



Base de Datos Drugstore 24 horas.

Guaimás Rosado Cesar Gabriel

—

Tutor: Sergio Occhipinti

CoderHouse - Comisión 34940

CONTENIDO

Introducción	3
Objetivo	4
Situación problemática	5
Modelo de Negocio	6
Diagrama Entidad - Relación	7
Diagrama de Base de Datos V1.0	8
Diagrama de Base de Datos V2.0	9
Listado de Tablas	10
Listado de Vistas (Views), Funciones, Stored Procedures y Triggers.	14
Código SQL para la creación del schema Drugstore	24
Tecnologías Utilizadas	17

Introducción:

Este escrito se elabora como documentación del desarrollo de una base de datos, utilizando la tecnología de Oracle “MySQL Community Server”, el cual se entregará a los dueños de un **Drugstore 24 horas** (establecimiento comercial de venta al público) llamado **San Silvestre 24**.

Se muestra de manera detallada el desarrollo de la base de datos (prototipo versión 1.0).

La base de datos en su versión final cumplirá con los estándares y requerimientos necesarios para su óptimo funcionamiento.

Para esta base de datos se optó por el motor de base de datos **MySQL Community Server** en su **versión 8.0.30** y su herramienta visual **MySQL Workbench** la cual integra componentes para el desarrollo, administración, diseño y gestión de bases de datos relacionales.

Se agradece al profesor **Cesar Aracena** y al mi tutor **Sergio Occhipinti** por sus clases, guía para la entrega y finalización de este trabajo final del curso de **SQL** brindado por la plataforma **CODERHOUSE**.

-- * Este documento fue confeccionado únicamente con fines
-- educativos y no representa una base de datos real ni
-- totalmente funcional. No debe usarse de ninguna manera para
-- fines lucrativos reales.

Objetivos:

Para una mejor administración del drugstore se puso en marcha este proyecto que tiene como fin cumplir las necesidades y demanda de datos de los dueños del negocio.

- Por seguridad sólo los usuarios seleccionados por el personal administrativo pertinente tendrán acceso a la base de datos previa firma de la documentación correspondiente a la aceptación de responsabilidades.
- Se hará entrega de la documentación correspondiente al cliente para que pueda ver la descripción detallada de la base de datos, así como los procedimientos almacenados, funciones y triggers.
- La base de datos deberá de cumplir con los requerimientos o estándares que el negocio tenga a su alcance.
- Esta documentación será enseñada de manera detallada para que se comprenda el desarrollo de la base de datos y la futura interfaz gráfica, la cual tiene como fin facilitar el uso de la información generada por el negocio.
- Se agilizarán las instancias de procesamiento de datos, búsqueda y obtención de información.

Situación problemática:

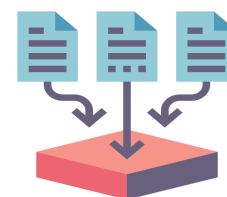
Los dueños del drugstore **San Silvestre 24** cuentan con una gran cantidad de datos, almacenados en más de 20 archivos de Microsoft Excel y planillas de proveedores escritas manualmente, los cuales son manejados por seis administrativos en tres locales comerciales. Así mismo, dichos datos son consultados libremente por 16 empleados que hacen turnos rotativos cumpliendo un horario de 8 horas cada uno y cubriendo las 24 horas del día.

Las planillas mencionadas se comparten mediante la tecnología brindada por Google, **"Google Drive"**, que es actualizada por el personal anteriormente nombrado. Al no poseer permisos específicos por el uso del servicio gratuito de la herramienta, estas planillas son modificadas por cualquier empleado que tenga acceso a las mismas, generando así faltantes en los productos y por ende, en el dinero generado por las ventas.

Otro problema que se genera del método de trabajo actual del drugstore es la falta de control de stock de productos costosos, como las hojas de coca, las cuales tienen mucha demanda y venta durante el día, Impidiendo el control exacto de los ingresos y egresos de los paquetes fraccionados del producto.

Por último cabe aclarar que se han generado muchas pérdidas de datos en los meses pasados y no pudieron ser recuperados porque no se cuenta con un sistema íntegro y confiable de base de datos y backups (copias de seguridad) periódicos que aseguren la integridad a dicha información.

Modelo de Negocio



Base de Datos de un Drugstore 24 horas (Negocio de venta)

Para mi proyecto escogí un negocio en el que trabajo llamado **"San Silvestre 24"** (Drugstore 24 horas), ya que me pareció interesante el generar una base de datos relacional mediante la herramienta MySQL Server y Workbench para solucionar problemas con los que se enfrentan día a día con el procesamiento de datos y obtención de la información necesaria.

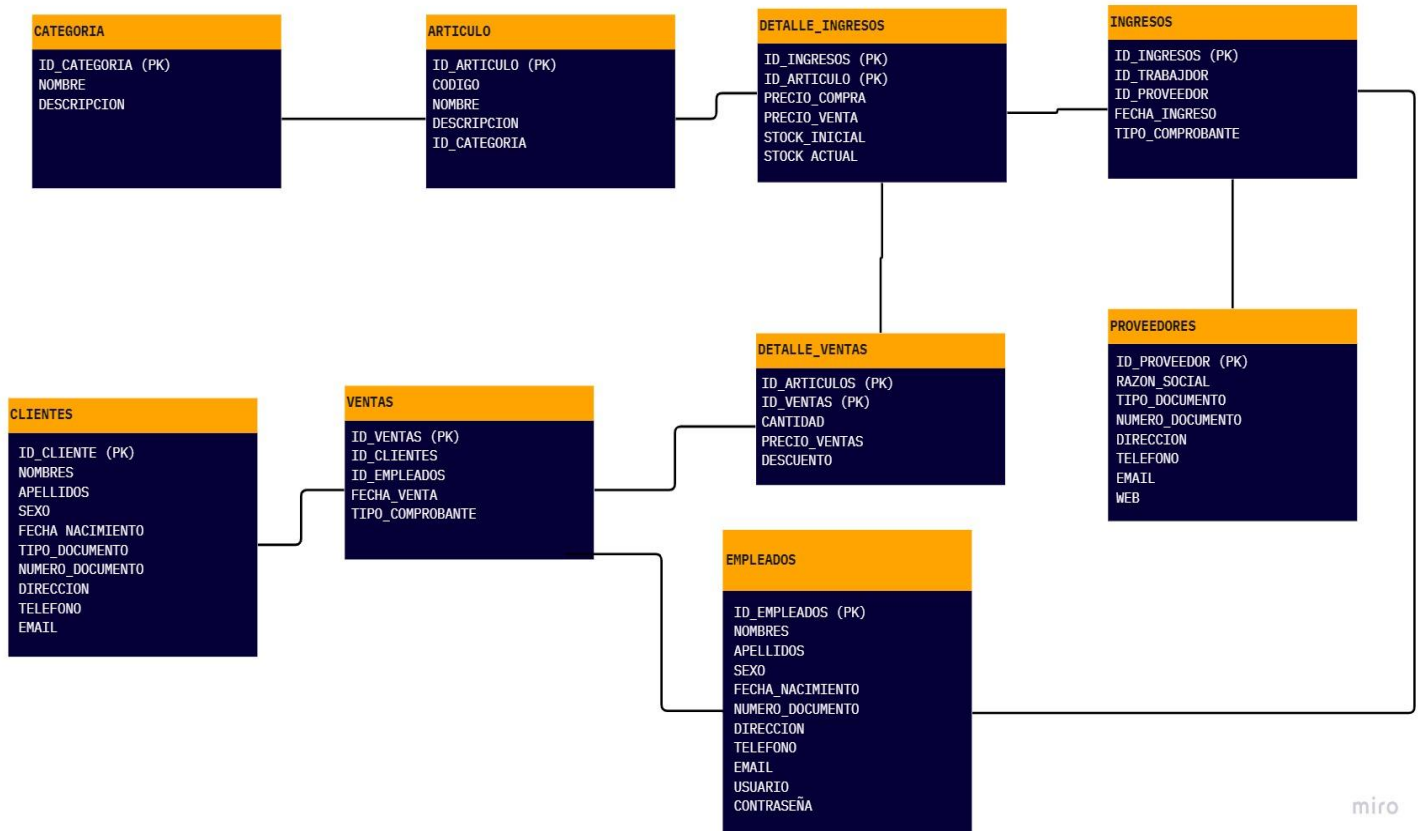
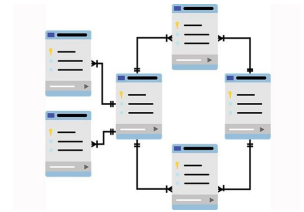
San Silvestre 24 es un drugstore/negocio abierto las 24 horas en el cual se venden diversos artículos al público, entre ellos comestibles, bebidas, golosinas, lácteos, fiambres, juguetes, hoja de coca fraccionada, entre otros.

Las ventas son realizadas por empleados de atención al público y cuentan con distintos medios de pago; efectivo, transferencia bancaria, MercadoPago, Uala, Pago24, tarjetas de débito y crédito.

Los proveedores y el stock son manejados por el personal administrativo correspondiente, se anotan los ingresos y egresos en planillas de Microsoft Excel y también de manera manual en planillas impresas.

En la base de datos se podría cargar información relacionada con **"Clientes"**, **"Ventas"**, **"Detalle de ventas"**, **"Empleados"**, **"Proveedores"**, **"Ingresos de mercadería"**, **"Detalles de Ingreso"**, **"Artículos"** y **"Categorías"**.

Diagrama de Base de Datos (Sin llaves)



miro

Diagrama de Base de Datos V1.0 (EER)

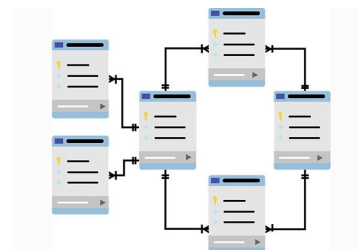


Diagrama EER Generado por MySQL Server

En la primera versión de este diagrama EER/DER se puede apreciar las primeras tablas generadas mediante código SQL. Se pueden apreciar las tablas Categorías, Artículos, Detalle_Ingresos, Ingresos, Ventas, Detalle_Ventas, Proveedores, Clientes, Contacto y Empleados. La descripción de las tablas, columnas y tipo de dato se encuentra en la hoja: [Documentacion DB Drugstore24](#)

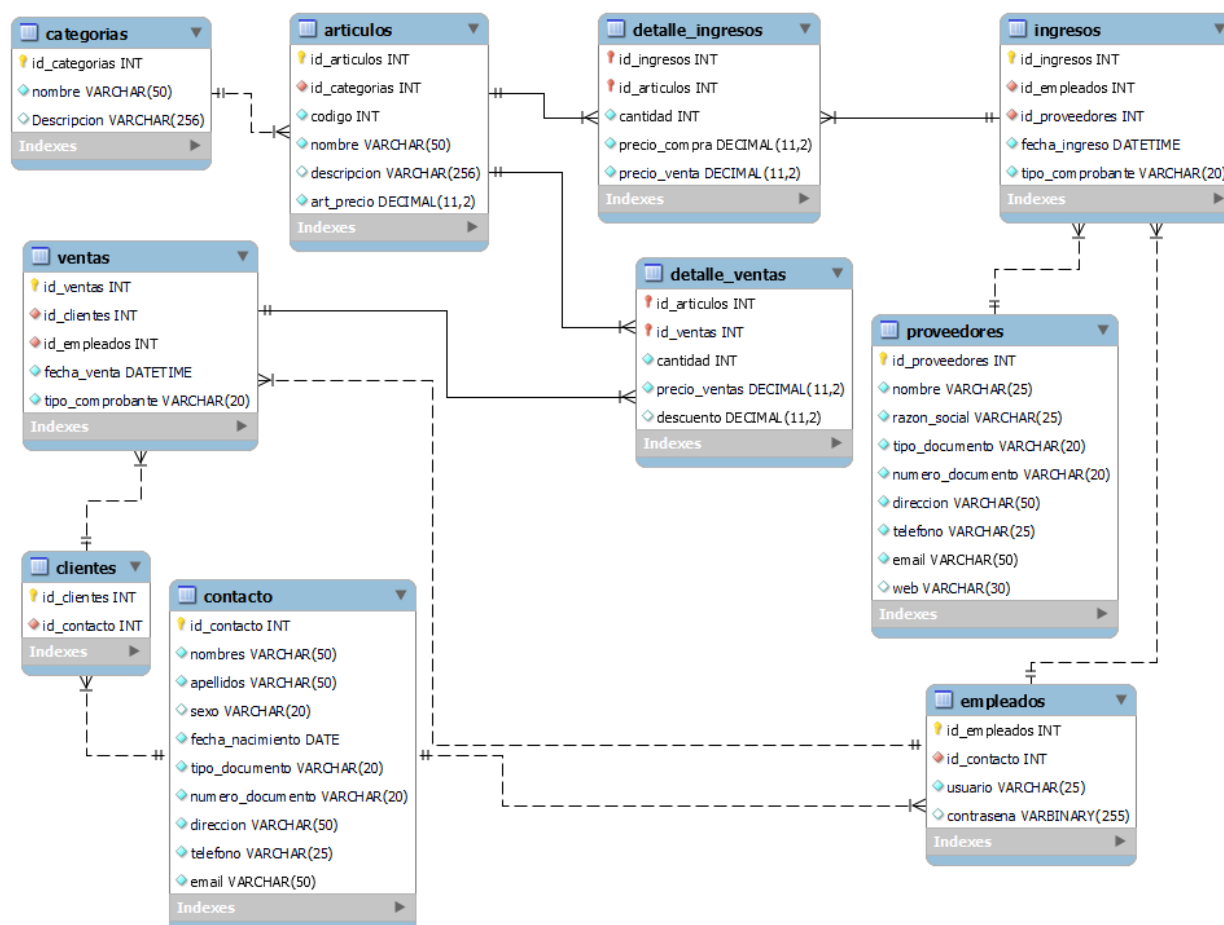


Diagrama de Base de Datos V2.0 (EER)

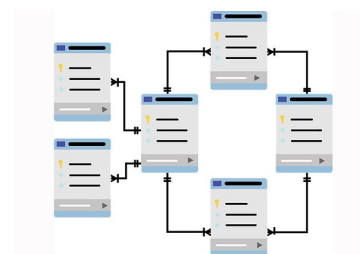


Diagrama EER Generado por MySQL Server

En la segunda versión de este diagrama EER/DER se puede apreciar, además de lo dicho en la primera versión, las tablas Log_Articulos (Artículos) y Log_Proveedores (Proveedores) las cuales se generan para poder auditar las tablas asociadas a las originales. Además se pueden observar al final del diagrama, las **vistas** (tablas virtuales) las cuales facilitan las consultas más frecuentes.

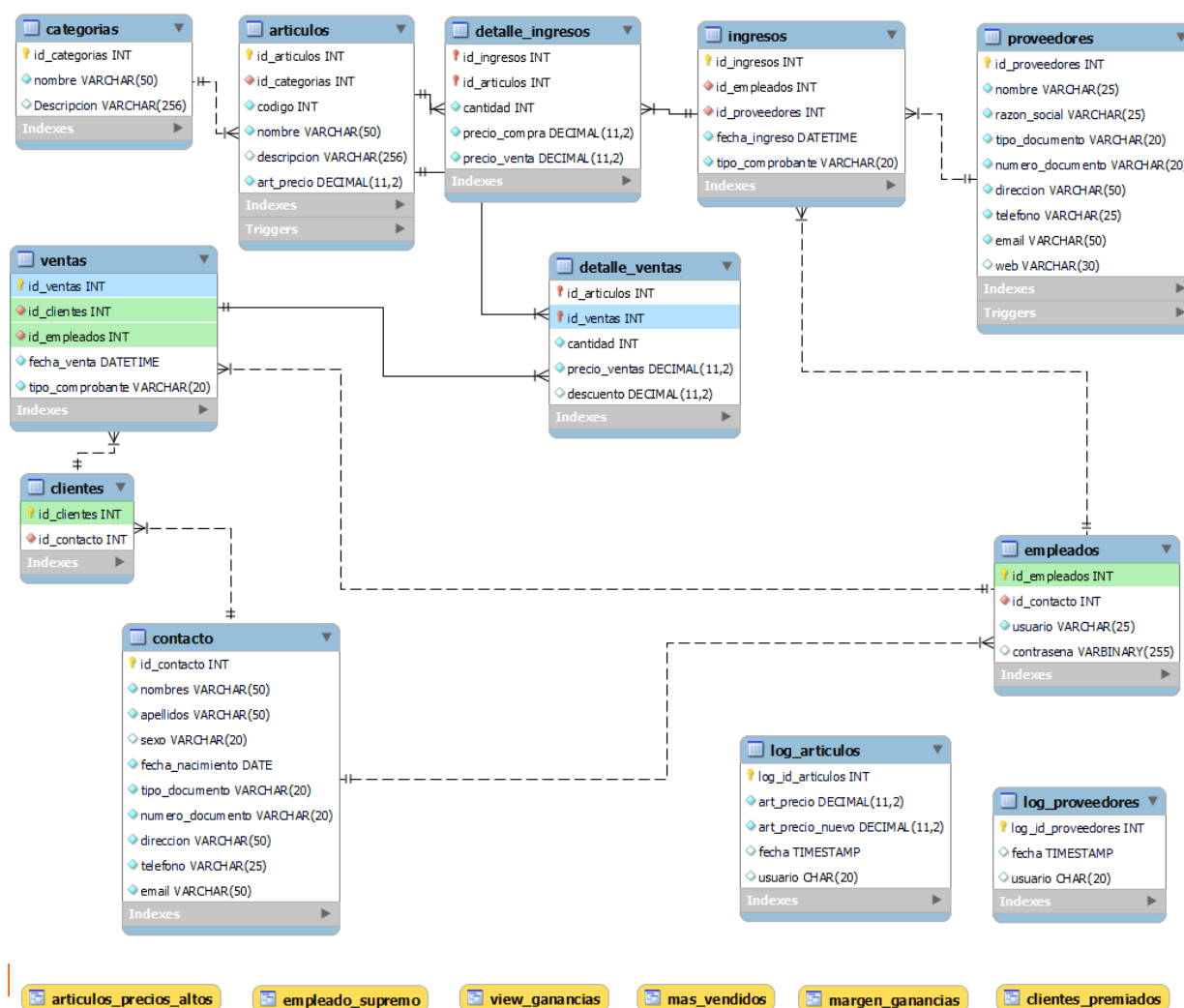


Tabla 1: Clientes

Tipo de clave	Campo	Tipo de campo
(PK)	Id_Cliente	INT
	Nombres	VARCHAR (50)
	Apellidos	VARCHAR (50)
	Sexo	VARCHAR (20)
	Fecha Nacimiento	DATETIME
	Tipo_Documento	VARCHAR (20)
	Numero_Documento	VARCHAR (20)
	Direccion	VARCHAR (50)
	Telefono	VARCHAR (25)
	Email	VARCHAR (50)

Tabla 2: Ventas

Tipo de clave	Campo	Tipo de campo
(PK)	Id_Ventas	INT
(FK)	Id_Cliente	INT
(FK)	Id_Empleado	INT
	Fecha_Venta	DATETIME
	Tipo_Comprobante	VARCHAR (20)

Tabla 3: Empleados

Tipo de clave	Campo	Tipo de campo
(PK)	Id_Empleado	INT
	Nombres	VARCHAR (50)
	Apellidos	VARCHAR (50)
	Sexo	VARCHAR (20)
	Fecha Nacimiento	DATETIME
	Tipo_Documento	VARCHAR (20)
	Numero_Documento	VARCHAR (20)
	Direccion	VARCHAR (50)
	Telefono	VARCHAR (25)
	Email	VARCHAR (50)
	Usuario	VARCHAR (25)
	Contrasena	VARBINARY (MAX)

Tabla 4: Detalle_Ventas

Tipo de clave	Campo	Tipo de campo
(PK)	id_articulos	INT
(PK)	Id_Ventas	INT
	Cantidad	INT
	Precio_Ventas	DECIMAL (11,2)
	Descuento	DECIMAL (11,2)

Tabla 5: Proveedores

Tipo de clave	Campo	Tipo de campo
(PK)	Id_Proveedores	INT
	Razon_Social	VARCHAR (25)
	Tipo_Documento	VARCHAR (20)
	Numero_Documento	VARCHAR (20)
	Direccion	VARCHAR (50)
	Telefono	VARCHAR (25)
	Email	VARCHAR (50)
	Web	VARCHAR (30)

Tabla 6: Ingresos

Tipo de clave	Campo	Tipo de campo
(PK)	Id_Ingresos	INT
(FK)	Id_Empleados	INT
(FK)	Id_Proveedores	INT
	Fecha_Ingreso	DATETIME
	Tipo_Comprobante	VARCHAR (20)

Tabla 7: Detalle_Ingresos

Tipo de clave	Campo	Tipo de campo
(PK)	Id_Ingresos	INT
(PK)	Id_Articulos	INT
	Precio_Compra	DECIMAL(11,2)
	Precio_Venta	DECIMAL(11,2)

Tabla 8: Artículos

Tipo de clave	Campo	Tipo de campo
(PK)	Id_Articulos	INT
(FK)	Id_Categorias	INT
	Codigo	INT
	Nombre	VARCHAR (50)
	Descripcion	VARCHAR (256)
	Tipo_Comprobante	VARCHAR (20)

Tabla 9: Categorías

Tipo de clave	Campo	Tipo de campo
(PK)	Id_Categorias	INT
	Nombre	VARCHAR (50)
	Descripcion	VARCHAR (256)

Listado de Vistas (Views), Funciones, Stored Procedures y Triggers.

Se muestran las vistas, funciones, stored procedures y triggers que forman parte de la base de datos a utilizar en el sistema de gestión del drugstore.

Vistas:

- **articulos_precios_altos:** Esta vista muestra los nombres de las categorías y los precios superiores a \$1000 (pesos mil). La composición está dada por las tablas **Categorías** y **Artículos**.
- **margen_ganancias:** Esta vista muestra cual es el proveedor que deja el mayor margen de ganancias. Esto facilita escoger el mejor proveedor. La composición está dada por las tablas: **Proveedores**, **Ingresos** y **Detalle_Ingresos**.
- **empleado_supremo:** Esta vista muestra el empleado que más ventas ha realizado hasta el día de la fecha. Puede usarse para darle premios de incentivo. La composición está dada por las tablas: **Empleados**, **Ventas** y **Contacto**.
- **mas_vendidos:** Esta vista muestra el top de los 5 artículos más vendidos hasta el día de la fecha. Puede usarse para analizar qué artículos deben ser estoqueados más frecuentemente. La composición está dada por las tablas: **Artículos** y **Detalle_Ventas**.
- **clientes_premiados:** Esta vista muestra las últimas dos ventas para premiar a dichos clientes. Muestra el número de teléfono de contacto en caso de necesitar dar con ellos. La composición está dada por las tablas: **Cliente**, **Contacto** y **Ventas**.

Funciones:

- **func_calcularGanancias:** Esta función muestra automáticamente las ganancias que genera cada producto que está en el stock actual del drugstore. Para probar esta función se puso en práctica una vista adicional llamada **view_ganancias** la cual lleva integrada la función.
 - **view_ganancias:** Esta vista nos deja ver las ganancias generadas por cada artículo del Drugstore utilizando la función almacenada **func_calcularGanancias**. Está compuesta por las tablas: **Detalle_Ingresos** y **Artículos**.
- **func_cantvent:** Esta función muestra la cantidad total de ventas por categorías hasta el día de la fecha. Esta información se obtiene al ingresar el nombre de la categoría. Está compuesta por las tablas: **Detalle_Ventas**, **Artículos** y **Categorías**.

Stored Procedures:

- **pr_limite_stock:** Este procedimiento almacenado permite tener un control sobre todos los productos que tengan menos de 10 unidades disponibles para así agilizar los procesos de restock. Se compone por las tablas: **Detalle_Ingresos** y **Artículos**.
- **pr_insert_prov:** Este procedimiento almacenado permite insertar datos nuevos en la tabla proveedores, de esta manera se evita escribir a mano las sentencias necesarias para el ingreso de nuevos contratistas. Impacta únicamente sobre la tabla **Proveedores**.

Triggers:

- **tr_log_proveedores:** Este trigger nos permite auditar la tabla **Proveedores**. Luego de ingresar nuevos valores. Ya que se desea saber cuáles proveedores fueron recientemente agregados, la fecha, la hora y quien realizó dicha modificación mediante el uso de la sentencia **SESSION_USER()**. Para esto se creó una tabla llamada **log_proveedores**.

Tabla Log_Proveedores:

Tipo de clave	Campo	Tipo de campo
(PK)	log_id_proveedores	INT
	fecha	TIMESTAMP
	usuario	VARCHAR (20)

- **tr_log_articulos:** Este trigger nos permite auditar la tabla **Artículos**. Se generó una nueva tabla llamada **log_proveedores** en la cual se guardarán, antes de la actualización (UPDATE) del precio, el precio anterior, el precio nuevo, la fecha, la hora y quien realizó dicha actualización mediante el uso de la sentencia **SESSION_USER()**.

Tabla Log_Articulos:

Tipo de clave	Campo	Tipo de campo
(PK)	log_id_articulos	INT
	art_precio	DECIMAL(11,2)
	art_precio_nuevo	DECIMAL(11,2)
	fecha	TIMESTAMP
	usuario	VARCHAR (20)

Código SQL para la creación del schema Drugstore 24



Link a Drugstore24-Schema.sql: [drugstore24-schema.sql](#)

Link a Drugstore24-Data.sql: [drugstore24-data.sql](#)

Link a Drugstore24-Backup: [drugstore24-backup.sql](#)

Tecnologías utilizadas:

1. Visual Studio Code. [Visual Studio Code](#)

Editor de código. Utilizado para escribir el código SQL necesario para la creación de todos los componentes de la base de datos.

2. Dbeaver. [DBeaver Community](#)

Administrador de base de datos.

3. Miro. [La plataforma visual de colaboración para los equipos | Miro](#)

Aplicación de pizarra infinita para desarrollar flujos de trabajo.

4. MySQL Server Community 8.0.30 CE + Workbench. [MySQL Community Server](#)

Gestor de base de datos y herramienta visual de diseño de base de datos.

5. Microsoft Excel 365

Generación de gráficos para la visualización de informes.