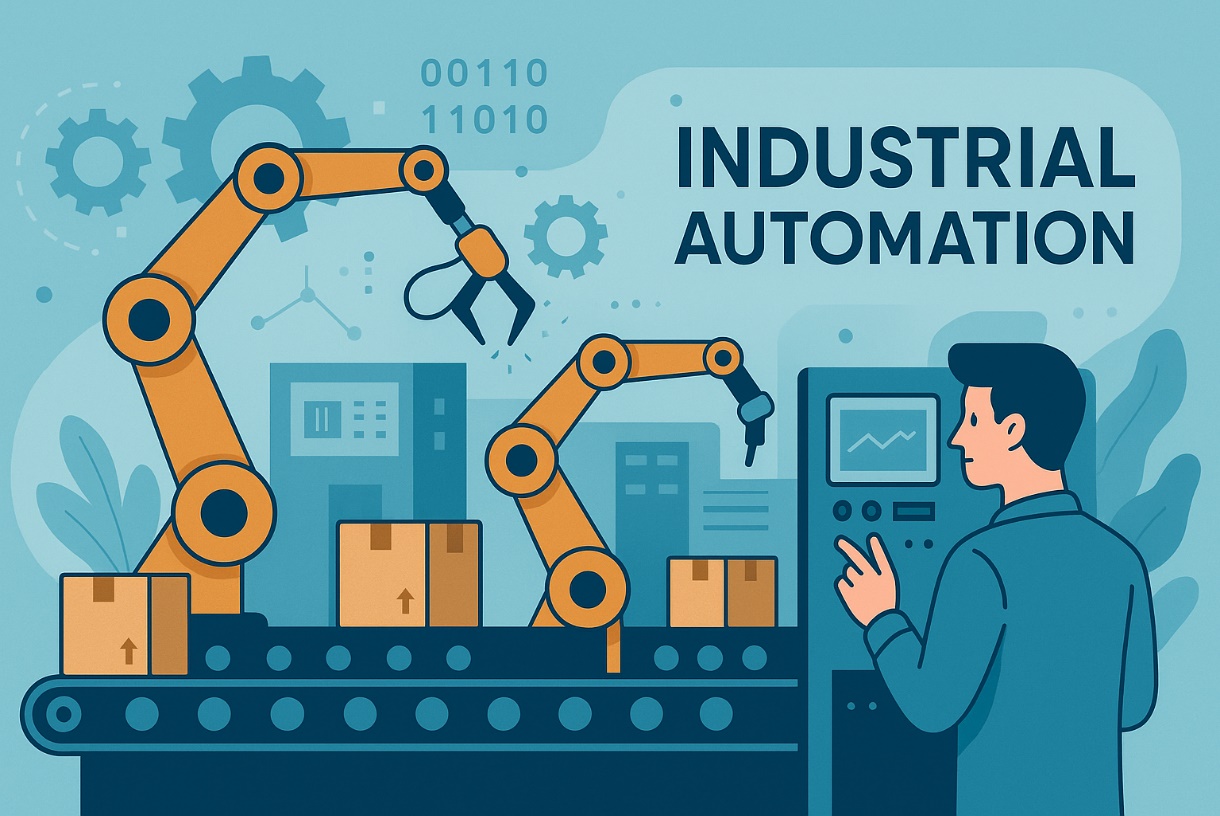
EMPRESA AUTOMATIZACIÓN INDUSTRIAL



***https://github.com/Xurxo777/Proyecto\_BBDD\_XurxoBargeBlanco.git***

**Nome Alumno/a: XURXO BARGE BLANCO**

***Xurxo barge blanco***

**Curso: *1º DAM* Materia:** ***Bases de Datos – Proyecto Final 24/25***

Contido

[1. Introducción 2](#_Toc199846372)

[2. Descripción del Problema / Requisitos 2](#_Toc199846373)

[3. Modelo Conceptual 3](#_Toc199846374)

[4. Modelo Relacional 4](#_Toc199846375)

[5. Proceso de Normalización 5](#_Toc199846376)

[6. Script de Creación de la Base de Datos 6](#_Toc199846377)

[7. Carga de Datos Inicial 16](#_Toc199846378)

[8. Funciones y Procedimientos Almacenados 25](#_Toc199846379)

[9. Triggers 32](#_Toc199846380)

[10. Consultas SQL 36](#_Toc199846381)

[11. Casos de Prueba y Simulación 44](#_Toc199846382)

[12. Resultados y Verificación 51](#_Toc199846383)

[13. Conclusiones y Mejoras Futuras 52](#_Toc199846384)

[14. Enlace al Repositorio en GitHub 53](#_Toc199846385)

# Introducción

O proxecto consiste nunha base de datos para xestionar os rexistros, datos de clientes y empregados y todo tipo de información de una empresa que se dedica a automatización industrial , facendo maquinas automatizadas, automatizando procesos en planta e ofrecendo un mantemento de estas ou de maquinas axenas a la empresa.

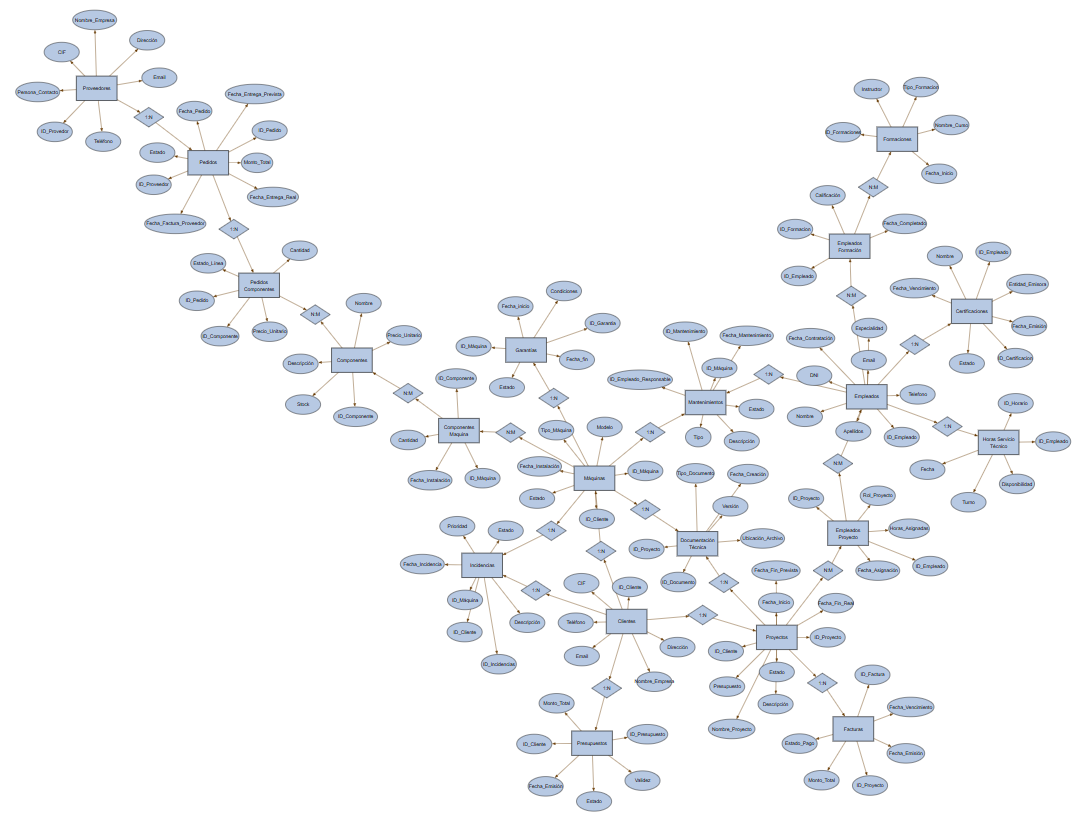
# Descripción del Problema / Requisitos

O principal problema e unha mellor xestión da empresa xa que precisa xestionar gran cantidade de datos.

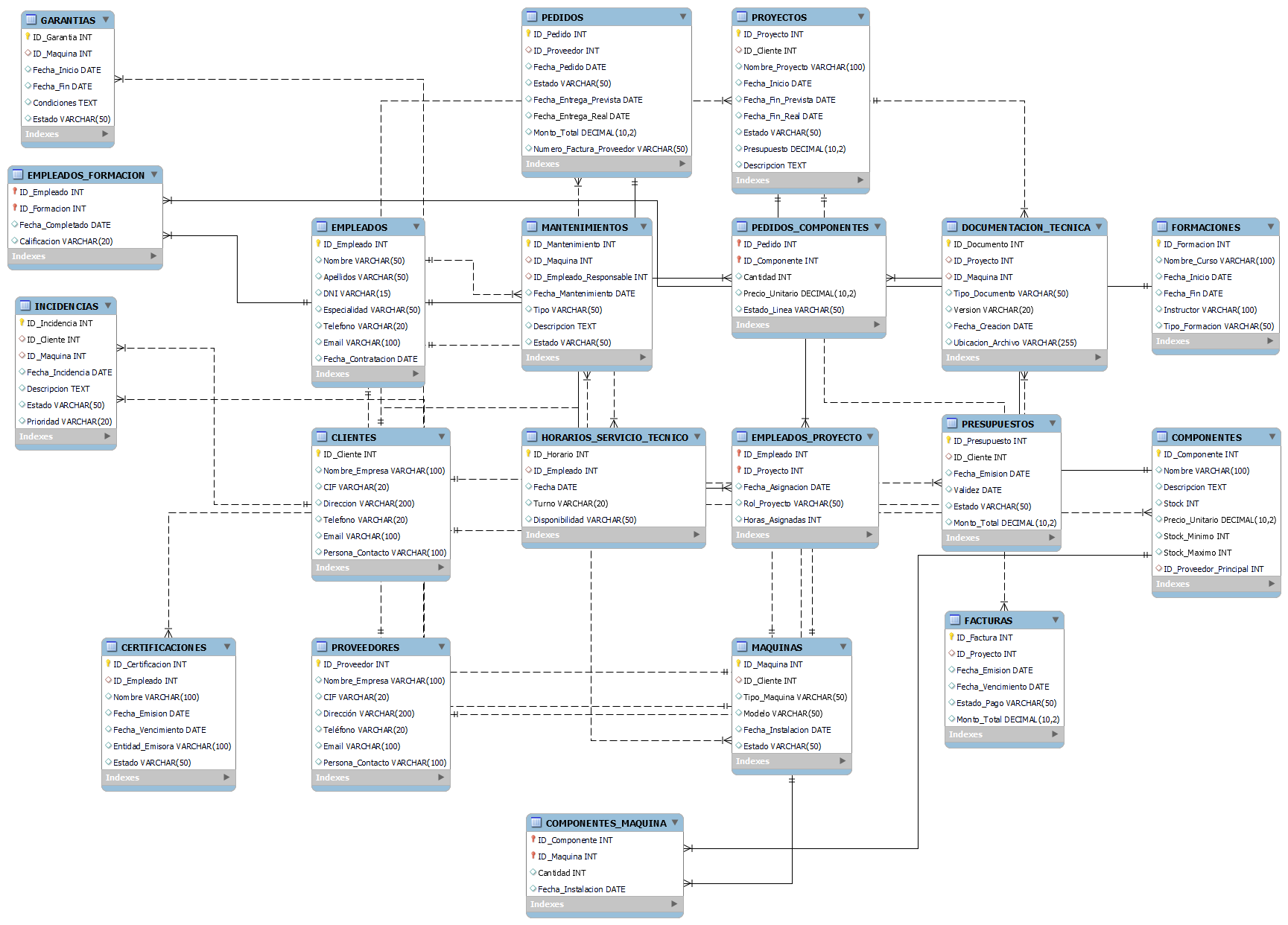
Características clave do negocio:

* Servicios de consultoría e diseño de solucions.
* Implementación de sistemas de control industrial.
* Programación de PLC’s e robots.
* Mantemento preventivo e correctivo.
* Capacitación do persoal do cliente.
* Servicio de soporte técnico 24/7.

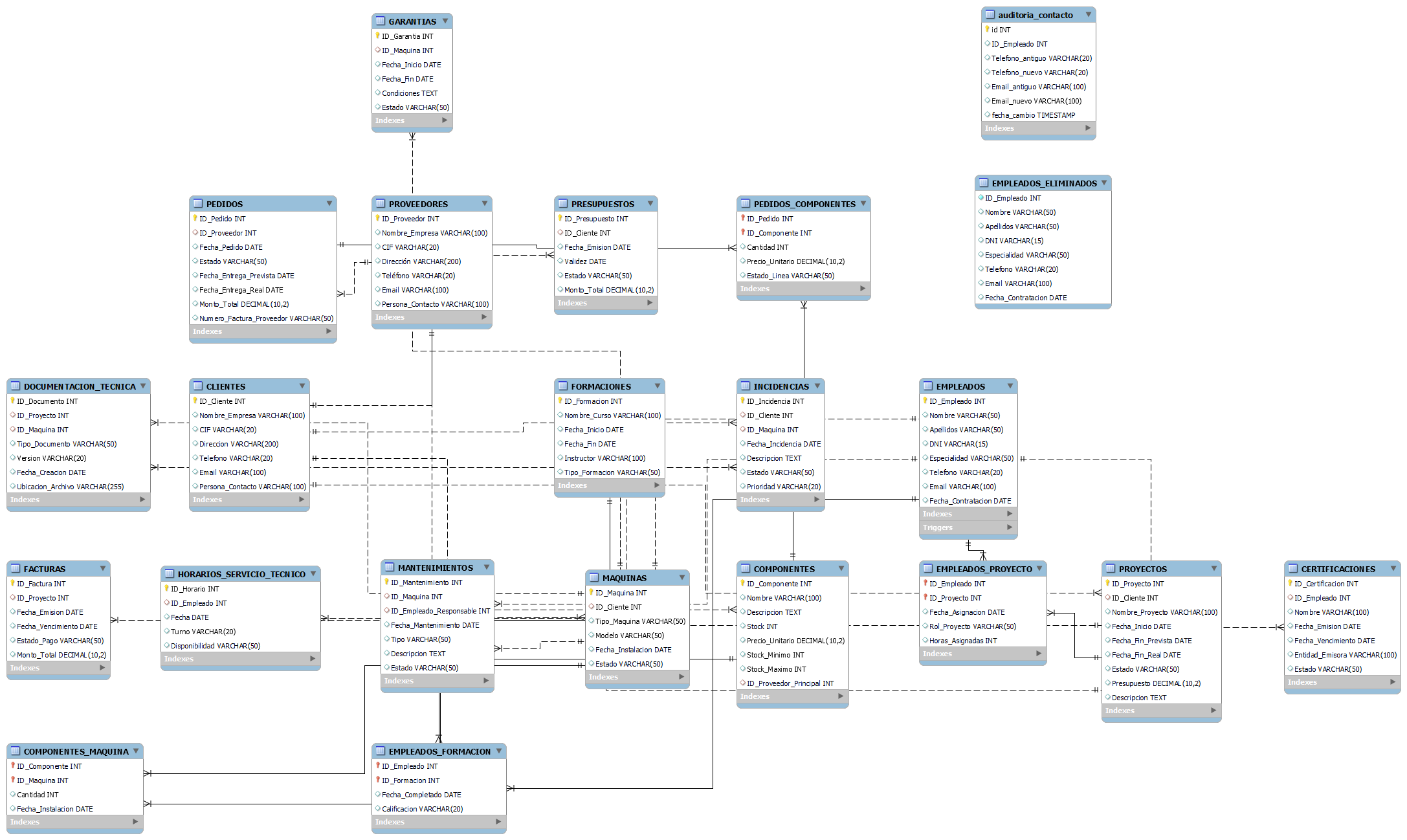
# Modelo Conceptual



# Modelo Relacional



En este modelo añadense 2 tablas inplementadas na parte de triggers.



# Proceso de Normalización

O proceso de normalización e unha técnica fundamental no diseño de bases de datos relacionais que buscan minimizar a redundancia de datos y asegurar a integridad da información. A través de diferentes formas normales (1FN, 2FN, 3FN...), garantizase que a estructura da base de datos sea eficiente, coherente e escalable.

En este proyecto realizouse unha análisis das seguientes relacions: CLIENTES, PROYECTOS, EMPLEADOS, EMPLEADOS\_PROYECTO, MÁQUINAS, COMPONENTES, COMPONENTES\_MÁQUINA, MANTENIMIENTOS, INCIDENCIAS, DOCUMENTACIÓN\_TÉCNICA, PRESUPUESTOS, FACTURAS, PROVEEDORES, PEDIDOS, PEDIDOS\_COMPONENTES, GARANTIAS, FORMACIONES, EMPLEADOS\_FORMACION, CERTIFICACIONES y HORARIOS\_SERVICIO\_TÉCNICO.

#### **1ª Forma Normal (1FN)**

Todas as tablas cumplen coa 1FN ao garantizar que cada atributo conten un valor atómico e indivisible, sen repeticions de grupos de datos nin listas. Esto permite que cada celda da tabla contenña únicamente unha valor.

#### **2ª Forma Normal (2FN)**

Ao ter todas as tablas unha clave primaria claramente definida e non existir dependencias parciales (e decir, que un atributo dependa solo dunha parte de unha clave composta), podemos afirmar que tamen se cumpre a 2FN. Aquelas tablas con claves compostas, como EMPLEADOS\_PROYECTO, solo teñen atributos que dependen da totalidad da clave.

#### **3ª Forma Normal (3FN)**

As relacions tamen cumpren coa 3FN, xa que no existen dependencias transitivas (un atributo non clave que dependa doutro atributo non clave). A información está correctamente descomposta entre entidades separadas, como PROVEEDORES, COMPONENTES o PROYECTOS, evitando duplicidades e redundancia innecesaria.

#### **Aplicación práctica**

A normalización permitiu organizar a base de datos nuha estructura clara donde:

* Os **clientes** e os **proyectos** están correctamente relacionados mediante claves externas.
* Os **empleados** poden participar en múltiples proxectos e formacions, mantendo a trazabilidad das suas actividades.
* As **máquinas**, **componentes** e **mantenimientos** están claramente separados pero interconectados, facilitando a sua xestión técnica.
* As **incidencias**, **presupuestos**, **facturas** y **pedidos** permiten un seguimento completo da operativa e da relación comercial coa clientela e proveedores.
* As **garantías**, **documentación técnica** e **certificaciones** están asociadas aos seus elementos correspondientes, garantizando o control de calidad e trazabilidad normativa.

# Script de Creación de la Base de Datos

**CREATE DATABASE AUTOMATIZACION\_INDUSTRIAL;**

* ***Comando de creación de la base de datos***

**USE AUTOMATIZACION\_INDUSTRIAL;**

* ***Comando de selección de la base de datos***

**CREATE TABLE CLIENTES (**

**ID\_Cliente INT PRIMARY KEY,**

**Nombre\_Empresa VARCHAR(100),**

**CIF VARCHAR(20),**

**Direccion VARCHAR(200),**

**Telefono VARCHAR(20),**

**Email VARCHAR(100),**

**Persona\_Contacto VARCHAR(100)**

**);**

* ***Comando para la creación de la tabla “Clientes ”, variables con tipo de variable acorde a la variable.***

**CREATE TABLE PROYECTOS (**

**ID\_Proyecto INT PRIMARY KEY,**

**ID\_Cliente INT,**

**Nombre\_Proyecto VARCHAR(100),**

**Fecha\_Inicio DATE,**

**Fecha\_Fin\_Prevista DATE,**

**Fecha\_Fin\_Real DATE,**

**Estado VARCHAR(50),**

**Presupuesto DECIMAL(10,2),**

**Descripcion TEXT,**

**FOREIGN KEY (ID\_Cliente) REFERENCES CLIENTES(ID\_Cliente)**

**ON DELETE CASCADE ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Proyectos ”, referenciada a la tabla clientes para saber los proyectos de cada cliente, uso de borrado y actualización en cascada.***

**CREATE TABLE EMPLEADOS (**

**ID\_Empleado INT PRIMARY KEY,**

**Nombre VARCHAR(50),**

**Apellidos VARCHAR(50),**

**DNI VARCHAR(15),**

**Especialidad VARCHAR(50),**

**Telefono VARCHAR(20),**

**Email VARCHAR(100),**

**Fecha\_Contratacion DATE**

**);**

* ***Comando para la creación de la tabla “Empleados ”.***

**CREATE TABLE EMPLEADOS\_PROYECTO (**

**ID\_Empleado INT,**

**ID\_Proyecto INT,**

**Fecha\_Asignacion DATE,**

**Rol\_Proyecto VARCHAR(50),**

**Horas\_Asignadas INT,**

**PRIMARY KEY (ID\_Empleado, ID\_Proyecto),**

**FOREIGN KEY (ID\_Empleado) REFERENCES EMPLEADOS(ID\_Empleado)**

**ON DELETE CASCADE ON UPDATE CASCADE,**

**FOREIGN KEY (ID\_Proyecto) REFERENCES PROYECTOS(ID\_Proyecto)**

**ON DELETE CASCADE ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Empleados\_Proyecto ”, referenciada a la tabla Empleados y referencia a la tabla Proyecto para saber empleados en cada proyecto , uso de borrado y actualización en cascada.***

**CREATE TABLE MAQUINAS (**

**ID\_Maquina INT PRIMARY KEY,**

**ID\_Cliente INT,**

**Tipo\_Maquina VARCHAR(50),**

**Modelo VARCHAR(50),**

**Fecha\_Instalacion DATE,**

**Estado VARCHAR(50),**

**FOREIGN KEY (ID\_Cliente) REFERENCES CLIENTES(ID\_Cliente)**

**ON DELETE CASCADE ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Maquinas ”, referenciada a la tabla clientes para saber las maquinas de los proyectos de cada cliente, uso de borrado y actualización en cascada.***

**CREATE TABLE COMPONENTES (**

**ID\_Componente INT PRIMARY KEY,**

**Nombre VARCHAR(100),**

**Descripcion TEXT,**

**Stock INT,**

**Precio\_Unitario DECIMAL(10,2),**

**Stock\_Minimo INT,**

**Stock\_Maximo INT,**

**ID\_Proveedor\_Principal INT,**

**FOREIGN KEY (ID\_Proveedor\_Principal) REFERENCES PROVEEDORES(ID\_Proveedor)**

**ON DELETE SET NULL ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Componentes ”, referenciada a la tabla provedor para saber a que provedor se adquirió cada pieza en caso de hacer una devolución, uso de actualización en cascada y eliminación dejando a null.***

**CREATE TABLE COMPONENTES\_MAQUINA (**

**ID\_Componente INT,**

**ID\_Maquina INT,**

**Cantidad INT,**

**Fecha\_Instalacion DATE,**

**PRIMARY KEY (ID\_Componente, ID\_Maquina),**

**FOREIGN KEY (ID\_Componente) REFERENCES COMPONENTES(ID\_Componente)**

**ON DELETE CASCADE ON UPDATE CASCADE,**

**FOREIGN KEY (ID\_Maquina) REFERENCES MAQUINAS(ID\_Maquina)**

**ON DELETE CASCADE ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Componentes\_Maquina ”, referenciada a la tabla Componentes para llevar un listado de los componentes de cada máquina, uso de borrado y actualización en cascada.***

**CREATE TABLE MANTENIMIENTOS (**

**ID\_Mantenimiento INT PRIMARY KEY,**

**ID\_Maquina INT,**

**ID\_Empleado\_Responsable INT,**

**Fecha\_Mantenimiento DATE,**

**Tipo VARCHAR(50),**

**Descripcion TEXT,**

**Estado VARCHAR(50),**

**FOREIGN KEY (ID\_Maquina) REFERENCES MAQUINAS(ID\_Maquina)**

**ON DELETE CASCADE ON UPDATE CASCADE,**

**FOREIGN KEY (ID\_Empleado\_Responsable) REFERENCES EMPLEADOS(ID\_Empleado)**

**ON DELETE SET NULL ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Mantenimientos ”, referenciada a la tabla Maquinas y Empleados para llevar un listado de los empleados que hacen un mantenimiento y a que máquina en concreto, uso de actualización en cascada y eliminación dejando a null.***

**CREATE TABLE INCIDENCIAS (**

**ID\_Incidencia INT PRIMARY KEY,**

**ID\_Cliente INT,**

**ID\_Maquina INT,**

**Fecha\_Incidencia DATE,**

**Descripcion TEXT,**

**Estado VARCHAR(50),**

**Prioridad VARCHAR(20),**

**FOREIGN KEY (ID\_Cliente) REFERENCES CLIENTES(ID\_Cliente)**

**ON DELETE SET NULL ON UPDATE CASCADE,**

**FOREIGN KEY (ID\_Maquina) REFERENCES MAQUINAS(ID\_Maquina)**

**ON DELETE SET NULL ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Incidencias ”, referenciada a la tabla Clientes y Máquinas para llevar listado a que cliente y máquinas corresponde cada incidencia, uso de actualización en cascada y eliminación dejando a null.***

**CREATE TABLE DOCUMENTACION\_TECNICA (**

**ID\_Documento INT PRIMARY KEY,**

**ID\_Proyecto INT,**

**ID\_Maquina INT,**

**Tipo\_Documento VARCHAR(50),**

**Version VARCHAR(20),**

**Fecha\_Creacion DATE,**

**Ubicacion\_Archivo VARCHAR(255),**

**FOREIGN KEY (ID\_Proyecto) REFERENCES PROYECTOS(ID\_Proyecto)**

**ON DELETE SET NULL ON UPDATE CASCADE,**

**FOREIGN KEY (ID\_Maquina) REFERENCES MAQUINAS(ID\_Maquina)**

**ON DELETE SET NULL ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Documentación\_Técnica ”, referenciada a la tabla Proyecto y Máquinas para llevar listado las documentaciones de los proyectos y sus máquinas, uso de actualización en cascada y eliminación dejando a null.***

**CREATE TABLE PRESUPUESTOS (**

**ID\_Presupuesto INT PRIMARY KEY,**

**ID\_Cliente INT,**

**Fecha\_Emision DATE,**

**Validez DATE,**

**Estado VARCHAR(50),**

**Monto\_Total DECIMAL(10,2),**

**FOREIGN KEY (ID\_Cliente) REFERENCES CLIENTES(ID\_Cliente)**

**ON DELETE SET NULL ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Presupuestos ”, referenciada a la tabla Clientes para saber a que cliente corresponde cada presupuesto de un proyecto, uso de actualización en cascada y eliminación dejando a null.***

**CREATE TABLE FACTURAS (**

**ID\_Factura INT PRIMARY KEY,**

**ID\_Proyecto INT,**

**Fecha\_Emision DATE,**

**Fecha\_Vencimiento DATE,**

**Estado\_Pago VARCHAR(50),**

**Monto\_Total DECIMAL(10,2),**

**FOREIGN KEY (ID\_Proyecto) REFERENCES PROYECTOS(ID\_Proyecto)**

**ON DELETE SET NULL ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Facturas ”, referenciada a la tabla Proyectos para guardar las facturas de los proyectos, uso de actualización en cascada y eliminación dejando a null.***

**CREATE TABLE PROVEEDORES (**

**ID\_Proveedor INT PRIMARY KEY,**

**Nombre\_Empresa VARCHAR(100),**

**CIF VARCHAR(20),**

**Dirección VARCHAR(200),**

**Teléfono VARCHAR(20),**

**Email VARCHAR(100),**

**Persona\_Contacto VARCHAR(100)**

**);**

* ***Comando para la creación de la tabla “Proveedores ”,***

**CREATE TABLE PEDIDOS (**

**ID\_Pedido INT PRIMARY KEY,**

**ID\_Proveedor INT,**

**Fecha\_Pedido DATE,**

**Estado VARCHAR(50),**

**Fecha\_Entrega\_Prevista DATE,**

**Fecha\_Entrega\_Real DATE,**

**Monto\_Total DECIMAL(10,2),**

**Numero\_Factura\_Proveedor VARCHAR(50),**

**FOREIGN KEY (ID\_Proveedor) REFERENCES PROVEEDORES(ID\_Proveedor)**

**ON DELETE SET NULL ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Proveedores ”, referenciada a la tabla Proveedor para saber quien es el vendedor de cada pedido, uso de actualización en cascada y eliminación dejando a null.***

**CREATE TABLE PEDIDOS\_COMPONENTES (**

**ID\_Pedido INT,**

**ID\_Componente INT,**

**Cantidad INT,**

**Precio\_Unitario DECIMAL(10,2),**

**Estado\_Linea VARCHAR(50),**

**PRIMARY KEY (ID\_Pedido, ID\_Componente),**

**FOREIGN KEY (ID\_Pedido) REFERENCES PEDIDOS(ID\_Pedido)**

**ON DELETE CASCADE ON UPDATE CASCADE,**

**FOREIGN KEY (ID\_Componente) REFERENCES COMPONENTES(ID\_Componente)**

**ON DELETE CASCADE ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Pedidos\_Componentes ”, referenciada a la tabla Pedidos y Componentes para saber que componentes pertenecen a cada pedido, uso de actualización y eliminación en cascada.***

**CREATE TABLE GARANTIAS (**

**ID\_Garantia INT PRIMARY KEY,**

**ID\_Maquina INT,**

**Fecha\_Inicio DATE,**

**Fecha\_Fin DATE,**

**Condiciones TEXT,**

**Estado VARCHAR(50),**

**FOREIGN KEY (ID\_Maquina) REFERENCES MAQUINAS(ID\_Maquina)**

**ON DELETE CASCADE ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Garantías ”, referenciada a la tabla Máquina para saber a que máquina le pertenece cada garantía o su estado, uso de actualización y eliminación en cascada.***

**CREATE TABLE FORMACIONES (**

**ID\_Formacion INT PRIMARY KEY,**

**Nombre\_Curso VARCHAR(100),**

**Fecha\_Inicio DATE,**

**Fecha\_Fin DATE,**

**Instructor VARCHAR(100),**

**Tipo\_Formacion VARCHAR(50)**

**);**

* ***Comando para la creación de la tabla “Formaciones ”,***

**CREATE TABLE EMPLEADOS\_FORMACION (**

**ID\_Empleado INT,**

**ID\_Formacion INT,**

**Fecha\_Completado DATE,**

**Calificacion VARCHAR(20),**

**PRIMARY KEY (ID\_Empleado, ID\_Formacion),**

**FOREIGN KEY (ID\_Empleado) REFERENCES EMPLEADOS(ID\_Empleado)**

**ON DELETE CASCADE ON UPDATE CASCADE,**

**FOREIGN KEY (ID\_Formacion) REFERENCES FORMACIONES(ID\_Formacion)**

**ON DELETE CASCADE ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Empleados\_Formación ”, referenciada a la tabla***

***Empleados y Formaciones para saber que formación tiene cada empleado de la empresa, uso de actualización y eliminación en cascada.***

**CREATE TABLE CERTIFICACIONES (**

**ID\_Certificacion INT PRIMARY KEY,**

**ID\_Empleado INT,**

**Nombre VARCHAR(100),**

**Fecha\_Emision DATE,**

**Fecha\_Vencimiento DATE,**

**Entidad\_Emisora VARCHAR(100),**

**Estado VARCHAR(50),**

**FOREIGN KEY (ID\_Empleado) REFERENCES EMPLEADOS(ID\_Empleado)**

**ON DELETE CASCADE ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Certificaciones ”, referenciada a la tabla Empleados para saber si la formación de estes esta certificada correctamente, uso de actualización y eliminación en cascada.***

**CREATE TABLE HORARIOS\_SERVICIO\_TECNICO (**

**ID\_Horario INT PRIMARY KEY,**

**ID\_Empleado INT,**

**Fecha DATE,**

**Turno VARCHAR(20),**

**Disponibilidad VARCHAR(50),**

**FOREIGN KEY (ID\_Empleado) REFERENCES EMPLEADOS(ID\_Empleado)**

**ON DELETE CASCADE ON UPDATE CASCADE**

**);**

* ***Comando para la creación de la tabla “Horarios\_Servicio\_Técnico ”, referenciada a la tabla Empleados para tener un seguimento de las horas de trabajo de cada técnico, uso de actualización y eliminación en cascada.***

# Carga de Datos Inicial

* ***Inserción de clientes: almacena datos básicos das empresas clientes da empresa de automatización, incluíndo contacto e dirección.***

**INSERT INTO CLIENTES (ID\_Cliente, Nombre\_Empresa, CIF, Direccion, Telefono, Email, Persona\_Contacto) VALUES**

**(1, 'Empresa 1', 'A00000001', 'Rúa Exemplo 1, Cidade', '+34 612345678', 'contacto1@empresa.com', 'Persoa 1'),**

**(2, 'Empresa 2', 'A00000002', 'Rúa Exemplo 2, Cidade', '+34 623456789', 'contacto2@empresa.com', 'Persoa 2'),**

**(3, 'Empresa 3', 'A00000003', 'Rúa Exemplo 3, Cidade', '+34 634567890', 'contacto3@empresa.com', 'Persoa 3'),**

**(4, 'Empresa 4', 'A00000004', 'Rúa Exemplo 4, Cidade', '+34 645678901', 'contacto4@empresa.com', 'Persoa 4'),**

**(5, 'Empresa 5', 'A00000005', 'Rúa Exemplo 5, Cidade', '+34 656789012', 'contacto5@empresa.com', 'Persoa 5'),**

**(6, 'Empresa 6', 'A00000006', 'Rúa Exemplo 6, Cidade', '+34 656789045', 'contacto6@empresa.com', 'Persoa 6'),**

**(7, 'Empresa 7', 'A00000007', 'Rúa Exemplo 7, Cidade', '+34 667890123', 'contacto7@empresa.com', 'Persoa 7'),**

**(8, 'Empresa 8', 'A00000008', 'Rúa Exemplo 8, Cidade', '+34 678901234', 'contacto8@empresa.com', 'Persoa 8');**

* ***Inserción de empregados: información dos empregados da empresa, co seu DNI, especialidade, datos de contacto e data de contratación.***

**INSERT INTO EMPLEADOS (ID\_Empleado, Nombre, Apellidos, DNI, Especialidad, Telefono, Email, Fecha\_Contratacion) VALUES**

**(1, 'Empregado1', 'Apelido1', '12345678X', 'Electrónica', '+34 600111222', 'empregado1@empresa.com', '2019-03-15'),**

**(2, 'Empregado2', 'Apelido2', '23456789Y', 'Mecánica', '+34 600222333', 'empregado2@empresa.com', '2018-07-10'),**

**(3, 'Empregado3', 'Apelido3', '34567890Z', 'Programación', '+34 600333444', 'empregado3@empresa.com', '2020-11-01'),**

**(4, 'Empregado4', 'Apelido4', '45678901W', 'Robótica', '+34 600444555', 'empregado4@empresa.com', '2021-02-20'),**

**(5, 'Empregado5', 'Apelido5', '56789012V', 'Electrónica', '+34 600555666', 'empregado5@empresa.com', '2022-09-05'),**

**(6, 'Empregado6', 'Apelido6', '56789012M', 'Gestión', '+34 600545666', 'empregado6@empresa.com', '2027-06-02'),**

**(7, 'Empregado7', 'Apelido7', '67890123N', 'Automatización', '+34 600666777', 'empregado7@empresa.com', '2023-05-10'),**

**(8, 'Empregado8', 'Apelido8', '78901234P', 'Electrónica', '+34 600777888', 'empregado8@empresa.com', '2020-10-01');**

* ***Inserción de provedores: empresas que proveen compoñentes ou servizos, cos seus datos de contacto e persoas de referencia.***

**INSERT INTO PROVEEDORES (ID\_Proveedor, Nombre\_Empresa, CIF, Dirección, Teléfono, Email, Persona\_Contacto) VALUES**

**(1, 'Proveedor 1', 'P0000001', 'Rúa Proveedor 1', '+34 912345678', 'contacto@proveedor1.com', 'Contacto 1'),**

**(2, 'Proveedor 2', 'P0000002', 'Rúa Proveedor 2', '+34 923456789', 'contacto@proveedor2.com', 'Contacto 2'),**

**(3, 'Proveedor 3', 'P0000003', 'Rúa Proveedor 3', '+34 934567890', 'contacto@proveedor3.com', 'Contacto 3'),**

**(4, 'Proveedor 4', 'P0000004', 'Rúa Proveedor 4', '+34 945678901', 'contacto@proveedor4.com', 'Contacto 4'),**

**(5, 'Proveedor 5', 'P0000005', 'Rúa Proveedor 5', '+34 956789012', 'contacto@proveedor5.com', 'Contacto 5');**

* ***Inserción de compoñentes: detalles dos compoñentes dispoñibles en stock, prezo, stock mínimo/máximo e provedores principais.***

**INSERT INTO COMPONENTES (ID\_Componente, Nombre, Descripcion, Stock, Precio\_Unitario, Stock\_Minimo, Stock\_Maximo, ID\_Proveedor\_Principal) VALUES**

**(1, 'Componente A', 'Componente para control', 50, 25.50, 10, 100, 1),**

**(2, 'Componente B', 'Componente eléctrico', 30, 12.75, 5, 50, 2),**

**(3, 'Componente C', 'Sensor de temperatura', 40, 15.90, 10, 80, 3),**

**(4, 'Componente D', 'Relé de control industrial', 20, 9.80, 5, 40, 1),**

**(5, 'Componente E', 'Motor paso a paso', 15, 45.00, 3, 20, 4),**

**(6, 'Componente F', 'Pantalla HMI 7"', 10, 120.00, 2, 15, 5),**

**(7, 'Componente G', 'Variador de frecuencia', 8, 220.50, 2, 10, 3),**

**(8, 'Componente H', 'Interruptor de fin de carreira', 25, 6.50, 5, 30, 2),**

**(9, 'Componente I', 'Placa Arduino Mega', 35, 32.00, 10, 50, 4),**

**(10, 'Componente J', 'Módulo Wi-Fi ESP32', 60, 8.25, 15, 100, 5),**

**(11, 'Componente K', 'Fonte de alimentación 24V', 18, 55.30, 5, 25, 1),**

**(12, 'Componente L', 'Encoder rotativo óptico', 12, 27.80, 3, 20, 2);**

* ***Inserción de máquinas: máquinas instaladas nos clientes, co tipo, modelo, data de instalación e estado operativo.***

**INSERT INTO MAQUINAS (ID\_Maquina, ID\_Cliente, Tipo\_Maquina, Modelo, Fecha\_Instalacion, Estado) VALUES**

**(1, 1, 'Robot Industrial', 'Model X', '2023-01-15', 'Operativa'),**

**(2, 2, 'CNC', 'Model Y', '2022-11-10', 'Mantenimiento'),**

**(3, 3, 'Impresora 3D', 'Delta Pro', '2023-05-22', 'Operativa'),**

**(4, 1, 'PLC Siemens', 'S7-1200', '2021-09-30', 'Operativa'),**

**(5, 4, 'Brazo Robótico', 'UR5e', '2022-02-14', 'Mantenimiento'),**

**(6, 5, 'Sistema SCADA', 'WinCC Advanced', '2023-08-01', 'Operativa'),**

**(7, 2, 'Fresadora CNC', 'Model Z', '2020-12-20', 'Averiada'),**

**(8, 3, 'Robot Colaborativo', 'Cobot A1', '2024-01-10', 'Operativa'),**

**(9, 4, 'Controlador PID', 'Omron E5CC', '2022-06-18', 'Mantenimiento'),**

**(10, 5, 'Sistema de Visión', 'Cognex In-Sight', '2023-03-03', 'Operativa');**

* ***Inserción de proxectos: proxectos que a empresa está a levar a cabo para os clientes, con datas, estado, orzamento e descrición.***

**INSERT INTO PROYECTOS (ID\_Proyecto, ID\_Cliente, Nombre\_Proyecto, Fecha\_Inicio, Fecha\_Fin\_Prevista, Fecha\_Fin\_Real, Estado, Presupuesto, Descripcion) VALUES**

**(1, 1, 'Automatización Línea 1', '2024-01-10', '2024-06-10', NULL, 'En Progreso', 15000.00, 'Proyecto de instalación de robots en línea 1'),**

**(2, 2, 'Actualización Máquina CNC', '2023-09-05', '2024-03-01', '2024-02-28', 'Finalizado', 8000.00, 'Actualización de software y hardware CNC'),**

**(3, 3, 'Integración SCADA', '2024-02-01', '2024-07-01', NULL, 'En Progreso', 12000.00, 'Integración de SCADA en planta de produción'),**

**(4, 4, 'Instalación Brazo Robótico', '2023-06-15', '2023-12-15', '2023-12-20', 'Finalizado', 9500.00, 'Instalación e posta en marcha de robot colaborativo');**

* ***Relación entre empregados e proxectos: asignación de empregados a proxectos con datas, rol e horas asignadas.***

**INSERT INTO EMPLEADOS\_PROYECTO (ID\_Empleado, ID\_Proyecto, Fecha\_Asignacion, Rol\_Proyecto, Horas\_Asignadas) VALUES**

**(1, 1, '2024-01-15', 'Jefe de Proyecto', 120),**

**(2, 1, '2024-02-01', 'Técnico', 100),**

**(3, 2, '2023-09-10', 'Programador', 90),**

**(4, 3, '2024-02-05', 'Programador', 100),**

**(5, 4, '2023-06-20', 'Instalador', 80);**

* ***Mantementos realizados ou pendentes sobre máquinas, indicando quen é responsable, tipo, data e estado.***

**INSERT INTO MANTENIMIENTOS (ID\_Mantenimiento, ID\_Maquina, ID\_Empleado\_Responsable, Fecha\_Mantenimiento, Tipo, Descripcion, Estado) VALUES**

**(1, 1, 1, '2024-03-15', 'Preventivo', 'Revisión trimestral del robot', 'Realizado'),**

**(2, 2, 2, '2024-04-01', 'Correctivo', 'Reparación de fallo en eje', 'Pendiente'),**

**(3, 5, 5, '2024-05-10', 'Correctivo', 'Substitución de pezas desgastadas', 'Realizado'),**

**(4, 6, 6, '2024-05-18', 'Preventivo', 'Revisión mensual sistema SCADA', 'Pendiente');**

* ***Incidencias rexistradas para clientes e máquinas, con data, descrición, estado e prioridade.***

**INSERT INTO INCIDENCIAS (ID\_Incidencia, ID\_Cliente, ID\_Maquina, Fecha\_Incidencia, Descripcion, Estado, Prioridad) VALUES**

**(1, 1, 1, '2024-04-10', 'Fallo no sistema de control', 'Abierta', 'Alta'),**

**(2, 2, 2, '2024-04-12', 'Problema coa alimentación eléctrica', 'Cerrada', 'Media'),**

**(3, 3, 3, '2024-05-15', 'Problemas de calibrado', 'Abierta', 'Media'),**

**(4, 5, 6, '2024-05-19', 'Erro de conexión coa rede', 'Abierta', 'Alta');**

* ***Documentación técnica asociada a proxectos e máquinas, incluíndo tipo de documento, versión, data e ubicación do arquivo.***

**INSERT INTO DOCUMENTACION\_TECNICA (ID\_Documento, ID\_Proyecto, ID\_Maquina, Tipo\_Documento, Version, Fecha\_Creacion, Ubicacion\_Archivo) VALUES**

**(1, 1, 1, 'Manual Técnico', 'v1.0', '2024-01-05', '/docs/manual\_robot\_x.pdf'),**

**(2, 2, 2, 'Plano Mecánico', 'v2.1', '2023-08-20', '/docs/plano\_cnc\_y.pdf'),**

**(3, 3, 6, 'Manual SCADA', 'v1.1', '2024-02-05', '/docs/manual\_scada.pdf'),**

**(4, 4, 5, 'Esquema Eléctrico', 'v3.0', '2023-06-16', '/docs/esquema\_robot.pdf');**

* ***Presupostos emitidos a clientes, con data, validez, estado (aprobado, rexeitado) e importe total.***

**INSERT INTO PRESUPUESTOS (ID\_Presupuesto, ID\_Cliente, Fecha\_Emision, Validez, Estado, Monto\_Total) VALUES**

**(1, 1, '2023-12-01', '2024-03-01', 'Aprobado', 15000.00),**

**(2, 2, '2023-08-15', '2023-11-15', 'Rechazado', 8000.00);**

* ***Facturas xeradas por proxectos, con datas de emisión e vencemento, estado do pago e importe.***

**INSERT INTO FACTURAS (ID\_Factura, ID\_Proyecto, Fecha\_Emision, Fecha\_Vencimiento, Estado\_Pago, Monto\_Total) VALUES**

**(1, 1, '2024-03-01', '2024-04-01', 'Pagada', 15000.00),**

**(2, 2, '2024-01-15', '2024-02-15', 'Pendiente', 8000.00),**

**(3, 3, '2024-05-10', '2024-06-10', 'Pendiente', 12000.00),**

**(4, 4, '2023-12-21', '2024-01-21', 'Pagada', 9500.00);**

* ***Pedidos feitos a provedores, con datas, estado do pedido, datas previstas e reais de entrega, importe total e número de factura do provedor.***

**INSERT INTO PEDIDOS (ID\_Pedido, ID\_Proveedor, Fecha\_Pedido, Estado, Fecha\_Entrega\_Prevista, Fecha\_Entrega\_Real, Monto\_Total, Numero\_Factura\_Proveedor) VALUES**

**(1, 1, '2024-03-20', 'Entregado', '2024-04-05', '2024-04-04', 1275.00, 'F2024-001'),**

**(2, 2, '2024-04-01', 'En Proceso', '2024-04-20', NULL, 650.00, 'F2024-002'),**

**(3, 3, '2024-04-15', 'Entregado', '2024-04-30', '2024-04-29', 330.00, 'F2024-003'),**

**(4, 4, '2024-05-01', 'Pendiente', '2024-05-20', NULL, 890.00, 'F2024-004');**

* ***Detalle dos compoñentes incluídos en cada pedido, con cantidade, prezo unitario e estado da liña.***

**INSERT INTO PEDIDOS\_COMPONENTES (ID\_Pedido, ID\_Componente, Cantidad, Precio\_Unitario, Estado\_Linea) VALUES**

**(1, 1, 50, 25.50, 'Recibido'),**

**(2, 2, 30, 12.75, 'Pendiente'),**

**(3, 3, 20, 15.90, 'Recibido'),**

**(4, 4, 10, 9.80, 'Pendiente');**

* ***Garantías asociadas a máquinas, con datas de inicio e fin, condicións e estado da garantía.***

**INSERT INTO GARANTIAS (ID\_Garantia, ID\_Maquina, Fecha\_Inicio, Fecha\_Fin, Condiciones, Estado) VALUES**

**(1, 1, '2023-01-15', '2025-01-14', 'Garantía completa por dos anos', 'Activa'),**

**(2, 2, '2022-11-10', '2024-11-09', 'Garantía limitada por piezas', 'Expirada'),**

**(3, 3, '2024-01-10', '2026-01-10', 'Completa', 'Inclúe pezas e man de obra'),**

**(4, 5, '2023-07-01', '2025-07-01', 'Limitada', 'Só pezas principais');**

* ***Formacións dispoñibles, co nome, datas, instructor e tipo (técnica, seguridade, etc.).***

**INSERT INTO FORMACIONES (ID\_Formacion, Nombre\_Curso, Fecha\_Inicio, Fecha\_Fin, Instructor, Tipo\_Formacion) VALUES**

**(1, 'Formación en Robótica Industrial', '2023-09-01', '2023-09-15', 'Ing. Pérez', 'Técnica'),**

**(2, 'Seguridad en el Trabajo', '2024-01-10', '2024-01-20', 'Dr. López', 'Seguridad');**

* ***Relación entre empregados e formacións, indicando data de finalización e calificación obtida.***

**INSERT INTO EMPLEADOS\_FORMACION (ID\_Empleado, ID\_Formacion, Fecha\_Completado, Calificacion) VALUES**

**(1, 1, '2023-09-15', 'Aprobado'),**

**(2, 2, '2024-01-20', 'Sobresaliente');**

* ***Certificacións obtidas polos empregados, con nome, datas de emisión e vencemento, entidade emisora e estado.***

**INSERT INTO CERTIFICACIONES (ID\_Certificacion, ID\_Empleado, Nombre, Fecha\_Emision, Fecha\_Vencimiento, Entidad\_Emisora, Estado) VALUES**

**(1, 1, 'Certificación de Programación PLC', '2023-06-01', '2026-06-01', 'Entidad Certificadora A', 'Vigente'),**

**(2, 2, 'Certificación en Mantenimiento Industrial', '2022-11-15', '2025-11-15', 'Entidad Certificadora B', 'Vigente');**

* ***Horarios asignados aos empregados para o servizo técnico, con datas, turno e dispoñibilidade.***

**INSERT INTO HORARIOS\_SERVICIO\_TECNICO (ID\_Horario, ID\_Empleado, Fecha, Turno, Disponibilidad) VALUES**

**(1, 1, '2024-05-01', 'Mañana', 'Disponible'),**

**(2, 2, '2024-05-01', 'Tarde', 'No Disponible');**

**INSERT INTO COMPONENTES\_MAQUINA (ID\_Componente, ID\_Maquina, Cantidad, Fecha\_Instalacion) VALUES**

**(1, 1, 4, '2023-01-15'), *-- Componente A no Robot Industrial Model X***

**(2, 2, 6, '2022-11-10'), *-- Componente B na CNC Model Y***

**(3, 3, 2, '2023-05-22'), *-- Componente C na Impresora 3D Delta Pro***

**(4, 4, 3, '2021-09-30'), *-- Componente D no PLC Siemens S7-1200***

**(5, 5, 1, '2022-02-14'), *-- Componente E no Brazo Robótico UR5e***

**(6, 6, 1, '2023-08-01'), *-- Componente F no Sistema SCADA WinCC Advanced***

**(7, 7, 2, '2020-12-20'), *-- Componente G na Fresadora CNC Model Z***

**(8, 8, 5, '2024-01-10'), *-- Componente H no Robot Colaborativo Cobot A1***

**(9, 9, 2, '2022-06-18'), *-- Componente I no Controlador PID Omron E5CC***

**(10, 10, 3, '2023-03-03'),** ***-- Componente J no Sistema de Visión Cognex In-Sight***

**(11, 1, 1, '2023-01-15'),** ***-- Componente K tamén no Robot Industrial Model X***

**(12, 2, 1, '2022-11-10'); *-- Componente L tamén na CNC Model Y***

# Funciones y Procedimientos Almacenados

* **FUNCIONES ALMACENADAS**
  + ***Calculo de antiguedad en la empresa desde año de contratación introducido por teclado;***

**DELIMITER //**

**CREATE FUNCTION calcular\_antiguedad(p\_fecha DATE)**

**RETURNS INT**

**DETERMINISTIC**

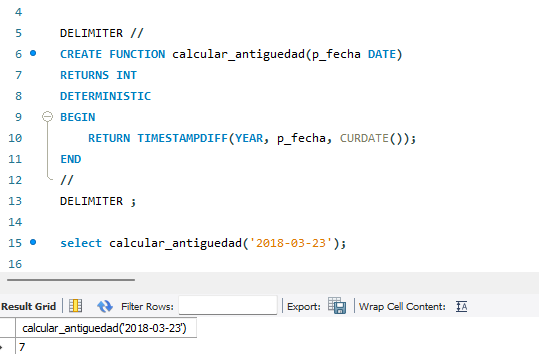
**BEGIN**

**RETURN TIMESTAMPDIFF(YEAR, p\_fecha, CURDATE());**

**END**

**//**

**DELIMITER ;**

****

* + ***Creación de el nombre completo de los empleados con su nombre y su apellido***

**DELIMITER //**

**CREATE FUNCTION nombre\_completo(p\_nombre VARCHAR(50), p\_apellidos VARCHAR(50))**

**RETURNS VARCHAR(101)**

**DETERMINISTIC**

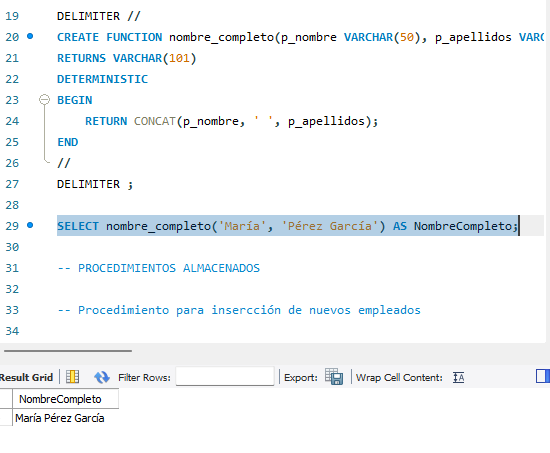
**BEGIN**

**RETURN CONCAT(p\_nombre, ' ', p\_apellidos);**

**END**

**//**

**DELIMITER ;**

****

* + ***Función que devolve un resumo da información dun empregado.***

**DELIMITER //**

**CREATE FUNCTION resumen\_empleado(p\_ID INT)**

**RETURNS TEXT**

**DETERMINISTIC**

**BEGIN**

**DECLARE v\_nombre VARCHAR(50);**

**DECLARE v\_apellidos VARCHAR(50);**

**DECLARE v\_email VARCHAR(100);**

**DECLARE v\_fecha DATE;**

**DECLARE v\_antiguedad INT;**

**DECLARE v\_resultado TEXT;**

**SELECT Nombre, Apellidos, Email, Fecha\_Contratacion**

**INTO v\_nombre, v\_apellidos, v\_email, v\_fecha**

**FROM EMPLEADOS**

**WHERE ID\_Empleado = p\_ID;**

**SET v\_antiguedad = TIMESTAMPDIFF(YEAR, v\_fecha, CURDATE());**

**SET v\_resultado = CONCAT('Empregado: ', v\_nombre, ' ', v\_apellidos,**

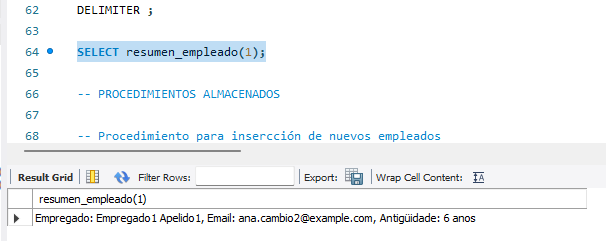
**', Email: ', v\_email,**

**', Antigüidade: ', v\_antiguedad, ' anos');**

**RETURN v\_resultado;**

**END //**

**DELIMITER ;**

****

* **PROCEDIMIENTOS ALMACENADOS**
  + ***Procedimiento para insercción de nuevos empleados***

**DELIMITER //**

**CREATE PROCEDURE insertar\_empleado(**

**IN p\_ID\_Empleado INT,**

**IN p\_Nombre VARCHAR(50),**

**IN p\_Apellidos VARCHAR(50),**

**IN p\_DNI VARCHAR(15),**

**IN p\_Especialidad VARCHAR(50),**

**IN p\_Telefono VARCHAR(20),**

**IN p\_Email VARCHAR(100),**

**IN p\_Fecha\_Contratacion DATE**

**)**

**BEGIN**

**INSERT INTO EMPLEADOS (ID\_Empleado, Nombre, Apellidos, DNI, Especialidad, Telefono, Email, Fecha\_Contratacion)**

**VALUES (p\_ID\_Empleado, p\_Nombre, p\_Apellidos, p\_DNI, p\_Especialidad, p\_Telefono, p\_Email, p\_Fecha\_Contratacion);**

**END**

**//**

**DELIMITER ;**

****

****

* + ***Procedimiento para actualizar el contacto de un empleado***

**DELIMITER //**

**CREATE PROCEDURE actualizar\_contacto(**

**IN p\_ID\_Empleado INT,**

**IN p\_Telefono VARCHAR(20),**

**IN p\_Email VARCHAR(100)**

**)**

**BEGIN**

**UPDATE EMPLEADOS**

**SET Telefono = p\_Telefono,**

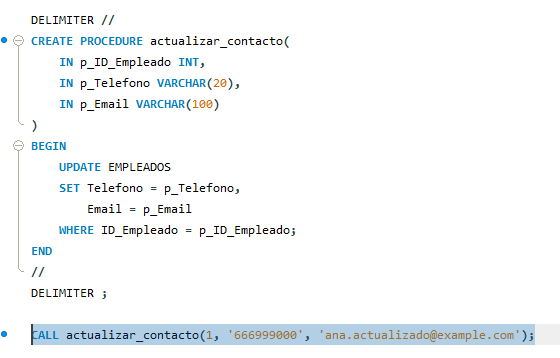
**Email = p\_Email**

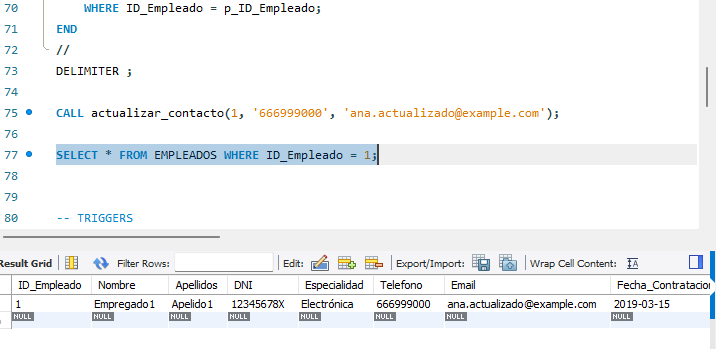
**WHERE ID\_Empleado = p\_ID\_Empleado;**

**END**

**//**

**DELIMITER ;**

****

****

* + ***Procedemento almacenado******que elimina empregados con máis de X anos de antigüidade (con backup previo).***

**CREATE TABLE IF NOT EXISTS EMPLEADOS\_ELIMINADOS AS**

**SELECT \* FROM EMPLEADOS WHERE 1=0;**

**DELIMITER //**

**CREATE PROCEDURE eliminar\_empregados\_antigos(IN p\_anhos INT)**

**BEGIN**

**DECLARE v\_data\_limite DATE;**

**SET v\_data\_limite = DATE\_SUB(CURDATE(), INTERVAL p\_anhos YEAR);**

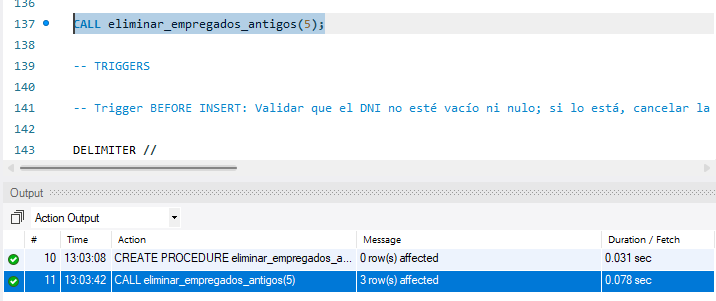
**INSERT INTO EMPLEADOS\_ELIMINADOS**

**SELECT \* FROM EMPLEADOS WHERE Fecha\_Contratacion < v\_data\_limite;**

**DELETE FROM EMPLEADOS WHERE Fecha\_Contratacion < v\_data\_limite;**

**END //**

**DELIMITER ;**

**^**

****

# Triggers

**-- TRIGGERS**

**-- Trigger BEFORE INSERT: Validar que el DNI no esté vacío ni nulo; si lo está, cancelar la inserción.**

**DELIMITER //**

**CREATE TRIGGER validar\_DNI\_before\_insert**

**BEFORE INSERT ON EMPLEADOS**

**FOR EACH ROW**

**BEGIN**

**IF NEW.DNI IS NULL OR NEW.DNI = '' THEN**

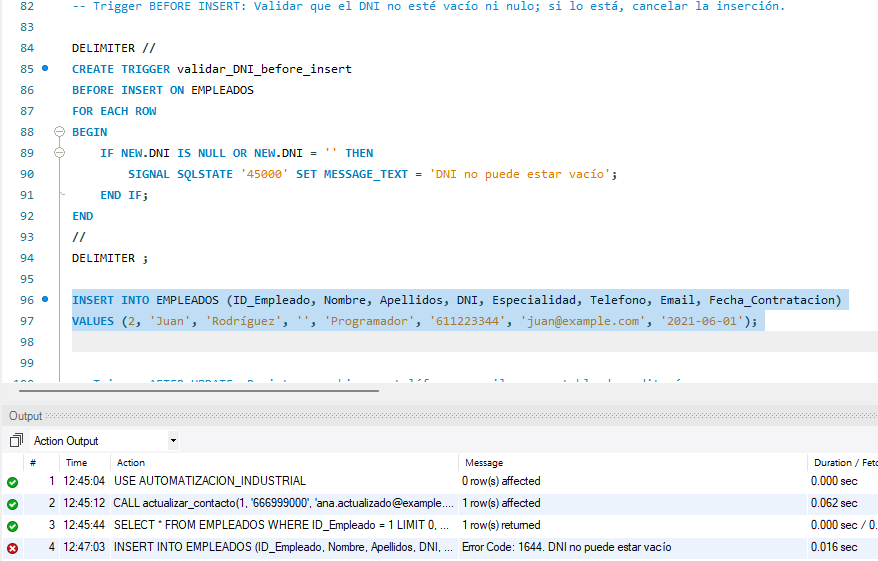
**SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'DNI no puede estar vacío';**

**END IF;**

**END**

**//**

**DELIMITER ;**

****

**-- Trigger AFTER UPDATE: Registrar cambios en teléfono o email en una tabla de auditoría**

**CREATE TABLE auditoria\_contacto (**

**id INT AUTO\_INCREMENT PRIMARY KEY,**

**ID\_Empleado INT,**

**Telefono\_antiguo VARCHAR(20),**

**Telefono\_nuevo VARCHAR(20),**

**Email\_antiguo VARCHAR(100),**

**Email\_nuevo VARCHAR(100),**

**fecha\_cambio TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

**DELIMITER //**

**CREATE TRIGGER auditar\_cambios\_contacto\_after\_update**

**AFTER UPDATE ON EMPLEADOS**

**FOR EACH ROW**

**BEGIN**

**IF OLD.Telefono <> NEW.Telefono OR OLD.Email <> NEW.Email THEN**

**INSERT INTO auditoria\_contacto (ID\_Empleado, Telefono\_antiguo, Telefono\_nuevo, Email\_antiguo, Email\_nuevo)**

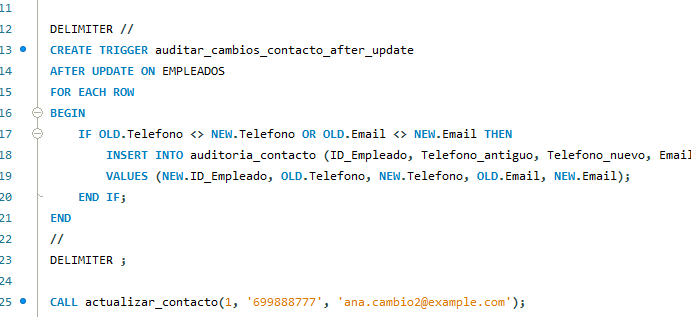
**VALUES (NEW.ID\_Empleado, OLD.Telefono, NEW.Telefono, OLD.Email, NEW.Email);**

**END IF;**

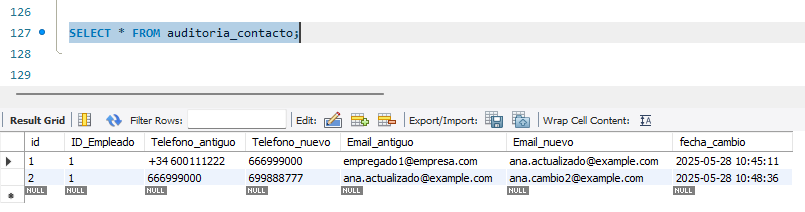
**END**

**//**

**DELIMITER ;**

****

****

****

* + ***Un trigger que evita eliminar empregados da especialidade “Director”.***

**DELIMITER //**

**CREATE TRIGGER evitar\_eliminar\_director**

**BEFORE DELETE ON EMPLEADOS**

**FOR EACH ROW**

**BEGIN**

**IF OLD.Especialidad = 'Director' THEN**

**SIGNAL SQLSTATE '45000'**

**SET MESSAGE\_TEXT = 'Non se pode eliminar un Director.';**

**END IF;**

**END //**

**DELIMITER ;**

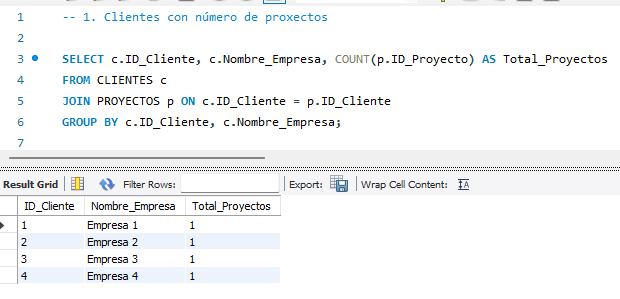
****

* + Como debería ser ao intentar eliminar o empleado coa categoria “Director”

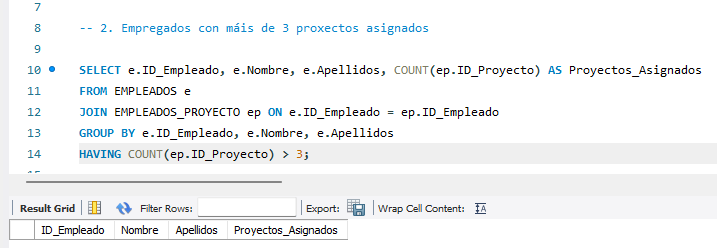
lanzase unha mensaxe que nos di que non e posible a accion e non a executa

# Consultas SQL

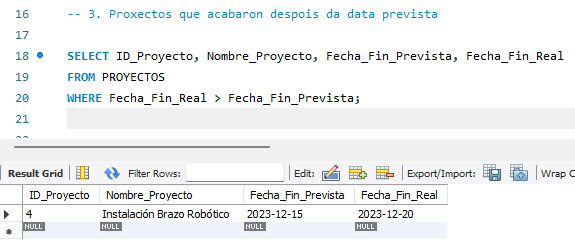
Listar os clientes o nome da sua empresa e o numero total de proxectos de cada cliente.



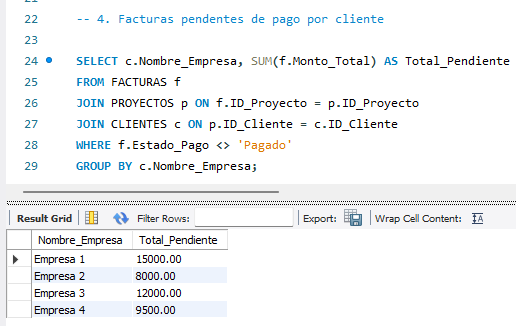
Listar os empregados con mais de 3 proyectos asignados (en este caso nadie ten mais de 3 proxectos asigando).



Listar os proxectos que por retrasos ou outros casos acabaron despois da fecha prevista.

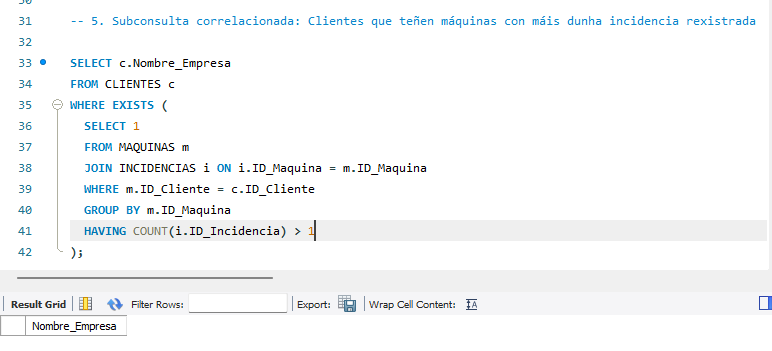


Listar as empresas que teñen pendente o pago.

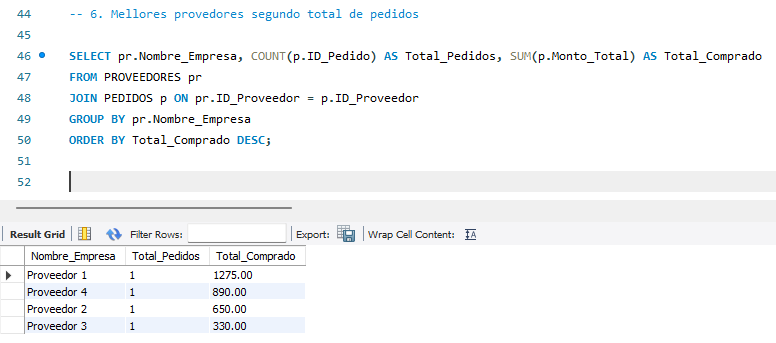


Polo que parece a empresa non ten presa polos cobros.

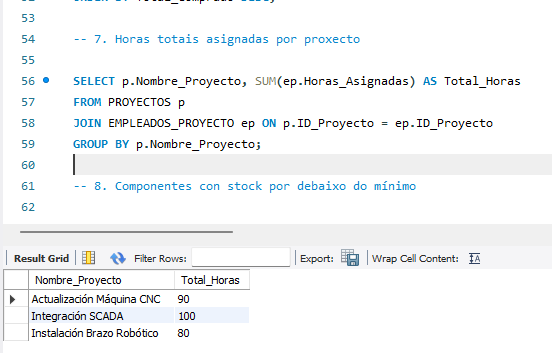
Listar as empresas que teñen maquinas con mais de unha incidencia:



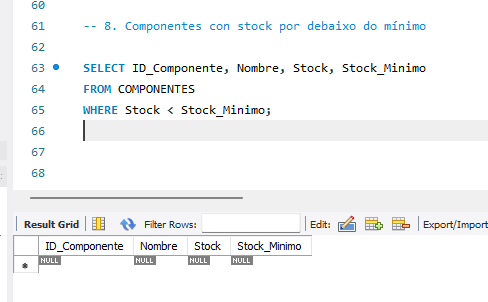
Listar os proveedores en funcion de pedidos.



Horas totales asignadas por proyecto

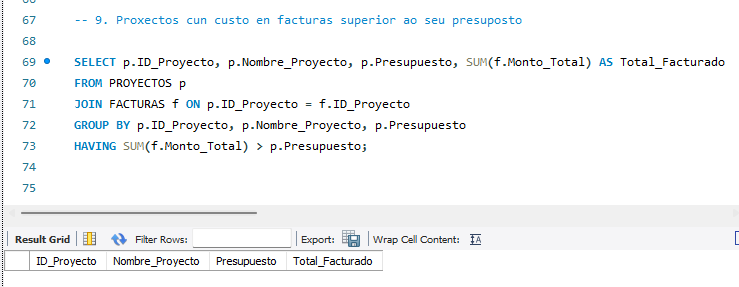


Componentes con stock por debaixo do mínimo

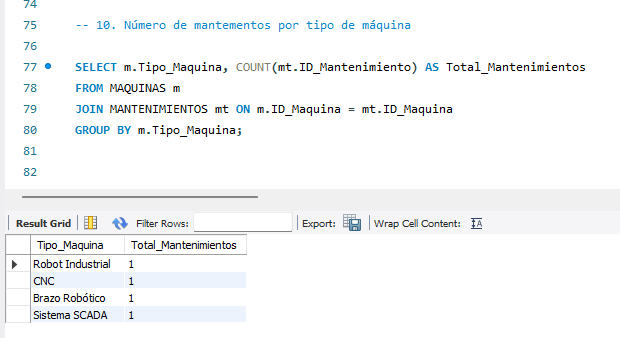


Non hai compoñentes con stock por debaixo do minimo

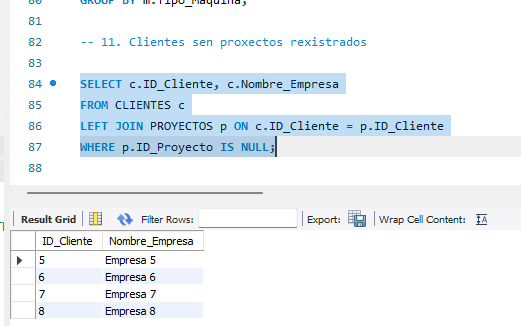
Proxectos cun custo en facturas superior ao seu presuposto (Non hai)



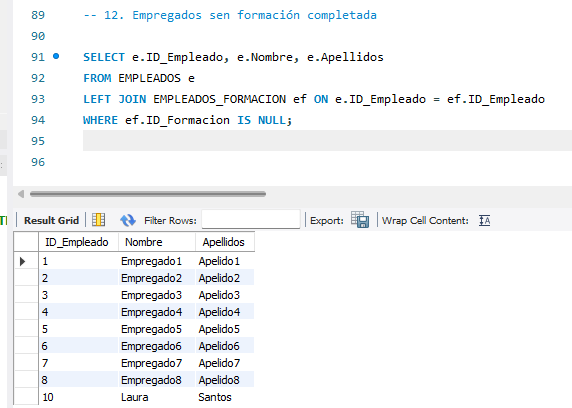
Número de mantementos por tipo de máquina



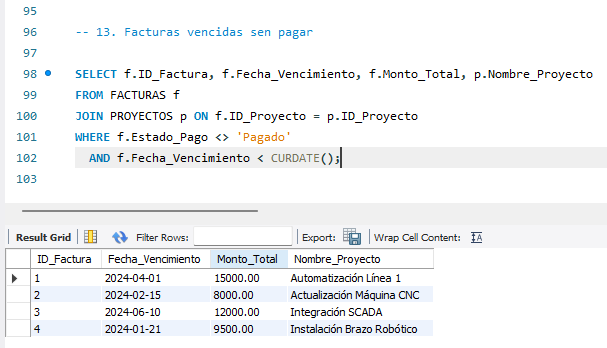
Clientes sen proxectos rexistrados



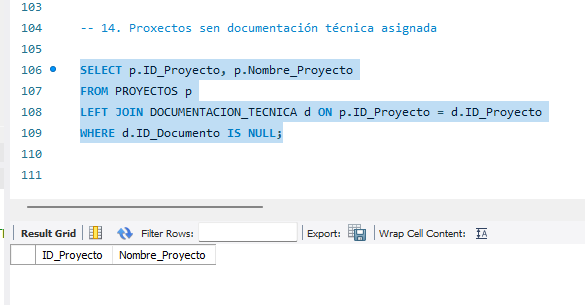
Empregados sen formación completada



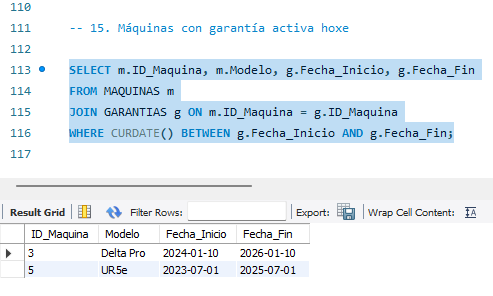
Facturas vencidas sen pagar



Proxectos sen documentación técnica asignada



Máquinas con garantía activa hoxe

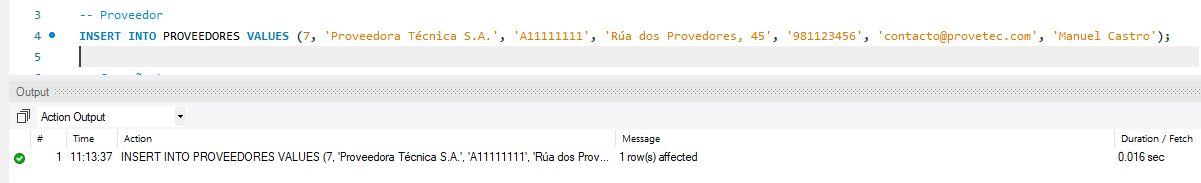


# Casos de Prueba y Simulación

|  |  |  |  |
| --- | --- | --- | --- |
| **Nº** | **PROBA** | **ACCIÓN** | **ESPERADO** |
| **1** | |  | | --- | |  |  |  | | --- | | Alta válida | | |  | | --- | |  |  |  | | --- | | Inserir un novo **compoñente** cun provedor existente | | Inserción exitosa |
| **2** | Alta válida | Crear un **pedido** e engadir liñas de compoñentes | Pedido e liñas creadas correctamente |
| **3** | |  | | --- | |  |  |  |  |  | | --- | --- | --- | | |  | | --- | |  |  |  | | --- | | Alta válida | | | Rexistrar unha **máquina** e asociarlle compoñentes | Asociación correcta |
| **4** | Alta válida | Crear un **mantemento** asignado a un técnico | Mantemento creado con FK válida |
| **5** | Alta válida | Rexistrar unha **formación** e asociar a un empregado | Relación formación-empregado rexistrada correctamente |
| **6** | Alta válida | |  | | --- | |  |  |  | | --- | | Emitir unha **certificación** para un empregado | | Certificación rexistrada correctamente |
| **7** | |  | | --- | |  |  |  | | --- | | Alta válida | | Rexistrar unha **incidencia** cun cliente e máquina existentes | Incidencia rexistrada correctamente |
| **8** | |  | | --- | |  |  |  | | --- | | Erro por FK | | |  | | --- | |  |  |  | | --- | | Asociar un compoñente a unha máquina inexistente | | Erro por FK (restrición de integridade referencial) |
| **9** | |  | | --- | |  |  |  | | --- | | Erro por FK | | Crear unha incidencia cun cliente inexistente | Erro por FK (cliente non existente) |
| **10** | |  | | --- | |  |  |  | | --- | | Erro por PK duplicada | | Inserir dúas veces o mesmo rexistro en EMPLEADOS\_FORMACION | Erro por clave primaria composta duplicada |
| **11** | Borrado con SET NULL | Eliminar un empregado con mantementos asignados | Campo ID\_Empleado\_Responsable en MANTENIMIENTOS pasa a NULL |
| **12** | Borrado con CASCADE | Eliminar unha máquina asociada a compoñentes e mantementos | Eliminación en cascada de asociacións en COMPONENTES\_MAQUINA e outros |

**-- Proveedor**

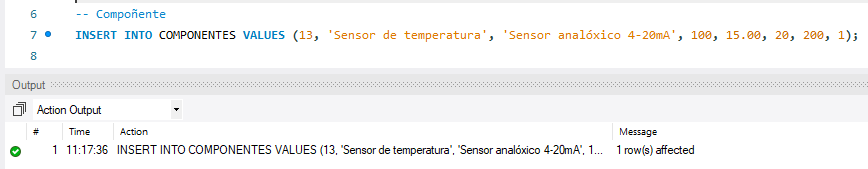
**INSERT INTO PROVEEDORES VALUES (7, 'Proveedora Técnica S.A.', 'A11111111', 'Rúa dos Provedores, 45', '981123456', 'contacto@provetec.com', 'Manuel Castro');**

****

**Como se mostra na tabla a inserción e permitida pola BD.**

**-- Compoñente**

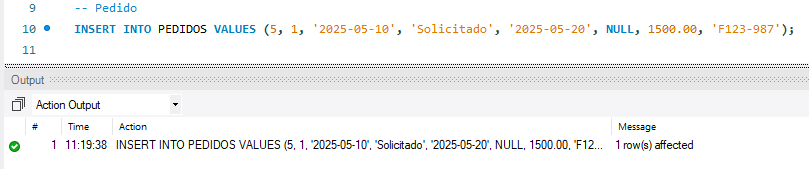
**INSERT INTO COMPONENTES VALUES (13, 'Sensor de temperatura', 'Sensor analóxico 4-20mA', 100, 15.00, 20, 200, 1);**

****

**Como se mostra na tabla a inserción e permitida pola BD.**

**-- Pedido**

**INSERT INTO PEDIDOS VALUES (5, 1, '2025-05-10', 'Solicitado', '2025-05-20', NULL, 1500.00, 'F123-987');**

****

**Como se mostra na tabla a inserción e permitida pola BD.**

**-- Pedido de compoñente**

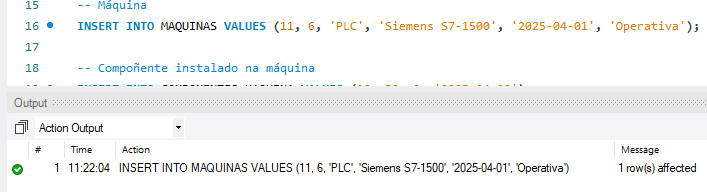
**INSERT INTO PEDIDOS\_COMPONENTES VALUES (5, 10, 100, 15.00, 'Pendiente');**

****

**Como se mostra na tabla a inserción e permitida pola BD.**

**-- Máquina**

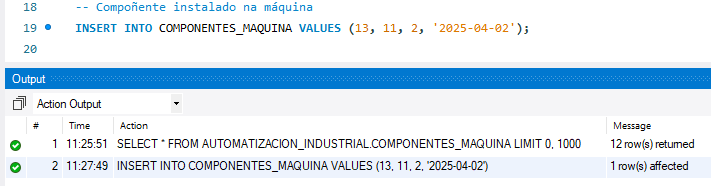
**INSERT INTO MAQUINAS VALUES (11, 6, 'PLC', 'Siemens S7-1500', '2025-04-01', 'Operativa');**

****

**Como se mostra na tabla a inserción e permitida pola BD.**

**-- Compoñente instalado na máquina**

**INSERT INTO COMPONENTES\_MAQUINA VALUES (13, 11, 2, '2025-04-02');**

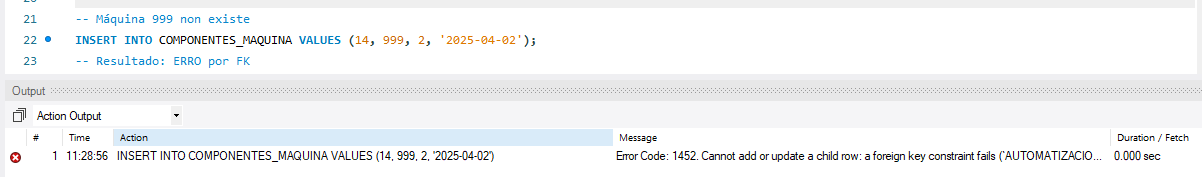
****

**Como se mostra na tabla a inserción e permitida pola BD.**

**-- Máquina 999 non existe**

**INSERT INTO COMPONENTES\_MAQUINA VALUES (14, 999, 2, '2025-04-02');**

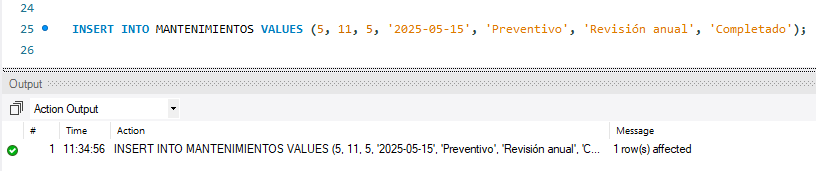
**-- Resultado: ERRO por FK**

****

**Como o id da maquina non corresponde a ningunha maquina non e posible a inserrción.**

**--Mantenimiento a la maquina añadida**

**INSERT INTO MANTENIMIENTOS VALUES (5, 11, 300, '2025-05-15', 'Preventivo', 'Revisión anual', 'Completado');**

****

**Como se mostra na tabla a inserción e permitida pola BD a maquina introducida anteriormente.**

**-- Curso**

**INSERT INTO FORMACIONES VALUES (3, 'Curso PLC avanzado', '2025-01-15', '2025-02-15', 'Xosé López', 'Presencial');**

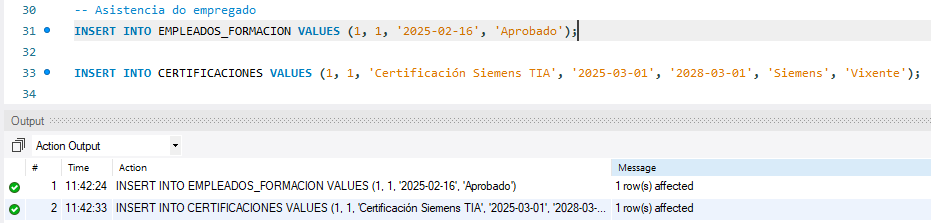
****

**Insercción de novo curso permitido**

**-- Asistencia do empregado**

**INSERT INTO EMPLEADOS\_FORMACION VALUES (1, 1, '2025-02-16', 'Aprobado');**

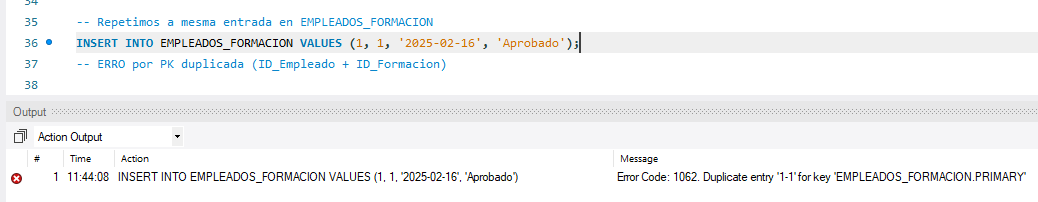
**INSERT INTO CERTIFICACIONES VALUES (1, 1, 'Certificación Siemens TIA', '2025-03-01', '2028-03-01', 'Siemens', 'Vixente');**

****

**Inserccións permitidas pola base de datos.**

**-- Repetimos a mesma entrada en EMPLEADOS\_FORMACION**

**INSERT INTO EMPLEADOS\_FORMACION VALUES (1, 1, '2025-02-16', 'Aprobado');**

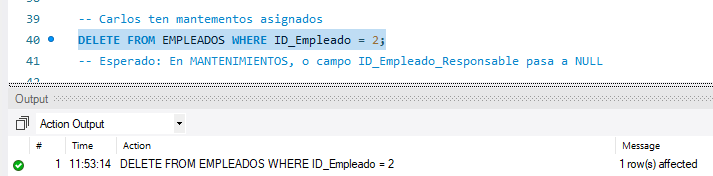
****

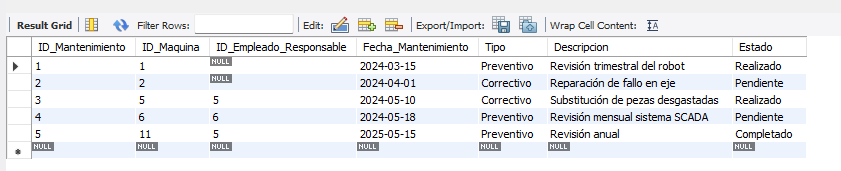
**ERRO por PK duplicada (ID\_Empleado + ID\_Formacion)**

**-- Carlos ten mantementos asignados**

**DELETE FROM EMPLEADOS WHERE ID\_Empleado = 1;**

**-- Esperado: En MANTENIMIENTOS, o campo ID\_Empleado\_Responsable pasa a NULL**

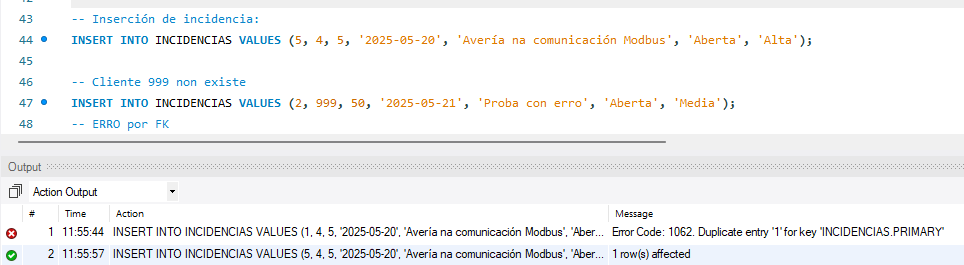
****

****

**O empleado con id 2 tiña asociado 2 mantementos;**

**-- Inserción de incidencia:**

**INSERT INTO INCIDENCIAS VALUES (5, 4, 5, '2025-05-20', 'Avería na comunicación Modbus', 'Aberta', 'Alta');**

****

**Inserccións permitidas pola base de datos.**

**-- Cliente 999 non existe**

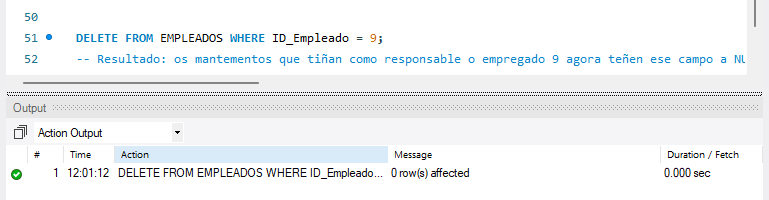
**INSERT INTO INCIDENCIAS VALUES (2, 999, 50, '2025-05-21', 'Proba con erro', 'Aberta', 'Media');**

****

**ERRO por FK**

**DELETE FROM EMPLEADOS WHERE ID\_Empleado = 9;**

**-- Resultado: os mantementos que tiñan como responsable o empregado 9 agora teñen ese campo a NULL**

****

**Neste caso parece que casualmente o empleado 9 non tiña mantementos asociados**

# Resultados y Verificación

Unha vez feita toda a base de datos, cos seus datos cargados e relacións definidas, fixéronse varias probas para comprobar que todo funciona como debería. A idea era ver se o sistema responde ben tanto nas operacións normais como cando se lle meten datos erróneos ou se fan accións complexas.

### **Comprobación da integridade dos datos**

* As relacións entre táboas funcionan ben: cando un dato depende doutro (por exemplo, un proxecto dun cliente ou un mantemento dunha máquina), os valores están ben ligados e cumpren as regras de integridade.
* Probouse a eliminación de datos con distintas configuracións:
  + Cando se borra un empregado con mantementos asignados, o campo de responsable pasa a NULL, tal e como se esperaba (ON DELETE SET NULL).
  + Cando se elimina unha máquina, tamén se borran automaticamente os compoñentes instalados nela e os mantementos rexistrados (ON DELETE CASCADE).

### **Altas, baixas e modificacións**

* Insertáronse clientes, proxectos, máquinas, compoñentes, empregados e relacións entre eles sen erros.
* Tamén se fixeron modificacións, como cambiar teléfonos ou emails de empregados, e comprobouse que se gardaban ben.
* Probouse a eliminación de empregados e máquinas, comprobando que os datos relacionados se actualizaban ou borraban correctamente.

### **Probas con erros (intencionados)**

* Intentouse inserir datos con claves foráneas erróneas (por exemplo, asociar un compoñente a unha máquina que non existe), e o sistema rexeitouno, como era de esperar.
* Tamén se fixo unha proba de duplicar rexistros en táboas con claves primarias compostas (como empregados en formacións), e saltou o erro correspondente.
* Os triggers tamén responderon ben:
  + O que impide inserir empregados sen DNI funcionou correctamente.
  + O que evita borrar directores tamén.
  + O que garda cambios de contacto nunha táboa de auditoría tamén se executou cando se fixo un UPDATE.

### **Resultados das consultas**

As consultas preparadas permitiron extraer datos útiles como:

* Clientes con máis proxectos.
* Empregados que traballan en varios proxectos.
* Incidencias abertas ou prioritarias.
* Proxectos que superan o orzamento previsto.
* Compoñentes con stock baixo.
* Empregados que aínda non fixeron cursos de formación.

### **Resumo final**

A base de datos responde ben, mantén a coherencia dos datos, reacciona ante erros e permite obter información importante para a empresa. O deseño parece axeitado e está preparado para un uso real con datos dinámicos. Pódense facer operacións complexas sen que haxa inconsistencias, e os procedementos, funcións e triggers aportan valor engadido.

# Conclusiones y Mejoras Futuras

A base de datos deseñada para a empresa de automatización industrial cumpre cos obxectivos iniciais do proxecto. A estrutura creada permite:

* Xestionar de maneira organizada os **clientes**, **proxectos**, **empregados**, **máquinas**, **compoñentes** e **incidencias**.
* Levar un control detallado de **mantementos**, **formacións**, **certificacións** e **servizos técnicos**, algo fundamental neste tipo de negocio.
* Garantir a **integridade dos datos** mediante claves foráneas, restricións e accións automáticas (como CASCADE e SET NULL) que aseguran que non haxa información illada ou errónea.
* Mellorar a funcionalidade e seguridade coa implementación de **funcións**, **procedementos almacenados** e **triggers**, que automatizan certas tarefas e preveñen erros comúns.

Durante o desenvolvemento realizáronse probas completas (altas, baixas, modificacións e situacións con erros) e o comportamento do sistema foi o esperado en todos os casos. Ademais, as consultas SQL creadas permiten extraer información útil para a toma de decisións.

### **Melloras futuras**

Aínda que a base de datos está funcional e lista para uso real, hai algunhas melloras que se poderían aplicar nun futuro:

* **Interfaz de usuario**: Desenvolver unha aplicación web ou de escritorio que se conecte coa base de datos para facilitar o uso a usuarios non técnicos.
* **Xestión de permisos**: Engadir roles de usuario na base de datos (administrador, técnico, comercial…) con diferentes niveis de acceso.
* **Auditoría avanzada**: Crear un sistema máis completo para rexistrar accións importantes (insercións, modificacións e eliminacións), con datas e usuarios.
* **Validacións máis completas**: Engadir máis CHECK constraints ou procedementos para controlar valores lóxicos (ex. stock negativo, datas erróneas, etc.).
* **Historial de mantementos e incidencias**: Gardar rexistros históricos cando se faga un cambio de estado para manter trazabilidade completa.
* **Documentación técnica**: Incorporar unha mellor organización e almacenamento dos documentos técnicos asociados a máquinas e proxectos.
* **Exportación de datos**: Preparar informes en PDF ou Excel para facilitar a comunicación cos clientes ou provedores.

# Enlace al Repositorio en GitHub

[**REPOSITORIO GIT**](https://github.com/Xurxo777/Proyecto_BBDD_XurxoBargeBlanco.git)