Week 9 - LAQ's

Example of Executing a Stored Procedure Using SQLJ

To execute a stored procedure using SQLJ, you typically follow a structured approach that involves embedding SQL statements within your Java code. Below is an example illustrating how to call a stored procedure using SQLJ.

Step-by-Step Example

1. Define the Stored Procedure:

First, ensure that you have a stored procedure defined in your database. For example, let's create a simple stored procedure that retrieves employee details based on their ID.

```
CREATE PROCEDURE GetEmployeeDetails(IN empId INT)
BEGIN
   SELECT name, position FROM Employees WHERE id = empId;
END;
```

2. SQLJ Code to Execute the Stored Procedure:

In your Java application, you can use SQLJ to call this stored procedure. Here's how you would write the code:

```
import sqlj.runtime.ref.DefaultContext;

public class EmployeeDetails {
   public static void main(String[] args) {
      // Connection setup (assumes proper driver and connection details)
```

```
    DefaultContext context = new DefaultContext();

    // Declare host variables
    int empId = 101; // Example employee ID
    String name;
    String position;

    // Call the stored procedure using SQLJ
    #sql {
      CALL GetEmployeeDetails(:empId)
    };

    // Retrieve results into host variables
    #sql {
      SELECT name, position INTO :name, :position FROM Employees WHERE id = :empId
    };

    // Output the results
    System.out.println("Employee Name: " + name);
    System.out.println("Position: " + position);

    // Close the context (cleanup)
    context.close();
  }
```

```
}
```

Explanation of the Code

- Connection Setup: The DefaultContext is used to establish a connection to the database. Ensure that you have the necessary JDBC driver and connection parameters configured.

- Host Variables: The variables empId, name, and position are declared to hold the input parameter and output values.

- Calling the Stored Procedure: The #sql { CALL GetEmployeeDetails(:empId) } statement invokes the stored procedure, passing in the employee ID as an input parameter.

- Retrieving Results: After executing the stored procedure, another SQLJ statement retrieves the employee's name and position into the declared host variables.

- Output: Finally, the results are printed to the console.