By OnlineInterviewQuestions.com

JDBC Interview Questions For Freshers

Best JDBC Interview Questions For Freshers | Experienced

JDBC is an important aspect in terms of interview questions for the Java programmers. The candidates who want to be hired by good companies then they should start preparing themselves. Here we present you some interview questions that may help in your interviews.

Finally, Practice Best **JDBC Interview Questions and Answers** for the best preparaton of the JDBC Interview. These JDBC Interview Questions are very popular and asked various times in JDBC Interview. So, Practice these questions to check your final preparaton for your Interview.

Q1. What is the full form of JDBC and what is its purpose?

JDBC stands for Java Database connectivity which is an application that helps in connection for the programming of Java language. It is primarily helpful for the clients by guiding them on how to access a database. This java technology helps in connectivity of java database. It comes from Java Standard Edition platform, from Oracle Corporation.

Q2. What do you understand by JDBC driver and explain its types?

JDBC drivers are software that enables the Java application to connect with the database. For a better understanding, if you want to connect an individual database you need a driver for each database. JDBC drivers are very much like to ADO.NET data provider, OLE DB provider, ODBC driver. It has connections and classes that are very useful in Java application and its database.

It has four types and these are: -

- **JDBC-ODBC bridge driver**:
  This bridge is primarily used for the connection of ODBC driver to the database. As the name suggests, this driver converts JDBC method calls into ODBC calls. This driver is highly efficient as it is easy to use. This can be connected to any of the databases very easily. This driver is currently disheartened because of the thin wire
- **Native-API driver or partially java driver**:
  This driver is far better than the JDBC-ODBC driver with respect to performance. This driver utilizes the client-side libraries of the database. As the name suggests, it is partially written in java language. The native- API driver is used to convert the JDBC calls into native calls of the database API. The only quibble of this driver is that it is necessary to be installed on each client machine. It is the fastest driver of all of the drivers.

- **Network Protocol driver or fully java driver**:
  This driver is used to convert the JDBC calls into the vendor-specific database protocol directly or indirectly. It does not require any client-side library. This driver is able to perform many tasks such as logging, load balancing, auditing etc.
- **Thin driver or fully java driver**:
  As the name suggests, this driver converts the JDBC calls into the vendor-specific database protocol directly. It is more efficient in performance than those of other drivers rather than that is depends on the database.

Q3. What are the different types of interfaces in the API component?

The different interfaces are given below:

- **Connection:**
  getConnection() method is used to create this Connection object. DriverManager has a great number of connections.
- **Statement:**
  createStatement() method of connection class is used to create the Statement object. The Connection interface has a great number of Statement.
- **PreparedStatement:**
  prepareStatement() method of Connection classes used to create the PreparedStatement. This method is efficient in the performance of the parameterized query.
- **ResultSet:**
  Its object is to maintain and manage the cursor that points to a row of a table. ResultSet object is returned by the executeQuery() method of Statement interface.
- **ResultSetMetaData:**
  It contains a lot of important information which is about the data such as column type, column name, number of columns etc. ResultSetMetaData's object is returned by the getMetaData() of ResultSet.
- **DatabaseMetaData:**
  This interface is highly used to get metadata of a database. It provides information like database product version, driver name, database product name, the name of the total number of views, the name of the total no. of tables and much more. DatabaseMetaData's object is returned by the getMetaData() method of Connection interface.
- **CallableStatement:**
  It is used to call the reserved procedure and functions. Its performance is quite good because of the business logic on the database by using these reserved procedures and function. It can be done because these are precompiled. CallableStatement is returned by the prepareCall() method of Connection interface.

Q4. What are the different types of classes in the API component?

The classes are:

- **DriverManager:**
  The linkage between the user and driver is provided by the DriverManager. IT is used to keep track of the drivers which are available. It also performs the function of handling by building a relationship between

an appropriate driver and the database. This driver comprises many methods that are efficient to maintain the linkage between the drivers and the users.

- **Blob:**
  Blob servers as Binary Large OBject which signifies the factory of binary data stores as a single entity in the database management. Blobs are basically is the images, audio, multimedia objects, or may be executable code.
- **Club:**
  Club serves as Character large object. The main purpose of the Club is to store the character data files. Clob is used by several database management. The main difference between the clob and blob is that blob represents binary data while clob represent character files.
- **SQLException:**
  SQLException is an Exception class which provides information about the data access error.

Q5. What are the advantages of using PreparedStatement over Statement?

There are some advantages of the PreparedStatement over Statement which are given below: -

- The PreparedStatement's performance is far better than the Statement as it compiled once and then perform only on the runtime which is not like Statement which compiled every time at the time of running the code.
- Another important benefit of PreparedStatement is that it is able to execute the Parameterized query while Statement is able to run static queries.
- The query that is used in PreparedStatement is believed to similar every time. By this, a database is able to reuse the previous access plan while the Statement inlines the parameter into the string, Because of it, the query does not seem the same every time which obstruct the reuse of cache.

Q6. What is a lock in JDBC?

The lock is a type of software mechanism which is used as a restrictor to block the other users so that they can't use the data resource.

Q7. Explain different types of locks in JDBC?

Four types of locks are available in JDBC. These are: -

- **Row and Key Locks**: This kind of lock is used to update the rows.
- **Page Locks**: This kind of lock is used for a page. It is used in the case when a transaction remains in the process and continuously updated, inserting, deleting the data in a row of the table. Page locks are performed its action by locking the entire page. But it is only used once by the database users.
- **Table locks**: As the name suggests, Table locks are utilized to the table. This can be possibly done by the two methods that are shared and exclusive. A shared lock allows the other transactions to see the table but not let them update it. Nonetheless, the exclusive lock blocks others transaction from reading and writing the table.

- **Database locks**: It is the lock that is used when the database is open and it helps to prevent the read and update access from other transactions. This kind of lock blocks the other transaction to perform an operation when there is any database opened.

Q8. Differentiate between stored procedure and functions?

There are several differences between stored procedures and functions. These are: -

- Stored Procedure is very suitable to perform business logic while functions are utilized to do the calculation.
- Stored Procedure need not hold the return type while Function obligates the return type.
- A stored procedure can be returned 0 or more values while a function may state just one value.
- The Stored procedure upholds input and output parameters while the function upholds only input parameter.
- The stored procedure may be used try/catch for exception handling but it is not be used in a function.

Q9. Where JDBC DriverManager class is used?

- It is the main source of class by which it is easy to get the Database connection object. At the time of loading of JDBC Driver class, it automatically registered to DriverManager. To ensure it, Client can make check it on JDBC Driver classes.
- It is now possible to get utilized this registered driver to get the connection and make it useful for the caller programme. It can be so with the help of DriveManager.getConnection() method.

Q10. What do you mean by database connection pooling?

Connection pooling is storage of database connection in the cache. It is very useful for the reuse of connection in the future at the time of requirement of the database. It is also better for the executing commands as it increases its performance on a database.

Q11. What are the advantages of database connection pooling?

There are some advantages of the database connection pooling and these are: -

- The main advantage of using database connection pooling is that it is very fast.
- It is very easy to examine and diagnose the database connection.

Q12. What are the measures to connect to the DB using JDBC?

It can be done by two possible methods and these are: -

- Drive manager: With the help of DriverManager you can connect to the DB.
  - The driver class is loaded by making use of Class.forName() and class.forName.
  - It then transfers the control to DriverManager by post loading.
  - The next step is the creation of the connection and it is done by DriverManager.getConnection(). It is an important step to access the database.
- DataSource: It is another method to connect to DB using JDBC. In this case, it is not required to use DriverManager with the help of JNDI which will take care of the Data source from naming service server. To return Connection object to the DB, Datasource.getConnection() method is used.

Q13. What do you mean by JDBC Batch Processing and what are the advantages of using JDBC Batch Processing?

JDBC branching is used to send a bulk of SQL statement in just one call to the database. It enhances the performance as you forward A great number of a similar statement in a batch in a call and reducing the connection overhead. It can be done by the options such as Statement or PreparedStatement addBatch() and executeBatch() methods.

Advantages: -

- It is faster.
- JDBC API motivates batch processing.

Q14. What do you understand by JDBC DataSource?

JDBC DataSource is believed to be more powerful than the DriverManager in the case of the database connection. It is used to build a database connection. Drive implementation classes are very efficient to perform the job of getting the connection. It acts as an interface that is defined in the javax.sql package.

Q15. What are the advantages of using JDBC DataSource?

- It uses PreparedStatement for speedy and better processing
- It also provides connection timeout settings.
- It also provides features like logging.
- It has features like resultset maximum size threshold.
- It also supports connection Pooling in a servlet container with the help of JNDI support.

Q16. List some new features available in JDBC 4.0?

The new features that are available in JDBC 4.0 are: -

A new feature is available in JDBC 4.0 is automatically loading by JDBC driver class

- It increases Connection Management
- It allows RowId SQL.
- SQL implemented Dataset with the help of Annotations
- It also Improves SQL exception handling
- It also additional supports of files like SQL XML.

Q17. What are the basic exceptions in JDBC?

There are mainly four exceptions which are:

1. Java.sql.SQLWarning: This is the message popped for warning in SQL operations.
2. Java.sql.DataTruncations: This is the exception when the data value is unexpectedly other that the exceeded MaxFieldSize.
3. Java.sql.SQLException: It is the basic class for Java exceptions.
4. Java.sql.BatchUpdateException: When the Batch operation fails, this exception is thrown.

Please Visit OnlineInterviewquestions.com to download more pdfs