

# **JDBC Interview questions and answers**

-----

## **1. What do you mean by JDBC?**

**JDBC (Java Database Connectivity) is a Java API that enables Java applications to interact with databases. It allows developers to execute SQL queries, update databases,**

**retrieve results, and manage database connections.**

**2. What are the steps to connect to a database in Java?**

**To connect to a database in Java:**

**Load the database driver  
(Class.forName("driverName")).**

**Establish a connection using  
DriverManager.getConnection(url,  
username, password).**

**Create a Statement or  
PreparedStatement to execute  
SQL queries.**

**Execute the SQL query using `executeQuery()` or `executeUpdate()`.**

**Process the `ResultSet` (for `SELECT` queries).**

**Close the connection using `connection.close()`.**

**3. What are the JDBC API components?**

**JDBC API components include:**

**Driver: Provides a mechanism to establish a connection to the database.**

**Connection: Manages the connection to the database.**

**Statement: Allows you to execute SQL queries.**

**PreparedStatement: A precompiled SQL statement for improved performance.**

**ResultSet: Holds the result set of a query.**

**SQLException: Provides error and exception handling.**

**DriverManager: Manages the list of database drivers.**

**4. How do you load the drivers in JDBC?**

**You load the driver using  
Class.forName() method:**

**java**

**Copy code**

```
Class.forName("com.mysql.cj.jdbc  
.Driver");
```

**5. What is the purpose of JDBC  
ResultSet interface?**

**The ResultSet interface provides  
methods to retrieve data from a  
database query. It represents a set  
of rows returned by a SQL query,  
and you can iterate over it to  
extract data.**

## **6. What is JDBC**

**ResultSetMetaData interface?**

**ResultSetMetaData provides information about the structure of a ResultSet such as column names, column types, and the number of columns.**

## **7. What is JDBC**

**DatabaseMetaData interface?**

**DatabaseMetaData provides information about the database itself, such as the database**

**product name, version, supported SQL features, and available tables.**

**8. What do you mean by batch updates?**

**Batch updates allow you to group multiple SQL statements and execute them as a single batch, reducing the number of database round trips and improving performance.**

**9. Explain how you can establish a connection?**

**You can establish a connection using `DriverManager.getConnection()` by passing the URL of the database, username, and password:**

**java**

**Copy code**

```
Connection conn =  
DriverManager.getConnection("jd  
bc:mysql://localhost:3306/mydb",  
"username", "password");
```

**10. What are the different types of statements in JDBC?**



**There are three types of statements in JDBC:**

**Statement: Used for simple SQL queries without parameters.**

**PreparedStatement: Used for SQL queries with parameters, precompiled for performance.**

**CallableStatement: Used to execute stored procedures in the database.**

**11. Have you used prepared statements? Where have you used prepared statements?**

**Yes, PreparedStatement is used to execute parameterized SQL queries to prevent SQL injection and improve performance.**

**Example:**

**java**

**Copy code**

```
PreparedStatement pstmt =  
conn.prepareStatement("SELECT  
* FROM users WHERE username =  
?");  
  
pstmt.setString(1, "john_doe");  
  
ResultSet rs =  
pstmt.executeQuery();
```

## **12. How do you create JDBC statements?**

**Statements are created using the Connection object:**

**java**

**Copy code**

```
Statement stmt =  
conn.createStatement();
```

```
PreparedStatement pstmt =  
conn.prepareStatement("SELECT  
* FROM users WHERE id = ?");
```

```
CallableStatement cstmt =  
conn.prepareCall("{call  
my_procedure(?)}");
```

### **13. What is connection pooling?**

**Connection pooling is a technique where a pool of database connections is created and managed. These connections can be reused, reducing the overhead of opening and closing connections for each database operation.**

### **14. What is the difference between a Statement and a PreparedStatement?**

**Statement: Executes static SQL queries with no parameters, and is vulnerable to SQL injection.**

**PreparedStatement: Executes dynamic SQL queries with parameters, improving performance by precompiling the SQL statement.**

**15. What are callable statements?**

**CallableStatement is used to execute SQL stored procedures. It can handle input and output parameters and execute precompiled stored procedure logic.**

**16. What are types of JDBC drivers?**

**JDBC drivers are classified into four types:**

**Type-1: JDBC-ODBC Bridge driver.**

**Type-2: Native-API driver.**

**Type-3: Network Protocol driver.**

**Type-4: Thin driver (Pure Java driver).**

**17. How do you retrieve data from a result set?**

**You can retrieve data from a ResultSet using methods like getString(), getInt(), etc. Example:**

**java**

**Copy code**

```
while (rs.next()) {  
    String name =  
rs.getString("name");  
    int age = rs.getInt("age");  
}
```

**18. Explain about stored procedure?**

**A stored procedure is a precompiled SQL code that can be executed by the database server. It typically accepts input parameters and returns output or performs database operations.**

## **19. What are the different tasks of JDBC?**

### **Key tasks of JDBC:**

**Establishing a connection.**

**Executing SQL queries.**

**Retrieving results.**

**Handling exceptions and transactions.**

**Closing database resources.**

## **20. When do we look for batch updates?**



**Batch updates are useful when you have multiple similar SQL statements (e.g., multiple INSERT statements) and want to send them to the database in a single batch for improved performance.**

**21. When do we set `setAutoCommit(false)`?**

**`setAutoCommit(false)` is used when you want to manually manage transactions, ensuring that changes are committed only after executing a set of operations.**

## **22. What is connection pooling?**

**Connection pooling allows you to reuse database connections from a pool rather than opening and closing them repeatedly, improving performance and resource management.**

## **23. What is the return type of Class.forName() method?**

**The return type of Class.forName() is Class<?>, which represents the class object of the specified class name.**

## **24. How can we execute stored procedures using CallableStatement?**

**Example of executing a stored procedure:**

**java**

**Copy code**

```
CallableStatement stmt =  
conn.prepareCall("{call  
procedure_name(?, ?)}");  
stmt.setInt(1, value1);  
stmt.setString(2, value2);  
ResultSet rs =  
stmt.executeQuery();
```

**25. What is the role of the JDBC DriverManager class?**

**DriverManager manages a list of database drivers and establishes a connection to the database based on the available driver.**

**26. What are the functions of the JDBC Connection interface?**

**Functions include:**

**Establishing and closing connections.**

**Creating Statement, PreparedStatement, and**

**CallableStatement objects.**

**Managing transactions (commit, rollback).**

**Setting auto-commit mode.**

**27. Which interface is responsible for transaction management in JDBC?**

**The Connection interface manages transactions, providing methods like commit(), rollback(), and setAutoCommit().**

**28. What is batch processing and how to perform batch processing in JDBC?**

**Batch processing is the execution of multiple SQL statements in a single batch. Example:**

**java**

**Copy code**

```
Statement stmt =  
conn.createStatement();  
stmt.addBatch("INSERT INTO  
users (name) VALUES ('Alice')");  
stmt.addBatch("INSERT INTO  
users (name) VALUES ('Bob')");  
stmt.executeBatch();
```

**29. What are CLOB and BLOB data types in JDBC?**

**CLOB (Character Large Object):**  
Stores large text data.

**BLOB (Binary Large Object):**  
Stores large binary data like  
images or audio files.

**30. How can we store and retrieve  
images from the database?**

**Images can be stored as BLOB  
data types. Use**

**PreparedStatement to insert  
images and ResultSet to retrieve  
them as streams:**

**java**

**Copy code**

**// Storing**

```
PreparedStatement pstmt =  
conn.prepareStatement("INSERT  
INTO images (img) VALUES (?");  
pstmt.setBinaryStream(1, new  
FileInputStream("image.jpg"));  
pstmt.executeUpdate();
```

**java**

**Copy code**

**// Retrieving**

```
ResultSet rs =  
pstmt.executeQuery();  
  
InputStream in =  
rs.getBinaryStream("img");
```



**BufferedImage image =  
ImageIO.read(in);**

**31. What are the differences  
between stored procedure and  
functions?**

**Stored Procedure: Can perform  
actions (e.g., insert, update) and  
return multiple values.**

**Function: Typically used to return  
a single value and must always  
return a result.**

**32. What is the JDBC Rowset?**

**A RowSet is a wrapper around a  
ResultSet that makes it more  
portable and interactive,  
supporting features like**

**disconnected operation, scrolling, and update capabilities.**

**33. What is the major difference between java.util.Date and java.sql.Date data type?**

**java.util.Date: Includes both date and time.**

**java.sql.Date: Only includes the date portion (year, month, and day).**

**34. What does JDBC setMaxRows() method do?**

**The setMaxRows() method limits the number of rows a ResultSet**

**can return. It is useful when you only want to fetch a certain number of rows from a large result set.**