

# **Análisis del Modelo de Base de Datos para el Consultorio Odontológico.**

**YULIET FAIZULI PACHON CARO  
NÉSTOR FABIAN GUTIERREZ SABOGAL  
JORGE MILLER GUTIERREZ OSPINA**

**SENA  
ANALISIS Y DESARROLLO DE SOFTWARE – 2721520  
Ivan Leonardo Medina Gomez  
Abril 2024**

## Introducción

El presente documento describe y analiza el modelo de base de datos diseñado para el consultorio odontológico “Maria Jose Velez”. Este modelo tiene como objetivo gestionar la información relacionada con usuarios, pacientes, odontólogos, consultas, citas, y otros aspectos relevantes para la administración y operación del consultorio odontológico.

## Objetivo

El objetivo principal de este modelo de base de datos es proporcionar una estructura organizada y eficiente que permita almacenar, gestionar y consultar la información necesaria para el funcionamiento del consultorio odontológico, facilitando la administración de usuarios, el seguimiento de pacientes, la programación de consultas y citas, y la gestión de facturación, entre otras funcionalidades.

## Análisis del Modelo.

### Características Generales

- **Caracteres Especiales:** Se ha seleccionado el conjunto de caracteres `utf8mb4` para admitir caracteres especiales, permitiendo una mayor flexibilidad en el almacenamiento de datos multilingües.
- **Collation:** Se ha elegido `utf8mb4\_spanish\_ci` como collation para la base de datos, lo que facilita el ordenamiento y comparación de cadenas de texto en español.

## Tablas Principales

## **1. Usuarios:**

- Almacena información básica de los usuarios, como nombre de usuario y contraseña.

## **2. TiposUsuario:**

- Define los tipos de usuarios (auxiliar u odontólogo) y establece la relación con la tabla Usuarios mediante claves foráneas.

## **3. Auxiliar:**

- Contiene detalles específicos de los auxiliares, como nombre, apellido, dirección, teléfono, ciudad y correo electrónico. Está relacionada con la tabla Usuarios mediante una clave foránea.

## **4. Pacientes:**

- Guarda información detallada de los pacientes, incluyendo datos personales, información de contacto y detalles relacionados con el auxiliar asignado. Establece relaciones con las tablas Usuarios y Auxiliar mediante claves foráneas.

## **5. Odontologos:**

- Almacena datos de los odontólogos, incluyendo especialidad, dirección, teléfono y correo electrónico. Se relaciona con la tabla Usuarios mediante una clave foránea.

## **6. Consultas:**

- Registra información sobre las consultas realizadas, incluyendo paciente, odontólogo, fecha, motivo, diagnóstico y tratamiento.

Establece relaciones con las tablas Pacientes y Odontólogos mediante claves foráneas.

### **7. Citas:**

- Guarda detalles de las citas programadas, como paciente, odontólogo, fecha, hora y motivo. Se relaciona con las tablas Pacientes y Odontólogos mediante claves foráneas.

### **8. Agenda:**

- Gestiona la programación de agendas para los odontólogos, incluyendo fecha, hora de inicio y hora de fin. Establece una relación con la tabla Odontólogos mediante una clave foránea.

### **9. Historial:**

- Almacena el historial médico de los pacientes, incluyendo fecha y descripción del evento. Se relaciona con la tabla Pacientes mediante una clave foránea.

### **10. Configuración:**

- Guarda la configuración general de la clínica, como nombre, dirección, teléfono, correo electrónico y horario de atención.

### **11. Facturas:**

- Registra detalles de las facturas emitidas por las consultas realizadas, incluyendo fecha y total. Se relaciona con la tabla Consultas mediante una clave foránea.

### **Características Adicionales.**

**Claves Primarias y Foráneas:** Se han definido claves primarias para identificar de forma única cada registro en las tablas, así como claves foráneas para establecer relaciones entre las tablas y garantizar la integridad referencial de los datos.

**Restricciones:** Se han establecido restricciones en las claves foráneas para controlar el borrado restrictivo y mantener la consistencia de los datos entre las tablas relacionadas.

**Tipos de Datos:** Se han utilizado diversos tipos de datos, como INT, VARCHAR, DATE, TIME, ENUM y DECIMAL, para almacenar diferentes tipos de información según las necesidades de cada campo.

En este modelo de base de datos para el consultorio odontológico, se han establecido varios tipos de relaciones entre las tablas para mantener la integridad referencial de los datos y garantizar la coherencia y consistencia de la información almacenada. A continuación, se detallan los tipos de relaciones :

## **1. Relación Uno a Uno (1:1)**

### **Usuarios - TiposUsuario:**

Un usuario puede tener un único tipo de usuario (auxiliar u odontólogo).

Un tipo de usuario pertenece a un único usuario.

## **2. Relación Uno a Muchos (1:N)**

### **Usuarios - Auxiliar:**

Un usuario (auxiliar) puede estar asociado con varios registros en la tabla Auxiliar, pero cada registro en Auxiliar pertenece a un único usuario.

### **Usuarios - Odontologos:**

Un usuario (odontólogo) puede estar asociado con varios registros en la tabla Odontologos, pero cada registro en Odontologos pertenece a un único usuario.

### **Usuarios - Pacientes:**

Un usuario (paciente) puede estar asociado con varios registros en la tabla Pacientes, pero cada registro en Pacientes pertenece a un único usuario.

### **Odontologos - Consultas:**

Un odontólogo puede tener varias consultas, pero cada consulta está asociada a un único odontólogo.

### **Pacientes - Consultas:**

Un paciente puede tener varias consultas, pero cada consulta está asociada a un único paciente.

### **Pacientes - Citas:**

Un paciente puede tener varias citas, pero cada cita está asociada a un único paciente.

### **Odontologos - Citas:**

Un odontólogo puede tener varias citas, pero cada cita está asociada a un único odontólogo.

### **Odontologos - Agenda:**

Un odontólogo puede tener varias entradas en la agenda, pero cada entrada en la agenda está asociada a un único odontólogo.

### **Pacientes - Historial:**

Un paciente puede tener varios registros en el historial, pero cada registro en el historial está asociado a un único paciente.

## **3. Relación Muchos a Muchos (N:N)**

### **TiposUsuario - Auxiliar:**

Un tipo de usuario (auxiliar) puede estar asociado con varios auxiliares y, a su vez, un auxiliar puede tener un tipo de usuario diferente o el mismo.

### **TiposUsuario - Odontologos:**

Un tipo de usuario (odontólogo) puede estar asociado con varios odontólogos y, a su vez, un odontólogo puede tener un tipo de usuario diferente o el mismo. La normalización es un proceso de diseño de bases de datos que se utiliza para organizar los datos en tablas de tal manera que se reduzca la redundancia y se mantenga la integridad de los datos. El objetivo principal de la normalización es eliminar las redundancias y asegurar que cada tabla represente una única entidad o relación.

## **análisis de la normalización aplicada en este modelo:**

### **1. Primera Forma Normal (1NF)**

### **Usuarios:**

Cumple con 1NF al no tener campos repetidos o grupos de campos que puedan ser divididos en tablas separadas.

### **TiposUsuario:**

También cumple con 1NF al tener campos atómicos y sin repetición.

### **Auxiliar, Odontologos, Pacientes, Consultas, Citas, Agenda, Historial, Configuración, Facturas:**

Todas estas tablas cumplen con 1NF al no tener campos repetidos y tener campos atómicos.

## **2. Segunda Forma Normal (2NF)**

### **Auxiliar, Odontologos, Pacientes, Consultas, Citas, Agenda, Historial, Facturas:**

Estas tablas cumplen con 2NF al no tener dependencias parciales de las claves primarias compuestas. Los campos dependen completamente de las claves primarias.

## **3. Tercera Forma Normal (3NF)**

### **Usuarios, TiposUsuario, Auxiliar, Odontologos, Pacientes, Consultas, Citas, Agenda, Historial, Configuración, Facturas:**

Todas estas tablas cumplen con 3NF al no tener dependencias transitivas de las claves primarias. Los campos no dependen de otros campos que no sean claves primarias.



## **Conclusión**

El modelo de base de datos del consultorio odontológico sigue las formas normales desde 1NF,2NF,3NF. La estructura de las tablas y las relaciones entre ellas están diseñadas para minimizar la redundancia y mantener la integridad de los datos.

Sin embargo, es importante destacar que la normalización puede ser un proceso iterativo y continuo, y puede haber oportunidades para optimizar aún más el diseño de la base de datos según las necesidades específicas del consultorio odontológico y los requisitos de negocio.

## **Script base de datos.**

**-- Crear base de datos seleccionamos el tipo de caracteres especiales**

**CREATE DATABASE consultorio\_odontologico CHARACTER SET utf8mb4 COLLATE utf8mb4\_spanish\_ci;**

**-- Advertencia este comando borra toda la base de datos**

**-- DROP DATABASE consultorio\_odontologico;**

**USE consultorio\_odontologico; -- Seleccionar la base de datos**

**SELECT DATABASE(); -- Nos muestra cual base de datos tenemos seleccionada**

**# Algunos comandos utilizados en esta base de datos**

**SHOW TABLES; -- Muestra todas las tablas creadas**

**DESCRIBE Usuarios; -- Muestra la tabla con sus tipos de datos**

**SHOW COLUMNS FROM Pacientes; -- Muestra las columnas de la tabla**

**RENAME TABLE Usuarios to Usuario; -- Renombrar una tabla**

**# -----**

**-- Crear tabla Usuarios**

**CREATE TABLE Usuarios (**

**userID INT PRIMARY KEY AUTO\_INCREMENT, -- Identificador  
    único de usuario, auto incremental**

**username VARCHAR(50), -- Nombre de usuario**

**password VARCHAR(255) -- Contraseña**

**);**

**-- Ejemplo para Insertar registros en la tabla Usuarios**

**INSERT INTO `Usuarios` (username,password) VALUES  
('Yuliet','123456'),('Fabian','789654');**

**# -----**

**-- Crear tabla TiposUsuario**

**CREATE TABLE TiposUsuario (**

**tipoID INT PRIMARY KEY AUTO\_INCREMENT, -- Identificador  
    único de tipo de usuario**

```
    userID INT, -- Clave foránea que referencia la tabla Usuarios
    userType ENUM('auxiliar', 'odontologo'), -- Tipo de usuario
    (paciente u odontólogo)
    FOREIGN KEY (userID) REFERENCES Usuarios(userID) -- Clave
    foránea que referencia la tabla Usuarios
);
```

```
# -----
```

```
-- Crear tabla Auxiliar
```

```
CREATE TABLE Auxiliar (
```

```
    auxiliarID INT PRIMARY KEY AUTO_INCREMENT, --
    Identificador único de auxiliar
```

```
    userID INT, -- Clave foránea que referencia la tabla Usuarios
```

```
    nombre VARCHAR(50), -- Nombre del auxiliar
```

```
    apellido VARCHAR(50), -- Apellido del auxiliar
```

```
    direccion VARCHAR(100), -- Dirección del auxiliar
```

```
    telefono VARCHAR(15), -- Número de teléfono del auxiliar
```

```
    ciudad VARCHAR(20), -- Ciudad de residencia
```

```
    correo_electronico VARCHAR(100), -- Correo electrónico del
    auxiliar
```

```
    FOREIGN KEY (userID) REFERENCES Usuarios(userID) -- Clave
    foránea que referencia la tabla Usuarios
);
```

```
# -----
```

```
-- Crear tabla Pacientes
```

```
CREATE TABLE Pacientes (
```

**pacienteID INT PRIMARY KEY AUTO\_INCREMENT, --  
Identificador único de paciente**

**userID INT, -- Clave foránea que referencia la tabla Usuarios**

**auxiliarID INT, -- Clave foránea que referencia la tabla Auxiliar**

**nombre VARCHAR(50), -- Nombre del paciente**

**apellido VARCHAR(50), -- Apellido del paciente**

**fecha\_nacimiento DATE, -- Fecha de nacimiento del paciente**

**direccion VARCHAR(100), -- Dirección del paciente**

**telefono VARCHAR(15), -- Número de teléfono del paciente**

**ciudad VARCHAR(20),**

**correo\_electronico VARCHAR(100), -- Correo electrónico del  
paciente**

**FOREIGN KEY (userID) REFERENCES Usuarios(userID), --  
Clave foránea que referencia la tabla Usuarios**

**FOREIGN KEY (auxiliarID) REFERENCES Auxiliar(auxiliarID) ON  
DELETE RESTRICT -- Clave foránea que referencia la tabla  
Auxiliar con restricción restrictiva de borrado**

**);**

**-- Agregamos la columna auxiliarID en la tabla pacientes y le  
decimos que la cree despues de auxiliarID**

**ALTER TABLE Pacientes ADD COLUMN odontologoID INT AFTER  
auxiliarID;**

**-- Borramos la columna odontologoID**

**ALTER TABLE Pacientes DROP COLUMN odontologoID;**

-- Agregamos una restriccion de uno a muchos en la tabla  
pacientes donde un odontologo tiene muchos pacientes

**ALTER TABLE Pacientes ADD CONSTRAINT fk\_odontologID  
FOREIGN KEY (odontologID) REFERENCES  
Odontologos(odontologID) ON DELETE RESTRICT;**

-- Eliminar la restricción de clave externa de la tabla TiposUsuario

**ALTER TABLE Pacientes DROP FOREIGN KEY fk\_odontologID;**

-- Advertencia Elimina la tabla Pacientes

**-- DROP TABLE Pacientes;**

**# -----**

-- Crear tabla Odontologos

**CREATE TABLE Odontologos (**

**odontologID INT PRIMARY KEY AUTO\_INCREMENT, --**  
    **Identificador único de odontólogo**

**userID INT, -- Clave foránea que referencia la tabla Usuarios**

**nombre VARCHAR(50), -- Nombre del odontólogo**

**apellido VARCHAR(50), -- Apellido del odontólogo**

**especialidad VARCHAR(100), -- Especialidad del odontólogo**

**direccion VARCHAR(100), -- Dirección del odontólogo**

**telefono VARCHAR(15), -- Número de teléfono del odontólogo**

**ciudad VARCHAR(20), -- Ciudad de residencia del odontologo**

**correo\_electronico VARCHAR(100), -- Correo electrónico del**  
    **odontólogo**

**FOREIGN KEY (userID) REFERENCES Usuarios(userID) -- Clave**  
    **foránea que referencia la tabla Usuarios**

**);**

**SHOW CREATE TABLE Odontologos; -- Muestra el codigo SQL como se creo la tabla**

**-- Agregamos la columna pais en la tabla odontologos**

**ALTER TABLE Odontologos ADD COLUMN pais VARCHAR(15) NULL;**

**-- Podemos borrar la columna pais**

**ALTER TABLE Odontologos DROP COLUMN pais;**

**# -----**

**-- Crear tabla Consultas**

**CREATE TABLE Consultas (**

**consultaID INT PRIMARY KEY AUTO\_INCREMENT, --  
Identificador único de consulta**

**pacienteID INT, -- Clave foránea que referencia la tabla  
Pacientes**

**odontologoID INT, -- Clave foránea que referencia la tabla  
Odontólogos**

**fecha DATE, -- Fecha de la consulta**

**motivo VARCHAR(255), -- Motivo de la consulta**

**diagnostico TEXT, -- Diagnóstico de la consulta**

**tratamiento TEXT, -- Tratamiento de la consulta**

**FOREIGN KEY (pacienteID) REFERENCES  
Pacientes(pacienteID), -- Clave foránea que referencia la tabla  
Pacientes**

**FOREIGN KEY (odontologoID) REFERENCES  
Odontologos(odontologoID) -- Clave foránea que referencia la  
tabla Odontólogos**

);

# -----

-- Crear tabla Citas

**CREATE TABLE Citas (**

**citaID INT PRIMARY KEY AUTO\_INCREMENT, -- Identificador  
    único de cita**

**pacienteID INT, -- Clave foránea que referencia la tabla  
Pacientes**

**odontologoID INT, -- Clave foránea que referencia la tabla  
Odontólogos**

**fecha DATE, -- Fecha de la cita**

**hora TIME, -- Hora de la cita**

**motivo VARCHAR(255), -- Motivo de la cita**

**FOREIGN KEY (pacienteID) REFERENCES  
Pacientes(pacienteID), -- Clave foránea que referencia la tabla  
Pacientes**

**FOREIGN KEY (odontologoID) REFERENCES  
Odontologos(odontologoID) -- Clave foránea que referencia la  
tabla Odontólogos**

);

# -----

-- Crear tabla Agenda

**CREATE TABLE Agenda (**

**agendaID INT PRIMARY KEY AUTO\_INCREMENT, --  
Identificador único de agenda**

```
    odontologoID INT, -- Clave foránea que referencia la tabla
Odontólogos
    fecha DATE, -- Fecha de la agenda
    hora_inicio TIME, -- Hora de inicio de la agenda
    hora_fin TIME, -- Hora de finalización de la agenda
    FOREIGN KEY (odontologoID) REFERENCES
Odontologos(odontologoID) -- Clave foránea que referencia la
tabla Odontólogos
);
```

```
# -----
```

```
-- Crear tabla Historial
```

```
CREATE TABLE Historial (
```

```
    historialID INT PRIMARY KEY AUTO_INCREMENT, --
Identificador único de historial
```

```
    pacienteID INT, -- Clave foránea que referencia la tabla
Pacientes
```

```
    fecha DATE, -- Fecha del historial
```

```
    descripcion TEXT, -- Descripción del historial
```

```
    FOREIGN KEY (pacienteID) REFERENCES
Pacientes(pacienteID) -- Clave foránea que referencia la tabla
Pacientes
);
```

```
# -----
```

```
-- Crear tabla Configuración
```

```
CREATE TABLE Configuracion (
```



**configID INT PRIMARY KEY AUTO\_INCREMENT, -- Identificador  
único de configuración**

**nombre\_clinica VARCHAR(100), -- Nombre de la clínica**

**direccion VARCHAR(100), -- Dirección de la clínica**

**telefono VARCHAR(15), -- Número de teléfono de la clínica**

**ciudad VARCHAR(20),**

**correo\_electronico VARCHAR(100), -- Correo electrónico de la  
clínica**

**horario\_atencion VARCHAR(100) -- Horario de atención de la  
clínica**

**);**

**# -----**

**-- Crear tabla Facturas**

**CREATE TABLE Facturas (**

**facturaID INT PRIMARY KEY AUTO\_INCREMENT, -- Identificador  
único de factura**

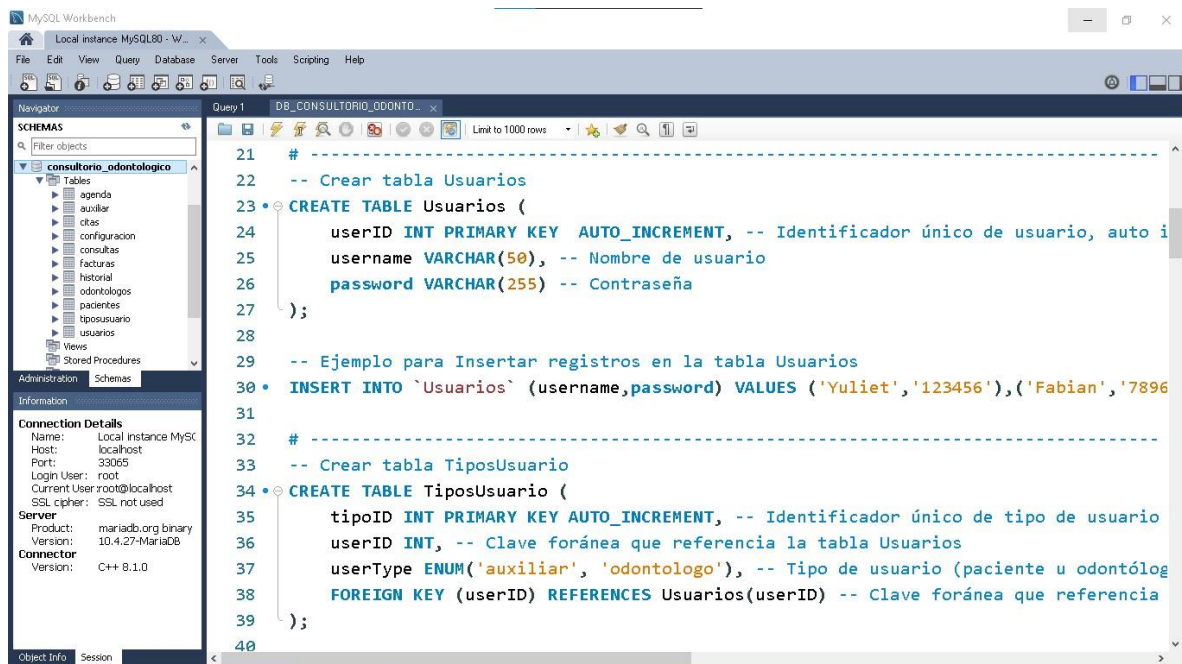
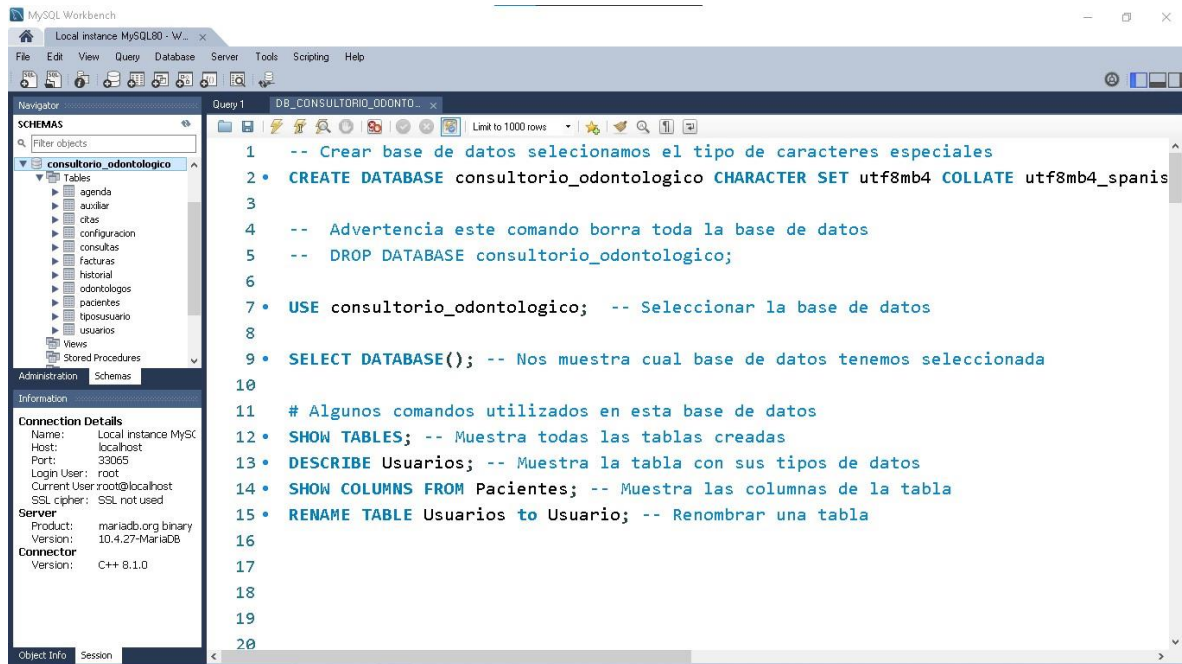
**consultaID INT, -- Clave foránea que referencia la tabla  
Consultas**

**fecha DATE, -- Fecha de la factura**

**total DECIMAL(10, 2), -- Total de la factura**

**FOREIGN KEY (consultaID) REFERENCES  
Consultas(consultaID) -- Clave foránea que referencia la tabla  
Consultas**

**);**



MySQL Workbench

Local instance MySQL80 - W...

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

consultorio\_odontologico

Tables

- agenda
- auxiliar
- citas
- configuracion
- consultas
- facturas
- historial
- odontologos
- pacientes
- tiposusuario
- usuarios

Views

Stored Procedures

Administration Schemas

Information

Connection Details

Name: Local instance MySQL

Host: localhost

Port: 33065

Login User: root

Current User: root@localhost

SSL cipher: SSL not used

Server

Product: mariadb.org binary

Version: 10.4.27-MariaDB

Connector

Version: C++ 8.1.0

Object Info Session

Query 1 DB\_CONSULTORIO\_ODONTO...

Limit to 1000 rows

```
41 # -----
42 -- Crear tabla Auxiliar
43 CREATE TABLE Auxiliar (
44     auxiliarID INT PRIMARY KEY AUTO_INCREMENT, -- Identificador único de auxiliar
45     userID INT, -- Clave foránea que referencia la tabla Usuarios
46     nombre VARCHAR(50), -- Nombre del auxiliar
47     apellido VARCHAR(50), -- Apellido del auxiliar
48     direccion VARCHAR(100), -- Dirección del auxiliar
49     telefono VARCHAR(15), -- Número de teléfono del auxiliar
50     ciudad VARCHAR(20), -- Ciudad de residencia
51     correo_electronico VARCHAR(100), -- Correo electrónico del auxiliar
52     FOREIGN KEY (userID) REFERENCES Usuarios(userID) -- Clave foránea que referencia
53 );
54
55
56
57
58
59
60
```

MySQL Workbench

Local instance MySQL80 - W...

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

consultorio\_odontologico

Tables

- agenda
- auxiliar
- citas
- configuracion
- consultas
- facturas
- historial
- odontologos
- pacientes
- tiposusuario
- usuarios

Views

Stored Procedures

Administration Schemas

Information

Connection Details

Name: Local instance MySQL

Host: localhost

Port: 33065

Login User: root

Current User: root@localhost

SSL cipher: SSL not used

Server

Product: mariadb.org binary

Version: 10.4.27-MariaDB

Connector

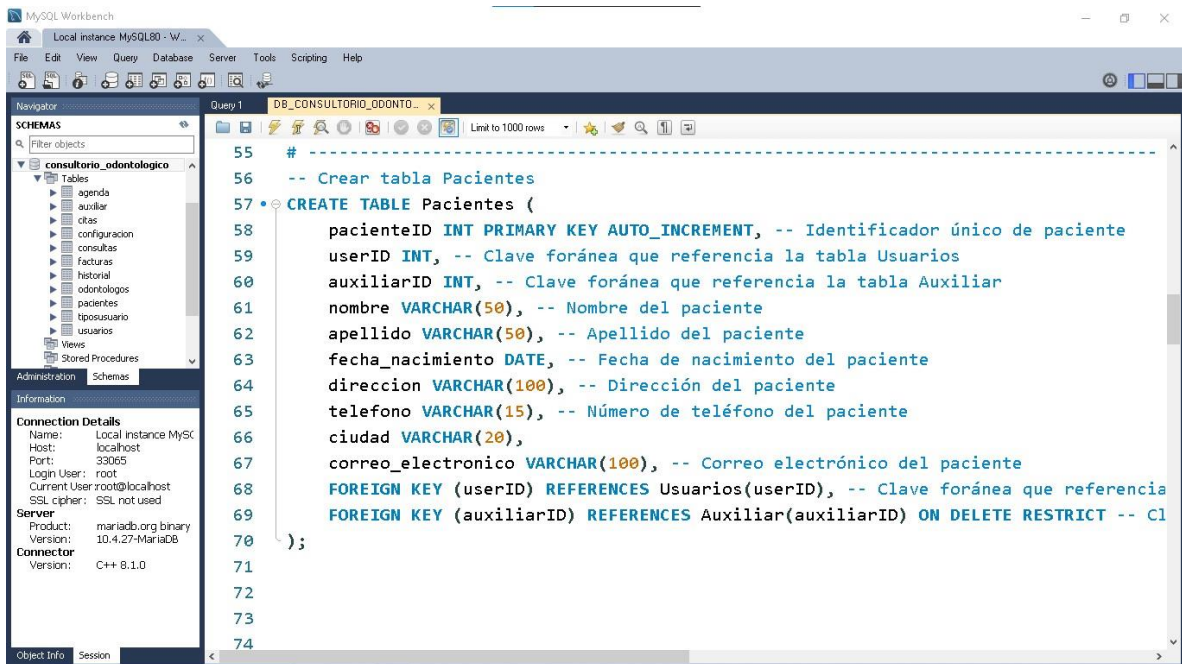
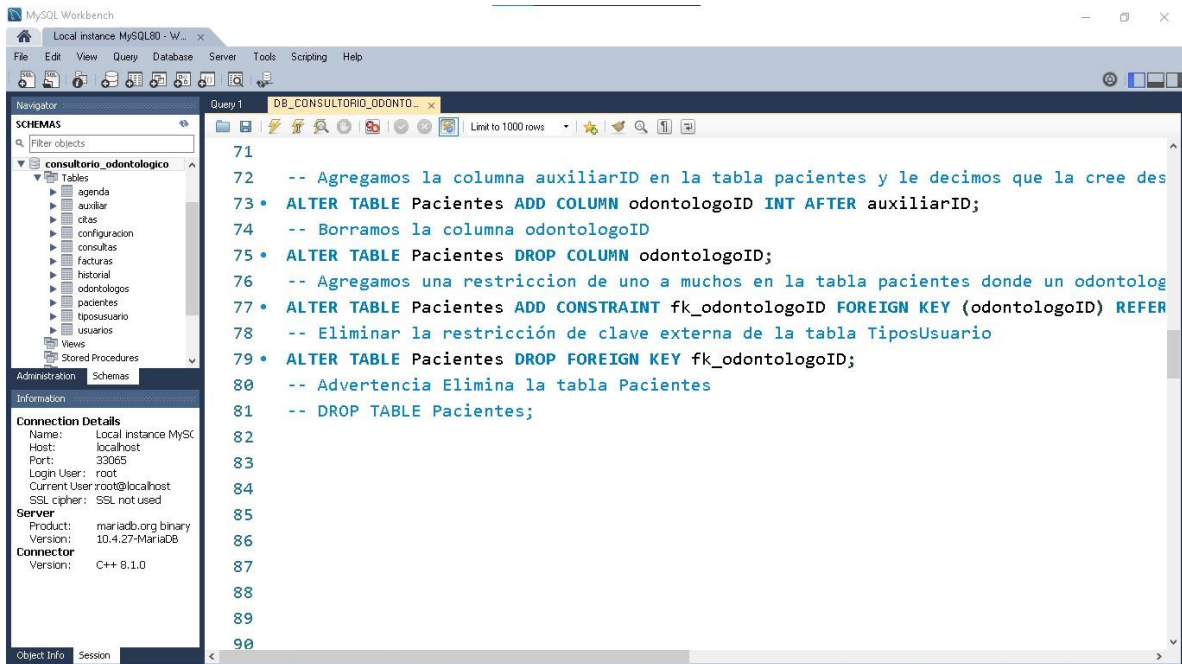
Version: C++ 8.1.0

Object Info Session

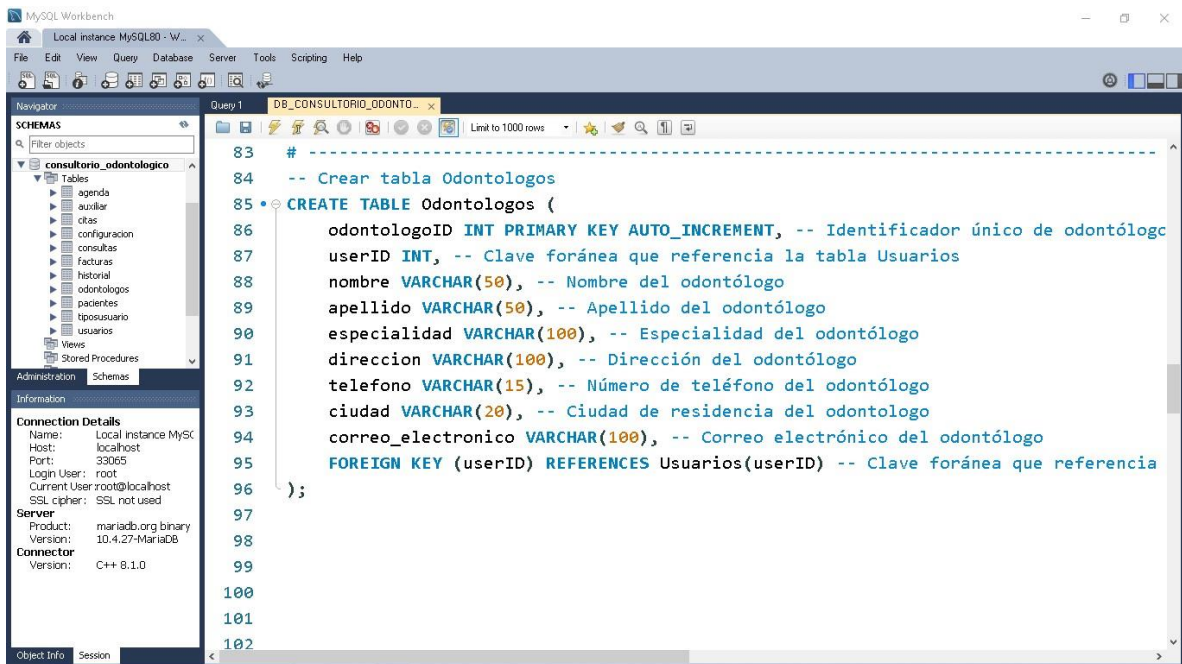
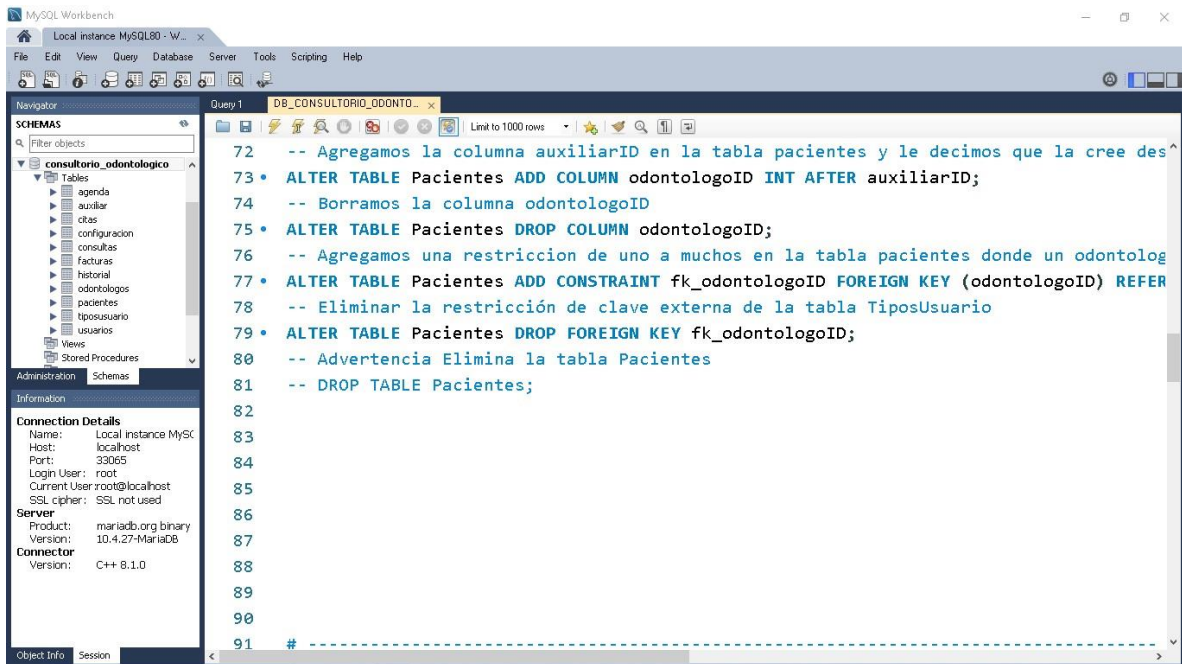
Query 1 DB\_CONSULTORIO\_ODONTO...

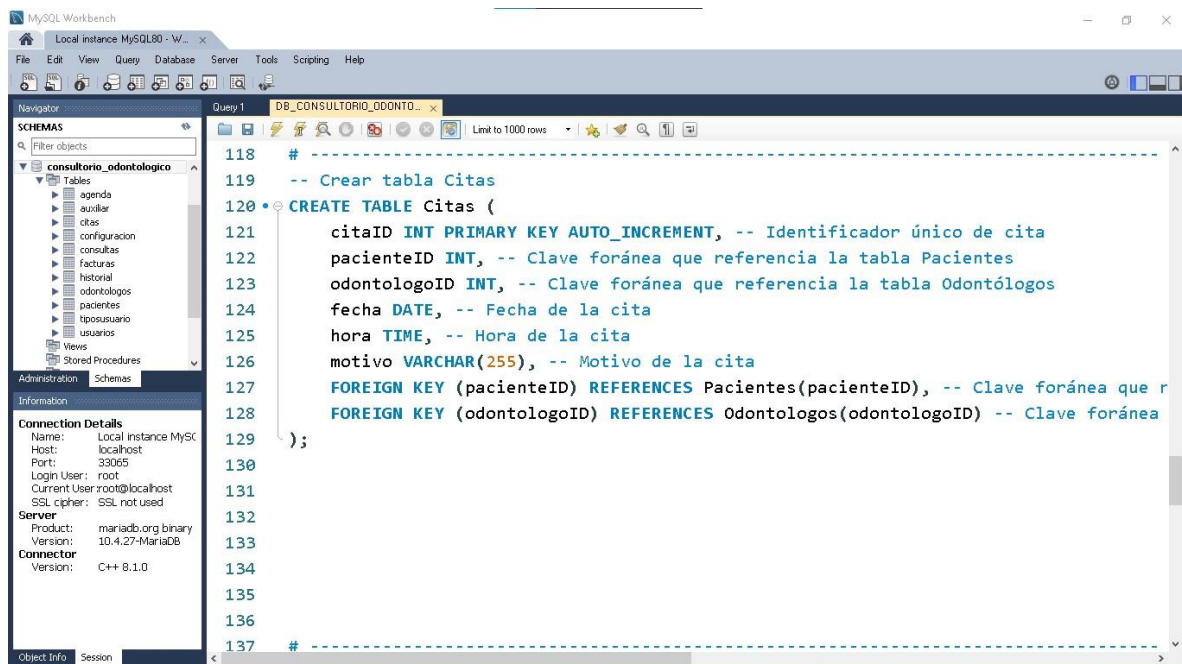
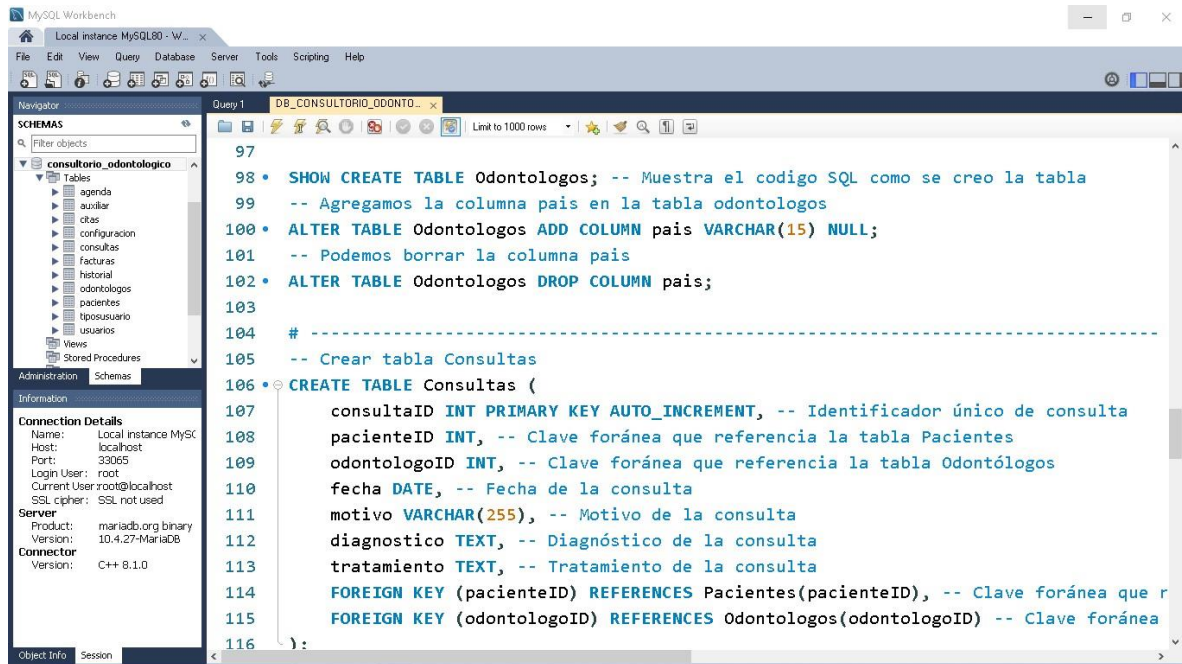
Limit to 1000 rows

```
55 # -----
56 -- Crear tabla Pacientes
57 CREATE TABLE Pacientes (
58     pacienteID INT PRIMARY KEY AUTO_INCREMENT, -- Identificador único de paciente
59     userID INT, -- Clave foránea que referencia la tabla Usuarios
60     auxiliarID INT, -- Clave foránea que referencia la tabla Auxiliar
61     nombre VARCHAR(50), -- Nombre del paciente
62     apellido VARCHAR(50), -- Apellido del paciente
63     fecha_nacimiento DATE, -- Fecha de nacimiento del paciente
64     direccion VARCHAR(100), -- Dirección del paciente
65     telefono VARCHAR(15), -- Número de teléfono del paciente
66     ciudad VARCHAR(20),
67     correo_electronico VARCHAR(100), -- Correo electrónico del paciente
68     FOREIGN KEY (userID) REFERENCES Usuarios(userID), -- Clave foránea que referencia
69     FOREIGN KEY (auxiliarID) REFERENCES Auxiliar(auxiliarID) ON DELETE RESTRICT -- Cl
70 );
71
72
73
74
```









The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' pane displays a tree view for the 'consultorio\_odontologico' database, listing tables like 'agenda', 'auxiliar', 'citas', 'configuracion', 'consultas', 'facturas', 'historial', 'odontologos', 'pacientes', 'tipousuario', and 'usuarios'. The 'Information' pane shows connection details for 'Local instance MySQL80'. The main editor window, titled 'Query 1', contains SQL code for creating two tables. The first table, 'Agenda', has columns: 'agendaID' (INT PRIMARY KEY AUTO\_INCREMENT), 'odontologoID' (INT, foreign key to 'Odontologos'), 'fecha' (DATE), 'hora\_inicio' (TIME), and 'hora\_fin' (TIME). The second table, 'Historial', has columns: 'historialID' (INT PRIMARY KEY AUTO\_INCREMENT), 'pacienteID' (INT, foreign key to 'Pacientes'), 'fecha' (DATE), and 'descripcion' (TEXT). The code is as follows:

```
132 -- Crear tabla Agenda
133 CREATE TABLE Agenda (
134     agendaID INT PRIMARY KEY AUTO_INCREMENT, -- Identificador único de agenda
135     odontologoID INT, -- Clave foránea que referencia la tabla Odontólogos
136     fecha DATE, -- Fecha de la agenda
137     hora_inicio TIME, -- Hora de inicio de la agenda
138     hora_fin TIME, -- Hora de finalización de la agenda
139     FOREIGN KEY (odontologoID) REFERENCES Odontologos(odontologoID) -- Clave foránea
140 );
141
142 # -----
143 -- Crear tabla Historial
144 CREATE TABLE Historial (
145     historialID INT PRIMARY KEY AUTO_INCREMENT, -- Identificador único de historial
146     pacienteID INT, -- Clave foránea que referencia la tabla Pacientes
147     fecha DATE, -- Fecha del historial
148     descripcion TEXT, -- Descripción del historial
149     FOREIGN KEY (pacienteID) REFERENCES Pacientes(pacienteID) -- Clave foránea que re
150 );
151
```

The screenshot shows the MySQL Workbench interface, similar to the first one. The 'SCHEMAS' pane shows the same database structure. The main editor window, titled 'Query 1', contains SQL code for creating the 'Configuracion' table. The table has columns: 'configID' (INT PRIMARY KEY AUTO\_INCREMENT), 'nombre\_clinica' (VARCHAR(100)), 'direccion' (VARCHAR(100)), 'telefono' (VARCHAR(15)), 'ciudad' (VARCHAR(20)), 'correo\_electronico' (VARCHAR(100)), and 'horario\_atencion' (VARCHAR(100)). The code is as follows:

```
152 # -----
153 -- Crear tabla Configuración
154 CREATE TABLE Configuracion (
155     configID INT PRIMARY KEY AUTO_INCREMENT, -- Identificador único de configuración
156     nombre_clinica VARCHAR(100), -- Nombre de la clínica
157     direccion VARCHAR(100), -- Dirección de la clínica
158     telefono VARCHAR(15), -- Número de teléfono de la clínica
159     ciudad VARCHAR(20),
160     correo_electronico VARCHAR(100), -- Correo electrónico de la clínica
161     horario_atencion VARCHAR(100) -- Horario de atención de la clínica
162 );
163
164
165
166
167
168
169
170
171 # -----
```

MySQL Workbench

Local instance MySQL80 - W...

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

consultorio\_odontologico

Tables

agenda

auxiliar

cit

configurac

consultas

facturas

historial

odontologos

pacientes

tiposusuario

usuarios

Views

Stored Procedures

Administration Schemas

Information

Connection Details

Name: Local instance MySQL

Host: localhost

Port: 33065

Login User: root

Current User: root@localhost

SSL cipher: SSL not used

Server

Product: mariadb.org binary

Version: 10.4.27-MariaDB

Connector

Version: C++ 8.1.0

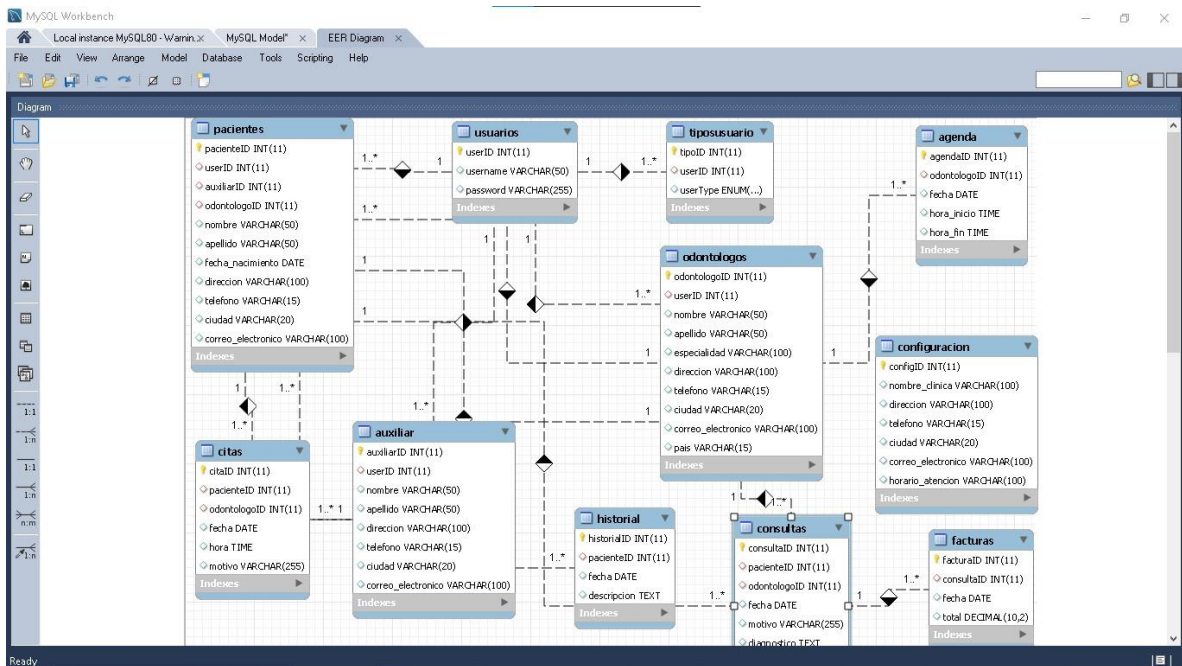
Object Info Session

Query 1

DB\_CONSULTORIO\_ODONTO...

Limit to 1000 rows

```
163
164 # -----
165 -- Crear tabla Facturas
166 CREATE TABLE Facturas (
167     facturaID INT PRIMARY KEY AUTO_INCREMENT, -- Identificador único de factura
168     consultaID INT, -- Clave foránea que referencia la tabla Consultas
169     fecha DATE, -- Fecha de la factura
170     total DECIMAL(10, 2), -- Total de la factura
171     FOREIGN KEY (consultaID) REFERENCES Consultas(consultaID) -- Clave foránea que re
172 );
173
174
175
176
177
178
179
180
181
```





phpMyAdmin

Reciente Favoritas

bd

consultorio

consultorio

consultorio\_odontologico

Nueva

agenda

auxiliar

citas

configuracion

consultas

facturas

historial

odontologos

pacientes

tiposusuario

usuarios

empresa

escuela

information\_schema

mysql

performance\_schema

phpmyadmin

posventa\_mvc

Sender: 127.0.0.1:33065 > Base de datos: consultorio\_odontologico

Estructura SQL Buscar Generar una consulta Exportar Importar Operaciones Privilegios Rutinas Más

Filtros

Que contengan la palabra:

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
<input type="checkbox"/> agenda	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> auxiliar	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> citas	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_general_ci	48.0 KB	-
<input type="checkbox"/> configuracion	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/> consultas	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_general_ci	48.0 KB	-
<input type="checkbox"/> facturas	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> historial	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> odontologos	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> pacientes	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_general_ci	64.0 KB	-
<input type="checkbox"/> tiposusuario	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> usuarios	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_general_ci	16.0 KB	-
11 tablas	Número de filas	0	InnoDB	utf8mb4_general_ci	384.0 KB	0 B

Consola Seleccionar todo Para los elementos que están marcados: ▼