



DESARROLLO DE SOFTWARE I

Eric Gustavo Coronel Castillo

gcoronelc.blogspot.com

ecoronel@uch.edu.pe

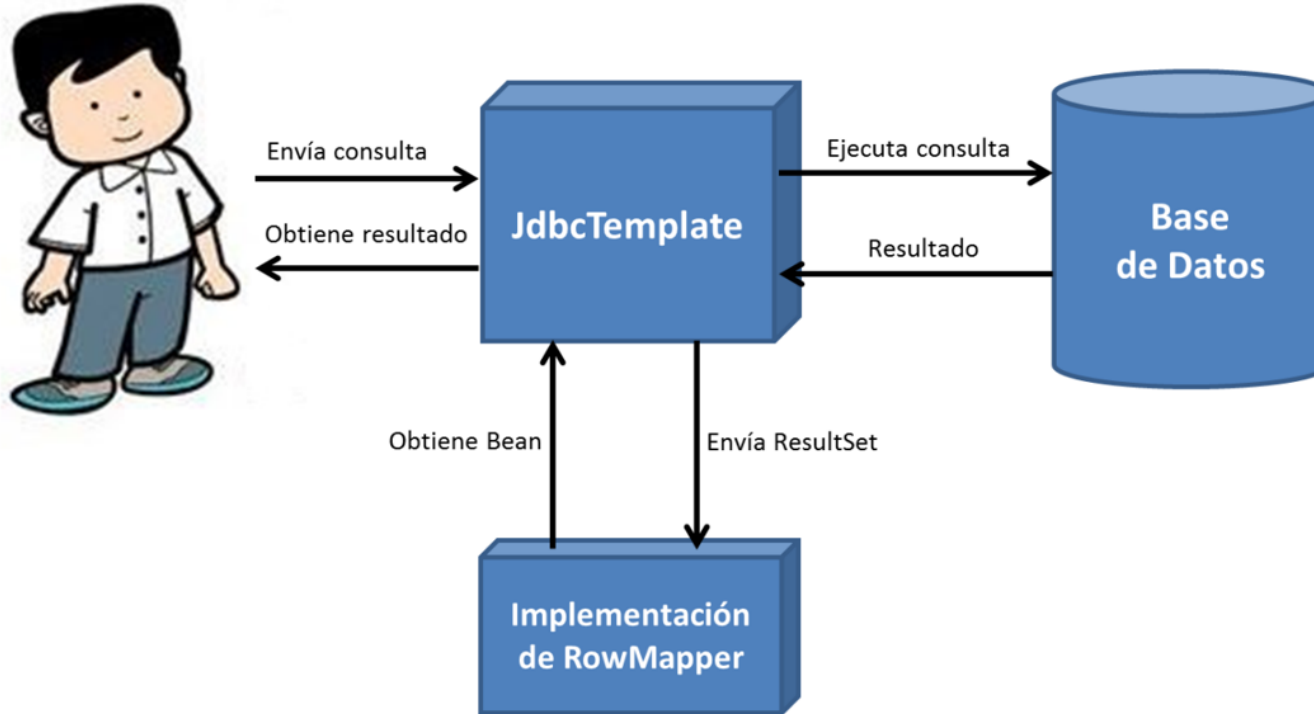


TEMA: Uso de RowMapper



Ejecutar consultas con
Spring JDBC usando
RowMapper.







Definición del bean

```
public class ClienteBean {  
  
    private String codigo;  
    private String paterno;  
    private String materno;  
    private String nombre;  
    private String dni;  
    private String ciudad;  
    private String direccion;  
    private String telefono;  
    private String email;  
  
    public ClienteBean () {  
    }  
  
    // Aquí debes crear los métodos getter y setter  
  
}
```



Implementación de RowMapper

```
public class ClienteMapper implements RowMapper<ClienteBean> {  
  
    public ClienteBean mapRow(ResultSet rs, int rowNum) throws SQLException {  
        Cliente clienteBean = new ClienteBean();  
        clienteBean.setCodigo(rs.getString("chr_cliecodigo"));  
        clienteBean.setPaterno(rs.getString("vch_cliepaterno"));  
        clienteBean.setMaterno(rs.getString("vch_cliematerno"));  
        clienteBean.setNombre(rs.getString("vch_clienombre"));  
        clienteBean.setDni(rs.getString("chr_cliedni"));  
        clienteBean.setCiudad(rs.getString("vch_clieciudad"));  
        clienteBean.setDireccion(rs.getString("vch_cliedireccion"));  
        clienteBean.setTelefono(rs.getString("vch_clietelefono"));  
        clienteBean.setEmail(rs.getString("vch_clieemail"));  
        return clienteBean;  
    }  
}
```



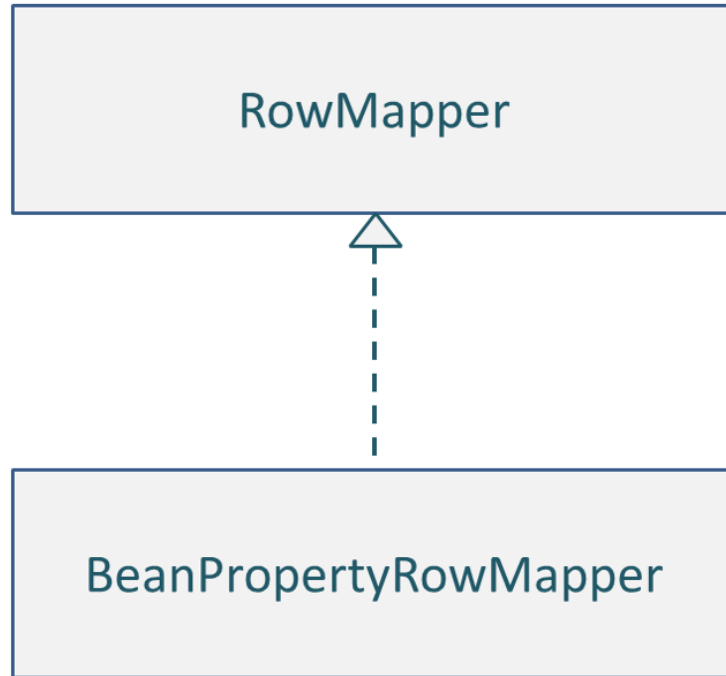
Consultando una fila

```
String sql = "select chr_cliecodigo, vch_cliepaterno, "  
    + "vch_cliematerno, vch_clienombre, chr_cliedni, "  
    + "vch_clieciudad, vch_cliedireccion, "  
    + "vch_clietelefono, vch_clieemail "  
    + "from cliente where chr_cliecodigo = ?";  
Object[] parms = {"00001"};  
ClienteBean clienteBean = jdbcTemplate.queryForObject(sql, parms,  
    new ClienteMapper() );  
System.out.println("Nombre: " + clienteBean.getNombre());  
System.out.println("Paterno: " + clienteBean.getPaterno());  
System.out.println("Materno: " + clienteBean.getMaterno());
```



Consultando varias filas

```
String sql = "select chr_emplcodigo, vch_emplpaterno, vch_emplmaterno, "  
            + "vch_emplnombre, vch_emplciudad, vch_empldireccion, "  
            + "vch_emplusuario from empleado where vch_emplpaterno like ?";  
Object[] parms = {"R%"};  
EmpleadoMapper mapper = new EmpleadoMapper();  
List<EmpleadoBean> lista = jdbcTemplate.query(sql, parms, mapper );  
  
for(EmpleadoBean empleadoBean: lista){  
    System.out.println( empleadoBean.getCodigo() + " "  
        + empleadoBean.getNombre() + " "  
        + empleadoBean.getPaterno() + " "  
        + empleadoBean.getMaterno() );  
}
```

Consultando una fila

```
String sql = "select "  
    + "chr_cliecodigo      codigo,"  
    + "vch_cliepaterno    paterno,"  
    + "vch_cliematerno    materno,"  
    + "vch_clienombre     nombre,"  
    + "chr_cliedni        dni,"  
    + "vch_clieciudad     ciudad,"  
    + "vch_cliedireccion  direccion,"  
    + "vch_clietelefono   telefono,"  
    + "vch_clieemail      email "  
    + "from cliente "  
    + "where chr_cliecodigo = ?";  
Object[] args = {codigo};  
ClienteBean clienteBean = jdbcTemplate.queryForObject(sql, args,  
    new BeanPropertyRowMapper(ClienteBean.class));
```

Consultando varias filas

```
String sql = "select "  
    + "chr_cliecodigo      codigo,"  
    + "vch_cliepaterno    paterno,"  
    + "vch_cliematerno    materno,"  
    + "vch_clienombre     nombre,"  
    + "chr_cliedni        dni,"  
    + "vch_clieciudad     ciudad,"  
    + "vch_cliedireccion  direccion,"  
    + "vch_clietelefono   telefono,"  
    + "vch_clieemail      email "  
    + "from cliente "  
    + "where vch_cliepaterno like ?";  
Object[] args = { "R%" };  
List<ClienteBean> lista;  
lista = jdbcTemplate.query(sql, args,  
    new BeanPropertyRowMapper(ClienteBean.class));
```