

BASE DE DATOS II

PARCIAL #1

Modalidad: Trabajo en equipo (máximo 2 estudiantes)

Duración: 2 semanas

Entrega: Moodle

Competencias a evaluar

El estudiante será capaz de:

- Diseñar modelos de datos complejos aplicando cardinalidad correcta.
- Construir diagramas entidad-relación.
- Aplicar normalización hasta 3FN.
- Implementar bases de datos usando Docker.
- Utilizar DDL, DML y CRUD.
- Utilizar VISTAS SQL.
- Justificar decisiones técnicas como DBA.

PARTE I – Diagrama Entidad Relación (10 pts)

Usted ha sido contratado como **Administrador de Base de Datos (DBA)** para diseñar la solución de datos de una plataforma empresarial.

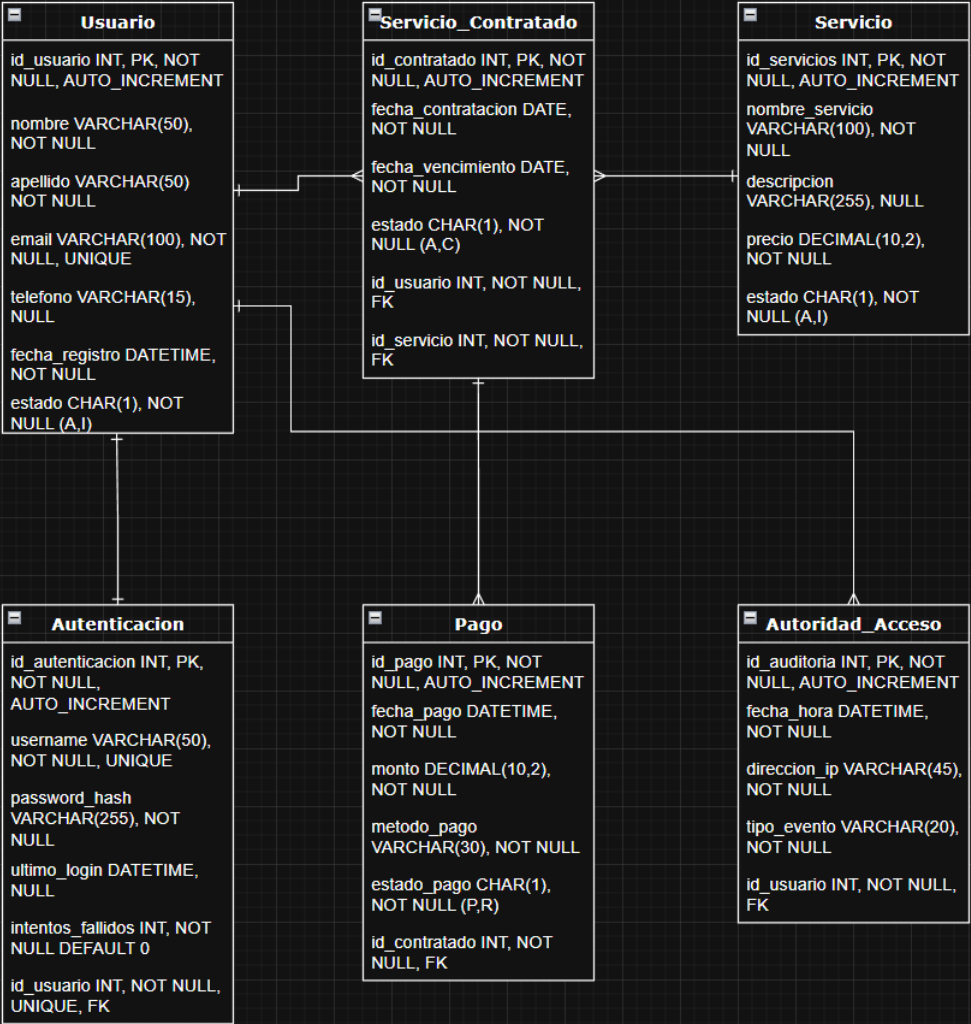
Caso 1: Sistema Empresarial de Servicios Digitales

El sistema debe cubrir:

- Gestión de usuarios
- Autenticación
- Servicios contratados
- Pagos
- Auditoría de accesos

Actividades:

- Diseñar un diagrama entidad relación(**DER**) **completo** que incluya:
 - a) Entidades
 - b) Atributos
 - c) Relaciones
 - d) Cardinalidades (1:1, 1:N, N:M)
 - e) Entidades asociativas cuando aplique
- El modelo debe reflejar **reglas de negocio reales**



PARTE II – Implementación de Base de Datos con Docker (30 pts)

La empresa exige que toda la infraestructura de base de datos sea desplegada mediante Docker, garantizando portabilidad y estandarización.

Uso de Base de Datos con Docker (10 pts)

Requerimientos obligatorios

- Crear un contenedor Docker que levante un motor de base de datos relacional (MySQL o MariaDB).
- Utilizar los comandos de docker visto en clase de:
 - Ejecución de contenedor
 - Creación de contenedor
 - La asignación de nombre, variable, puerto e imagen
 - Estado actual del contenedor
 - Visualizar los eventos de logs
 - Conectar al contenedor por CLI
- En la creación del contenedor definir:
 - Nombre del contenedor
 - Usuario
 - Contraseña
 - Puerto expuesto

Entregables

- Comando Docker documentado
- Evidencia de que la base de datos está operativa

NOTA: No se aceptan instalaciones locales fuera de Docker

Usage: docker run [OPTIONS] IMAGE [COMMAND] [ARG...]

See 'docker run --help' for more information
PS C:\Users\nmcom> docker run --name empresa_db -p 3309:3306 -d --env MARIADB_ROOT_PASSWORD=12345 --env MARIADB_USER=admin_empresa --env MARIADB_PASSWORD=Admin123! mariadb
>>

base_de_datos

3ca7e0428ac9

mariadb

3309:3306

0.06%

22 seconds

PS C:\Users\nmcom> docker ps

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
3ca7e0428ac9	mariadb	"docker-entrypoint.s..."	About a minute ago	Up About a minute	0.0.0.0:3309->3306/tcp, [::]:3309->3306/tcp	empresa_db

PS C:\Users\nmcom>

PS C:\Users\nmcom> docker logs empresa_db

```
2026-02-23 22:06:59 0 [Note] InnoDB: Compressed tables use zlib 1.3
2026-02-23 22:06:59 0 [Note] InnoDB: Number of transaction pools: 1
2026-02-23 22:06:59 0 [Note] InnoDB: Using crc32 + pclmulqdq instructions
2026-02-23 22:06:59 0 [Note] mariadbd: O_TMPFILE is not supported on /tmp (disabling future attempts)
2026-02-23 22:06:59 0 [Warning] mariadbd: io_uring_queue_init() failed with EPERM: sysctl kernel.io_uring_disabled has the value 2, or 1 and the user of the process is not a member of sysctl kernel.io_uring_group. (see man 2 io_uring_setup).
create_uring failed, falling back to libaio
2026-02-23 22:06:59 0 [Note] InnoDB: Using Linux native AIO
2026-02-23 22:06:59 0 [Note] InnoDB: InnoDB_buffer_pool_size_max=128M, innodb_buffer_pool_size=128M
2026-02-23 22:06:59 0 [Note] InnoDB: Completed initialization of buffer pool
2026-02-23 22:06:59 0 [Note] InnoDB: File system buffers for log disabled (block size=4096 bytes)
2026-02-23 22:06:59 0 [Note] InnoDB: End of log at LSN=46100
2026-02-23 22:06:59 0 [Note] InnoDB: Opened 3 undo tablespaces
2026-02-23 22:06:59 0 [Note] InnoDB: 128 rollback segments in 3 undo tablespaces are active.
2026-02-23 22:06:59 0 [Note] InnoDB: Setting file './ibtmp1' size to 12.000MiB. Physically writing the file full; Please wait ...
2026-02-23 22:06:59 0 [Note] InnoDB: File './ibtmp1' size is now 12.000MiB.
2026-02-23 22:06:59 0 [Note] InnoDB: log sequence number 46100; transaction id 14
2026-02-23 22:06:59 0 [Note] Plugin 'FEDERATED' is disabled.
2026-02-23 22:06:59 0 [Note] Plugin 'winp-provider' is disabled.
2026-02-23 22:06:59 0 [Note] InnoDB: Loading buffer pool(s) from /var/lib/mysql/ib_buffer_pool
2026-02-23 22:06:59 0 [Note] InnoDB: Buffer pool(s) load completed at 260223 22:06:59
2026-02-23 22:07:00 0 [Note] Server socket created on IP: '0.0.0.0', port: '3306'.
2026-02-23 22:07:00 0 [Note] Server socket created on IP: '::', port: '3306'.
2026-02-23 22:07:00 0 [Note] mariadbd: Event Scheduler: loaded 0 events
2026-02-23 22:07:00 0 [Note] mariadbd: ready for connections.
Version: '12.1.2-MariaDB-ubu2404' socket: /run/mysqld/mysqld.sock port: 3306 mariadb.org binary distribution
PS C:\Users\nmcom>
```

```
PS C:\Users\nmcom> docker exec -it empresa_db mariadb -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 6
Server version: 12.1.2-MariaDB-ubu2404 mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

```
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 7
Server version: 12.1.2-MariaDB-ubu2404 mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

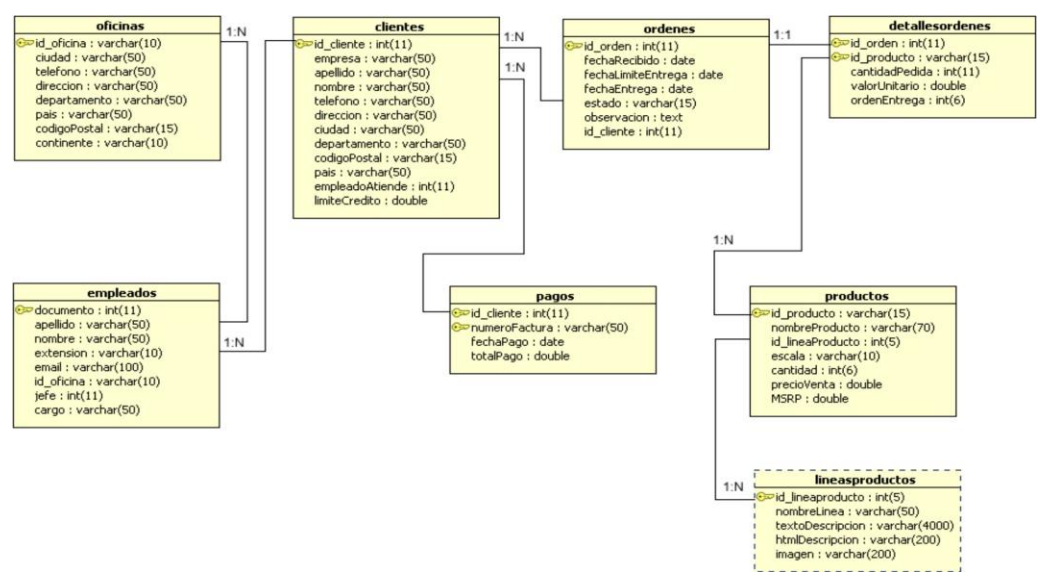
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.006 sec)

MariaDB [(none)]> S
```

Caso 2: Sistema de punto de ventas (20 pts)

Basándonos en el siguiente diagrama entidad relación, realice las siguientes bases de datos.



El diseño base de datos debe tener como mínimo:

- Creación de las estructuras de tablas y relaciones.
- Se deberá agregar por cada tabla, 2 columnas adicionales, qué maneje los diferentes tipos de datos como: tipo lista, tipo si/no y columnas únicas.
- Se debe contar con un mínimo de 10 registros o más por cada tabla.
- Ejecutar con el comando SQL la estructura o descripción de cada tabla.

Tablas creadas:

```
SistemaDePuntoDeVentas.sql U X
C: > Users > nmcom > Desktop > SistemaDePuntoDeVentas.sql > ...
  ▸ Run |  Active: 127.0.0.1 | punto_ventas
1  CREATE DATABASE punto_ventas;
  ▸ Run
2  USE punto_ventas;
3
4  -- Tabla de oficinas
  ▸ Run | Select
5  CREATE TABLE oficinas (
6      id_oficina VARCHAR(10) PRIMARY KEY,
7      ciudad VARCHAR(50),
8      telefono VARCHAR(50),
9      direccion VARCHAR(50),
10     departamento VARCHAR(50),
11     pais VARCHAR(50),
12     codigoPostal VARCHAR(15),
13     continente VARCHAR(10),
14
15     tipo_oficina ENUM('Principal','Sucursal','Regional') NOT NULL,
16     activa BOOLEAN NOT NULL,
17     codigo_unico VARCHAR(20) UNIQUE
18 );
19
```

```
19
20 -- Tabla de empleados
  ▸ Run | Select
21 CREATE TABLE empleados (
22     documento INT PRIMARY KEY,
23     apellido VARCHAR(50),
24     nombre VARCHAR(50),
25     extension VARCHAR(10),
26     email VARCHAR(100) UNIQUE,
27     id_oficina VARCHAR(10),
28     jefe INT,
29     cargo VARCHAR(50),
30
31     tipo_contrato ENUM('Fijo','Temporal','Servicios') NOT NULL,
32     activo BOOLEAN NOT NULL,
33
34     FOREIGN KEY (id_oficina) REFERENCES oficinas(id_oficina)
35 );
36
```

```
36
37 -- Tabla de clientes
  ▸ Run | Select
38 CREATE TABLE clientes (
39     id_cliente INT PRIMARY KEY,
40     empresa VARCHAR(50),
41     apellido VARCHAR(50),
42     nombre VARCHAR(50),
43     telefono VARCHAR(50),
44     direccion VARCHAR(50),
45     ciudad VARCHAR(50),
46     departamento VARCHAR(50),
47     codigoPostal VARCHAR(15),
48     pais VARCHAR(50),
49     empleadoAtiende INT,
50     limiteCredito DOUBLE,
51
52     tipo_cliente ENUM('Minorista','Mayorista','Corporativo') NOT NULL,
53     activo BOOLEAN NOT NULL,
54     email VARCHAR(100) UNIQUE,
55
56     FOREIGN KEY (empleadoAtiende) REFERENCES empleados(documento)
57 );
58
```

```

58
59  -- Tabla de ordenes
    ▷Run | ⌵Select
60  CREATE TABLE ordenes (
61      id_orden INT PRIMARY KEY,
62      fechaRecibido DATE,
63      fechaLimiteEntrega DATE,
64      fechaEntrega DATE,
65      estado VARCHAR(15),
66      observacion TEXT,
67      id_cliente INT,
68
69      prioridad ENUM('Alta','Media','Baja') NOT NULL,
70      facturada BOOLEAN NOT NULL,
71
72      FOREIGN KEY (id_cliente) REFERENCES clientes(id_cliente)
73  );
74

```

```

74
75  --Tabla de detallesordenes
    ▷Run | ⌵Select
76  CREATE TABLE detallesordenes (
77      id_orden INT,
78      id_producto VARCHAR(15),
79      cantidadPedida INT,
80      valorUnitario DOUBLE,
81      ordenEntrega INT,
82
83      tipo_descuento ENUM('Ninguno','Promocion','ClienteFrecuente') NOT NULL,
84      aplica_impuesto BOOLEAN NOT NULL,
85
86      PRIMARY KEY (id_orden, id_producto),
87      FOREIGN KEY (id_orden) REFERENCES ordenes(id_orden),
88      FOREIGN KEY (id_producto) REFERENCES productos(id_producto)
89  );
90

```

```

90
91  -- Tabla de lineasproductos
    ▷Run | ⌵Select
92  ✓ CREATE TABLE lineasproductos (
93      id_lineaproducto INT PRIMARY KEY,
94      nombreLinea VARCHAR(50),
95      textoDescripcion VARCHAR(4000),
96      htmlDescripcion VARCHAR(200),
97      imagen VARCHAR(200),
98
99      categoria ENUM('Electrico','Manual','Industrial') NOT NULL,
100     disponible BOOLEAN NOT NULL,
101     codigo_linea VARCHAR(20) UNIQUE
102 );
103

```

```
104 -- Tabla de productos
105 CREATE TABLE productos (
106     id_producto VARCHAR(15) PRIMARY KEY,
107     nombreProducto VARCHAR(70),
108     id_lineaProducto INT,
109     escala VARCHAR(10),
110     cantidad INT,
111     precioventa DOUBLE,
112     MSRP DOUBLE,
113
114     tipo_producto ENUM('Fisico','Digital','Servicio') NOT NULL,
115     activo BOOLEAN NOT NULL,
116     codigo_barra VARCHAR(50) UNIQUE,
117
118     FOREIGN KEY (id_lineaProducto) REFERENCES lineasproductos(id_lineaproducto)
119 );
120
```

```
121 -- Tabla de pagos
122 CREATE TABLE pagos (
123     id_pago INT PRIMARY KEY,
124     id_cliente INT,
125     numeroFactura VARCHAR(50) UNIQUE,
126     fechaPago DATE,
127     totalPago DOUBLE,
128
129     metodo_pago ENUM('Efectivo','Tarjeta','Transferencia') NOT NULL,
130     confirmado BOOLEAN NOT NULL,
131
132     FOREIGN KEY (id_cliente) REFERENCES clientes(id_cliente)
133 );
134
```

Selects

SistemaDePuntoDeVentas.sql U oficinas X

Properties DATA Log ER Monitor punto_ventas

SELECT * FROM oficinas LIMIT 100

Search Results

Cost: 5ms < 1 > Total 10

	id_oficina varchar(10)	ciudad varchar(50)	telefono varchar(50)	direccion varchar(50)	departamento varchar(50)	pais varchar(50)	codigoPostal varchar(15)	continente varchar(10)	* tipo_oficina enum('Principal','Sucu'	* activo tinyint(1)	codigo_unico varchar(20)
>	OF1	Panama	2222-1111	Via España	Panama	Panama	0801	America	Principal	1	OF-001
>	OF10	Anraijan	2222-0000	Centro	Panama Oeste	Panama	1201	America	Sucursal	1	OF-010
>	OF2	Colon	2222-2222	Zona Libre	Colon	Panama	0301	America	Sucursal	1	OF-002
>	OF3	David	2222-3333	Centro	Chiriqui	Panama	0401	America	Regional	1	OF-003
>	OF4	Santiago	2222-4444	Central	Veraguas	Panama	0501	America	Sucursal	1	OF-004
>	OF5	Chitre	2222-5555	Principal	Herrera	Panama	0601	America	Sucursal	1	OF-005
>	OF6	Penonome	2222-6666	Centro	Cocle	Panama	0701	America	Regional	1	OF-006
>	OF7	Bocas	2222-7777	Isla	Bocas	Panama	0901	America	Sucursal	1	OF-007
>	OF8	Las Tablas	2222-8888	Centro	Los Santos	Panama	1001	America	Sucursal	1	OF-008
>	OF9	Aguadulce	2222-9999	Principal	Cocle	Panama	1101	America	Regional	1	OF-009

SistemaDePuntoDeVentas.sql U empleados X

Properties DATA Log ER Monitor punto_ventas

SELECT * FROM empleados LIMIT 100

Search Results

Cost: 4ms < 1 > Total 10

	document int(11)	apellido varchar(50)	nombre varchar(50)	extension varchar(10)	email varchar(100)	id_oficina varchar(10)	jefe int(11)	cargo varchar(50)	* tipo_contrato enum('Fijo','Temporal'	* activo tinyint(1)
>	1	Perez	Juan	101	juan@empresa.com	OF1	(NULL)	Gerente	Fijo	1
>	2	Lopez	Ana	102	ana@empresa.com	OF2	1	Supervisor	Fijo	1
>	3	Gomez	Luis	103	luis@empresa.com	OF3	1	Vendedor	Temporal	1
>	4	Diaz	Maria	104	maria@empresa.com	OF4	2	Vendedor	Servicios	1
>	5	Torres	Carlos	105	carlos@empresa.com	OF5	2	Vendedor	Fijo	1
>	6	Ruiz	Laura	106	laura@empresa.com	OF6	1	Supervisor	Fijo	1
>	7	Mendez	Jose	107	jose@empresa.com	OF7	6	Vendedor	Temporal	1
>	8	Castro	Elena	108	elena@empresa.com	OF8	6	Vendedor	Servicios	1
>	9	Vega	Mario	109	mario@empresa.com	OF9	1	Supervisor	Fijo	1
>	10	Rojas	Sofia	110	sofia@empresa.com	OF10	9	Vendedor	Temporal	1

SistemaDePuntoDeVentas.sql U															
clientes															
Properties DATA Log ER Monitor punto_ventas															
SELECT * FROM clientes LIMIT 100															
Search Results															
Cost: 13ms < Total 10															
id_cliente empresa apellido nombre telefono direccion ciudad departamento codigoPostal pais empleadoAtiende limiteCredito tipo_cliente activo email															
int(11) varchar(50) varchar(50) varchar(50) varchar(50) varchar(50) varchar(50) varchar(30) varchar(15) varchar(50) int(11) double enum('Minorista', 'May') tinyint(1) varchar(100)															
> 1 TechCorp Santos Miguel 6000-0001 Dir1 Panama Panama 0801 Panama 3 5000 Corporativo 1 c1@gmail.com															
> 2 Comercial SA Diaz Pedro 6000-0002 Dir2 Colon Colon 0301 Panama 4 3000 Mayorista 1 c2@gmail.com															
> 3 Retail PTY Lopez Ana 6000-0003 Dir3 David Chiriqui 0401 Panama 5 2000 Minorista 1 c3@gmail.com															
> 4 Distribuidora X Perez Luis 6000-0004 Dir4 Santiago Veraguas 0501 Panama 7 4000 Mayorista 1 c4@gmail.com															
> 5 Global Inc Martinez Rosa 6000-0005 Dir5 Chitre Herrera 0601 Panama 8 3500 Corporativo 1 c5@gmail.com															
> 6 Empresa 6 Ruiz Mario 6000-0006 Dir6 Penonomie Cocle 0701 Panama 3 2500 Minorista 1 c6@gmail.com															
> 7 Empresa 7 Gomez Laura 6000-0007 Dir7 Bocas Bocas 0901 Panama 4 1500 Mayorista 1 c7@gmail.com															
> 8 Empresa 8 Torres Jose 6000-0008 Dir8 Las Tablas Los Santos 1001 Panama 5 2800 Minorista 1 c8@gmail.com															
> 9 Empresa 9 Castro Elena 6000-0009 Dir9 Aguadulce Cocle 1101 Panama 7 3200 Corporativo 1 c9@gmail.com															
> 10 Empresa 10 Rojas Sofia 6000-0010 Dir10 Arrajan Panama Oeste 1201 Panama 8 4100 Mayorista 1 c10@mail.com															

SistemaDePuntoDeVentas.sql U

ordenes

Properties

DATA

Log

ER

Monitor

punto_ventas

SELECT * FROM ordenes LIMIT 100

Q

Search Results

</

SistemaDePuntoDeVentas.sql U

detallesordenes

Properties

DATA

Log

ER

Monitor

punto_ventas

SELECT * FROM detallesordenes LIMIT 100

Search Results

Cost: 5ms

Total 10

	id_orde int(11)	id_producto varchar(15)	cantidadPedida int(11)	valorUnitario double	ordenEntrega int(11)	* tipo_descuento enum('Ninguno';Prom	* aplica_impuesto tinyint(1)
>	1	P1	2	800	1	Ninguno	1
>	2	P2	5	20	1	Promocion	1
>	3	P3	1	150	1	Ninguno	1
>	4	P4	3	120	1	ClienteFrecuente	1
>	5	P5	1	500	1	Ninguno	0
>	6	P6	2	200	1	Promocion	1
>	7	P7	4	30	1	Ninguno	1
>	8	P8	1	90	1	ClienteFrecuente	1
>	9	P9	10	15	1	Ninguno	1
>	10	P10	1	1000	1	Promocion	0

SistemaDePuntoDeVentas.sql U

lineasproductos X

Properties

DATA

Log

ER

Monitor

punto_ventas

SELECT * FROM lineasproductos LIMIT 100

Search Results

Export

Cost: 9ms < 1 > Total 10

	id_lineaproducto int(11)	nombreLinea varchar(50)	textoDescripcion varchar(4000)	htmlDescripcion varchar(200)	imagen varchar(200)	* categoria enum('Electrico','Manu...)	* disponible tinyint(1)	codigo_linea varchar(20)
>	1	Electronica	Desc1	html1	img1	Electrico	1	L-001
>	2	Hogar	Desc2	html2	img2	Manual	1	L-002
>	3	Industrial	Desc3	html3	img3	Industrial	1	L-003
>	4	Oficina	Desc4	html4	img4	Manual	1	L-004
>	5	Tecnologia	Desc5	html5	img5	Electrico	1	L-005
>	6	Automotriz	Desc6	html6	img6	Industrial	1	L-006
>	7	Salud	Desc7	html7	img7	Manual	1	L-007
>	8	Construccion	Desc8	html8	img8	Industrial	1	L-008
>	9	Deportes	Desc9	html9	img9	Manual	1	L-009
>	10	Servicios	Desc10	html10	img10	Electrico	1	L-010

SistemaDePuntoDeVentas.sql U

productos X

Properties

DATA

Log

ER

Monitor

punto_ventas

SELECT * FROM productos LIMIT 100

Search Results

Export

Cost: 4ms

1

Total 10

	id_producto varchar(15)	nombreProducto varchar(70)	id_lineaProducto int(11)	escala varchar(10)	cantidad int(11)	precioventa double	MSRP double	tipo_producto enum('Fisico','Digital')	activo tinyint(1)	codigo_barra varchar(50)
>	P1	Laptop	1	Alta	50	800	900	Fisico	1	CB001
>	P10	Consultoria	10	N/A	0	1000	1200	Servicio	1	CB010
>	P2	Mouse	1	Media	100	20	25	Fisico	1	CB002
>	P3	Impresora	4	Alta	30	150	180	Fisico	1	CB003
>	P4	Taladro	8	Media	40	120	150	Fisico	1	CB004
>	P5	Servicio IT	10	N/A	0	500	600	Servicio	1	CB005
>	P6	Monitor	5	Alta	60	200	250	Fisico	1	CB006
>	P7	Teclado	1	Media	80	30	40	Fisico	1	CB007
>	P8	Silla Oficina	4	Alta	20	90	110	Fisico	1	CB008
>	P9	Aceite Motor	6	Media	200	15	20	Fisico	1	CB009

SistemaDePuntoDeVentas.sql U

pagos

Properties

DATA

Log

ER

Monitor

punto_ventas

SELECT * FROM pagos LIMIT 100

Search Results

Export

Cost: 8ms

1

Total 10

	id_pag int(11)	id_cliente int(11)	numeroFactura varchar(50)	fechaPago date	totalPago double	* metodo_pago enum('Efectivo','Tarjeta')	* confirmado tinyint(1)
	1	1	F001	2026-01-08	1600	Tarjeta	1
	2	2	F002	2026-01-09	100	Efectivo	1
	3	3	F003	2026-01-10	150	Transferencia	1
	4	4	F004	2026-01-11	360	Tarjeta	1
	5	5	F005	2026-01-12	500	Efectivo	1
	6	6	F006	2026-01-13	400	Transferencia	1
	7	7	F007	2026-01-14	120	Tarjeta	1
	8	8	F008	2026-01-15	90	Efectivo	1
	9	9	F009	2026-01-16	150	Transferencia	1
	10	10	F010	2026-01-17	1000	Tarjeta	1

Los puntos (III y IV) se basará bajo el contexto del **caso 1**
“Sistema Empresarial de Servicios Digitales”

PARTE III - Normalización y diseño lógico (20 puntos)

Normalización de la Base de Datos (10 pts)

- Aplicar normalización hasta Tercera Forma Normal (3FN).
- **1FN: Se separó los datos en entidades individuales (usuario, servicios, pago, auditoría y autenticación), básicamente cada tabla representa una sola entidad específica y cada atributo contiene su único valor.**
- **2FN: En el sistema tenía una relación muchos a muchos entre el usuario y el servicios, para resolverlo cree una entidad llamada Servicio_Contratado, y la tabla almacena su propia información del contrato (fecha_contratacion, fecha_vencimiento. estado.)**
- **3FN: Luego de todo esto en el diseño final hice en base a la 3FN: cada tabla depende únicamente de su clave primaria, no existen atributos que dependen de otros que no sean atributos clave, y la información esta separada por responsabilidad.**
- Explicar con sus propias palabras:
 - Qué problemas se resolvieron en cada etapa
 - **1FN: Se eliminaron los grupos que se repetían y también se garantizó que cada una de los campos tenga un único valor.**
 - **2FN: Se resolvió el problema de la relación de muchos a muchos, cree la entidad Servicio_Contratado y también se eliminaron las dependencias parciales.**
 - **3FN: Se redujo las redundancias de datos, se aseguro cada atributo dependiente únicamente de la clave primaria y se eliminaron dependencias transitivas.**
 - Por qué el diseño final es óptimo
 - **Porque se redujeron las redundancias de datos, evitamos la inconsistencia de actualizaciones, elimine anomalías de inserción y eliminación, mejore la integridad referencial, separe las responsabilidades de identidades y cumple con la tercera forma normal.**

NOTA: Respuestas copiadas de internet o sin justificación técnica no suman puntos.

Implementación SQL – DDL (10 pts)

- Crear todas las tablas del modelo usando SQL.
- Cada tabla debe incluir:
 - Clave primaria
 - Claves foráneas
 - Restricciones (NOT NULL, UNIQUE, CHECK)
- Por cada tabla:
 - 1 campo tipo lista (ENUM o tabla catálogo)
 - 1 campo tipo booleano
 - 1 restricción UNIQUE

C: > Users > nmcom > Desktop > EmpresaServicios.sql > ...

▷Run | ☞ Active: 127.0.0.1 | ☐ punto_ventas

1 CREATE DATABASE empresa_servicios;

▷Run

2 USE empresa_servicios;

3

4 -- Tabla de usuarios

▷Run | ☐Select

5 CREATE TABLE usuario (

6 id_usuario INT AUTO_INCREMENT PRIMARY KEY,

7 nombre VARCHAR(50) NOT NULL,

8 apellido VARCHAR(50) NOT NULL,

9 email VARCHAR(100) NOT NULL UNIQUE,

10 telefono VARCHAR(15),

11 fecha_registro DATETIME NOT NULL,

12 estado ENUM('Activo','Inactivo') NOT NULL,

13 verificado BOOLEAN NOT NULL,

14 codigo_cliente VARCHAR(20) UNIQUE

15);

16

6

7 -- Tabla de autenticacion

▷Run | ☐Select

8 CREATE TABLE autenticacion (

9 id_autenticacion INT AUTO_INCREMENT PRIMARY KEY,

0 username VARCHAR(50) NOT NULL UNIQUE,

1 password_hash VARCHAR(255) NOT NULL,

2 ultimo_login DATETIME,

3 intentos_fallidos INT DEFAULT 0 CHECK (intentos_fallidos >= 0),

4 tipo_usuario ENUM('Admin','Cliente') NOT NULL,

5 bloqueado BOOLEAN NOT NULL,

6 id_usuario INT NOT NULL UNIQUE,

7 FOREIGN KEY (id_usuario) REFERENCES usuario(id_usuario)

8);

9

9

0 -- Tabla de servicio

▷Run | ☐Select

1 CREATE TABLE servicio (

2 id_servicio INT AUTO_INCREMENT PRIMARY KEY,

3 nombre_servicio VARCHAR(100) NOT NULL UNIQUE,

4 descripcion VARCHAR(255),

5 precio DECIMAL(10,2) NOT NULL CHECK (precio >= 0),

6 categoria ENUM('Streaming','Cloud','Soporte','Software') NOT NULL,

7 activo BOOLEAN NOT NULL,

8 codigo_servicio VARCHAR(20) UNIQUE

9);

0

```

40
41 -- Tabla servicio_contratado
  ▶Run | ◻Select
42 ✓ CREATE TABLE servicio_contratado (
43     id_contrato INT AUTO_INCREMENT PRIMARY KEY,
44     fecha_contratacion DATE NOT NULL,
45     fecha_vencimiento DATE NOT NULL,
46     estado ENUM('Activo','Cancelado','Suspendido') NOT NULL,
47     renovacion_automatica BOOLEAN NOT NULL,
48     numero_contrato VARCHAR(30) UNIQUE,
49     id_usuario INT NOT NULL,
50     id_servicio INT NOT NULL,
51     FOREIGN KEY (id_usuario) REFERENCES usuario(id_usuario),
52     FOREIGN KEY (id_servicio) REFERENCES servicio(id_servicio)
53 );
54

```

```

4
5 -- Tabla de pago
  ▶Run | ◻Select
6 CREATE TABLE pago (
7     id_pago INT AUTO_INCREMENT PRIMARY KEY,
8     fecha_pago DATETIME NOT NULL,
9     monto DECIMAL(10,2) NOT NULL CHECK (monto > 0),
10    metodo_pago ENUM('Tarjeta','Transferencia','PayPal') NOT NULL,
11    confirmado BOOLEAN NOT NULL,
12    numero_factura VARCHAR(30) UNIQUE,
13    id_contrato INT NOT NULL,
14    FOREIGN KEY (id_contrato) REFERENCES servicio_contratado(id_contrato)
15 );
16

```

```

7
8 --Tabla de autoridad_acceso
  ▶Run | ◻Select
9 ✓ CREATE TABLE auditoria_acceso (
10     id_auditoria INT AUTO_INCREMENT PRIMARY KEY,
11     fecha_hora DATETIME NOT NULL,
12     direccion_ip VARCHAR(45) NOT NULL,
13     tipo_evento ENUM('Login','Logout','IntentoFallido') NOT NULL,
14     exitoso BOOLEAN NOT NULL,
15     codigo_evento VARCHAR(30) UNIQUE,
16     id_usuario INT NOT NULL,
17     FOREIGN KEY (id_usuario) REFERENCES usuario(id_usuario)
18 ); 26ms
19

```

PARTE IV – DML, CRUD y Vistas (40 puntos)

Inserción de Datos – DML (10 pts)

- Insertar **mínimo 10 registros** por tabla.
- Los datos deben ser:
 - Realistas
 - Coherentes
 - Relacionados correctamente

EmpresaServicios.sql u

servicio_contratado

PropertiesDATALogERMonitorempresa_servicios

SELECT * FROM servicio_contratado LIMIT 100

Search Results

Cost: 6ms1Total 10

	id_contrato int(11)	nombre date	apellido date	email enum('Activo', 'Cancela	telefono tinyint(1)	estado varchar(30)	fecha_registro int(11)	fecha_hora int(11)
>	1	2026-01-01	2026-12-31	Activo	1	CON-001	1	1
>	2	2026-01-02	2026-12-31	Activo	1	CON-002	2	2
>	3	2026-01-03	2026-12-31	Activo	0	CON-003	3	3
>	4	2026-01-04	2026-12-31	Suspendido	0	CON-004	4	4
>	5	2026-01-05	2026-12-31	Activo	1	CON-005	5	5
>	6	2026-01-06	2026-12-31	Activo	1	CON-006	6	6
>	7	2026-01-07	2026-12-31	Activo	0	CON-007	7	7
>	8	2026-01-08	2026-12-31	Cancelado	0	CON-008	8	8
>	9	2026-01-09	2026-12-31	Activo	1	CON-009	9	9
>	10	2026-01-10	2026-12-31	Activo	1	CON-010	10	10

EmpresaServicios.sql u

usuario

PropertiesDATALogERMonitorempresa_servicios

SELECT * FROM usuario LIMIT 100

Search Results

Cost: 5ms1Total 10

	id_usuario int(11)	nombre varchar(50)	apellido varchar(50)	email varchar(100)	telefono varchar(15)	fecha_registro datetime	estado enum('Activo', 'Inactive	verificado tinyint(1)	codigo_cliente varchar(20)
>	1	Carlos	Poveda	carlos@empresa.com	6000-1001	2026-02-24 23:56:50	Activo	1	CLI-001
>	2	Ana	Gomez	ana@empresa.com	6000-1002	2026-02-24 23:56:50	Activo	1	CLI-002
>	3	Luis	Martinez	luis@empresa.com	6000-1003	2026-02-24 23:56:50	Activo	0	CLI-003
>	4	Maria	Lopez	maria@empresa.com	6000-1004	2026-02-24 23:56:50	Activo	1	CLI-004
>	5	Pedro	Sanchez	pedro@empresa.com	6000-1005	2026-02-24 23:56:50	Inactivo	0	CLI-005
>	6	Laura	Diaz	laura@empresa.com	6000-1006	2026-02-24 23:56:50	Activo	1	CLI-006
>	7	Jorge	Ruiz	jorge@empresa.com	6000-1007	2026-02-24 23:56:50	Activo	1	CLI-007
>	8	Elena	Torres	elena@empresa.com	6000-1008	2026-02-24 23:56:50	Activo	0	CLI-008
>	9	Miguel	Castro	miguel@empresa.com	6000-1009	2026-02-24 23:56:50	Activo	1	CLI-009
>	10	Sofia	Rojas	sofia@empresa.com	6000-1010	2026-02-24 23:56:50	Activo	1	CLI-010

EmpresaServicios.sql u

auditoria_acceso

PropertiesDATALogERMonitorempresa_servicios

SELECT * FROM auditoria_acceso LIMIT 100

Search Results

Cost: 7ms1Total 10

	id_auditoria int(11)	fecha_hora datetime	direccion_ip varchar(45)	tipo_evento enum('Login', 'Logout',	exitoso tinyint(1)	codigo_evento varchar(30)	id_usuario int(11)
>	1	2026-02-24 23:59:40	192.168.1.1	Login	1	EV-001	1
>	2	2026-02-24 23:59:40	192.168.1.2	Login	1	EV-002	2
>	3	2026-02-24 23:59:40	192.168.1.3	IntentoFallido	0	EV-003	3
>	4	2026-02-24 23:59:40	192.168.1.4	Login	1	EV-004	4
>	5	2026-02-24 23:59:40	192.168.1.5	Logout	1	EV-005	5
>	6	2026-02-24 23:59:40	192.168.1.6	Login	1	EV-006	6
>	7	2026-02-24 23:59:40	192.168.1.7	IntentoFallido	0	EV-007	7
>	8	2026-02-24 23:59:40	192.168.1.8	Login	1	EV-008	8
>	9	2026-02-24 23:59:40	192.168.1.9	Logout	1	EV-009	9
>	10	2026-02-24 23:59:40	192.168.1.10	Login	1	EV-010	10

EmpresaServicios.sql u

autenticacion

PropertiesDATALogERMonitorempresa_servicios

SELECT * FROM autenticacion LIMIT 100

Search Results

Cost: 9ms1Total 10

	id_autenticacion int(11)	username varchar(50)	password_hash varchar(255)	ultimo_login datetime	intentos_fallidos int(11)	tipo_usuario enum('Admin', 'Cliente	bloqueado tinyint(1)	id_usuario int(11)
>	1	carlos_admin	hash1	2026-02-24 23:59:05	0	Admin	0	1
>	2	ana_user	hash2	2026-02-24 23:59:05	0	Cliente	0	2
>	3	luis_user	hash3	2026-02-24 23:59:05	1	Cliente	0	3
>	4	maria_user	hash4	2026-02-24 23:59:05	0	Cliente	0	4
>	5	pedro_user	hash5	2026-02-24 23:59:05	2	Cliente	0	5
>	6	laura_user	hash6	2026-02-24 23:59:05	0	Cliente	0	6
>	7	jorge_user	hash7	2026-02-24 23:59:05	0	Cliente	0	7
>	8	elena_user	hash8	2026-02-24 23:59:05	1	Cliente	0	8
>	9	miguel_user	hash9	2026-02-24 23:59:05	0	Cliente	0	9
>	10	sofia_user	hash10	2026-02-24 23:59:05	0	Cliente	0	10

EmpresaServicios.sql U

pago

PropertiesDATALogERMonitor

empresa_servicios

SELECT * FROM pago LIMIT 100

Search Results

Cost: 9ms1Total 10

	id_pago int(11)	fecha_pago datetime	monto decimal(10,2)	metodo_pago enum('Tarjeta','Transfe	confirmado tinyint(1)	numero_factura varchar(30)	id_contrato int(11)
>	1	2026-02-24 23:59:35	15.99	Tarjeta	1	FAC-001	1
>	2	2026-02-24 23:59:35	9.99	Transferencia	1	FAC-002	2
>	3	2026-02-24 23:59:35	19.99	Tarjeta	1	FAC-003	3
>	4	2026-02-24 23:59:35	29.99	PayPal	1	FAC-004	4
>	5	2026-02-24 23:59:35	12.50	Tarjeta	1	FAC-005	5
>	6	2026-02-24 23:59:35	99.99	Transferencia	1	FAC-006	6
>	7	2026-02-24 23:59:35	8.99	Tarjeta	1	FAC-007	7
>	8	2026-02-24 23:59:35	14.99	PayPal	1	FAC-008	8
>	9	2026-02-24 23:59:35	45.00	Tarjeta	1	FAC-009	9
>	10	2026-02-24 23:59:35	11.00	Transferencia	1	FAC-010	10

EmpresaServicios.sql U

servicio

PropertiesDATALogERMonitor

empresa_servicios

SELECT * FROM servicio LIMIT 100

Search Results

Cost: 5ms1Total 10

	id_servicio int(11)	nombre_servicio varchar(100)	descripcion varchar(255)	precio decimal(10,2)	categoria enum('Streaming','Cloi	activo tinyint(1)	codigo_servicio varchar(20)
>	1	Streaming Pro	Plataforma streaming HD	15.99	Streaming	1	SER-001
>	2	Cloud Basic	Almacenamiento 100GB	9.99	Cloud	1	SER-002
>	3	Cloud Premium	Almacenamiento 1TB	19.99	Cloud	1	SER-003
>	4	Soporte 24/7	Soporte tecnico permanent	29.99	Soporte	1	SER-004
>	5	Antivirus	Proteccion empresarial	12.50	Software	1	SER-005
>	6	ERP Empresarial	Sistema ERP completo	99.99	Software	1	SER-006
>	7	Backup Online	Respaldo automatico	8.99	Cloud	1	SER-007
>	8	Hosting Web	Hosting empresarial	14.99	Cloud	1	SER-008
>	9	Licencia Office	Licencia anual office	45.00	Software	1	SER-009
>	10	VPN Empresarial	Conexion segura remota	11.00	Software	1	SER-010

Operaciones CRUD (15 pts)

Seleccionar una entidad principal del sistema y realizar:

- CREATE → Inserción
- READ → Consulta filtrada
- UPDATE → Actualización controlada
- DELETE → Eliminación lógica o física (justificar)

Se deben entregar los **scripts SQL** y explicar el propósito de cada operación.

```
158
159 --Opreaciones CRUD
160
161 -- CREATE Registra un nuevo usuario en la tabla de usuarios
162 >Run
163 INSERT INTO usuario (nombre,apellido,email,telefono,fecha_registro,estado,verificado,codigo_cliente)
164 VALUES ('Nuevo','Cliente','nuevo@empresa.com','6000-2020',NOW(),'Activo',1,'CLI-011');
165
166 -- READ Consulta usuarios activos, por ejemplo para analisis comercial
167 >Run | +Tab | JSON
168 SELECT nombre, apellido, email
169 FROM usuario
170 WHERE estado = 'Activo';
171
172 -- UPDATE desactiva usuario sin eliminarlo, por ejemplo para mantener historial de contratos
173 >Run | Select
174 UPDATE usuario
175 SET estado = 'Inactivo'
176 WHERE id_usuario = 3;
177
178 --DELETE no elimina fisicamente un usuario, sino que lo marca como inactivo para preservar integridad referencial
179 >Run | Select
180 UPDATE usuario
181 SET estado = 'Inactivo'
182 WHERE id_usuario = 5;
```

- CREATE Registra un nuevo usuario en la tabla de usuarios
- READ Consulta usuarios activos, por ejemplo para analisis comercial
- UPDATE desactiva usuario sin eliminarlo, por ejemplo para mantener historial de contratos
- DELETE no elimina fisicamente un usuario, sino que lo marca como inactivo para preservar integridad referencial

Creación y Uso de VISTAS SQL (15 pts)

Requerimientos obligatorios

Crear **mínimo 3 VISTAS**, cumpliendo:

- 1. **Vista de negocio**
 - Consolidación de datos relevantes para toma de decisiones
 - Ejemplo: usuarios + servicios + pagos
- 2. **Vista de seguridad**
 - Oculta datos sensibles (contraseñas, tokens, etc.)
- 3. **Vista de auditoría**
 - Muestra accesos, fechas, estado de usuarios o transacciones

```
179
180  -- VISTAS SQLS
181
182  -- Vista de negocio / utilidad: analiza el ingreso por usuario y servicio
183  ▷Run| ⌵Select
184  CREATE VIEW vista_negocio AS
185  SELECT
186    u.nombre,
187    u.apellido,
188    s.nombre_servicio,
189    p.monto,
190    p.fecha_pago
191  FROM usuario u
192  JOIN servicio_contratado sc ON u.id_usuario = sc.id_usuario
193  JOIN servicio s ON sc.id_servicio = s.id_servicio
194  JOIN pago p ON sc.id_contrato = p.id_contrato;
195
196  --Vista de seguridad / utilidad: muestra la informacion sin exponer la password_hash
197  ▷Run| ⌵Select
198  CREATE VIEW vista_seguridad AS
199  SELECT
200    u.id_usuario,
201    u.nombre,
202    u.email,
203    a.username,
204    a.bloqueado
205  FROM usuario u
206  JOIN autenticacion a ON u.id_usuario = a.id_usuario;
207
208  -- Vista de auditoria / utilidad: monitorea los eventos sospechosos y tambien los accesos
209  ▷Run| ⌵Select
210  CREATE VIEW vista_auditoria AS
211  SELECT
212    u.nombre,
213    a.fecha_hora,
214    a.tipo_evento,
215    a.exitoso
216  FROM auditoria_acceso a
217  JOIN usuario u ON a.id_usuario = u.id_usuario;
```

- **Vista de negocio / utilidad: analiza el ingreso por usuario y servicio**
- **Vista de seguridad / utilidad: muestra la informacion sin exponer la password_hash**
- **Vista de auditoria / utilidad: monitorea los eventos sospechosos y tambien los accesos**

<

EmpresaServicios.sql

u

vista_negocio

x

Properties

DATA

Log

ER

Monitor

empresa_servicios

SELECT * FROM vista_negocio LIMIT 100

Q

Search Results

EmpresaServicios.sql u

vista_seguridad X

Properties

DATA

Log

ER

Monitor

empresa_servicios

SELECT * FROM vista_seguridad LIMIT 100

Search Results

Export

Cost: 11ms

1

Total 10

	* id_usuario int(11)	* nombre varchar(50)	* email varchar(100)	* username varchar(50)	* bloqueado tinyint(1)
>	1	Carlos	carlos@empresa.com	carlos_admin	0
>	2	Ana	ana@empresa.com	ana_user	0
>	3	Luis	luis@empresa.com	luis_user	0
>	4	Maria	maria@empresa.com	maria_user	0
>	5	Pedro	pedro@empresa.com	pedro_user	0
>	6	Laura	laura@empresa.com	laura_user	0
>	7	Jorge	jorge@empresa.com	jorge_user	0
>	8	Elena	elena@empresa.com	elena_user	0
>	9	Miguel	miguel@empresa.com	miguel_user	0
>	10	Sofia	sofia@empresa.com	sofia_user	0

Evaluación

- Correcta construcción de la vista
- Uso real (SELECT sobre la vista)
- Justificación de su utilidad

Lineamiento e información a entregar

1. Desarrollar cada uno de los puntos
2. Crear el repositorio de carpeta ***“PARCIAL_01”***
3. Subir la copia de toda la base de datos
4. Capturas de cada de los procesos (Sentencia SQL y resultado)
5. Documentar todo el proceso y agregar las lecciones aprendidas.

Se puede realizar individual o en equipo con **un máximo de 2 integrantes**, con la condición que por cada integrante adicional se deben agregar 10 registros en cada módulo, en caso de no cumplir con esta cláusula, se le baja automáticamente **-20 puntos de la nota final**.

Se debe subir a la plataforma el link o todos los documentos.

¡Buena Suerte!