

Chapter 33 Advanced Java Database Programming

1. Batch processing enables SQL nonselect statements to be processed all together. Therefore, it is more efficient than processing each statement individually.
2. To add an SQL statement to a batch, you add it to a Statement object using the `addBatch(sqlString)` method. Use `executeBatch()` method to execute it.
3. No, you cannot execute a SELECT statement in a batch.
- 4.

To find out whether a driver supports batch updates, invoke `supportsBatchUpdates()` on a `DatabaseMetaData` instance. If the driver supports batch updates, it will return true. The JDBC drivers for MySQL, Access, and Oracle all support batch updates.

5. A scrollable result set allows you to move the cursor in any order and place it anywhere. By default, a result set is non-scrollable and you can only traverse it forward from beginning to end. A scrollable result set may be updateable, i.e., insert, delete, update rows in the table through a result set.
- 6.

You use the following statement to create a scrollable and updateable result set:

```
Statement statement = connection.createStatementResultSet.TYPE_SCROLL_INSENSITIVE,  
ResultSet.CONCUR_UPDATABLE)
```

- 7.

You can use `supportsResultSetType(int type)` and `supportsResultSetConcurrency(int type, int concurrency)` in the `DatabaseMetaData` interface to find out which result type and concurrency modes are supported by the JDBC driver.

8. Advantages: 1) Simple. RowSet combines Connection, Statement, and ResultSet into one interface and thus makes programming simple. 2) Good for visual GUI programming. Concrete classes of RowSet can be used as JavaBeans component in a visual GUI development environment and RowSet objects can be configured at design time. 3) RowSet fires RowSetEvent when cursor moved, row changed, or the entire row set changed. This event can be used to synchronize a `RowSet` with the components that rely on the `RowSet`.

9. `JdbcRowSet` and `CachedRowSet` are subinterfaces of `RowSet`. `JdbcRowSet` is connected, meaning that a `RowSet` object makes a

connection with a data source and maintains that connection throughout its life cycle. CachedRowSet is disconnected, meaning that a RowSet object makes a connection with a data source, executes a query to get data from the data source, and then closes the connection. A disconnected rowset may make changes to its data while it is disconnected and then send the changes back to the original source of the data, but it must reestablish a connection to do so. Also, JdbcRowSet is neither serializable nor cloneable, while CachedRowSet is serializable and cloneable.

10. Sun provides a reference implementation for JdbcRowSet and CachedRowSet. You can create an instance of JdbcRowSet using JdbcRowSetImpl and create an instance of CachedRowSet using CachedRowSetImpl.

11. A RowSet object is scrollable and updateable. To write changes back to the database for a CachedRowSet, invoke the acceptChanges() method on a CachedRowSet.

12. There are three handlers in RowSetListener. 1) cursorMoved: invoked after a cursor moved. 2) rowChanged: invoked after a row changed. 3) rowSetChanged: invoked after the entire row set has changed.

13.

You have to use the SQL3 new data type BLOB to declare a column in the table, and use JDBC to prepared statement to prepare an INSERT statement. To set the data to the field, use the setBinaryStream method.

14.

To retrieve it, use the getBlob() method to obtain the binary data in the Blob type and use the getBytes method to return an array of bytes from the Blob value. To create an image, use ImageIcon(byte[]).

15.

Oracle supports SQL3 BLOB and CLOB types. MySQL support CLOB. Access supports none.