Presentado por: Santiago Ramirez Arenas- Training SofkaU

**Taller BD 03**

**Comandos SQL (DDL DML)**

[*https://www.w3schools.com/mysql/mysql\_insert.asp*](https://www.w3schools.com/mysql/mysql_insert.asp)

***Primera actividad***

1. Elaborar el diagrama E-R del ejercicio de la librería (anexo)
2. Elaborar el modelo relación del punto 1.
3. Diagramar en workbench el punto 2.
4. Escribir con sentencias de SQL la representación del modelo relacional del punto 2.

***Segunda actividad***

1. Ingrese al repositorio de su compañeros según el listado excel adjunto.
2. Descargue el archivo PDF.
3. Corregir el diagrama en casa de considerar que tenga algun error.
4. Diseñar nuevamente el Modelo relacional en caso de que el punto 3 haya sufrido una modificación.
5. Escribir los cambios realizados en el paso 3, o un comentario de que no se cambia nada.
6. Diagramar en workbench el punto.
7. Escribir con sentencias de SQL la representación del modelo del 6.

***Tercera actividad.***

1. Poblar la base de datos creada de las actividades 1 y 2 con mínimo 2 registros por tabla. empleando instrucciones de SQL o por medio de Workbench

***Condiciones de la entrega***

1. PR apuntando a la rama solución
2. PR con su nombre completo
3. documento PDF del repositorio de su compañero y uno propio con las conclusiones o el análisis
4. mínimo 6 commits
5. Hora de entrega 11.55 PM

"we make it simple"

**Solución punto uno**

Explicación relaciones y entidades identificados en el ejercicio

* Una editorial puede publicar uno o más libros, mientras que un libro puede ser publicado por una única editorial, por lo tanto tiene una cardinalidad de (1,N).
* Un libro puede ser comprado por uno o varios clientes, mientras que un cliente puede comprar uno o varios libros, por lo tanto tiene una cardinalidad de (M,N).
* Un autor puede escribir uno o varios libros, mientras que un libro puede ser escrito por uno o varios autores (en caso de existir una colaboración) , por lo tanto tiene una cardinalidad de (M,N).

A continuación, se mostrará la solución del punto 1

Imagen 1: Modelo entidad relación ejercicio librería

Diagrama

Descripción generada automáticamente

Se crean las dos tablas correspondientes a Escribir y Comprar, dado que se generan a partir de una relación muchos a muchos (N,M) y se asignan las llaves primarias de las dos entidades que participan en cada relación mencionada anteriormente en el modelo entidad relación, además se crea la tabla TelefonoCliente dado que surge a partir de una atributo multivaluado.

Imagen 2 : Modelo relacional ejercicio librería

Diagrama

Descripción generada automáticamente

Normalización

La base de datos anterior cumple con la normalización porque se han implementado las siguientes reglas de normalización:

1. Primera Forma Normal (1NF): Cada tabla tiene una llave primaria única que identifica de manera única a cada registro.
2. Segunda Forma Normal (2NF): Cada tabla está libre de dependencias parciales, lo que significa que cada atributo en una tabla depende completamente de la llave primaria de esa tabla.
3. Tercera Forma Normal (3NF): No hay dependencias transitivas, lo que significa que no hay atributos que dependan de otro atributo en lugar de depender de la llave primaria.

Además, se han creado tablas adicionales para mantener la integridad de los datos, lo que ayuda a evitar la redundancia y el desperdicio de espacio en la base de datos. Por ejemplo, la tabla "Escribir" se usa para mantener la información sobre los autores y los libros que escribieron, mientras que la tabla "Comprar" se usa para mantener la información sobre los clientes y los libros que compraron.

En resumen, esta base de datos se ha diseñado siguiendo los principios de normalización, lo que ayuda a garantizar la integridad de los datos, mejorar la eficiencia de la base de datos y evitar problemas como la redundancia y la pérdida de información.

Principio del formulario

Final del formulario

Se realiza el modelo relacional en Workbench teniendo en cuenta las conexiones con las llaves foráneas. Para dar practicidad al ejercicio se le asignó a los campos de VAR CHAR un valor de 45, en un ejercicio real tendría que adaptarse este valor de una mejor forma a cada contexto, asignando el valor máximo según los datos esperados de quien use la base de datos.

Imagen 3: Modelo relacional ejercicio librería en Workbench

Interfaz de usuario gráfica, Diagrama

Descripción generada automáticamente

Código generado a partir de la opción *Forward Engineer* que proporciona *Workbench*

-- MySQL Workbench Forward Engineering

**SET** @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=**0**;

**SET** @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=**0**;

**SET** @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

-- -----------------------------------------------------

-- Schema LibreriaModeloRelacionalWorbench

-- -----------------------------------------------------

-- -----------------------------------------------------

-- Schema LibreriaModeloRelacionalWorbench

-- -----------------------------------------------------

**CREATE** **SCHEMA** IF **NOT** **EXISTS** `LibreriaModeloRelacionalWorbench` **DEFAULT** CHARACTER **SET** utf8 ;

USE `LibreriaModeloRelacionalWorbench` ;

-- -----------------------------------------------------

-- Table `LibreriaModeloRelacionalWorbench`.`Editorial`

-- -----------------------------------------------------

**CREATE** **TABLE** IF **NOT** **EXISTS** `LibreriaModeloRelacionalWorbench`.`Editorial` (

`Nombre` VARCHAR(**45**) **NOT** **NULL**,

`ciudad` VARCHAR(**45**) **NOT** **NULL**,

`complemento` VARCHAR(**45**) **NOT** **NULL**,

**PRIMARY** **KEY** (`Nombre`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `LibreriaModeloRelacionalWorbench`.`Libro`

-- -----------------------------------------------------

**CREATE** **TABLE** IF **NOT** **EXISTS** `LibreriaModeloRelacionalWorbench`.`Libro` (

`ISBN` VARCHAR(**45**) **NOT** **NULL**,

`NombreEditorial` VARCHAR(**45**) **NOT** **NULL**,

`NumerPaginas` INT **NOT** **NULL**,

`FechaPublicacion` VARCHAR(**45**) **NOT** **NULL**,

`Titulo` VARCHAR(**45**) **NOT** **NULL**,

**PRIMARY** **KEY** (`NombreEditorial`),

**INDEX** `NombreEditorial\_idx` (`ISBN` **ASC**) VISIBLE,

**CONSTRAINT** `NombreEditorial`

**FOREIGN** **KEY** (`ISBN`)

**REFERENCES** `LibreriaModeloRelacionalWorbench`.`Editorial` (`Nombre`)

**ON** **DELETE** **NO** ACTION

**ON** **UPDATE** **NO** ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `LibreriaModeloRelacionalWorbench`.`Cliente`

-- -----------------------------------------------------

**CREATE** **TABLE** IF **NOT** **EXISTS** `LibreriaModeloRelacionalWorbench`.`Cliente` (

`IdCliente` VARCHAR(**45**) **NOT** **NULL**,

`FechaNacimiento` VARCHAR(**45**) **NOT** **NULL**,

`Apeliido` VARCHAR(**45**) **NOT** **NULL**,

`Nombre` VARCHAR(**45**) **NOT** **NULL**,

**PRIMARY** **KEY** (`IdCliente`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `LibreriaModeloRelacionalWorbench`.`Comprar`

-- -----------------------------------------------------

**CREATE** **TABLE** IF **NOT** **EXISTS** `LibreriaModeloRelacionalWorbench`.`Comprar` (

`ISBNLibroComprar` VARCHAR(**45**) **NOT** **NULL**,

`IdClienteComprar` VARCHAR(**45**) **NOT** **NULL**,

**PRIMARY** **KEY** (`IdClienteComprar`, `ISBNLibroComprar`),

**INDEX** `ISBNLibro\_idx` (`ISBNLibroComprar` **ASC**) VISIBLE,

**CONSTRAINT** `ISBNLibroComprar`

**FOREIGN** **KEY** (`ISBNLibroComprar`)

**REFERENCES** `LibreriaModeloRelacionalWorbench`.`Libro` (`ISBN`)

**ON** **DELETE** **NO** ACTION

**ON** **UPDATE** **NO** ACTION,

**CONSTRAINT** `IdClienteComprar`

**FOREIGN** **KEY** (`IdClienteComprar`)

**REFERENCES** `LibreriaModeloRelacionalWorbench`.`Cliente` (`IdCliente`)

**ON** **DELETE** **NO** ACTION

**ON** **UPDATE** **NO** ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `LibreriaModeloRelacionalWorbench`.`Autor`

-- -----------------------------------------------------

**CREATE** **TABLE** IF **NOT** **EXISTS** `LibreriaModeloRelacionalWorbench`.`Autor` (

`CCAutor` VARCHAR(**45**) **NOT** **NULL**,

`Apeliido` VARCHAR(**45**) **NOT** **NULL**,

`Nombre` VARCHAR(**45**) **NOT** **NULL**,

`FechaNacimiento` VARCHAR(**45**) **NOT** **NULL**,

**PRIMARY** **KEY** (`CCAutor`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `LibreriaModeloRelacionalWorbench`.`Escribir`

-- -----------------------------------------------------

**CREATE** **TABLE** IF **NOT** **EXISTS** `LibreriaModeloRelacionalWorbench`.`Escribir` (

`ISBNLibroEscribir` VARCHAR(**45**) **NOT** **NULL**,

`CCAutorEscribir` VARCHAR(**45**) **NOT** **NULL**,

**PRIMARY** **KEY** (`ISBNLibroEscribir`, `CCAutorEscribir`),

**INDEX** `CCAutorEscribir\_idx` (`CCAutorEscribir` **ASC**) VISIBLE,

**CONSTRAINT** `ISBNLibroEscribir`

**FOREIGN** **KEY** (`ISBNLibroEscribir`)

**REFERENCES** `LibreriaModeloRelacionalWorbench`.`Libro` (`ISBN`)

**ON** **DELETE** **NO** ACTION

**ON** **UPDATE** **NO** ACTION,

**CONSTRAINT** `CCAutorEscribir`

**FOREIGN** **KEY** (`CCAutorEscribir`)

**REFERENCES** `LibreriaModeloRelacionalWorbench`.`Autor` (`CCAutor`)

**ON** **DELETE** **NO** ACTION

**ON** **UPDATE** **NO** ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `LibreriaModeloRelacionalWorbench`.`TelefonoCliente`

-- -----------------------------------------------------

**CREATE** **TABLE** IF **NOT** **EXISTS** `LibreriaModeloRelacionalWorbench`.`TelefonoCliente` (

`idTelefonoCliente` VARCHAR(**15**) **NOT** **NULL**,

`TelefonoCliente` VARCHAR(**45**) **NOT** **NULL**,

**PRIMARY** **KEY** (`idTelefonoCliente`, `TelefonoCliente`),

**CONSTRAINT** `idTelefonoCliente`

**FOREIGN** **KEY** (`idTelefonoCliente`)

**REFERENCES** `LibreriaModeloRelacionalWorbench`.`Cliente` (`IdCliente`)

**ON** **DELETE** **NO** ACTION

**ON** **UPDATE** **NO** ACTION)

ENGINE = InnoDB;

**SET** SQL\_MODE=@OLD\_SQL\_MODE;

**SET** FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

**SET** UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

**CÓDIGO CREADO DE FORMA MANUAL**

**CREATE** **DATABASE** IF **NOT** **EXISTS** LibreriaSanty;

USE LibreriaSanty;

--USE base\_peliculas;

**CREATE** **TABLE** Autor (

CCAutor VARCHAR(**45**) **PRIMARY** **KEY**,

Apellido VARCHAR(**45**) **NOT** **NULL**,

Nombre VARCHAR(**45**) **NOT** **NULL**,

FechaNacimiento VARCHAR(**45**) **NOT** **NULL**

);

**CREATE** **TABLE** Editorial (

Nombre VARCHAR(**45**) **PRIMARY** **KEY**,

Ciudad VARCHAR(**45**) **NOT** **NULL**,

Complemento VARCHAR(**45**) **NOT** **NULL**

);

**CREATE** **TABLE** Libro (

ISBN VARCHAR(**45**) **PRIMARY** **KEY**,

NombreEditorial VARCHAR(**45**) **NOT** **NULL**,

NumeroPaginas INT **NOT** **NULL**,

FechaPublicacion DATE **NOT** **NULL**,

Titulo VARCHAR(**45**) **NOT** **NULL**,

**FOREIGN** **KEY** (NombreEditorial) **REFERENCES** Editorial(Nombre)

);

**CREATE** **TABLE** Escribir (

ISBNLibroEscribir VARCHAR(**45**) **NOT** **NULL**,

CCAutorEscribir VARCHAR(**45**) **NOT** **NULL**,

**PRIMARY** **KEY** (ISBNLibroEscribir, CCAutorEscribir),

**FOREIGN** **KEY** (ISBNLibroEscribir) **REFERENCES** Libro(ISBN),

**FOREIGN** **KEY** (CCAutorEscribir) **REFERENCES** Autor(CCAutor)

);

**CREATE** **TABLE** Cliente (

IdCliente VARCHAR(**45**) **PRIMARY** **KEY**,

FechaNacimiento DATE **NOT** **NULL**,

Apellido VARCHAR(**45**) **NOT** **NULL**,

Nombre VARCHAR(**45**) **NOT** **NULL**

);

**CREATE** **TABLE** Comprar (

ISBNLibroComprar VARCHAR(**45**) **NOT** **NULL**,

IdClienteComprar VARCHAR(**45**) **NOT** **NULL**,

**PRIMARY** **KEY** (ISBNLibroComprar, IdClienteComprar),

**FOREIGN** **KEY** (ISBNLibroComprar) **REFERENCES** Libro (ISBN),

**FOREIGN** **KEY** (IdClienteComprar) **REFERENCES** Cliente (IdCliente)

);

**CREATE** **TABLE** TelefonoCliente (

TelefonoCliente VARCHAR(**45**) **NOT** **NULL**,

IdCliente VARCHAR(**45**) **NOT** **NULL**,

**PRIMARY** **KEY** (TelefonoCliente),

**FOREIGN** **KEY** (IdCliente) **REFERENCES** Cliente (IdCliente)

);

**INSERCIÓN DE REGISTROS DE FORMA MANUAL**

-- Tabla Autor

**INSERT** **INTO** Autor (CCAutor, Apellido, Nombre, FechaNacimiento)

**VALUES**

("A001", "Martinez", "Juan", "1980-01-01"),

("A002", "Gonzalez", "Pedro", "1982-03-05"),

("A003", "Rodriguez", "Miguel", "1983-04-08"),

("A004", "Perez", "Luis", "1981-02-11"),

("A005", "Sanchez", "Jorge", "1979-12-25");

-- Tabla Editorial

**INSERT** **INTO** Editorial (Nombre, Ciudad, Complemento)

**VALUES**

("Editorial 1", "Madrid", "España"),

("Editorial 2", "Barcelona", "España"),

("Editorial 3", "Valencia", "España"),

("Editorial 4", "Sevilla", "España"),

("Editorial 5", "Bilbao", "España");

-- Tabla Libro

**INSERT** **INTO** Libro (ISBN, NombreEditorial, NumeroPaginas, FechaPublicacion, Titulo)

**VALUES**

("L001", "Editorial 1", **300**, "2010-01-01", "Libro 1"),

("L002", "Editorial 2", **250**, "2011-02-01", "Libro 2"),

("L003", "Editorial 3", **200**, "2012-03-01", "Libro 3"),

("L004", "Editorial 4", **350**, "2013-04-01", "Libro 4"),

("L005", "Editorial 5", **400**, "2014-05-01", "Libro 5");

-- Tabla Escribir

**INSERT** **INTO** Escribir (ISBNLibroEscribir, CCAutorEscribir)

**VALUES**

("L001", "A001"),

("L002", "A002"),

("L003", "A003"),

("L004", "A004"),

("L005", "A005");

-- Tabla Cliente

**INSERT** **INTO** Cliente (IdCliente, FechaNacimiento, Apellido, Nombre)

**VALUES**

("C001", "1985-01-01", "Lopez", "Juan"),

("C002", "1986-02-01", "Garcia", "Pedro"),

("C003", "1987-03-01", "Ruiz", "Miguel"),

("C004", "1988-04-01", "Jimenez", "Luis"),

("C005", "1989-05-01", "Ramirez", "Jorge");

-- Tabla Comprar

**INSERT** **INTO** Comprar (ISBNLibroComprar, IdClienteComprar)

**VALUES**

("L001", "C001"),

("L002", "C002"),

("L003", "C003"),

("L004", "C004"),

("L005", "C005");

-- Tabla TelefonoCliente

**INSERT** **INTO** TelefonoCliente (TelefonoCliente, IdCliente)

**VALUES**

("555-555-5555", "C001"),

("555-555-5556", "C002"),

("555-555-5557", "C003"),

("555-555-5558", "C004"),

("555-555-5559", "C005");

Para evidenciar la creación de la base de datos y la creación de los registros adjunto captura de pantalla(Solo se adjunta captura de pantalla de una sola tabla ya que son demasiadas y eso solo generaría un desorden innecesario)

**Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamenteImagen 4:** Creación de las tablas a partir de sentencias SQL

**Imagen 5:** Registros creados exitosamente a partir de sentencias SQL en la tabla Libro

**Tabla

Descripción generada automáticamente**

**SOLUCIÓN PUNTO DOS**

Nombre del compañero al que debo analizar el trabajo realizado: **Eros José Adarraga**

Diagrama

Descripción generada automáticamente**Imagen 6:** Modelo entidad relación Eros José Adarraga

Revisando el modelo entidad relación hay cierto orden de las relaciones que no concuerdan con mi lógica, dado que el planteamiento de una base de datos es muy conceptual decidí preguntarle con la lógica con la cual la había planteado.

En su momento yo había imaginado que una vez que un cliente compra el brazalete se le daba acceso a las atracciones, mi compañero me explicó su idea y es que una vez que un cliente compra un brazalete puede entrar a la atracción por medio de este brazalete (como si un empleado estuviera en la entrada revisando), por lo tanto su idea me parece acertada y no cambiaría esto dado que yo lo pensé más desde el lado del proceso de compra del brazalete y mi compañero lo vio desde el lado de acceso a las atracciones por medio del brazalete, es decir del funcionamiento como tal del parque.

Otro punto es que yo percibí que un técnico recibía la orden de servicio y después ese mismo técnico realizaba lo que dice la orden de servicio, mi compañero lo interpretó como que el mismo técnico tiene la autonomía de generar la orden de servicio y después por medio de la orden se genera el mantenimiento, así que también me parece que es un punto acertado por parte de mi compañero.

Un aspecto a cambiar es el verbo de la relación entre proveedor y atracción, en lugar de comprar, pondría “Proveer” ya que suena un poco fuera de lugar leer “Un proveedor compra una atracción” suena más natural el verbo “Proveer”, ya que esa es su función, proveer.

Y por último, en el atributo **Tipo**, mi compañero propuso en el documento que fuera nacional o internacional, pero más adelante en el diagrama relacional puso Nacionalidad, por lo tanto el atributo debe ser marcada como multivalor y cambiar su nombre a Nacionalidad , ya que un proveedor bien puede tener varias nacionalidad (Una multinacional).

En general está muy bien, entender como planteó mi compañero el ejercicio me fue de mucha ayuda, tan uno unos errores muy pequeños de gramática en cuanto a palabras que pueden resultar un poco confusas.

En cuanto a la cardinalidad, los atributos y la dependencia veo que todo está muy bien.

**Imagen 7:** Observaciones modelo entidad relación Eros José Adarraga

Diagrama

Descripción generada automáticamente

**Imagen 8:** Modelo relacional Eros José Adarraga

Diagrama

Descripción generada automáticamente

Se verificó el cambio del nombre del atributo y se creó la tabla para el atributo multivalor, además se borró la tabla “Nombre Completo” y se agregó como un atributo simple ya que en lo personal prefiero tener el nombre como un atributo simple, es decir Primer Nombre, Segundo Nombre, Primer Apellido, Segundo Apellido, dentro de la entidad misma. Para el ejercicio simplemente se dejará el atributo “Nombre” por cuestión de tiempo, **resaltando que es un atributo simple** y que solo se pondrá el nombre por cuestión de tiempo.

**Imagen 9:** Nuevo modelo relacional Eros José Adarraga

Diagrama

Descripción generada automáticamente

**Normalización**

La base de datos anterior cumple con la Primera Forma Normal (1FN) porque cada tabla tiene una llave primaria única que identifica cada registro de manera única y cada columna en una tabla contiene valores atómicos, es decir, no se repiten o están divididos en otras columnas.

Además, cumple con la Segunda Forma Normal (2FN) porque cada no-llave primaria depende funcionalmente de toda la llave primaria y no de una parte de ella. Esto significa que los atributos de una tabla están relacionados directamente con la llave primaria y no con otras columnas que no son parte de la llave primaria.

La base de datos cumple con la tercera forma normal ya que no tiene valores dependientes transitorios, es decir, un valor en una columna no debe ser determinado por un valor en otra columna y además no contiene valores dependientes no primarios, es decir, un valor en una columna no debe ser determinado por un valor en otra columna que no sea la llave primaria.

Diagrama

Descripción generada automáticamente**Imagen 10:** Diagrama relacional en Workbench

**CODIGO GENERADO MANUALMENTE**

**CREATE** **DATABASE** IF **NOT** **EXISTS** ParqueSanty;

USE ParqueSanty;

**CREATE** **TABLE** Tecnico (

TecnicoID varchar(**20**) **PRIMARY** **KEY**,

Nombre varchar(**20**) **NOT** **NULL**,

Telefono varchar(**20**) **NOT** **NULL**

);

**CREATE** **TABLE** OrdenServicio (

IdOrden varchar(**20**) **PRIMARY** **KEY**,

Fecha varchar(**20**) **NOT** **NULL**,

Duracion varchar(**20**) **NOT** **NULL**,

TecnicoID varchar(**20**) **NOT** **NULL**,

**FOREIGN** **KEY** (TecnicoID) **REFERENCES** Tecnico(TecnicoID)

);

**CREATE** **TABLE** Reparar (

IdOrden varchar(**20**) **NOT** **NULL**,

IdAtraccion varchar(**20**) **NOT** **NULL**,

**PRIMARY** **KEY** (IdOrden, IdAtraccion),

**FOREIGN** **KEY** (IdOrden) **REFERENCES** OrdenServicio(IdOrden),

**FOREIGN** **KEY** (IdAtraccion) **REFERENCES** Atraccion(IdAtraccion)

);

**CREATE** **TABLE** Atraccion (

IdAtraccion varchar(**20**) **PRIMARY** **KEY**,

Marca varchar(**20**) **NOT** **NULL**,

Costo varchar(**20**) **NOT** **NULL**,

Nombre varchar(**20**) **NOT** **NULL**,

FechaAdquision varchar(**20**) **NOT** **NULL**,

IdEmpleado varchar(**20**) **NOT** **NULL**,

IdProveedor varchar(**20**) **NOT** **NULL**,

**FOREIGN** **KEY** (IdEmpleado) **REFERENCES** Empleado(IdEmpleado),

**FOREIGN** **KEY** (IdProveedor) **REFERENCES** Proveedor(IdProveedor)

);

**CREATE** **TABLE** Empleado (

IdEmpleado varchar(**20**) **PRIMARY** **KEY**,

Telefono varchar(**20**) **NOT** **NULL**,

Nombre varchar(**20**) **NOT** **NULL**,

Direccion varchar(**20**) **NOT** **NULL**,

AreaTrabajo varchar(**20**) **NOT** **NULL**

);

**CREATE** **TABLE** Proveedor (

IdProveedor varchar(**20**) **PRIMARY** **KEY**,

Nombre varchar(**20**) **NOT** **NULL**,

Direccion varchar(**20**) **NOT** **NULL**,

Telefono varchar(**20**) **NOT** **NULL**

);

**CREATE** **TABLE** Nacionalidad (

IdNacionalidad varchar(**20**) **PRIMARY** **KEY**,

Colombia varchar(**20**) **NOT** **NULL**,

Mexico varchar(**20**) **NOT** **NULL**,

**FOREIGN** **KEY** (IdNacionalidad) **REFERENCES** Proveedor(IdProveedor)

);

**CREATE** **TABLE** Entrar (

IdEntrar varchar(**20**) **PRIMARY** **KEY**,

IdBrazalete varchar(**20**) **NOT** **NULL**,

IdAtraccion varchar(**20**) **NOT** **NULL**,

FechaIngreso varchar(**20**),

**FOREIGN** **KEY** (IdAtraccion) **REFERENCES** Atraccion(IdAtraccion),

**FOREIGN** **KEY** (IdBrazalete) **REFERENCES** Brazalete(IdBrazalete)

);

**CREATE** **TABLE** Brazalete (

IdBrazalete VARCHAR(**20**) **PRIMARY** **KEY**,

Precio VARCHAR(**20**),

FechaCompra VARCHAR(**20**),

Tipo VARCHAR(**20**),

IdCliente VARCHAR(**20**),

**FOREIGN** **KEY** (IdCliente) **REFERENCES** Cliente(IdCliente)

);

**CREATE** **TABLE** Cliente (

IdCliente VARCHAR(**20**) **PRIMARY** **KEY**,

Edad VARCHAR(**20**),

Altura VARCHAR(**20**),

Nombre VARCHAR(**20**)

);

**CREATE** **TABLE** EspecializacionTecnico (

ID varchar(**20**) **PRIMARY** **KEY**,

Electrico varchar(**20**) **NOT** **NULL**,

Mecanico varchar(**20**) **NOT** **NULL**,

TecnicoID varchar(**20**) **NOT** **NULL**,

**FOREIGN** **KEY** (TecnicoID) **REFERENCES** Tecnico(TecnicoID)

);

**INGRESO DE REGISTROS**

-- Inserciones de registros

**INSERT** **INTO** Cliente (IdCliente, Edad, Altura, Nombre) **VALUES** ('C001', '30', '1.70', 'Pedro');

**INSERT** **INTO** Cliente (IdCliente, Edad, Altura, Nombre) **VALUES** ('C002', '25', '1.60', 'Ana');

**INSERT** **INTO** Brazalete (IdBrazalete, Precio, FechaCompra, IdCliente)**VALUES** ('B001', '10', '2022-01-01', 'C001');

**INSERT** **INTO** Brazalete (IdBrazalete, Precio, FechaCompra, IdCliente) **VALUES** ('B002', '20', '2022-01-01', 'C002');

**INSERT** **INTO** Tecnico (TecnicoID, Nombre, Telefono) **VALUES** ('1', 'Juan Perez', '3001234567');

**INSERT** **INTO** Tecnico (TecnicoID, Nombre, Telefono) **VALUES** ('2', 'Maria Rodriguez', '3002345678');

**INSERT** **INTO** OrdenServicio (IdOrden, Fecha, Duracion, TecnicoID) **VALUES** ('1', '2022-01-01', '5 horas', '1');

**INSERT** **INTO** OrdenServicio (IdOrden, Fecha, Duracion, TecnicoID) **VALUES** ('2', '2022-02-01', '7 horas', '2');

**INSERT** **INTO** Reparar (IdOrden, IdAtraccion) **VALUES** ('1', '1');

**INSERT** **INTO** Reparar (IdOrden, IdAtraccion) **VALUES** ('2', '2');

**INSERT** **INTO** Atraccion (IdAtraccion, Marca, Costo, Nombre, FechaAdquision, IdEmpleado, IdProveedor) **VALUES** ('1', 'Disney', '1000000', 'Montaña Rusa', '2020-01-01', '1', '1');

**INSERT** **INTO** Atraccion (IdAtraccion, Marca, Costo, Nombre, FechaAdquision, IdEmpleado, IdProveedor) **VALUES** ('2', 'Universal', '2000000', 'Piscina de olas', '2021-01-01', '2', '2');

**INSERT** **INTO** Empleado (IdEmpleado, Telefono, Nombre, Direccion, AreaTrabajo) **VALUES** ('1', '3009876543', 'Pedro Martinez', 'Calle 123', 'Mantenimiento');

**INSERT** **INTO** Empleado (IdEmpleado, Telefono, Nombre, Direccion, AreaTrabajo) **VALUES** ('2', '3006789012', 'Sofia Gomez', 'Calle 456', 'Atención al cliente');

**INSERT** **INTO** Proveedor (IdProveedor, Nombre, Direccion, Telefono) **VALUES** ('1', 'Proveedor 1', 'Calle 789', '3003456789');

**INSERT** **INTO** Proveedor (IdProveedor, Nombre, Direccion, Telefono) **VALUES** ('2', 'Proveedor 2', 'Calle 101112', '3004567890');

**INSERT** **INTO** Nacionalidad (IdNacionalidad, Colombia, Mexico) **VALUES** ('1', 'Si', 'No');

**INSERT** **INTO** Nacionalidad (IdNacionalidad, Colombia, Mexico) **VALUES** ('2', 'No', 'Si');

**INSERT** **INTO** Entrar (IdEntrar, IdAtraccion, IdBrazalete) **VALUES** ('1', '1', '1');

**INSERT** **INTO** Entrar (IdEntrar, IdAtraccion, IdBrazalete) **VALUES** ('2', '2', '2');

**INSERT** **INTO** EspecializacionTecnico(ID, Electrico,Mecanico , TecnicoID) **VALUES** ('1-23A', 'si', 'no', '1');

**INSERT** **INTO** EspecializacionTecnico (ID,Electrico, Mecanico, TecnicoID) **VALUES** ('1-23-B', 'no', 'si', '2');

Para términos de practicidad se adjunta captura de la una sola tabla, en el código anteriormente visto se puede tener una referencia de todos los registros ingresados.

**Imagen 11:** Captura de los registros ingresados a una de las tablas

