Parameterized Queries:

Explain how to execute parameterized queries using JdbcTemplate in Spring. Provide examples of using named parameters (SqlParameterSource) and positional parameters (Object[]) to pass parameters to SQL queries.

Parameterized queries are a way to execute SQL queries with dynamic parameters in a secure and efficient manner. They help prevent SQL injection attacks by separating the SQL code from the data. In Spring, you can execute parameterized queries using the JdbcTemplate class.

Using Named Parameters (SqlParameterSource)

Named parameters are a way to pass parameters to a SQL query using a map of parameter names to values. You can use the SqlParameterSource interface to create a map of named parameters.

import org.springframework.jdbc.core.namedparam.NamedParameterJdbcTemplate;

import org.springframework.jdbc.core.namedparam.SqlParameterSource;

// Create a NamedParameterJdbcTemplate instance

NamedParameterJdbcTemplate jdbcTemplate = new NamedParameterJdbcTemplate(dataSource);

// Create a SqlParameterSource instance

SqlParameterSource parameters = new MapSqlParameterSource()

.addValue("username", "johnDoe")

.addValue("email", "johndoe@example.com");

// Execute the query

String query = "SELECT \* FROM users WHERE username = :username AND email = :email";

List<User> users = jdbcTemplate.queryForList(query, parameters);

In this example, we create a SqlParameterSource instance using a MapSqlParameterSource object, which is a implementation of SqlParameterSource. We add the parameter values to the map using the addValue() method. Then, we execute the query using the queryForList() method, passing the query and the SqlParameterSource instance as arguments.

Using Positional Parameters (Object[])

Positional parameters are a way to pass parameters to a SQL query using an array of values. You can use the JdbcTemplate class to execute queries with positional parameters.

import org.springframework.jdbc.core.JdbcTemplate;

// Create a JdbcTemplate instance

JdbcTemplate jdbcTemplate = new JdbcTemplate(dataSource);

// Execute the query

String query = "SELECT \* FROM users WHERE username =? AND email =?";

Object[] parameters = new Object[] { "johnDoe", "johndoe@example.com" };

List<User> users = jdbcTemplate.queryForList(query, parameters);