

```
Import java.sql.*;
```

```
Public class EventManagementSystem {
```

```
    Private static final String JDBC_URL = "jdbc:mysql://localhost:3306/event_management";
```

```
    Private static final String USERNAME = "your_username";
```

```
    Private static final String PASSWORD = "your_password";
```

```
    Public static void main(String[] args) {
```

```
        Try {
```

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

```
            Connection connection = DriverManager.getConnection(JDBC_URL, USERNAME, PASSWORD);
```

```
            // Create table if it doesn't exist
```

```
            createTable(connection);
```

```
            // Insert a new event
```

```
            insertEvent(connection, "Conference", "2023-06-15", "New York");
```

```
            // Retrieve and display all events
```

```
            displayEvents(connection);
```

```
            // Update an event
```

```
            updateEvent(connection, 1, "Conference", "2023-06-16", "San Francisco");
```

```
            // Delete an event
```

```
            deleteEvent(connection, 1);
```

```
            connection.close();
```

```

    } catch (Exception e) {
        e.printStackTrace();
    }
}

```

Private static void createTable(Connection connection) throws SQLException {

```

    String createTableQuery = "CREATE TABLE IF NOT EXISTS events (" +
        "id INT PRIMARY KEY AUTO_INCREMENT," +
        "name VARCHAR(100) NOT NULL," +
        "date DATE NOT NULL," +
        "location VARCHAR(100) NOT NULL" +
        ")";

```

```

    Statement statement = connection.createStatement();

```

```

    statement.executeUpdate(createTableQuery);

```

```

    statement.close();

```

```

}

```

Private static void insertEvent(Connection connection, String name, String date, String location) throws SQLException {

```

    String insertQuery = "INSERT INTO events (name, date, location) VALUES (?, ?, ?)";

```

```

    PreparedStatement preparedStatement = connection.prepareStatement(insertQuery);

```

```

    preparedStatement.setString(1, name);

```

```

    preparedStatement.setString(2, date);

```

```

    preparedStatement.setString(3, location);

```

```

    preparedStatement.executeUpdate();

```

```

    preparedStatement.close();

```

```

}

```

Private static void displayEvents(Connection connection) throws SQLException {

```

String selectQuery = "SELECT * FROM events";

Statement statement = connection.createStatement();

ResultSet resultSet = statement.executeQuery(selectQuery);

While (resultSet.next()) {
    Int id = resultSet.getInt("id");

    String name = resultSet.getString("name");

    String date = resultSet.getString("date");

    String location = resultSet.getString("location");


    System.out.println("Event ID: " + id);

    System.out.println("Name: " + name);

    System.out.println("Date: " + date);

    System.out.println("Location: " + location);

    System.out.println("-----");
}

resultSet.close();

statement.close();
}

Private static void updateEvent(Connection connection, int eventId, String name, String date, String
location) throws SQLException {

    String updateQuery = "UPDATE events SET name = ?, date = ?, location = ? WHERE id = ?";

    PreparedStatement preparedStatement = connection.prepareStatement(updateQuery);

    preparedStatement.setString(1, name);

    preparedStatement.setString(2, date);

    preparedStatement.setString(3, location);

    preparedStatement.setInt(4, eventId);

```

```
    preparedStatement.executeUpdate();  
    preparedStatement.close();  
}
```

```
Private static void deleteEvent(Connection connection, int eventId) throws SQLException {  
    String deleteQuery = "DELETE FROM events WHERE id = ?";  
    PreparedStatement preparedStatement = connection.prepareStatement(deleteQuery);  
    preparedStatement.setInt(1, eventId);  
    preparedStatement.executeUpdate();  
    preparedStatement.close();  
}  
}
```