

- [SQL Queries Documentation \(Java Project\)](#)
  - [Introduction](#)
  - [1. Langileak \(Employees\)](#)
  - [2. Bezeroak \(Customers\)](#)
  - [3. Produktuak \(Products\)](#)
  - [4. Eskaerak \(Orders\)](#)
  - [5. Konponketak \(Repairs\)](#)
  - [6. General Infrastructure Queries](#)
- Generated by Antigravity AI on 2026-02-12

# SQL Queries Documentation (Java Project)

---

This document provides a comprehensive list of all SQL queries used within the Java application, extracted from `PreparedStatement` usages.

## Introduction

---

The Java application interacts with a MySQL database (`birtek_db`) using JDBC. Most queries are parameterized using `PreparedStatement` to ensure security and performance.

---

## 1. Langileak (Employees)

---

Operation	SQL Query	File
Login	<pre>SELECT * FROM langileak l JOIN langile_sailak s ON l.saila_id = s.id_saila WHERE eruptnan = ? AND pasahitza = ? AND aktibo = 1</pre>	SaioaHastekoPanela.java

Operation	SQL Query	File
<b>Update Own Data</b>	<pre>UPDATE langileak SET izena = ?, abizena = ?, jaiotza_data = ?, sexua = ?, herria_id = ?, posta_kodea = ?, telefonoa = ?, emaila = ?, pasahitza = ? WHERE eruptnan = ?</pre>	NireDatuakDialog.java
<b>Insert</b>	<pre>INSERT INTO langileak (eruptnan, izena, abizena, jaiotza_data, sexua, helbidea, herria_id, posta_kodea, telefonoa, emaila, hizkuntza, saila_id, pasahitza, aktibo) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)</pre>	AdministrariLangilea.java
<b>Update</b>	<pre>UPDATE langileak SET izena=?, abizena=?, jaiotza_data=?, sexua=?, helbidea=?, herria_id=?, posta_kodea=?, telefonoa=?, emaila=?, hizkuntza=?, saila_id=?, pasahitza=?, aktibo=? WHERE eruptnan=?</pre>	AdministrariLangilea.java
<b>Delete (Soft)</b>	<pre>UPDATE langileak SET aktibo = 0 WHERE eruptnan = ?</pre>	AdministrariLangilea.java
<b>Check Clock-in Status</b>	<pre>SELECT * FROM fitxaketak WHERE langilea_nan = ? ORDER BY data DESC, ordua DESC</pre>	Langilea.java
<b>All Employee Clock-ins</b>	<pre>SELECT f.id_fitxaketa, CONCAT(l.izena, ' ', l.abizena) AS langilea,</pre>	MenuAdministrazioa.java

Operation	SQL Query	File
	f.data, CAST(f.ordua AS CHAR) AS ordua, f.mota FROM fitxaketak f JOIN langileak l ON f.langilea_id = l.id_langilea ORDER BY f.id_fitxaketa DESC	
<b>List All</b>	SELECT * FROM	
<b>Departments</b>	langile_sailak	MenuAdministrazioa.java
<b>Delete</b>	DELETE FROM langile_sailak	
<b>Department</b>	WHERE id_saila = ?	MenuAdministrazioa.java

## 2. Bezeroak (Customers)

Operation	SQL Query	File
<b>Insert</b>	INSERT INTO bezeroak (izena_edo_soziala, abizena, ifz_nan, jaiotza_data, sexua, bezero_ordainketa_txartela, helbidea, herria_id, posta_kodea, telefona, emaila, hizkuntza, pasahitza, aktibo) VALUES (?, ?, , ?, ?, ?, ?, ?, ?, ?, ?, ?, , ?, ?)	SalmentaLangilea.java
<b>Update</b>	UPDATE bezeroak SET izena_edo_soziala=?, abizena=?, ifz_nan=?, jaiotza_data=?, sexua=?, bezero_ordainketa_txartela=?, helbidea=?, herria_id=?, posta_kodea=?, telefonoa=?, emaila=?, hizkuntza=?, pasahitza=?, aktibo=? WHERE id_bezeroa=?	SalmentaLangilea.java

Operation	SQL Query	File
Delete (Soft)	UPDATE bezeroak SET aktibo = 0 WHERE id_bezeroa = ?	SalmentaLangilea.java
List All	SELECT * FROM bezeroak	SalmentaLangilea.java
List Active	SELECT id_bezeroa, izena_edo_soziala FROM bezeroak WHERE aktibo = 1	EskaeraDialog.java

## 3. Produktuak (Products)

Operation	SQL Query	File
Complex Join	<pre>SELECT p.id_produktua, p.izena AS Produktua, b.izena AS Biltegia, s.id_sarrera AS 'Sarrera ID', sl.sarrera_lerro_egoera AS Egoera, s.data AS 'Sarrera Data', sl.id_sarrera_lerroa, p.produktu_egoera_oharra AS Oharra FROM sarrera_lerroak sl JOIN sarrerak s ON sl.sarrera_id = s.id_sarrera JOIN produktuak p ON sl.produktua_id = p.id_produktua JOIN biltegiak b ON p.biltegi_id = b.id_biltegia ORDER BY s.data DESC</pre>	MenuLogistika.java
Set Warehouse	UPDATE produktuak SET biltegi_id = ? WHERE id_produktua = ?	BiltegiLangilea.java
Update Price	UPDATE produktuak SET salmenta_presioa = ?, eskaintza = ? WHERE id_produktua = ?	TeknikariLangilea.java
Stock Check	SELECT id_produktua, izena, salmenta_presioa, stock,	EskaeraDialog.java

Operation	SQL Query	File
	eskaintza FROM produktuak WHERE salgai = 1	
Delete	DELETE FROM produktuak WHERE id_produktua = ?	TeknikariLangilea.java

## 4. Eskaerak (Orders)

Operation	SQL Query	File
Insert	INSERT INTO eskaerak (bezero_id, data, egoera, eskaera_presio_totala) VALUES (?, ?, ?, ?)	SalmentaLangilea.java
Update	UPDATE eskaerak SET bezero_id = ?, data = ?, egoera = ?, eskaera_presio_totala = ? WHERE id_eskaera = ?	SalmentaLangilea.java
List with Join	SELECT e.id_eskaera, b.izena_edo_soziala, e.data, e.eskaera_presio_totala, e.egoera FROM eskaerak e JOIN bezeroak b ON e.bezero_id = b.id_bezeroa WHERE e.egoera LIKE ?	BiltegiLangilea.java
Insert Line	INSERT INTO eskaera_lerroak (eskaera_id, produktua_id, kantitatea, unitate_presioa, eskaera_lerro_egoera) VALUES (?, ?, ?, ?, ?)	EskaeraLerroa.java
Update Line	UPDATE eskaera_lerroak SET eskaera_id=?, produktua_id=?, kantitatea=?, unitate_presioa=?,	EskaeraLerroa.java

Operation	SQL Query	File
	eskaera_lerro_egoera=? WHERE id_eskaera_lerroa=?	
<b>Delete Line</b>	DELETE FROM eskaera_lerroak WHERE id_eskaera_lerroa=?	EskaeraLerroa.java
<b>List Lines by Order</b>	SELECT * FROM eskaera_lerroak WHERE eskaera_id=?	EskaeraLerroa.java
<b>Get Completed Invoices</b>	SELECT id_eskaera FROM eskaerak WHERE eskaera_egoera = 'Osatua/Bidalita'	MenuSalmentak.java
<b>List Official Invoices</b>	SELECT e.id_eskaera AS id_faktura, e.faktura_zenbakia, CONCAT(e.id_eskaera, ' - ', b.izena_edo_soziala) AS eskaera, e.data, e.faktura_url AS fitxategia_url FROM eskaerak e JOIN bezeroak b ON e.bezeroa_id = b.id_bezeroa WHERE e.faktura_zenbakia IS NOT NULL AND e.faktura_zenbakia != '' ORDER BY e.id_eskaera DESC	MenuAdministrazioa.java

## 5. Konponketak (Repairs)

Operation	SQL Query	File
<b>Insert</b>	INSERT INTO konponketak (produktua_id, teknikaria_id, hasiera_data, amaiera_data, konponketa_egoera, akatsa_id, oharrak) VALUES	TeknikariLangilea.java

	<b>Operation</b>	<b>SQL Query</b>	<b>File</b>
		(?, ?, ?, ?, ?, ?, ?, ?)	
<b>Update Details</b>		UPDATE konponketak SET konponketa_egoera = ?, akatsa_id = ?, oharrak = ? WHERE id_konponketa = ?	KonponketaXehetasunaElkarritzeta.java
<b>List All</b>		SELECT * FROM konponketak	MenuTeknikoa.java

## 6. General Infrastructure Queries

	<b>Table</b>	<b>Operation</b>	<b>SQL Query</b>
<b>Herriak</b>		<b>Insert</b>	INSERT INTO herriak (izena, lurrealdea, nazioa) VALUES (?, ?, ?)
<b>Fitxaketak</b>		<b>Insert</b>	INSERT INTO fitxaketak (langilea_nan, mota, data, ordua) VALUES (?, ?, ?, ?)
<b>Biltegiak</b>		<b>Insert</b>	INSERT INTO biltegiak (izena, biltegi_sku) VALUES (?, ?)
<b>Sarrerak</b>		<b>List Join</b>	SELECT s.id_sarrera, h.izena_soziala, s.data, s.egoera FROM sarrerak s JOIN hornitzaireak h ON s.hornitzairea_id = h.id_hornitzairea