Parul® University

NAAC A++

Enterprise Programming using JAVA

Chapter-1: Foundation of Enterprise Programming

Prof. ARNIKA PATEL
Assistant Professor
Department of CSE

Parul® University

NAAC A++

Content

INDEX

Steps to Connect to MySQL Database Using JDBC

Step 1: Load the JDBC Driver

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

Step 2: Establish a Connection

```
Connection connection = DriverManager.getConnection(
"jdbc:mysql://localhost:3306/your_database",
"your_username",
"your_password"
);
```

Steps to Connect to MySQL Database Using JDBC

Step 3: Create a Statement

Statement statement = connection.createStatement();

Step 4: Execute a Query

String query = “INSERT INTO students (id, name) VALUES (101, ‘John Doe’)”;
int rowsAffected = statement.executeUpdate(query);
System.out.println(“Rows affected: ” + rowsAffected);

Steps to Connect to MySQL Database Using JDBC

Step 5: Close the Connection

statement.close();
connection.close();

Create a Simple JDBC Application

```
// Java program to implement a simple JDBC application
import java.sql.*;

public class Geeks {
public static void main(String[] args)
{
    // Database URL, username, and password

    // Replace with your database name
    String url
        = "jdbc:mysql://localhost:3306/your_database";
```

Create a Simple JDBC Application

```
// Replace with your MySQL username
String username = "your_username";

// Replace with your MySQL password
String password = "your_password";

// Updated query syntax for modern databases
String query
    = "INSERT INTO students (id, name) VALUES (109,
'bhatt')";
```

Create a Simple JDBC Application

```
// Establish JDBC Connection
        try {

                // Load Type-4 Driver
                // MySQL Type-4 driver class
                Class.forName("com.mysql.cj.jdbc.Driver");

                // Establish connection
                Connection c = DriverManager.getConnection(
                    url, username, password);
```

Create a Simple JDBC Application

```
// Create a statement
        Statement st = c.createStatement();

        // Execute the query
int count = st.executeUpdate(query);
System.out.println(
    "Number of rows affected by this query: "
    + count);
```

Create a Simple JDBC Application

```
// Close the connection
            st.close();
            c.close();
            System.out.println("Connection closed.");
        }
catch (ClassNotFoundException e) {
    System.err.println("JDBC Driver not found: "
                    + e.getMessage());
}
catch (SQLException e) {
    System.err.println("SQL Error: "
                    + e.getMessage());
}}}
```

PPT Content Resources Reference Sample:

1. **Book Reference**

   Jim Farley, William Crawford, David Flanagan. Java Enterprise in a Nutshell, O'Reilly

2. **Book Reference**

   Rocha, R., Purificação, J. (2018). Java EE 8 Design Patterns and Best Practices: Build Enterprise-ready Scalable Applications with Architectural Design Patterns. Germany: Packt Publishing..

3. **Website Reference**

   https://www.scribd.com/document/268349254/Java-8-Programming-Black-Book .

4. **Sources**

   https://developers.redhat.com/topics/enterprise-java

5. **Article**

   https://www.researchgate.net/publication/276412369_Advanced_Java_Programming

Parul® University | NAAC GRADE A++

https://paruluniversity.ac.in/