

Java JDBC Interview Questions Pdf

Question: 1

What is JDBC?

It is an API. The latest version of JDBC API is (3.0)

The JDBC 3.0 API is divided into two packages.

java.sql

javax.sql

Both packages are included in the J2SE and J2EE platforms.

Advantages:

The JDBC API can be used to interact with multiple data sources in a distributed heterogeneous environment.

It can connect to any of the database from java language.

It can switch over to any backend database without changing java code or by minute changes.

Question: 2

What are the two major components of JDBC?

One implementation interface for database manufacturers, the other implementation interface for application and applet writers.

Question: 3

What are the types of JDBC driver models?

There are two JDBC driver models and they are

Two tier model: In this method, Java applications together interact directly with the database.

A JDBC driver is required to communicate with the particular database management system that is being accessed.

SQL statements are sent to the database and the results are given to user.

This model is referred to as client/server configuration where user is the client and the machine that has the database is called as the server.

Three tier model: A middle tier is introduced in this model. The function of this model are

Collection of SQL statements from the client and handing it over to the database,

Receiving results from database to the client and

Maintaining control over accessing and updating of the above.

Question: 4

Which methods and classes are used for connection pooling?

JDBC provides an interface, which must be implemented by a JDBC driver supporting connection pooling.

This interface is called `ConnectionPoolDataSource`.

Most existing JDBC drivers (`DataSource` instances) implement this interface. It provides the following methods for obtaining `PooledConnections`:

`PooledConnection get PooledConnection()` throws `SQLException`;

`PooledConnection get PooledConnection (String user, String password)` throws `SQLException`;

Such connections can be closed by issuing `connection.close()`, but it's only a logical closing, not physical.

Physically, no connection is closed, it's just returned to the pool.

Question: 5

Why most of the classes given in JDBC API are interfaces?

Abstract class forces all sub classes to implement common method whichever are required implementations.

Only abstract method and class can do this job.

That's why most part of the JDBC API is a formation of interfaces.