



DESARROLLO DE SOFTWARE I

Eric Gustavo Coronel Castillo

gcoronelc.blogspot.com

ecoronel@uch.edu.pe



TEMA: Procedimientos Almacenados



Ejecutar procedimientos almacenados utilizando Spring JDBC.





FUNDAMENTOS

Puedes usar cualquiera de los métodos **query** de JdbcTemplate o el método **update** para ejecutar el procedimiento almacenado, teniendo en cuenta si retorna o no un resultado.

Eric Gustavo Coronel Castillo

En caso que no retorne ningún resultado

```
jdbcTemplate.update("call proc (?, ?, ...)", parámetros);
```

En caso retorne una sola fila de datos

```
Map<String, Object> rec;  
rec = jdbcTemplate.queryForMap("call proc (?, ?, ...)", parámetros);
```

En caso retorne varias filas

```
List<Map<String, Object>> lista;  
lista = jdbcTemplate.queryForList("call proc (?, ?, ...)", parámetros);
```

FUNDAMENTOS

La clase **StoredProcedure** te brinda mayores opciones de control para ejecutar procedimientos almacenados.

Para implementarlo debes crear una clase que herede de **StoredProcedure**.

Eric Gustavo Coronel Castillo



Ejemplo

```
@Repository
public class ProceduresSaldoCuenta extends StoredProcedure {

    private static final String PROC_NAME = "usp_saldo_cuenta";

    @Autowired
    public ProceduresSaldoCuenta(JdbcTemplate jdbcTemplate) {
        super(jdbcTemplate, PROC_NAME);
        setFunction(false);
        declareParameter(new SqlParameter("p_cuenta", Types.VARCHAR));
        declareParameter(new SqlOutParameter("p_saldo", Types.DECIMAL));
        compile();
    }

    public double ejecutar(String cuenta) {
        Map<String, Object> rec = super.execute(cuenta);
        if (rec.get("p_saldo") == null) {
            throw new RuntimeException("Cuenta no existe.");
        }
        return Double.parseDouble(rec.get("p_saldo").toString());
    }
}
```

Eric Gustavo Coronel Castillo