

# JDBC INTERVIEW QUESTIONS

[LEAVE A COMMENT](#)

## How does the JDBC work?

The Java Database Connectivity (JDBC) is used to whenever a Java application should communicate with a relational database for which a JDBC driver exists.

Main JDBC classes:

DriverManager. Manages a list of database drivers.

Driver. The database communications link, handling all communication with the database.

Connection. Interface with all the methods for contacting a database

Statement. Encapsulates an SQL statement which is passed to the database to be parsed, compiled, planned and executed.

ResultSet. The answer/result from a statement. A ResultSet is a list which encapsulates all outgoing results from a given SQL query.

## Explain the ways in which we can get runtime information about the JDBC Driver?

Use the following DatabaseMetaData methods:

`getDriverMajorVersion()`

`getDriverMinorVersion()`

`getDriverName()`

`getDriverVersion()`

## Explain whether the JDBC-ODBC Bridge multi-threaded?

No. The JDBC-ODBC Bridge does not support multi threading. The JDBC-ODBC

Bridge uses synchronized methods to serialize all of the calls that it makes to ODBC. Multi-threaded Java programs may use the Bridge, but they won't get the advantages of multi-threading.

### **Explain the difference between cold backup, hot backup, warm backup recovery?**

Cold backup means all these files must be backed up at the same time, before

the database is restarted. Hot backup (official name is 'online backup' ) is a backup taken of

each tablespace while the database is running and is being accessed by the users

### **What are the advantage of denormalization?**

Data denormalization is reverse procedure, carried out purely for reasons of

improving performance. It maybe efficient for a high-throughput system to replicate data for certain data.

### **How do you handle your own transaction ?**

Connection Object has a method called setAutocommit ( boolean flag) . For

handling our own transaction we can set the parameter to false and begin your transaction .

Finally commit the transaction by calling the commit method.

### **Explain whether there is any practical limit for the number of SQL statements that can be**

### **added to an instance of a Statement object ?**

While the specification makes no mention of any size limitation for

Statement.addBatch(), this seems to be dependent, as usual, on the driver. Among other things,

it depends on the type of container/collection used. I know of at least one driver that uses a Vector and grows as needed. I've seen questions about another driver that appears to peak somewhere between 500 and 1000 statements. Unfortunately, there doesn't appear to be any metadata information regarding possible limits. Of course, in a production quality driver, one would expect an exception from an `addBatch()` invocation that went beyond the command list's limits.

## **How can we ensure that the Statement and its ResultSet will be closed on a commit or rollback?**

Use the `DatabaseMetaData` methods `supportsOpenStatementsAcrossCommit()` and `supportsOpenStatementsAcrossRollback()`.

## **How can we create an updatable ResultSet?**

Just as is required with a scrollable `ResultSet`, the `Statement` must be capable of returning an updatable `ResultSet`. This is accomplished by asking the `Connection` to return the appropriate type of `Statement` using `Connection.createStatement(int resultSetType, int resultSetConcurrency)`. The `resultSetConcurrency` parameter must be `ResultSet.CONCUR_UPDATABLE`.

The actual code would look like this:

```
Statement stmt = con.createStatement( ResultSet.TYPE_SCROLL_SENSITIVE,  
ResultSet.CONCUR_UPDATABLE );
```

Note that the spec allows a driver to return a different type of

`Statement/ResultSet` than that requested, depending on capabilities and circumstances, so the

actual type returned should be checked with `ResultSet.getConcurrency()`.

### Related Tutorials :

1. [What is JDBC](#)
2. [Spring JDBC](#)
3. [Accessing JDBC from EJBs](#)
4. [Spring Interview Questions](#)
5. [Java Interview Questions](#)

