSet的优点外，还可以将WebRowSet输出成XML，也可以将XML转换成WebRowSet，更加适合在Web环境中使用。标准的WebRowSet XML模式定义位于URI <http://java.sun.com/xml/ns/jdbc/webrowset.xsd>。将WebRowSet保存为XML的代码事例如下：
`wrs.setCommand("select id, salary, name from employees").wrs.execute().wrs.writeXml(new FileWriter(new File("D:\\employees.xml")))`。更多优质资料尽在百考试题论坛 百考试题在线题库 java认证更多详细资料 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