**UNIVERSIDAD DE CARTAGENA**

**Facultad de Ingeniería - Programa de Ingeniería de Software**

**Asignatura: Bases de Datos 1 - Unidad 1**

**Actividad de Laboratorio Práctico**

**Estudiante:** Oscar Andres Mercado Rodelo  
**Docente:** John Carlos Arrieta Arrieta  
**Fecha:** abril 2025

**Tabla de Contenido**

1. Introducción
2. Objetivos
3. Justificación
4. Instalación de motores de base de datos
5. Desarrollo de Evidencias
6. Bibliografía

**1. Introducción**

Este documento presenta el desarrollo de la parte práctica de la Unidad 1 de la asignatura Bases de Datos 1. A partir del modelo relacional definido para un sistema de información hospitalario, se ejecutan sentencias SQL en el entorno PostgreSQL, documentando cada operación mediante capturas de pantalla, comandos utilizados y su respectiva explicación.

**2. Objetivos**

* Implementar el modelo relacional en PostgreSQL y MySQL.
* Documentar paso a paso las acciones realizadas en la terminal.
* Comprender el funcionamiento de las instrucciones SQL básicas de administración de bases de datos.

**3. Justificación**

Este laboratorio permite aplicar los conocimientos teóricos sobre bases de datos relacionales en un entorno práctico realista. Facilita el dominio de comandos en línea de comandos (CLI) y fortalece el entendimiento de la estructura y relación entre tablas.

**4. Instalación de motores de bases de datos**

* **Instalación de motor PostgreSQL**

<https://youtu.be/vmOscfLIQAg>

* **Instalación de motor MySQL**

<https://youtu.be/7QrXVmCI2P0>

**4. DESARROLLO DE EVIDENCIAS**

1. **Identificar conjunto de entidades entidades y sus atributos**

**Entidad: MÉDICO**

* id\_medico (PK)
* nombre
* direccion
* telefono
* poblacion
* provincia
* codigo\_postal
* nif
* num\_seguridad\_social
* num\_colegiado
* categoria (titular, interino, sustituto)
* **Relación con horarios de consulta**
* **Relación con sustituciones (fechas alta/baja)**

**Entidad: SUSTITUCION**

* id\_sustitucion (PK)
* fecha\_alta
* fecha\_baja
* id\_medico (FK)

**Entidad: HORARIO\_CONSULTA**

* id\_horario (PK)
* dia\_semana
* hora\_inicio
* hora\_fin
* id\_medico (FK)

**Entidad: EMPLEADO**

* id\_empleado (PK)
* nombre
* direccion
* telefono
* poblacion
* provincia
* codigo\_postal
* nif
* num\_seguridad\_social
* tipo\_empleado (ATS, auxiliar, celador, administrativo, etc.)

**Entidad: VACACION**

* id\_vacacion (PK)
* fecha\_inicio
* fecha\_fin
* tipo (planificada/disfrutada)
* id\_medico (FK, opcional)
* id\_empleado (FK, opcional)

**Entidad: PACIENTE**

* id\_paciente (PK)
* nombre
* direccion
* telefono
* codigo\_postal
* nif
* num\_seguridad\_social
* id\_medico\_asignado (FK)

**Entidad: CONSULTORIO**

* id\_consultorio (PK)
* nombre o codigo

**Entidad: CITA\_MEDICA**

* id\_cita (PK)
* fecha
* hora
* id\_medico (FK)
* id\_paciente (FK)
* id\_consultorio (FK)

**Entidad: DIAGNOSTICO**

* id\_diagnostico (PK)
* nombre
* descripcion
* id\_cita (FK)

**Entidad: SINTOMA**

* id\_sintoma (PK)
* descripcion

**Relaciona muchos a muchos con ENFERMEDAD.**

**Entidad: ENFERMEDAD**

* id\_enfermedad (PK)
* nombre
* descripcion

**Entidad: MEDICAMENTO**

* id\_medicamento (PK)
* nombre\_comercial
* nombre\_generico
* presentacion
* es\_generico (booleano)
* dosis\_adulto
* dosis\_nino
* necesita\_receta (booleano)
* cobertura\_pos (booleano)
* unidades\_disponibles
* id\_laboratorio (FK)

**Entidad: PRINCIPIO\_ACTIVO**

* id\_principio (PK)
* nombre

**Entidad: CONTRAINDICACION**

* id\_contra (PK)
* descripcion

**Entidad: RECETA**

* id\_receta (PK)
* fecha\_inicio
* fecha\_fin
* id\_diagnostico (FK)

**Entidad: MEDICAMENTO\_RECETA**

(relación muchos a muchos entre medicamento y receta)

* id\_med\_rec (PK)
* id\_receta (FK)
* id\_medicamento (FK)
* cantidad
* dosis

1. **Identificar interrelaciones entre conjunto de entidades**
   1. medicamento ↔ principio activo (N:M)
   2. medicamento ↔ contraindicación (N:M)
   3. enfermedad ↔ síntoma (N:M)
   4. enfermedad ↔ medicamento (N:M)
2. **Diseñar los respectivos Diagramas Entidad Relación (DER)**

****

1. **Diseñar los respectivos Diagramas Entidad Relación (DER)**

ENTIDADES FUERTES

* [Medico](id\_medico, nombre, direccion, telefono, poblacion, provincia, codigo\_postal, nif, num\_seguridad\_social, num\_colegiado, categoria)
* [Empleado](id\_empleado, nombre, direccion, telefono, poblacion, provincia, codigo\_postal, nif, num\_seguridad\_social, tipo\_empleado)
* [Paciente](id\_paciente, nombre, direccion, telefono, codigo\_postal, nif, num\_seguridad\_social, id\_medico\_asignado)
* [Consultorio](id\_consultorio, nombre)
* [Enfermedad](id\_enfermedad, nombre, descripcion)
* [Sintoma](id\_sintoma, descripcion)
* [Laboratorio](id\_laboratorio, nombre, direccion)
* [Medicamento](id\_medicamento, nombre\_comercial, nombre\_generico, presentacion, es\_generico, dosis\_adulto, dosis\_nino, necesita\_receta, cobertura\_pos, unidades\_disponibles, id\_laboratorio)
* [PrincipioActivo](id\_principio, nombre)
* [Contraindicacion](id\_contra, descripcion)

RELACIONES DERIVADAS DE 1:N

* [Sustitucion](id\_sustitucion, fecha\_alta, fecha\_baja, id\_medico)
* [HorarioConsulta](id\_horario, dia\_semana, hora\_inicio, hora\_fin, id\_medico)
* [Vacacion](id\_vacacion, fecha\_inicio, fecha\_fin, tipo, id\_medico, id\_empleado)
* [CitaMedica](id\_cita, fecha, hora, id\_paciente, id\_medico, id\_consultorio)
* [Diagnostico](id\_diagnostico, nombre, descripcion, id\_cita, id\_enfermedad)
* [Receta](id\_receta, fecha\_inicio, fecha\_fin, id\_diagnostico)

RELACIONES DERIVADAS DE N:M

* [Enfermedad\_Sintoma](id\_enfermedad, id\_sintoma)
* [Medicamento\_Principio](id\_medicamento, id\_principio)
* [Medicamento\_Contraindicacion](id\_medicamento, id\_contra)
* [Medicamento\_Receta](id\_receta, id\_medicamento, cantidad, dosis)

**CLAVES PRIMARIAS Y FORÁNEAS**

* Las claves primarias (PK) están en los primeros campos de cada relación (si no es compuesta).
* Las claves foráneas (FK) se derivan de las relaciones entre entidades (por ejemplo, id\_medico en Paciente es FK hacia Medico).
* En las relaciones N:M, las claves primarias son compuestas por los FKs.

1. Identificar los atributos (Multivaluados o Compuestos) en el DER y transformarlos en Relaciones del MR.

**Atributos Compuestos**

Son atributos que contienen múltiples valores atómicos lógicamente relacionados. En este caso:

**Direccion** (en varias entidades como Medico, Empleado, Paciente, Consultorio, Laboratorio)

* + Podría dividirse en: calle, numero, ciudad, provincia, codigo\_postal.

**Transformación:**

[DIRECCION](id\_direccion, calle, numero, ciudad, provincia, codigo\_postal)

[MEDICO](id\_medico, nombre, telefono, ..., id\_direccion)

[EMPLEADO](id\_empleado, nombre, telefono, ..., id\_direccion)

[PACIENTE](id\_paciente, nombre, telefono, ..., id\_direccion)

[LABORATORIO](id\_laboratorio, nombre, id\_direccion)

**Atributos Multivaluados**

Un atributo es multivaluado cuando una entidad puede tener más de un valor para dicho atributo.

**Medicamento - PrincipioActivo**

* + Un medicamento puede tener múltiples principios activos.

**Ya está correctamente modelado con una relación intermedia:**

[MEDICAMENTO\_PRINCIPIO](id\_medicamento, id\_principio)

**Medicamento - Contraindicaciones**

* + Un medicamento puede tener varias contraindicaciones.

**También ya está modelado correctamente:**

[MEDICAMENTO\_CONTRAINDICACION](id\_medicamento, id\_contra)

**Enfermedad - Síntomas**

* + Una enfermedad puede estar relacionada con múltiples síntomas.

**Correctamente representado en:**

[ENFERMEDAD\_SINTOMA](id\_enfermedad, id\_sintoma)

1. **Identificar las llaves candidatas para cada Relación en MR a partir de los atributos identificativos en el DER, luego determinar cuál o cuáles campos del cada Relación del MAR serán definidos como Llave primaria**

**MEDICO**(id\_medico, nombre, direccion, telefono, poblacion, provincia, codigo\_postal, nif, num\_seguridad\_social, num\_colegiado, categoria)

* **Llaves candidatas:** id\_medico, nif, num\_seguridad\_social, num\_colegiado
* **PK seleccionada:** id\_medico *(por ser artificial y más manejable)*

**EMPLEADO**(id\_empleado, nombre, direccion, telefono, poblacion, provincia, codigo\_postal, nif, num\_seguridad\_social, tipo\_empleado)

* Llaves candidatas: id\_empleado, nif, num\_seguridad\_social
* PK: id\_empleado

**PACIENTE**(id\_paciente, nombre, direccion, telefono, codigo\_postal, nif, num\_seguridad\_social, id\_medico\_asignado)

* Llaves candidatas: id\_paciente, nif, num\_seguridad\_social
* PK: id\_paciente

**CONSULTORIO**(id\_consultorio, nombre)

* Llaves candidatas: id\_consultorio, posiblemente nombre
* PK: id\_consultorio

**ENFERMEDAD**(id\_enfermedad, nombre, descripcion)

* Llaves candidatas: id\_enfermedad, nombre
* PK: id\_enfermedad

**SINTOMA**(id\_sintoma, descripcion)

* Llaves candidatas: id\_sintoma, descripcion
* PK: id\_sintoma

**PRINCIPIO\_ACTIVO**(id\_principio, nombre)

* Llaves candidatas: id\_principio, nombre
* PK: id\_principio

**MEDICAMENTO**(id\_medicamento, nombre\_comercial, nombre\_generico, ..., id\_laboratorio)

* Llaves candidatas: id\_medicamento, nombre\_comercial
* PK: id\_medicamento

**LABORATORIO**(id\_laboratorio, nombre, direccion)

* Llaves candidatas: id\_laboratorio, nombre
* PK: id\_laboratorio

**DIAGNOSTICO**(id\_diagnostico, nombre, descripcion, id\_cita, id\_enfermedad)

* Llaves candidatas: id\_diagnostico
* PK: id\_diagnostico

**RECETA**(id\_receta, fecha\_inicio, fecha\_fin, id\_diagnostico)

* Llaves candidatas: id\_receta
* PK: id\_receta

**CONTRAINDICACION**(id\_contra, descripcion)

* Llaves candidatas: id\_contra, descripcion
* PK: id\_contra

**VACACION**(id\_vacacion, fecha\_inicio, fecha\_fin, tipo, id\_medico, id\_empleado)

* Llaves candidatas: id\_vacacion
* PK: id\_vacacion

**SUSTITUCION**(id\_sustitucion, fecha\_alta, fecha\_baja, id\_medico)

* Llaves candidatas: id\_sustitucion
* PK: id\_sustitucion

**HORARIO\_CONSULTA**(id\_horario, dia\_semana, hora\_inicio, hora\_fin, id\_medico)

* Llaves candidatas: id\_horario, o combinación (dia\_semana, hora\_inicio, hora\_fin, id\_medico)
* PK: id\_horario

**CITA\_MEDICA**(id\_cita, fecha, hora, id\_paciente, id\_medico, id\_consultorio)

* Llaves candidatas: id\_cita
* PK: id\_cita

**Relaciones N:M**

Estas son relaciones entre entidades con llaves primarias compuestas:

**ENFERMEDAD\_SINTOMA**(id\_enfermedad, id\_sintoma)

* PK: (id\_enfermedad, id\_sintoma)

**MEDICAMENTO\_PRINCIPIO**(id\_medicamento, id\_principio)

* PK: (id\_medicamento, id\_principio)

**MEDICAMENTO\_RECETA**(id\_receta, id\_medicamento, cantidad, dosis)

* PK: (id\_receta, id\_medicamento)

**MEDICAMENTO\_CONTRAINDICACION**(id\_medicamento, id\_contra)

* PK: (id\_medicamento, id\_contra)

1. Identificar las llaves foráneas en el MR, a partir de las Interrelaciones de uno a uno en el DER.

**Identificación de Relaciones 1:1 en el DER**

Al observar tu DER, la única interrelación que claramente se percibe como **uno a uno** es la siguiente:

**1. Relación entre MEDICO y SUSTITUCION**

* Cada **SUSTITUCION** se asocia con un único **MEDICO**.
* No se muestra que un médico tenga múltiples sustituciones activas a la vez → Podría interpretarse como una relación 1:1 si solo se permite una sustitución vigente por médico a la vez.

**Aplicación del Proceso de Transformación**

**[SUSTITUCION]--|---------|--[MEDICO]**

* Agregamos medico\_id a SUSTITUCION (ya está).
* Agregamos sustitucion\_id a MEDICO.

**Nueva estructura:**

SUSTITUCION(id\_sustitucion, fecha\_alta, fecha\_baja, medico\_id)

MEDICO(id\_medico, nombre, ..., sustitucion\_id)

**Llaves foráneas:**

* SUSTITUCION.medico\_id → FK hacia MEDICO.id\_medico
* MEDICO.sustitucion\_id → FK hacia SUSTITUCION.id\_sustitucion

**¿Hay más relaciones 1:1?**

No se visualizan otras relaciones estrictamente 1:1 en el diagrama. La mayoría son de:

* 1:N (por ejemplo, un médico tiene muchas citas, un diagnóstico tiene una receta)
* N:M (como medicamentos con contraindicaciones, enfermedades con síntomas)

**Resultado final**

[MEDICO](id\_medico, nombre, direccion, telefono, ..., sustitucion\_id)

[SUSTITUCION](id\_sustitucion, fecha\_alta, fecha\_baja, medico\_id)

1. Identificar las llaves foráneas en el MR, a partir de las Interrelaciones de uno a muchos en el DER.

**Identificación de Relaciones 1:N en el DER**

Vamos a recorrer el diagrama y extraer cada relación 1:N:

**MEDICO 1 -- N CITA\_MEDICA**

[MEDICO](id\_medico, ...)

[CITA\_MEDICA](id\_cita, fecha, hora, ..., medico\_id)

**PACIENTE 1 -- N CITA\_MEDICA**

[PACIENTE](id\_paciente, ...)

[CITA\_MEDICA](id\_cita, ..., paciente\_id)

**CONSULTORIO 1 -- N CITA\_MEDICA**

[CONSULTORIO](id\_consultorio, ...)

[CITA\_MEDICA](..., consultorio\_id)

**CITA\_MEDICA 1 -- N DIAGNOSTICO**

[CITA\_MEDICA](id\_cita, ...)

[DIAGNOSTICO](..., cita\_id)

**ENFERMEDAD 1 -- N DIAGNOSTICO**

[ENFERMEDAD](id\_enfermedad, ...)

[DIAGNOSTICO](..., enfermedad\_id)

**DIAGNOSTICO 1 -- N RECETA**

[DIAGNOSTICO](id\_diagnostico, ...)

[RECETA](..., diagnostico\_id)

**LABORATORIO 1 -- N MEDICAMENTO**

[LABORATORIO](id\_laboratorio, ...)

[MEDICAMENTO](..., laboratorio\_id)

**MEDICO 1 -- N HORARIO\_CONSULTA**

[MEDICO](id\_medico, ...)

[HORARIO\_CONSULTA](..., medico\_id)

**MEDICO 1 -- N VACACION**

**También se relaciona con EMPLEADO, pero eso parece opcional**

[MEDICO](id\_medico, ...)

[VACACION](..., medico\_id)

**EMPLEADO 1 -- N VACACION**

[EMPLEADO](id\_empleado, ...)

[VACACION](..., empleado\_id)

**MEDICO 1 -- N SUSTITUCION**

(Ya vista como posible 1:1, pero también puede ser 1:N según reglas de negocio)

[MEDICO](id\_medico, ...)

[SUSTITUCION](..., medico\_id)

**MEDICO 1 -- N PACIENTE**

(asignación de paciente a un médico)

[MEDICO](id\_medico, ...)

[PACIENTE](..., medico\_asignado\_id)

**Formato de Resultado**

Aquí las relaciones en el formato solicitado:

[MEDICO](id\_medico, nombre, ..., categoria)

[CITA\_MEDICA](id\_cita, fecha, hora, ..., medico\_id)

[PACIENTE](id\_paciente, nombre, ..., medico\_asignado\_id)

[CITA\_MEDICA](..., paciente\_id)

[CONSULTORIO](id\_consultorio, nombre)

[CITA\_MEDICA](..., consultorio\_id)

[CITA\_MEDICA](id\_cita, fecha, hora, ...)

[DIAGNOSTICO](..., cita\_id)

[ENFERMEDAD](id\_enfermedad, nombre, descripcion)

[DIAGNOSTICO](..., enfermedad\_id)

[DIAGNOSTICO](id\_diagnostico, nombre, ...)

[RECETA](..., diagnostico\_id)

[LABORATORIO](id\_laboratorio, nombre, direccion)

[MEDICAMENTO](..., laboratorio\_id)

[MEDICO](id\_medico, ...)

[HORARIO\_CONSULTA](..., medico\_id)

[MEDICO](id\_medico, ...)

[VACACION](..., medico\_id)

[EMPLEADO](id\_empleado, ...)

[VACACION](..., empleado\_id)

[MEDICO](id\_medico, ...)

[SUSTITUCION](..., medico\_id)

1. Identificar las llaves foráneas en el MR, a partir de las Interrelaciones de muchos a muchos en el DER

En el Modelo Relacional, **una relación N:M se transforma en una nueva tabla (relación intermedia)** que contiene:

* Las **llaves primarias** de ambas entidades como **llaves foráneas**.
* Estas llaves también forman la **clave primaria compuesta** de la tabla intermedia.
* Puede incluir **atributos adicionales** si los hay.

**Identificación de relaciones N:M en el DER**

Revisando el diagrama, encontramos las siguientes relaciones **N:M**:

**MEDICAMENTO N:M CONTRAINDICACION**

Tabla intermedia: MEDICAMENTO\_CONTRAINDICACION

[MEDICAMENTO\_CONTRAINDICACION](medicamento\_id, contra\_id)

FK: medicamento\_id → MEDICAMENTO.id\_medicamento

FK: contra\_id → CONTRAINDICACION.id\_contra

PK compuesta: (medicamento\_id, contra\_id)

**MEDICAMENTO N:M PRINCIPIO\_ACTIVO**

Tabla intermedia: MEDICAMENTO\_PRINCIPIO

[MEDICAMENTO\_PRINCIPIO](medicamento\_id, principio\_id)

FK: medicamento\_id → MEDICAMENTO.id\_medicamento

FK: principio\_id → PRINCIPIO\_ACTIVO.id\_principio

PK compuesta: (medicamento\_id, principio\_id)

**RECETA N:M MEDICAMENTO**

Tabla intermedia: MEDICAMENTO\_RECETA

[MEDICAMENTO\_RECETA](receta\_id, medicamento\_id, cantidad, dosis)

FK: receta\_id → RECETA.id\_receta

FK: medicamento\_id → MEDICAMENTO.id\_medicamento

PK compuesta: (receta\_id, medicamento\_id)

Otros atributos: cantidad, dosis

**ENFERMEDAD N:M SINTOMA**

Tabla intermedia: ENFERMEDAD\_SINTOMA

[ENFERMEDAD\_SINTOMA](enfermedad\_id, sintoma\_id)

FK: enfermedad\_id → ENFERMEDAD.id\_enfermedad

FK: sintoma\_id → SINTOMA.id\_sintoma

PK compuesta: (enfermedad\_id, sintoma\_id)

**Resultado en formato solicitado**

[MEDICAMENTO\_CONTRAINDICACION](medicamento\_id, contra\_id)

[MEDICAMENTO\_PRINCIPIO](medicamento\_id, principio\_id)

[MEDICAMENTO\_RECETA](receta\_id, medicamento\_id, cantidad, dosis)

[ENFERMEDAD\_SINTOMA](enfermedad\_id, sintoma\_id)

Estas tablas intermedias permiten modelar correctamente las relaciones de muchos a muchos (N:M) entre entidades.

1. Identificar las llaves foráneas en el MR, a partir de las Interrelaciones de Generalización |ISA> en DER Investigar y aplicar una de las tres formas de hacerlo.

**Llaves foráneas desde la Generalización (ISA)**

**En tu DER:**

Tienes una **generalización** clara:

EMPLEADO

├── MÉDICO

└── ADMINISTRATIVO

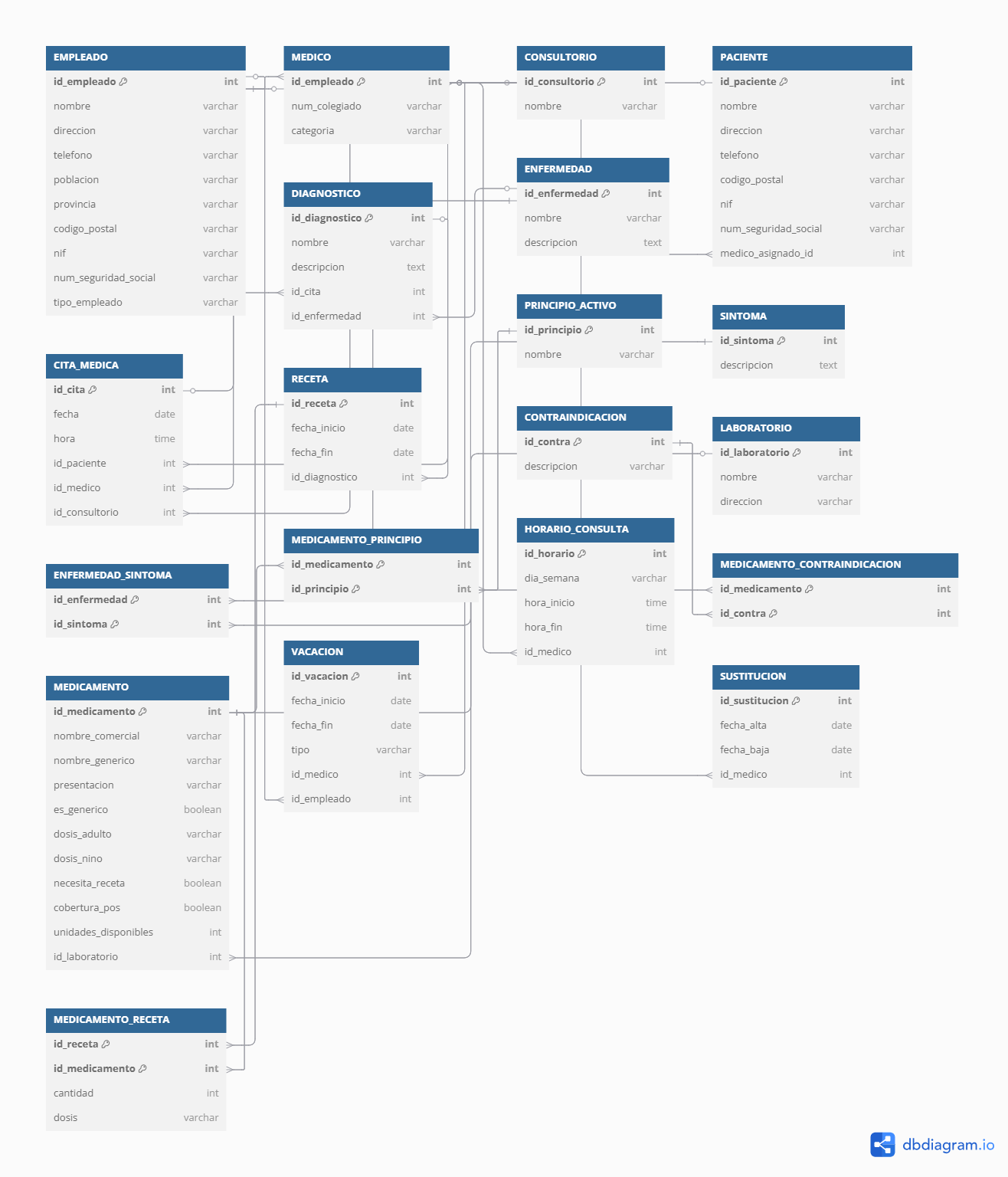
Esto es una relación tipo ISA.

**Forma de implementar ISA: Una tabla por entidad (también llamada *forma por defecto*)**

**¿Cómo funciona?**

* Creamos una tabla para la **superclase** (EMPLEADO)
* Creamos una tabla para cada **subclase** (MEDICO, ADMINISTRATIVO)
* Cada subclase hereda la PK de la superclase como su propia PK (y también como FK hacia la tabla padre)

1. Diseñar el respectivo Modelo Relacional (MR) a partir del DER



**PARTE 2. IMPLEMENTACIÓN DE LA BASE DE DATOS**

1. INSTALAR 2 MOTORES Y SERVIDORES DE BASES DE DATOS RELACIONALES
   1. Instalar **PostgreSQL** Server
   2. Instalar **MySQL** Server
   3. Como evidencia puede realizar un documento (o en su defecto un video con cámara y audio donde se vea y escuche tu voz, publicado en youtube (si desea puede ocultarlo) y anexar la URL en este documento) sobre el proceso de instalación paso a paso o idealmente.
2. Tomar como base el Modelo Relacional obtenido anteriormente (PARTE 1) a partir del Diagrama entidad Relación las siguientes operaciones:

**CONDICIONES PARA EL DESARROLLO DE LA ACTIVIDAD:**

* Para interactuar con el Servidor de BD se debe única y exclusivamente la **Terminal de Línea de Comando (CLI o Consola)**  cada Motor de Bases de Datos. (NO SE ADMITE EL USO DE APLICACIONES CON INTERFAZ GRÁFICA)
* De debe usar este mismo documento de word para dar respuesta a cada uno de los ITEM, colocando como respuesta:
  + La instrucción SQL correcta en formato texto
  + Un recorte del fragmento de la terminal CLI donde fue ingresada y ejecutada la instrucción y su respectiva salida. No colocar la pantalla completa porque se hace ilegible la imagen. Una explicación sobre el la instrucción y la salida del mismo
  + Por ejemplo:

INSTRUCCIÓN SQL:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA

Texto

Descripción generada automáticamente

EXPLICACIÓN.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Entrar al Motor de Bases** de datos usando la **Terminal (consola)** de Línea de Comandos usando el usuario creado durante la instrucción o el usuario por defecto con rol de super usuario.

INSTRUCCIÓN SQL:

Server [localhost]: localhost

Database [postgres]: postgres

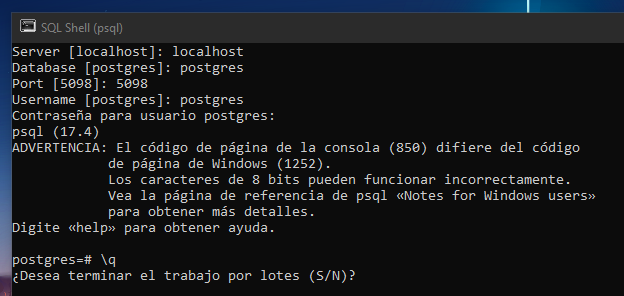
Port [5098]: 5098

Username [postgres]: postgres

Contraseña para usuario postgres:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN.

Al momento de iniciar la terminal de postgresql este nos pedirá que ingresemos el nombre del servidor, nombre de la base de datos, el puerto de escucha, nombre de usuario y la contraseña que fue declarada al momento de la instalación para el usuario root por defecto, que, en esta ocasión es el que se esta usando.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

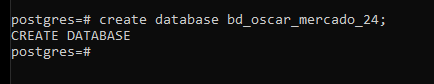
1. **Crear una base de datos** (tablespace en oracle) llamada **bd\_tu\_nombre\_tu\_apellido\_xyz**, donde debe reemplazarse numero\_nombre\_del\_archivo del ejercicio que le fue asignado.

INSTRUCCIÓN SQL:

create database bd\_oscar\_mercado\_24;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN.

Creación de base de datos inicial teniendo en cuenta las explicaciones, en este caso con simplemente con el comando “créate database” solo se crea la base de datos y se le asigna al usuario por defecto

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Crear** un nuevo **usuario** para poder entrar al Motor de BD, los datos del nuevo usuario deben ser:

Nombre: **user\_tu\_nombre\_tu\_apellido**

Clave: **AbcdeUdeC**

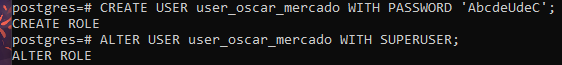
El nuevo usuario podrá conectarse desde cualquier dirección IP

INSTRUCCIÓN SQL:

CREATE USER user\_oscar\_mercado WITH PASSWORD 'AbcdeUdeC';

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN.

Se crea el usuario con nombre de estudiante y se le asignan permisos de superusuario

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Al nuevo usuario **user\_tu\_nombre\_tu\_apellido** creado anteriormente , se le deben **asignar Privilegios de Superusuario** sobre la base de datos **bd\_tu\_nombre\_tu\_apellido\_xyz**

INSTRUCCIÓN SQL:

GRANT ALL PRIVILEGES ON DATABASE bd\_oscar\_mercado\_24 TO user\_oscar\_mercado;

\c bd\_oscar\_mercado\_24

GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO user\_oscar\_mercado;

GRANT ALL PRIVILEGES ON ALL SEQUENCES IN SCHEMA public TO user\_oscar\_mercado;

GRANT ALL PRIVILEGES ON ALL FUNCTIONS IN SCHEMA public TO user\_oscar\_mercado;

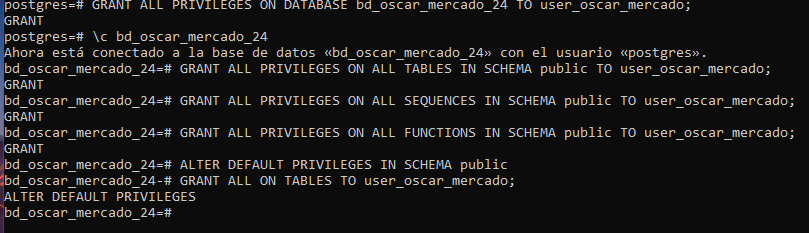
Darle privilegios futuros también

ALTER DEFAULT PRIVILEGES IN SCHEMA public

GRANT ALL ON TABLES TO user\_oscar\_mercado;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN.

Se otorga todos los privilegios sobre la base de datos, luego se otorga uso y control total sobre el esquema publico conectándose primero a la base de datos y para finalizar se otorgan privilegios futuros

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

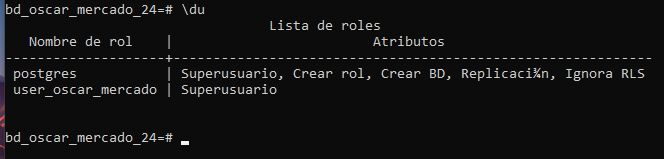
1. **Mostrar los usuarios** existentes

INSTRUCCIÓN SQL:

\du

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Muestra todos los usuarios

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

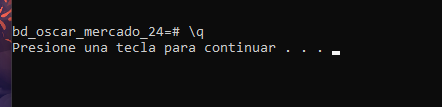
1. **Salir de la terminal** o consola del Motor de BD

INSTRUCCIÓN SQL:

\q

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Se sale de la terminal luego de presionar cualquier tecla

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Volver a **entrar** la Motor de Bases de datos **desde la terminal o consola** pero ahora usando las c**redenciales del nuevo usuario** **user\_tu\_nombre\_tu\_apellido** creado anteriormente-

INSTRUCCIÓN SQL:

Server [localhost]: localhost

Database [postgres]: bd\_oscar\_mercado\_24

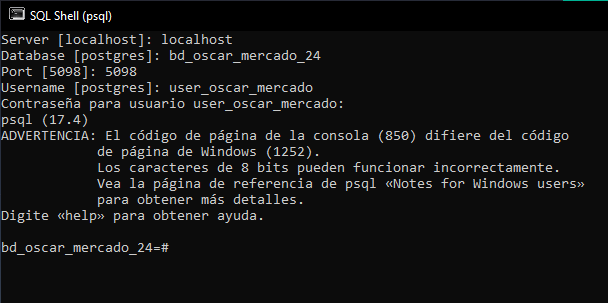
Port [5098]: 5098

Username [postgres]: user\_oscar\_mercado

Contraseña para usuario user\_oscar\_mercado: AbcdeUdeC

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Se ingresa a la base de datos creada con mi nombre y tambien con mi usuario

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Las siguientes operaciones SQL deben ser realizadas utilizando el nuevo usuario **user\_tu\_nombre\_tu\_apellido**.

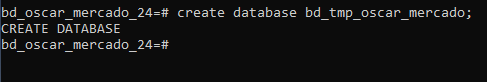
1. **Crear** una **nueva BD** llamada **bd\_tmp\_tu\_nombre\_tu\_apellido**

INSTRUCCIÓN SQL:

create database bd\_tmp\_oscar\_mercado;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Se crea la tabla nueva solicitada

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Mostrar** las **Bases de datos** que pertenecen al **user\_tu\_nombre\_tu\_apellido**

INSTRUCCIÓN SQL:

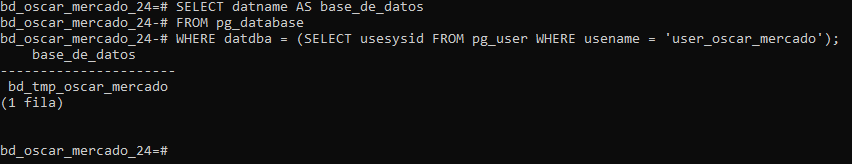
SELECT datname AS base\_de\_datos

FROM pg\_database

WHERE datdba = (SELECT usesysid FROM pg\_user WHERE usename = 'user\_oscar\_mercado');

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Muestra todas las bases de datos cuyo del usuario user\_oscar\_mercado.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

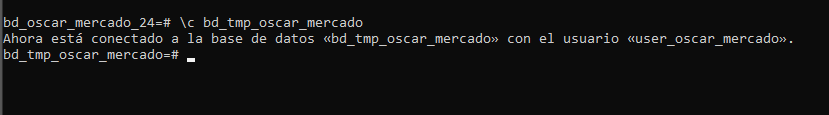
1. Entrar o usar la BD **bd\_tmp\_tu\_nombre\_tu\_apellido**

INSTRUCCIÓN SQL:

\c bd\_tmp\_oscar\_mercado

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Con este comando se accede a una base de datos especifica en este caso bd\_tmp\_oscar\_mercado

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Crear una tabla** llamada **bd\_tmp\_tu\_nombre\_tu\_apellido** con las siguiente estructura

**tabla\_tempora**

**id** auto\_incremental y clave primera

**columna2** de tipo Texto, no nula y con longitud de 100

**columna3** de tipo Entero

**columna4** de tipo Real

**columna5** de tipo Fecha

**columna6** de tipo FechaHora

**columna7** de tipo hora

**columna8** de tipo FechaHora automatica

**columna9** de tipo Texto con longitud variable

**columna10** de tipo archivo binario

INSTRUCCIÓN SQL:

CREATE TABLE tabla\_temporal (

id SERIAL PRIMARY KEY,

columna2 VARCHAR(100) NOT NULL,

columna3 INTEGER,

columna4 REAL,

columna5 DATE,

columna6 TIMESTAMP,

columna7 TIME,

columna8 TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

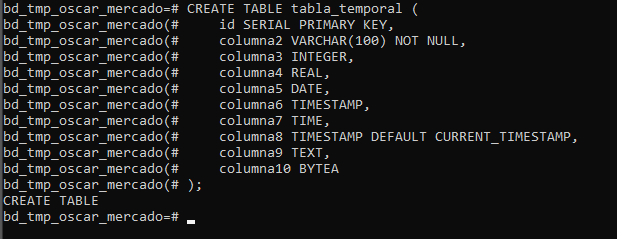
columna9 TEXT,

columna10 BYTEA

);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Tabla creada bajo el nombre de “tabla\_temporal” ya que el nombre solicitado es el mismo que el de la base de datos y en mi parecer eso puede generar confusiones.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

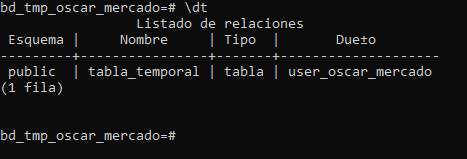
1. **Mostrar las tablas** de la **bd\_tmp\_tu\_nombre\_tu\_apellido**

INSTRUCCIÓN SQL:

\dt

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Con este comando se listan las tablas creadas en la base de datos

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

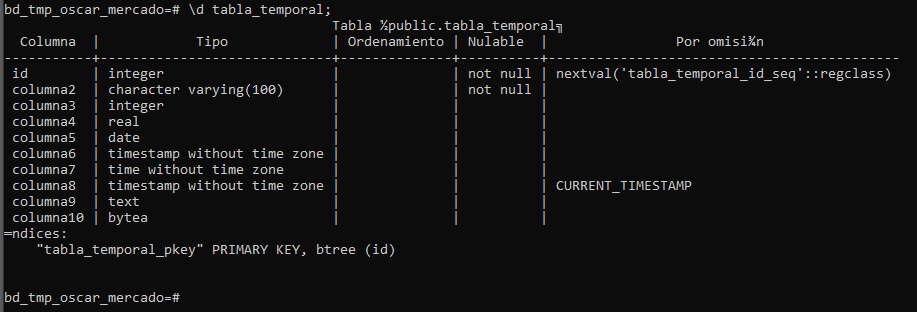
1. **Describir** la estructura de la **tabla\_tempora**

INSTRUCCIÓN SQL:

\d tabla\_temporal;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Para describir una tabla en PostgreSQL y obtener información sobre su estructura, puedes usar la instrucción \d en la consola de PostgreSQL. Este comando muestra los detalles de la tabla, como las columnas, sus tipos de datos y las restricciones.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

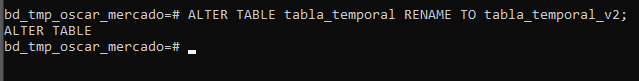
1. **Cambiar** el **nombre** de la **tabla\_tempora** por **tabla\_tempora\_v2**

INSTRUCCIÓN SQL:

ALTER TABLE tabla\_temporal RENAME TO tabla\_temporal\_v2;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Cambio de nombre exitoso

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

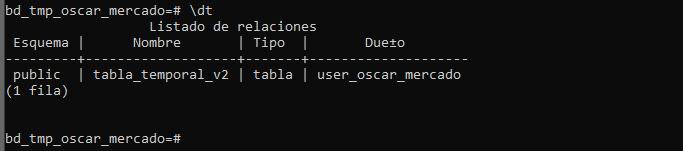
1. **Mostrar** las **tablas** de la **bd\_temporal**

INSTRUCCIÓN SQL:

\dt

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Como se puede apreciar el cambio de nombre se realizo

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

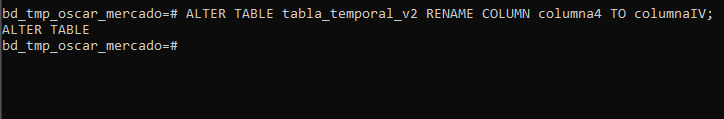
1. En la tabla **tabla\_tempora\_v2 cambiar el nombre** de la **culumna4** por **culumnaIV**

INSTRUCCIÓN SQL:

ALTER TABLE tabla\_temporal\_v2 RENAME COLUMN columna4 TO columnaIV;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Cambio de nombre realizado de manera exitosa

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. En la tabla **tabla\_tempora\_v2 cambiar el tipo** de dato de la **columna7** por el tipo **FechaHora**

INSTRUCCIÓN SQL:

ALTER TABLE tabla\_temporal\_v2

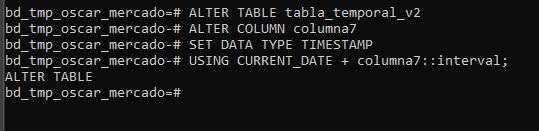
ALTER COLUMN columna7

SET DATA TYPE TIMESTAMP

USING CURRENT\_DATE + columna7::interval;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

CURRENT\_DATE: Devuelve la fecha actual sin la parte de la hora.

columna7::interval: Convierte el valor de la columna columna7 (que es de tipo TIME) en un intervalo de tiempo.

Al sumar CURRENT\_DATE con el intervalo, se obtiene un valor de tipo TIMESTAMP que tiene la fecha actual y la hora correspondiente.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. En la tabla **tabla\_tempora\_v2 agregar una columna** nueva llamada **email** de tipo Texto de longitud 70, que sea **única** y con valor por defecto **abc@udec.edu.co**

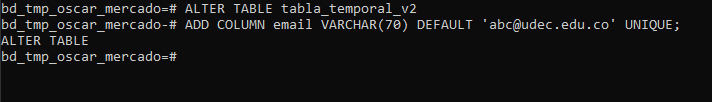
INSTRUCCIÓN SQL:

ALTER TABLE tabla\_temporal\_v2

ADD COLUMN email VARCHAR(70) DEFAULT 'abc@udec.edu.co' UNIQUE;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Creación de la nueva columna y limitación de longitud, como podemos apreciar no hay un error

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

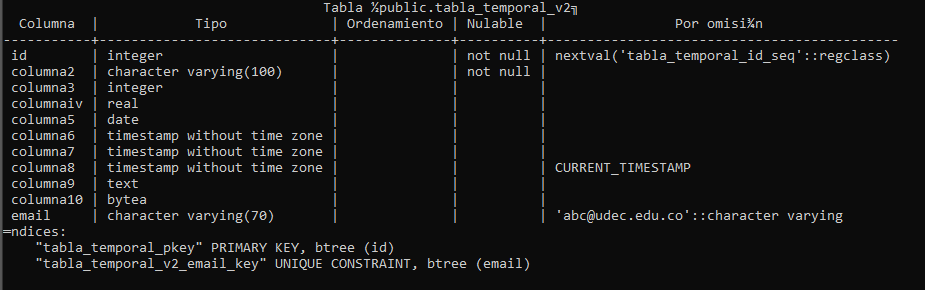
1. **Describir** la Estructura de la **tabla\_tempora\_v2** para observar los cambios

INSTRUCCIÓN SQL:

\d tabla\_temporal\_v2;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Podemos apreciar que todos los cambios fueron realizados sin ninguna complicacion

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Eliminar** la **columna6**

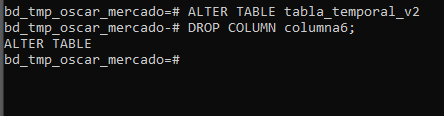
INSTRUCCIÓN SQL:

ALTER TABLE tabla\_temporal\_v2

DROP COLUMN columna6;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Se elimina la columna 6

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

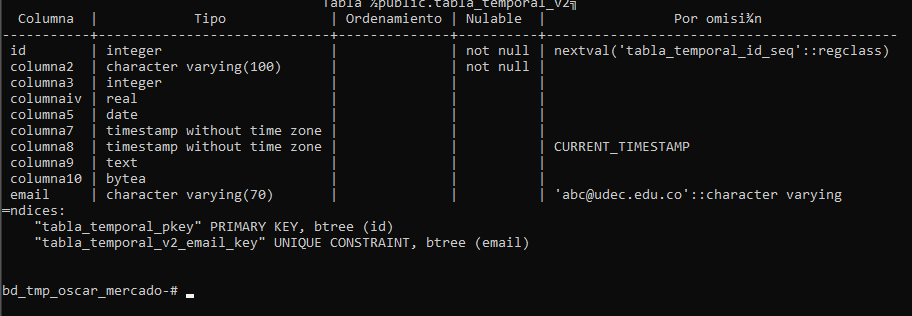
1. **Describir** la Estructura de la **tabla\_tempora\_v2** para observar los cambios

INSTRUCCIÓN SQL:

\d tabla\_temporal\_v2;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Alteraciones realizadas con exito

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

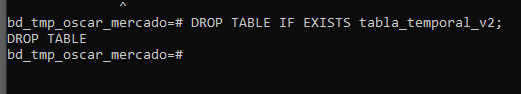
1. **Eliminar** la **tabla\_tempora\_v2**

INSTRUCCIÓN SQL:

DROP TABLE IF EXISTS tabla\_temporal\_v2;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Se elimina la tabla\_temporal\_v2 de manera permanente y a su vez sus datos, luego de correr este comando no se pueden recuperar los datos de la tabla.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Mostrar** las **tablas** de la **bd\_tmp\_tu\_nombre\_tu\_apellido**

INSTRUCCIÓN SQL:

\dt

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Podemos apreciar que no muestra ninguna relación ya que no hay tablas en la base de datos

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Cambiarle** el **nombre** de la **base de datos** **bd\_tmp\_tu\_nombre\_tu\_apellido** por **bd\_tmp\_tu\_nombre\_tu\_apellido\_v2**

Según el motor de BD utilizado esta operación no se realiza por CLI, si ese es su caso, entonces, debe explicar el proceso necesario para cambiar el nombre a la BD.

INSTRUCCIÓN SQL:

\c postgres

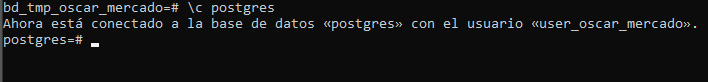
ALTER DATABASE bd\_tmp\_oscar\_mercado RENAME TO bd\_tmp\_tu\_nombre\_tu\_apellido\_v2;

\l

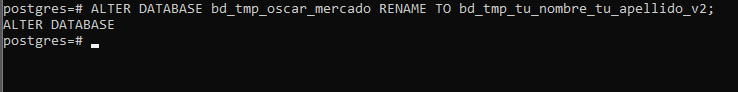
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA

En postgresql no se puede cambiar el nombre de la base de datos directamente cuando se esta conectado a ella, lo que se debe hacer es desconectarse de la base de datos y luego realizar el cambio

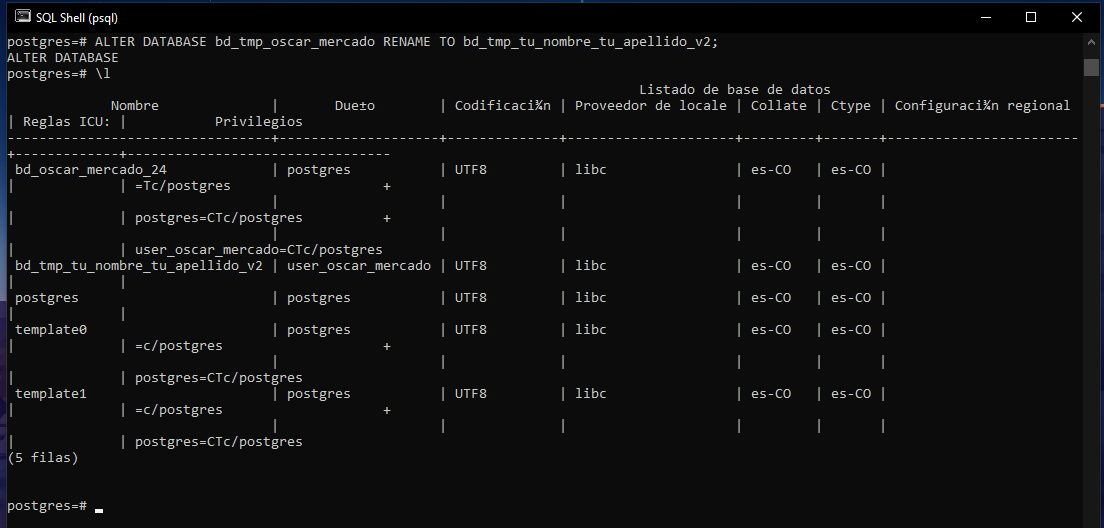


Con el comando \c postgres cambiamos la conexión a la base de datos por defecto postgres



Luego se realiza el cambio de nombre

Y para finalizar se listan las bases de datos para comprobar que se realizo el cambio exitosamente





Como podemos apreciar en el área subrayada, el nombre se cambio exitosamente

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

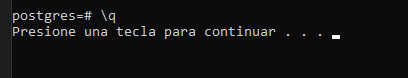
1. Salid de la CLI de la terminal del motor de BD

INSTRUCCIÓN SQL:

\q

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Luego de presionar cualquier tecla desconectará de la sesión de PostgreSQL

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Entrar al CLI de la terminal del motor de BD pero ahora utilizando las credenciales del usuario **user\_tu\_nombre\_tu\_apellido**

INSTRUCCIÓN SQL:

Server [localhost]: localhost

Database [postgres]: bd\_oscar\_mercado\_24

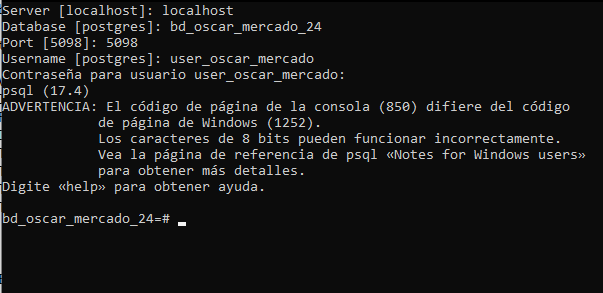
Port [5098]: 5098

Username [postgres]: user\_oscar\_mercado

Contraseña para usuario user\_oscar\_mercado: AbcdeUdeC

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Ingreso con mis credenciales y directamente a la base de datos creada anteriormente

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Mostrar** las **Bases de datos** que pertenecen al **user\_tu\_nombre\_tu\_apellido**

INSTRUCCIÓN SQL:

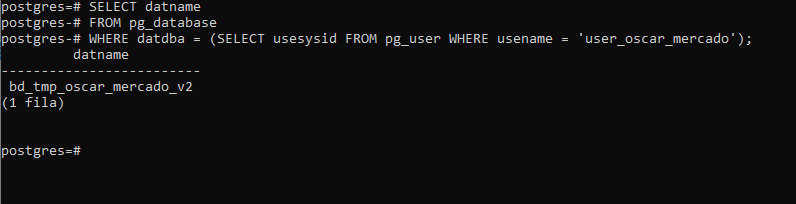
SELECT datname

FROM pg\_database

WHERE datdba = (SELECT usesysid FROM pg\_user WHERE usename = 'user\_oscar\_mercado');

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Con esta serie de comandos se pueden visualizar todas las bases de datos asignadas a un usuario en especifico.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Eliminar el usuario **user\_tmp\_tu\_nombre\_tu\_apellido**

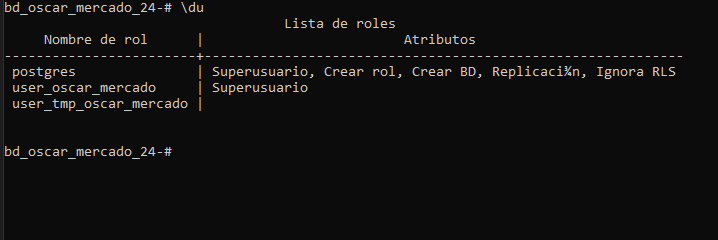
INSTRUCCIÓN SQL:

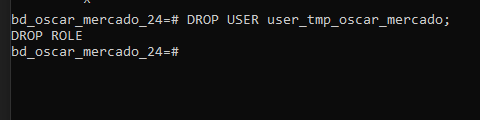
DROP USER user\_tmp\_oscar\_mercado;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA

Verificamos que el usuario esta creado





La consola dice que fue eliminado

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

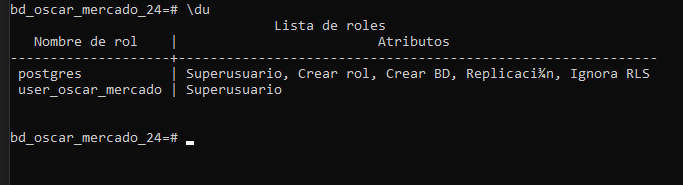
1. Mostrar los usuarios activos en el motor de bases de datos.

INSTRUCCIÓN SQL:

\du

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Podemos apreciar que se elimino con exito

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Las siguientes operaciones SQL se deben realizar tomando como base **Modelo Relacional** del ejercicio que le fue asignado:

1. **Entrar** o usar la base de datos **bd\_tu\_nombre\_tu\_apellido\_xyz** creada anteriormente.

INSTRUCCIÓN SQL:

Server [localhost]: localhost

Database [postgres]: bd\_oscar\_mercado\_24

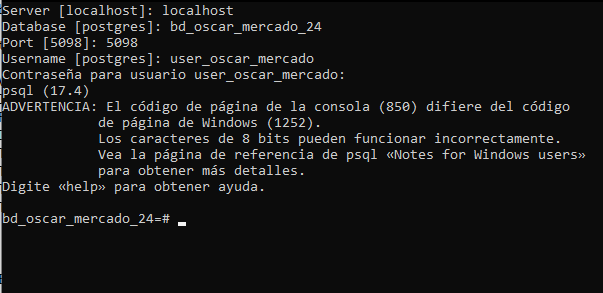
Port [5098]: 5098

Username [postgres]: user\_oscar\_mercado

Contraseña para usuario user\_oscar\_mercado: AbcdeUdeC

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Ingreso exitoso

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

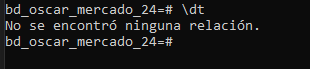
1. **Mostrar** las **tablas** de la **bd\_tu\_nombre\_tu\_apellido\_xyz**

INSTRUCCIÓN SQL:

\dt

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTE DE PANTALLA



EXPLICACIÓN

Dado a que no se ha creado ninguna tabla no es posible ver ninguna relacion

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Crear** las tablasde acuerdo a lo reflejado en el Modelo Relacional del ejercicio.

INSTRUCCIONES SQL PARA CREAR LAS TABLAS:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

CREATE TABLE Medico (

id\_medico SERIAL PRIMARY KEY,

nombre TEXT,

direccion TEXT,

telefono TEXT,

poblacion TEXT,

provincia TEXT,

codigo\_postal TEXT,

nif TEXT,

num\_seguridad\_social TEXT,

num\_colegiado TEXT,

categoria TEXT

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Sustitucion (

id\_sustitucion SERIAL PRIMARY KEY,

fecha\_alta DATE NOT NULL,

fecha\_baja DATE,

id\_medico INTEGER REFERENCES Medico(id\_medico)

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE HorarioConsulta (

id\_horario SERIAL PRIMARY KEY,

dia\_semana TEXT,

hora\_inicio TIME,

hora\_fin TIME,

id\_medico INTEGER REFERENCES Medico(id\_medico)

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Empleado (

id\_empleado SERIAL PRIMARY KEY,

nombre TEXT,

direccion TEXT,

telefono TEXT,

poblacion TEXT,

provincia TEXT,

codigo\_postal TEXT,

nif TEXT,

num\_seguridad\_social TEXT,

tipo\_empleado TEXT

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Vacacion (

id\_vacacion SERIAL PRIMARY KEY,

fecha\_inicio DATE,

fecha\_fin DATE,

tipo TEXT,

id\_medico INTEGER REFERENCES Medico(id\_medico),

id\_empleado INTEGER REFERENCES Empleado(id\_empleado)

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Paciente (

id\_paciente SERIAL PRIMARY KEY,

nombre TEXT,

direccion TEXT,

telefono TEXT,

codigo\_postal TEXT,

nif TEXT,

num\_seguridad\_social TEXT,

id\_medico\_asignado INTEGER REFERENCES Medico(id\_medico)

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Consultorio (

id\_consultorio SERIAL PRIMARY KEY,

nombre TEXT

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE CitaMedica (

id\_cita SERIAL PRIMARY KEY,

fecha DATE,

hora TIME,

id\_paciente INTEGER REFERENCES Paciente(id\_paciente),

id\_medico INTEGER REFERENCES Medico(id\_medico),

id\_consultorio INTEGER REFERENCES Consultorio(id\_consultorio)

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Enfermedad (

id\_enfermedad SERIAL PRIMARY KEY,

nombre TEXT,

descripcion TEXT

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Sintoma (

id\_sintoma SERIAL PRIMARY KEY,

descripcion TEXT

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Enfermedad\_Sintoma (

id\_enfermedad INTEGER REFERENCES Enfermedad(id\_enfermedad),

id\_sintoma INTEGER REFERENCES Sintoma(id\_sintoma),

PRIMARY KEY (id\_enfermedad, id\_sintoma)

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Diagnostico (

id\_diagnostico SERIAL PRIMARY KEY,

nombre TEXT,

descripcion TEXT,

id\_cita INTEGER REFERENCES CitaMedica(id\_cita),

id\_enfermedad INTEGER REFERENCES Enfermedad(id\_enfermedad)

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Receta (

id\_receta SERIAL PRIMARY KEY,

fecha\_inicio DATE,

fecha\_fin DATE,

id\_diagnostico INTEGER REFERENCES Diagnostico(id\_diagnostico)

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Laboratorio (

id\_laboratorio SERIAL PRIMARY KEY,

nombre TEXT,

direccion TEXT

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Medicamento (

id\_medicamento SERIAL PRIMARY KEY,

nombre\_comercial TEXT,

nombre\_generico TEXT,

presentacion TEXT,

es\_generico BOOLEAN,

dosis\_adulto TEXT,

dosis\_nino TEXT,

necesita\_receta BOOLEAN,

cobertura\_pos BOOLEAN,

unidades\_disponibles INTEGER,

id\_laboratorio INTEGER REFERENCES Laboratorio(id\_laboratorio)

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE PrincipioActivo (

id\_principio SERIAL PRIMARY KEY,

nombre TEXT

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Medicamento\_Principio (

id\_medicamento INTEGER REFERENCES Medicamento(id\_medicamento),

id\_principio INTEGER REFERENCES PrincipioActivo(id\_principio),

PRIMARY KEY (id\_medicamento, id\_principio)

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Contraindicacion (

id\_contra SERIAL PRIMARY KEY,

descripcion TEXT

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Medicamento\_Contraindicacion (

id\_medicamento INTEGER REFERENCES Medicamento(id\_medicamento),

id\_contra INTEGER REFERENCES Contraindicacion(id\_contra),

PRIMARY KEY (id\_medicamento, id\_contra)

);

Texto

El contenido generado por IA puede ser incorrecto.

CREATE TABLE Medicamento\_Receta (

id\_receta INTEGER REFERENCES Receta(id\_receta),

id\_medicamento INTEGER REFERENCES Medicamento(id\_medicamento),

cantidad INTEGER,

dosis TEXT,

PRIMARY KEY (id\_receta, id\_medicamento)

);

Texto

El contenido generado por IA puede ser incorrecto.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Describir** cada una de las **tablas** creadas creadas en la BD **bd\_tu\_nombre\_tu\_apellido\_xyz**

INSTRUCCIONES SQL PARA DESCRIBIR LAS TABLAS:

\d nombre\_de\_la\_tabla

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\d citamedica

Texto

El contenido generado por IA puede ser incorrecto.

\d medico

Texto, Escala de tiempo

El contenido generado por IA puede ser incorrecto.

\d sustitución

Texto

El contenido generado por IA puede ser incorrecto.

\d horarioconsulta

Texto

El contenido generado por IA puede ser incorrecto.

\d empleado

Texto

El contenido generado por IA puede ser incorrecto.

\d vacacion

Texto

El contenido generado por IA puede ser incorrecto.

\d paciente

Texto

El contenido generado por IA puede ser incorrecto.

\d consultorio

Texto

El contenido generado por IA puede ser incorrecto.

\d diagnostico

Texto

El contenido generado por IA puede ser incorrecto.

\d enfermedad

Texto

El contenido generado por IA puede ser incorrecto.

\d síntoma

Texto

El contenido generado por IA puede ser incorrecto.

\d enfermedad\_sintoma

Texto

El contenido generado por IA puede ser incorrecto.

\d laboratorio

Texto

El contenido generado por IA puede ser incorrecto.

\d medicamento

Texto

El contenido generado por IA puede ser incorrecto.

\d principioactivo

Texto

El contenido generado por IA puede ser incorrecto.

\d medicamento\_principio

Texto

El contenido generado por IA puede ser incorrecto.

\d contraindicación

Texto

El contenido generado por IA puede ser incorrecto.

\d medicamento\_contraindicacion

Texto

El contenido generado por IA puede ser incorrecto.

\d receta

Texto

El contenido generado por IA puede ser incorrecto.

\d medicamento\_receta

Texto

El contenido generado por IA puede ser incorrecto.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Si las tablas aún no tienen las columnas que deben actuar como llaves foráneas, entonces, usted debe agregar dichas columnas a en respectivas tablas.

Debido a que las tablas se crearon de manera tal que se establecieron las llaves foráneas no es necesario realizar este punto

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Agregar** los **índices** en las columnas que actuarán como llaves foráneas

En este caso postgresql se encarga de establecer los índices de manera automática así que no es necesario establecerlos de manera manual

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Agregar** restricciones de **llave foránea** en cada una de las tablas a las que haya lugar según el **Modelo Relacional**, cada llave foránea debe aplicar restricción de integridad referencial Restrictiva para operaciones de Eliminar y Cascada para Operaciones de Actualizar.

**(Estudiar e investigar qué es integridad referencial, como se aplica esto en las llaves foráneas)**

**OPCIONES COMUNES EN LLAVES FORÁNEAS**

| **Acción** | **Descripción** |
| --- | --- |
| ON DELETE RESTRICT | **No permite eliminar** el registro padre si hay hijos. |
| ON UPDATE CASCADE | Si se actualiza la PK en el padre, **propaga** el cambio a los hijos. |

INSTRUCCIONES SQL PARA AGREGAR LAS LLAVES FORÁNEAS:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTES DE PANTALLAS

FK en Paciente hacia Medico

ALTER TABLE paciente

ADD CONSTRAINT fk\_paciente\_medico

FOREIGN KEY (id\_medico\_asignado)

REFERENCES medico(id\_medico)

ON DELETE RESTRICT

ON UPDATE CASCADE;

Texto

El contenido generado por IA puede ser incorrecto.

FK en CitaMedica hacia Paciente

ALTER TABLE citamedica

ADD CONSTRAINT fk\_cita\_paciente

FOREIGN KEY (id\_paciente)

REFERENCES paciente(id\_paciente)

ON DELETE RESTRICT

ON UPDATE CASCADE;

Texto

El contenido generado por IA puede ser incorrecto.

FK en Receta hacia Diagnostico

ALTER TABLE receta

ADD CONSTRAINT fk\_receta\_diagnostico

FOREIGN KEY (id\_diagnostico)

REFERENCES diagnostico(id\_diagnostico)

ON DELETE RESTRICT

ON UPDATE CASCADE;

Texto

El contenido generado por IA puede ser incorrecto.

FK en Medicamento hacia Laboratorio

ALTER TABLE medicamento

ADD CONSTRAINT fk\_medicamento\_laboratorio

FOREIGN KEY (id\_laboratorio)

REFERENCES laboratorio(id\_laboratorio)

ON DELETE RESTRICT

ON UPDATE CASCADE;

Texto

El contenido generado por IA puede ser incorrecto.

EXPLICACIÓN

**¿Qué es integridad referencial?**

La **integridad referencial** asegura que los valores de una clave foránea (FK) **coincidan con un valor válido en la tabla referenciada** (PK), o bien sean nulos si la relación lo permite.  
Además, define qué pasa si se elimina o actualiza el registro padre.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Utilizar alguna herramienta para visualizar el Modelo Relacional a partir de la BD ya creada. Los tres motores de BD que usaron para el desarrollo de esta actividad ofrecen una herramienta que nos permite ingresar instrucciones SQL al motor de BD de forma rápida e intuitiva, gracias a que están desarrolladas usando Interfaz Gráfica de Usuario (GUI), también nos permiten gestionar cómodamente todos los componentes del servidor y los elementos de cualquier BD dentro de dicho motor, así como también, nos permiten generar automáticamente el MR (Modelo Relacional) seleccionado una BD previamente creará, o viceversa, nos permiten crear la BD a partir de un MR.

COLOCAR AQUÍ LA IMAGEN DEL MODELO RELACIONAL GENERADO AUTOMÁTICAMENTE

PASOS PARA GENERAR EL MODELO RELACIONAL A PARTIR DE LA BD CREADA EN EL RESPECTIVO MOTOR DE BD.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* 1. Abrir pgadmin4

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.



* 1. Seleccionar la base de datos

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.



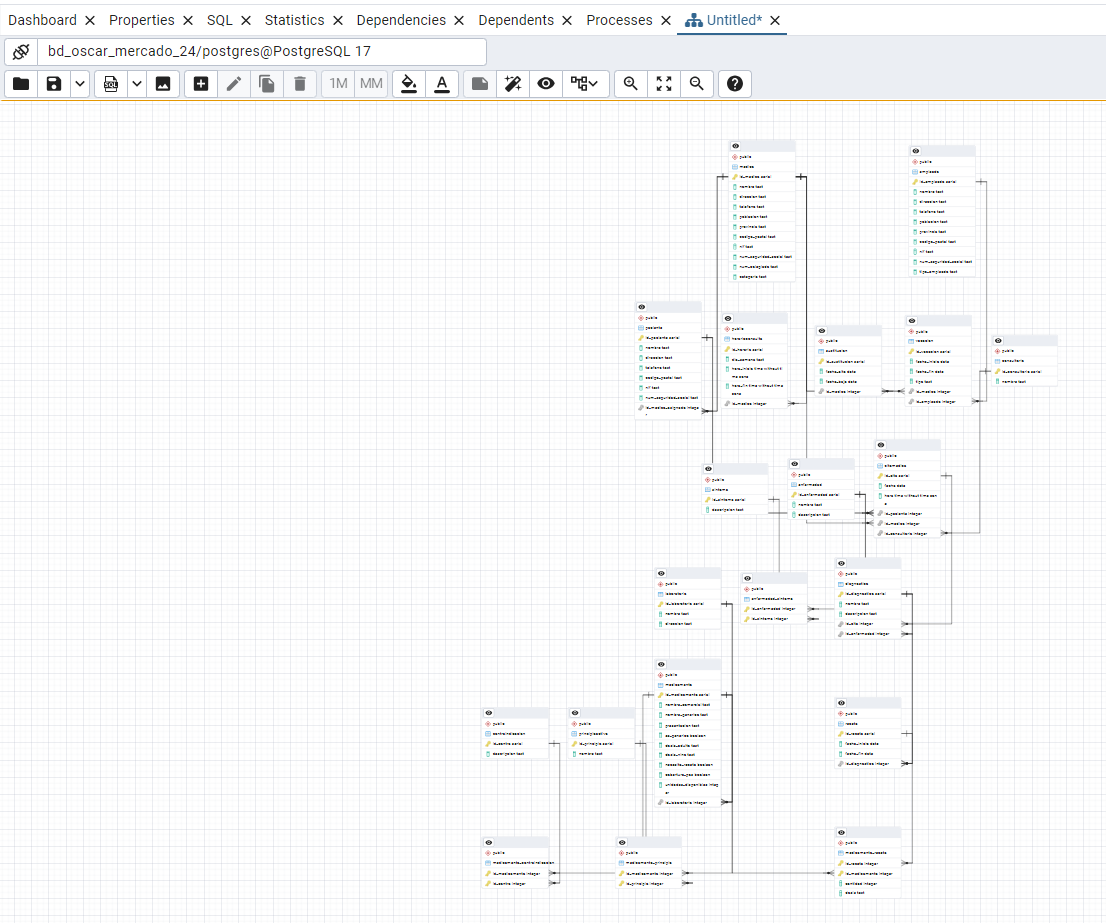
* 1. Click derecho en la base de datos y seleccionar generar ERD

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.



* 1. Podemos observar que se genera el modelo y esta seria la vista general



* 1. Seleccionamos descargar imagen.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.



* 1. Luego seleccionamos la ruta donde la queremos guardar y nombramos.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

El resultado final es.

Diagrama, Esquemático

El contenido generado por IA puede ser incorrecto.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Crear un **Backup** solo **de la estructura** de la **bd\_tu\_nombre\_tu\_apellido\_xyz** y guardar el resultado en un archivo llamado **backup\_bd\_tu\_nombre\_tu\_apellido\_xyz.sql**

PASOS PARA GENERAR EL BACKUP DE LA BD CREADA EN EL RESPECTIVO MOTOR DE BD. (IDEAL UTILIZAR EL COMANDO SQL APROPIADO SEGÚN CADA MOTOR DE BD Y OPCIONALMENTE TAMBIÉN PUEDEN HACERLO Y EXPLICARLO USANDO EL MODO GRÁFICO)

* 1. Abrir pgadmin4

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.



* 1. Seleccionar la base de datos

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.



* 1. Click derecho en la base de datos y seleccionar Backup.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.



* 1. Especificamos el nombre.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.



* 1. Y seleccionamos Guardar.

Tabla

El contenido generado por IA puede ser incorrecto.



1. Eliminar la Base de datos **bd\_tu\_nombre\_tu\_apellido\_xyz**

INSTRUCCIÓN SQL :

DROP DATABASE bd\_oscar\_mercado\_24;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTES DE PANTALLAS

Texto

El contenido generado por IA puede ser incorrecto.

EXPLICACIÓN

Se elimina completamente la base de datos bd\_oscar\_mercado\_24 del sistema PostgreSQL. Esta operación es irreversible, por eso es recomendable realizar previamente un backup de seguridad.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Mostrar las bases de datos del usuario **bd\_tu\_nombre\_tu\_apellido\_xyz**

INSTRUCCIÓN SQL :

\l

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTES DE PANTALLAS

Imagen que contiene Texto

El contenido generado por IA puede ser incorrecto.

EXPLICACIÓN

Este comando muestra todas las bases de datos creadas en el sistema PostgreSQL. Permite confirmar si la base bd\_oscar\_mercado\_24 está activa, o si ha sido eliminada con éxito.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Eliminar el Usuario **bd\_tu\_nombre\_tu\_apellido\_xyz**

INSTRUCCIÓN SQL:

DROP USER user\_oscar\_mercado;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTES DE PANTALLAS

* Asegúrate de que **no existan objetos (como bases de datos o tablas)** que dependan de este usuario.
* Si el usuario aún tiene **propiedades o privilegios**, la eliminación fallará.

REASSIGN OWNED BY user\_oscar\_mercado TO postgres;

DROP OWNED BY user\_oscar\_mercado;

Texto

El contenido generado por IA puede ser incorrecto.

DROP USER user\_oscar\_mercado;

Texto

El contenido generado por IA puede ser incorrecto.

El usuario no puede ser eliminado porque esta activo, debemos primero cerrar la sesión activa con el comando \q

Texto

El contenido generado por IA puede ser incorrecto.

Luego de iniciar sesión con el usuario root predeterminado podemos eliminar nuestro usuario.

Texto

El contenido generado por IA puede ser incorrecto.

DROP USER user\_oscar\_mercado;

Texto

El contenido generado por IA puede ser incorrecto.

eliminación exitosa

EXPLICACIÓN

Se elimina completamente el usuario del sistema de gestión de bases de datos PostgreSQL. Esta acción libera su espacio y evita futuras conexiones no autorizadas.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Utilizar el archivo de Backup para realizar restauración de la BD previamente eliminada.

INSTRUCCIÓN SQL :

createdb -U postgres bd\_oscar\_mercado\_24

pg\_restore -U postgres -d bd\_oscar\_mercado\_24 "C:\Users\Lenovo\Desktop\db\backup\_bd\_oscar\_mercado\_24.sql"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

RECORTES DE PANTALLAS

Creación de base de datos recipiente

Texto

El contenido generado por IA puede ser incorrecto.

Restauración de base de datos

Texto

El contenido generado por IA puede ser incorrecto.

EXPLICACIÓN

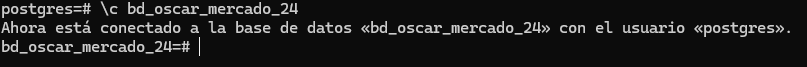
Primero creamos una base de datos donde se va a guardar la información del archivo backup

El archivo de respaldo se encontraba en una ruta diferente a la predeterminada del sistema. Se especificó la ruta completa al ejecutar la restauración para indicarle a psql dónde encontrar el archivo .sql.

También podemos observar que la nueva base de datos recipiente del backup genero problemas ya que en la información del backup hay ligaduras al usuario que habíamos creado anteriormente y que eliminamos casi al final de la actividad

Sin embargo la base de datos se restableció

Cambiamos a la nueva base de datos



Y con el comando \dt verificamos que todas las tablas se hayan restablecido

Imagen que contiene Texto

El contenido generado por IA puede ser incorrecto.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. De acuerdo a los dos motores de BD que usted ha elegido, al finalizar esta actividad usted debe estar en la capacidad de diligenciar la siguiente tabla:

|  |  |  |  |
| --- | --- | --- | --- |
| OPERACIÓN SQL | MySQL | PostgreSQL | SQLServer |
| Entrar a la terminal |  | psql -U postgres |  |
| Crear un usuario |  | CREATE USER nombre\_usuario WITH PASSWORD 'clave'; |  |
| Asignar permisos a un usuario |  | GRANT ALL PRIVILEGES ON DATABASE bd TO usuario; |  |
| Cambiar la clave a un usuario |  | ALTER USER nombre\_usuario WITH PASSWORD 'nueva'; |  |
| Quitar los permisos a un usuario |  | REVOKE ALL PRIVILEGES ON DATABASE bd FROM usuario; |  |
| Eliminar un usuario |  | DROP USER nombre\_usuario; |  |
| Mostrar los usuarios activos en el Motor de BD |  | \du |  |
| Salid de la terminal |  | \q |  |
| Mostrar las BD |  | \l ó \list |  |
| Entrar o usar una BD |  | \c nombre\_bd |  |
| Eliminar una BD |  | DROP DATABASE nombre\_bd; |  |
| Crear la tabla del punto # 12 |  | CREATE TABLE tabla\_temporal (  id SERIAL PRIMARY KEY,  columna2 VARCHAR(100) NOT NULL,  columna3 INTEGER,  columna4 REAL,  columna5 DATE,  columna6 TIMESTAMP,  columna7 TIME,  columna8 TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,  columna9 TEXT,  columna10 BYTEA  ); |  |
| Mostrar la estructura de una tabla |  | \d nombre\_tabla |  |
| Cambiar el nombre de una tabla |  | ALTER TABLE antigua RENAME TO nueva; |  |
| Agregar una columna a una tabla |  | ALTER TABLE tabla ADD COLUMN nueva tipo; |  |
| Cambiar el tipo de datos a una columna |  | ALTER TABLE tabla ALTER COLUMN col TYPE nuevo; |  |
| Cambiar el nombre de una columna |  | ALTER TABLE tabla RENAME COLUMN antigua TO nueva; |  |
| Eliminar una columna |  | ALTER TABLE tabla DROP COLUMN columna; |  |
| Tipos de datos Numéricos Enteros |  | SMALLINT, INTEGER, BIGINT |  |
| Tipos de datos numéricos reales o decimales |  | REAL, DOUBLE PRECISION, NUMERIC(p,s) |  |
| Tipos de datos caracteres de longitud fija |  | CHAR(n) |  |
| Tipos de datos caracteres de longitud variable |  | VARCHAR(n), TEXT |  |
| Tipos de datos Boolean |  | BOOLEAN |  |
| Tipos de datos Fecha |  | DATE, TIME, TIMESTAMP |  |
| Tipos de datos para guardar datos binarios grandes |  | BYTEA |  |
| Tipos de datos conjunto |  | No soportado directamente |  |
| Tipos de datos enumeración |  | CREATE TYPE estado AS ENUM ('activo', 'inactivo') |  |
| Columnas con valor autoincremental, secuencial o serial |  | SERIAL, BIGSERIAL, GENERATED ... AS IDENTITY |  |
| Cómo Eliminar una tabla |  | DROP TABLE nombre\_tabla; |  |
| Agregar una Llave foránea al crear la tabla |  | FOREIGN KEY (...) REFERENCES ... (...) |  |
| Agregar una llave foránea después de creada la tabla |  | ALTER TABLE ... ADD CONSTRAINT ... FOREIGN KEY ... |  |
| Crear un índice |  | CREATE INDEX nombre ON tabla(columna); |  |
| Mostrar los índice creados |  | \di |  |
| Mostrar las restricciones de llave foránea de una tabla |  | \d nombre\_tabla |  |
| Eliminar una restricción de llave foránea o cualquier otra restricción |  | ALTER TABLE tabla DROP CONSTRAINT nombre\_fk; |  |
| Hacer un backup |  | pg\_dump -U usuario -s -f archivo.sql base\_datos |  |
| Restaurar la BD |  | psql -U usuario -d base\_datos -f archivo.sql |  |

**5. Bibliografía**

* Documentación oficial de PostgreSQL: https://www.postgresql.org/docs/
* Normas APA: https://www.colconectada.com/normas-apa/
* Material de la asignatura Bases de Datos 1 - UdeC