Topic Related Questions

**1.** **What is JDBC?**

**Ans:-**

JDBC (Java Database Connectivity) is a Java API that allows Java programs to connect to and interact with relational databases using SQL.

**2.** **What is PreparedStatement?**

**Ans:-**

PreparedStatement is a precompiled SQL statement in JDBC that allows parameterized queries, improving performance and preventing SQL injection.

**3.** **What is ActionListener?**

**Ans:-**

| **Feature** | **Statement** | **PreparedStatement** |
| --- | --- | --- |

|  |  |  |
| --- | --- | --- |
| **SQL Type** | Static SQL queries | Parameterized (dynamic) SQL queries |

|  |  |  |
| --- | --- | --- |
| **Compilation** | Compiled every time it’s executed | Precompiled once, reused multiple times |

|  |  |  |
| --- | --- | --- |
| **Performance** | Slower for repeated execution | Faster for repeated execution |

|  |  |  |
| --- | --- | --- |
| **Parameters** | No parameters; SQL built using string concatenation | Uses ? placeholders for parameters |

**4.** **How do you handle SQL exceptions?**

**Ans:**

Use try-catch blocks to catch SQLException and log or handle errors:

try { ... } catch (SQLException e) { e.printStackTrace(); }

**5.** **How to prevent SQL Injection?**

**Ans:-**

Use **PreparedStatement** or **stored procedures** instead of string concatenation in SQL queries, as they safely bind parameters.  
Also validate and sanitize all user inputs before executing queries.

**6.** **What is JDBC DriverManager?**

**Ans:-**

DriverManager manages a list of database drivers and establishes a connection to a database.

**7.** **How to close connections?**

**Ans:-**

Always close Connection, Statement, and ResultSet in a finally block or use **try-with-resources** to ensure they close automatically, preventing memory leaks.

**8.** **What is a ResultSet?**

**Ans:-**

ResultSet is an object that stores the result of a SQL query and allows navigation through the retrieved rows.

**9.** **What is auto-commit in JDBC?**

**Ans:-**

Auto-commit mode automatically commits each SQL statement after execution; can be disabled for transactions.

**10.** **How to connect Java to MySQL?**

**Ans:-**

Load MySQL driver and use DriverManager.getConnection() with URL, username, and password:

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/db","user","pass");