Java JDBC Interview Questions and Answers Pdf

Question: 1

What is JDBC Driver Interface?

The JDBC Driver Interface provides vendor specific implementations of the abstract classes provided by the JDBC API.

Each Vendor's driver must provide implementations of the java.sql.Connection, Statement, PreparedStatement, CallableStatement, ResultSet and driver.

Question: 2

What packages are used in JDBC?

There are 8 packages:

java.sql.Driver,

Connection,

Statement,

PreparedStatement,

CallableStatement,

ResultSet,

ResultSetMetaData,

DatabaseMetaData.

Question: 3

What are the types of statements in JDBC?

Statement – to be used createStatement() method for executing single SQL statement.

PreparedStatement – To be used preparedStatement() method for executing same SQL statement over and over.

CallableStatement – To be used prepareCall() method for multiple SQL statements over and over.

What is the difference between JDBC 1.0 and JDBC 2.0?

The JDBC 2.0 API includes many new features in the java.sql package as well as the new Standard Extension package, javax.sql. This new JDBC API moves Java applications into the world of heavy duty database computing.

The new JDBC Standard Extension API, an integral part of Enterprise Java Beans (EJB) technology, allows you to write distributed transactions that use connection pooling, and it also makes it possible to connect to virtually any tabular data source, including files and spread sheets.

The JDBC 2.0 API includes many new features like

Scrollable result sets

Batch Updates

Connection Pooling

Distributed transactions

set autocomit()

What is JDBC?

JDBC is a set of Java API for executing SQL statements.

This API contains of a set of classes and interfaces to enable programs to write pure Java Database applications.

JDBC is a layer of abstraction that allows users to choose between database. It allows you to change to a different engine and to write to a Single API.

JDBC allows you to write database applications in Java without having to concern yourself with the underlying details of a particular database.

JDBC standards for Java Database Connectivity. It is a specification given by Sun Microsystems and standards followed by X/Open SAG (SQL Access Group) CLI (Call Level Interface) to interact with the DB.

How many JDBC Drivers are there?

There are four types of JDBC drivers.

JDBC – ODBC Bridge Driver (Type – 1 driver)

Native API Partly Java Driver (Type – 2 driver)

Net Protocol Pure Java Driver (Type – 3 driver)

Native Protocol Pure Java Driver (Type – 4 driver)

What is the difference between JDBC and ODBC?

ODBC is for Microsoft and JDBC is for Java applications.

ODBC can't be directly used with Java because it uses a C interface.

ODBC makes use of pointers which have been removed totally from Java.

ODBC mixes simple and advanced features together and has complex options for simple queries. But JDBC is designed to keep things simple while allowing advanced capabilities when required.

ODBC requires manual installation of the ODBC driver manager and driver on all client machines. JDBC drivers are written in Java and JDBC code is automatically installable, secure, and portable on all platforms.

JDBC API is a natural Java interface and is built on ODBC. JDBC retains some of the basic features of ODBC.

What are the different types of exceptions in JDBC?

batchUpdate Exception

Data Truncation

SQL Exception

SQL Warning

What are the steps for connecting to the database using JDBC?

Using DriverManager:

Load the driver class using class.forName(driverclass) and Class.forName() loads the driver class and passes the control to DriverManager class.

DriverManager.getConnection() creates the connection to the database.

Using DataSource:

DataSource is used instead of DriverManager in Distributed Environment with the help of JNDI.

Use JNDI to lookup the DataSource from Naming service server.

DataSource.getConnection method will return Connection object to the database.

What are the steps involved in establishing a connection?

This involves two steps:

loading the drivers and

making the connection