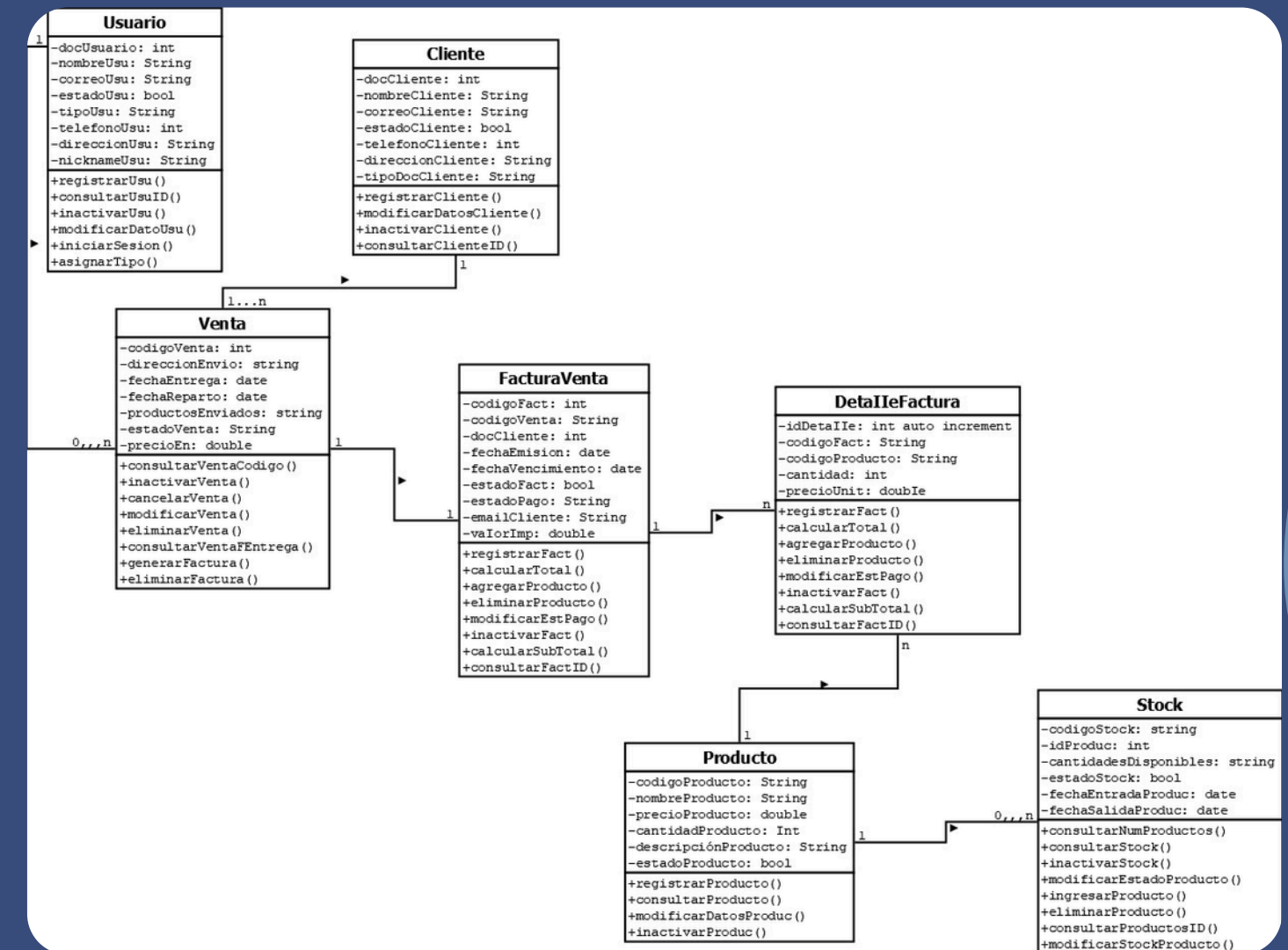
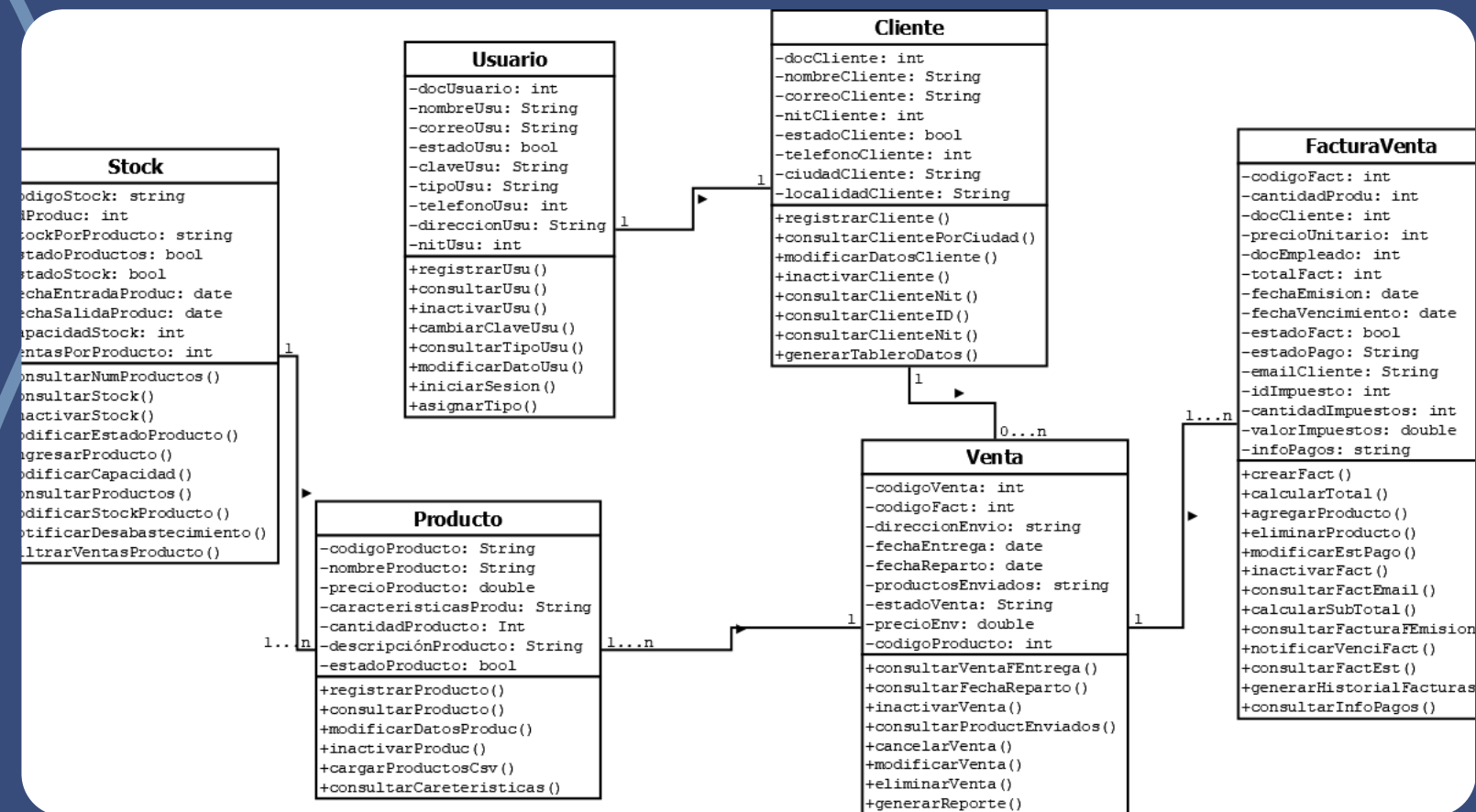




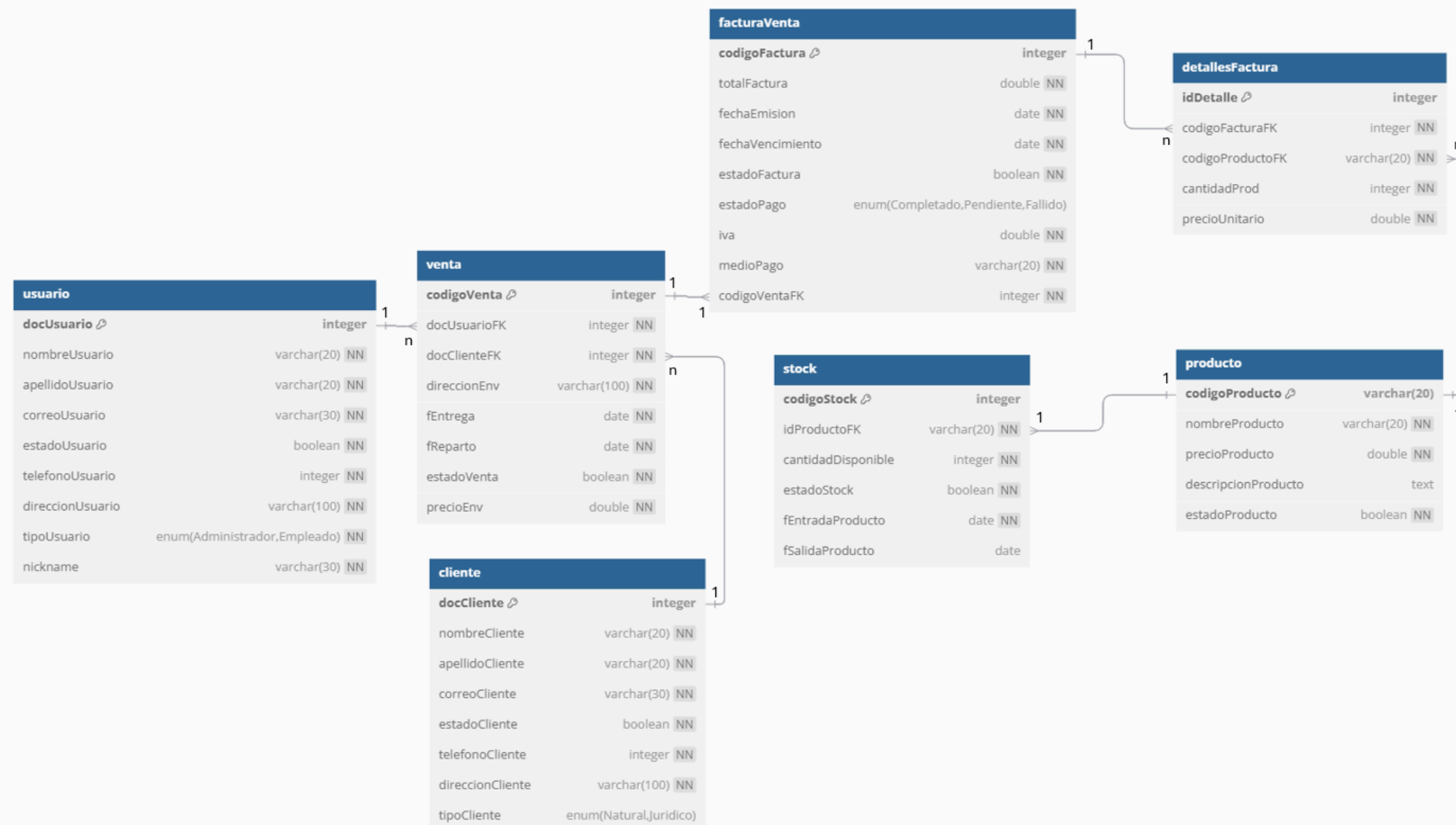
PALACIO EL TIEMPO



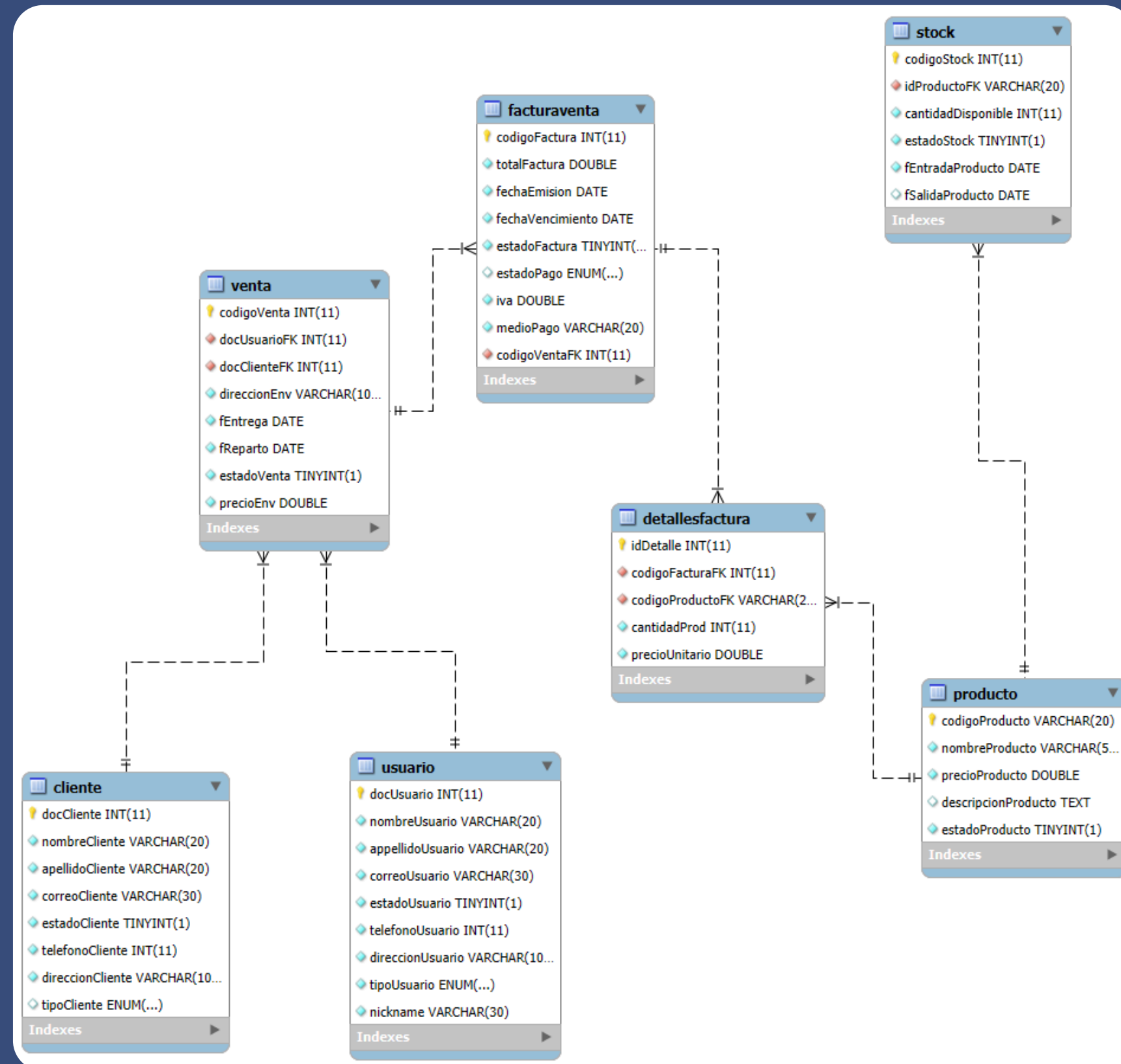
DIAGRAMAS DE CLASES



MODELO LÓGICO



MODELO FÍSICO



HISTORIAS DE USUARIO

✓

HU019: Modificar disponibilidad de producto

en la lista **DONE**

Notificaciones

Seguir

Descripción

Editar

Como vendedor necesito modificar la disponibilidad de los productos en el sistema, con el fin de mantener actualizada la información para los clientes

✓ criterios

Ocultar los elementos marcados

Eliminar

100%

✓

cambiar disponibilidad por ID

✓

Permitir filtrado según el estado

✓

No Permitir que haya un producto sin estado, o con 2 estados

Añade un elemento

Actividad

Mostrar detalles

DM

Escribe un comentario...

DM

DANIEL MORENO ha añadido esta tarjeta a Backlog

25 feb 2025, 21:51

Unirse

Miembros

Etiquetas

Checklist

Fechas

Adjunto

Portada

Campos personalizados

Power-Ups

Añadir Power-Ups


Automatización

Botón Añadir

Acciones

Mover

Done

Done

Done

3

✓ Correccion del diagrama de Clases

7 abr - 9 abr

MC

✓ Hacer Modelo Lógico

5 abr - 5 abr

5/5

LC

✓ Corrección Alcance

7 abr - 7 abr

3/3

DM

✓ Corrección Objetivos

7 abr - 8 abr

2/2

MC

DICCIONARIO DE DATOS

Nombre Tabla	Usuario			
Fecha	17/03/2025			
Descripción	Tabla que almacena los datos del usuario de la base de datos			
Campo	Tipo de Dato	Tamaño	Restricciones	Descripción
docUsuario	int		PK	Identificación del usuario y es llave primaria
nombreUsuario	varchar	20	not Null	Nombre del Usuario
apellidoUsuario	varchar	20	not Null	Apellido del Usuario
correoUsuario	varchar	50	not Null	Correo Usuario
estadoUsuario	bool		not Null	Indicador del estado del Usuario
telefonoUsuario	int unsigned		not Null	Telefono del usuario
direccionUsuario	varchar	100	not Null	Dirección vivienda usuario
tipoUsuario	enum		not Null	Indicador tipo de usuario, ya se administrador o empleado
nickname	varchar	30	not Null	Identificadord del usuario en la base de datos

Nombre Tabla	FacturaVenta			
Fecha	17/03/2025			
Descripción	Tabla que almacena los datos referentes a una factura de una venta			
Campo	Tipo de Dato	Tamaño	Restricciones	Descripción
codigoFactura	int		PK auto_increment	Identificador dde la factura y llave foránea
totalFactura	double		not null	Cantidad total a pagar
fechaEmision	date		not null	Fecha en la que se emite la factura
fechaVencimiento	date		not null	Fecha de vencimiento de la factura
estadoFactura	bool		not null	Indicador del estado de la factura: Activa o inactiva
estadoPago	enum		not null	Estado en el que se encuentra la factura: pendiente, pagado, cancelado
iva	double		not null	Sumatoria impuestos de los productos
medioPago	varchar	20	not null	Medio por el cual el cliente paga su compra
codigoVentaFK	int		FK	Codigo de la venta asociada

Nombre Tabla	Producto			
Fecha	17/03/2025			
Descripción	Tabla que almacena los datos referentes a un producto en específico			
Campo	Tipo de Dato	Tamaño	Restricciones	Descripción
codigoProducto	Varchar	20	PK	Identificador del producto y lleva primaria
nombreProducto	Varchar	20	not null	Nombre del Producto
precioProducto	double		not null	Costo o precio de producto en particular
descripcionProducto	text		null	Descripción características del producto
estadoProducto	bool		not null	Indicador estado producto ya sea activo o inactivo

Nombre Tabla	Venta			
Fecha	17/03/2025			
Descripción	Tabla que almacena los datos referentes a una venta en específico			
Campo	Tipo de Dato	Tamaño	Restricciones	Descripción
codigoVenta	int		PK auto_increment	Identificador de la venta y llave primaria
direccionEnv	Varchar	100	not null	Dirección vivienda o empresa cliente
fEntrega	date		not null	Fecha de entrega de productos de la venta al cliente
fReparto	date		not null	Fecha en la que el producto comienza el proceso de distribución hacia el cliente.
estadoVenta	bool		not null	Indicador estado venta ya sea activa o inactiva
precioEnv	double		not null	Indicador precio de una venta
docUsuarioFK	int		not null	Identificador foráneo del usuario
docClienteFK	int		FK	Identificador cliente de cierta venta

Nombre Tabla	Stock			
Fecha	17/03/2025			
Descripción	Tabla que almacena los datos referentes a la disponibilidad de un producto en específico			
Campo	Tipo de Dato	Tamaño	Restricciones	Descripción
codigoStock	int		PK auto_increment	Identificador del Stock
idProductoFK	Varchar	20	FK	Identificador Producto
cantidadDisponible	int		not null	Cantidad disponible del producto
estadoStock	bool		not null	Estado del stock, ya sea activo o inactivo.
fEntradaProducto	date		not null	Fecha de entrada de un producto al stock
fSalidaProducto	date		not null	Fecha de salida de un producto del stock

CREACIÓN DE LA BASE DE DATOS

```
create database palacioDelTiempo;  
  
use palacioDelTiempo;
```

	#	Time	Action
✓	1	09:29:14	create database palacioDelTiempo
✓	2	09:29:16	use palacioDelTiempo

CREACION TABLAS

```
create table FacturaVenta(  
  codigoFactura int auto_increment primary key,  
  totalFactura double not null,  
  fechaEmision date not null,  
  fechaVencimiento date not null,  
  estadoFactura bool not null,  
  estadoPago enum ('Completado', 'Pendiente', 'Fallido'),  
  iva double not null,  
  medioPago varchar(20) not null,  
  codigoVentaFK int not null,  
  foreign key (codigoVentaFK) references Venta(codigoVenta)  
  on delete cascade on update cascade  
);
```

```
create table Producto(  
  codigoProducto varchar(20) primary key,  
  nombreProducto varchar(20) not null,  
  precioProducto double not null,  
  descripcionProducto text null,  
  estadoProducto bool not null  
);
```

```
create table DetallesFactura(  
  idDetalle int auto_increment primary key,  
  codigoFacturaFK int not null,  
  codigoProductoFK varchar(20) not null,  
  cantidadProd int not null,  
  precioUnitario double not null,  
  foreign key (codigoFacturaFK) references FacturaVenta(codigoFactura)  
  on delete cascade on update cascade,  
  foreign key (codigoProductoFK) references Producto(codigoProducto)  
  on delete cascade on update cascade  
);
```

```
create table Stock(  
  codigoStock int auto_increment primary key,  
  idProductoFK varchar(20) not null,  
  cantidadDisponible int not null,  
  estadoStock bool not null,  
  fEntradaProducto date not null,  
  fSalidaProducto date,  
  foreign key (idProductoFK) references Producto(codigoProducto)  
  on delete cascade on update cascade  
);
```

CREACION TABLAS

```
create table Usuario(  
  docUsuario int primary key,  
  nombreUsuario varchar(20) not null,  
  apellidoUsuario varchar(20) not null,  
  correoUsuario varchar(50) not null,  
  estadoUsuario bool not null,  
  telefonoUsuario int unsigned not null,  
  direccionUsuario varchar(100) not null,  
  tipoUsuario enum('Administrador', 'Empleado') not null,  
  nickname varchar(30) not null  
);
```

```
create table Cliente(  
  docCliente int primary key,  
  nombreCliente varchar(20) not null,  
  apellidoCliente varchar(20) not null,  
  correoCliente varchar(50) not null,  
  estadoCliente bool not null,  
  telefonoCliente int unsigned not null,  
  direccionCliente varchar(100) not null,  
  tipoCliente enum('Natural', 'Juridico')  
);
```

```
create table Venta(  
  codigoVenta int auto_increment primary key,  
  docUsuarioFK int not null,  
  docClienteFK int not null,  
  direccionEnv varchar(100) not null,  
  fEntrega date not null,  
  fReparto date not null,  
  estadoVenta bool not null,  
  precioEnv double not null,  
  foreign key (docUsuarioFK) references Usuario(docUsuario)  
  on delete cascade on update cascade,  
  foreign key (docClienteFK) references Cliente(docCliente)  
  on delete cascade on update cascade  
);
```



CREACION TABLAS

✓	3	09:33:35	create table Usuario(docUsuario int primary key, nombreUsuario varchar(20) not null, apellidoUsuario varchar(20) not null, co
✓	4	09:33:40	create table Cliente(docCliente int primary key, nombreCliente varchar(20) not null, apellidoCliente varchar(20) not null,
✓	5	09:33:45	create table Venta(codigoVenta int auto_increment primary key, docUsuarioFK int not null, docClienteFK int not null, d

✓	5	09:33:45	create table Venta(codigoVenta int auto_increment primary key, docUsuarioFK int not null, docClienteFK int not null, direccionEnv varchar(100) not...
✓	6	09:34:22	create table Producto(codigoProducto varchar(20) primary key, nombreProducto varchar(50) not null, precioProducto double not null, descripcionP...
✓	7	09:34:25	create table FacturaVenta(codigoFactura int auto_increment primary key, totalFactura double not null, fechaEmision date not null, fechaVencimient...
✓	8	09:34:28	create table DetallesFactura(idDetalle int auto_increment primary key, codigoFacturaFK int not null, codigoProductoFK varchar(20) not null, cantida...
✓	9	09:34:32	create table Stock(codigoStock int auto_increment primary key, idProductoFK varchar(20) not null, cantidadDisponible int not null, estadoStock ...

REGISTROS

`insert into Usuario values`

```
(981234585, 'Clara Lucía', 'Morales González', 'clara.morales@hotmail.com', true, 30103456, 'Carrera 70 #45-67, Bogotá', 'Empleado', 'claralucia'),
(901234585, 'Antonio José', 'González Díaz', 'antonio.gonzalez@hotmail.com', true, 3010012345, 'Carrera 35 #23-45, Bogotá', 'Empleado', 'antoniojose'),
(109234595, 'Martín Felipe', 'Hernández Ruiz', 'martin.hernandez@hotmail.com', true, 3011012345, 'Avenida 12 #56-78, Bogotá', 'Empleado', 'martinfelipe'),
(881234575, 'Pedro Pablo', 'Vázquez Ruiz', 'pedropablo.vazquez@gmail.com', true, 3009012345, 'Carrera 45 #67-89, Bogotá', 'Empleado', 'pedropablo'),
(931234580, 'Miguel Ángel', 'Díaz Gómez', 'miguel.diaz@gmail.com', true, 3009567890, 'Calle 70 #56-78, Bogotá', 'Empleado', 'miguelangel'),
```

✓ 10 09:36:08 insert into Usuario values (981234585, 'Clara Lucía', 'Morales González', 'clara.morales@hotmail.com', true, 30103456, 'Carrera 70 #45-67, Bogotá', 'Emple...

```
insert into producto values ('P51', 'CORREA NATO ROJA', 28000.00, 'Correa de tela resistente para relojes deportivos', true),
('P52', 'RELOJ CASIO EDIFICE', 620000.00, 'Reloj elegante con cronómetro y correa metálica', true),
('P53', 'ESTUCHE PREMIUM MADERA', 97000.00, 'Estuche para colección de relojes con vidrio frontal', true),
('P54', 'KIT HERRAMIENTAS RELOJERO', 56000.00, 'Incluye destornilladores, pinzas y herramientas para ajuste', true),
('P55', 'GAFAS AVIADOR CLÁSICAS', 138000.00, 'Gafas con montura metálica estilo aviador', true),
('P56', 'RELOJ DIESEL RASP', 790000.00, 'Diseño moderno y robusto para uso diario', true),
('P57', 'CAJA RELOJES VIAJE', 45000.00, 'Estuche acolchado portátil para viajes', true),
('P58', 'CORREA PIEL MARRÓN', 35000.00, 'Correa de piel natural con hebilla clásica', true),
('P59', 'RELOJ SKAGEN HAGEN', 410000.00, 'Diseño delgado y minimalista con correa de malla', true);
```

✓ 11 09:37:00 insert into producto values ('P51', 'CORREA NATO ROJA', 28000.00, 'Correa de tela resistente para relojes deportivos', true), ('P52', 'RELOJ CASIO EDIF...

CONSULTAS BÁSICAS

-- Consultas basicas

```
select * from Cliente;
```

```
SELECT * FROM detallesfactura;
```

```
SELECT * FROM facturaVenta;
```

```
SELECT * FROM producto;
```

	codigoProducto	nombreProducto	precioProducto	descripcionProducto	estadoProducto
▶	P1	THOM GAFAS DE SOL迪特眼镜	26057.16	Gafas de sol	1
	P10	RELOJ ORIENT CLASSIC	790000	Reloj automático elegante para hombre	1
	P11	RELOJ DIESEL DZ4323	850000	Reloj cronógrafo para hombre	1
	P12	RELOJ FOSSIL CH2565	680000	Reloj deportivo de cuero	1
	P13	RELOJ SKAGEN ANITA	470000	Reloj minimalista para mujer	1
	P14	RELOJ MVMT CLASSIC	520000	Reloj moderno con diseño limpio	1
	P15	RELOJ TOMMY HILFIGER	650000	Reloj con cronógrafo y correa de acero	1
	P16	RELOJ GUESS GW0030L2	570000	Reloi de moda para mujer	1

✓	37	09:45:20	select * from Cliente LIMIT 0, 1000
✓	38	09:46:56	SELECT * FROM detallesfactura LIMIT 0, 1000
✓	39	09:47:02	SELECT * FROM facturaVenta LIMIT 0, 1000
✓	40	09:47:04	SELECT * FROM producto LIMIT 0, 1000

CONSULTAS ESPECIFICAS

```
-- Clientes con más de una venta
SELECT nombreCliente, apellidoCliente, COUNT(codigoVenta) AS 'TotalVentas'
FROM Cliente JOIN Venta ON Cliente.docCliente = Venta.docClienteFK
GROUP BY Cliente.docCliente
HAVING COUNT(Venta.codigoVenta) > 1;
```

✓ 42 09:50:55 SELECT nombreCliente, apellidoCliente, COUNT(codigoVenta) AS 'TotalVentas' FROM Cliente JOIN Venta ON Cliente.docCliente = Venta.docClienteF...

	nombreCliente	apellidoCliente	TotalVentas
▶	YEISON	ESTUPIÑAN TORRES	5

CONSULTAS ESPECIFICAS

```
select codigoProducto, nombreProducto, descripcionProducto from Producto
where descripcionProducto like '%estuche%';
```

✓ 43 09:52:49 select codigoProducto, nombreProducto, descripcionProducto from Producto where descripcionProducto like "%estuche%" LIMIT 0, 1000

	codigoProducto	nombreProducto	descripcionProducto
▶	P2	BOLSAS RAYBAN	estuche
	P21	ESTUCHE DUAL WATCH	Estuche para guardar dos relojes
	P53	ESTUCHE PREMIUM MADERA	Estuche para colección de relojes con vidrio fro...
	P57	CAJA RELOJES VIAJE	Estuche acolchado portátil para viajes
✱	NULL	NULL	NULL

MODIFICACIONES

```
update Producto  
set estadoProducto = false  
where codigoProducto = 'p28';
```

Actualizar (Inactiva)
estado producto

✓ 44 09:55:17 update Producto set estadoProducto = false where codigoProducto = 'p28'

Actualizar el stock de un
producto específico

```
UPDATE Stock SET cantidadDisponible = 50 WHERE idProductoFK = 'P2';
```

✓ 45 09:56:23 UPDATE Stock SET cantidadDisponible = 50 WHERE idProductoFK = 'P2'



ELIMINACIONES

```
delete from Producto  
where codigoProducto = 'P59';
```

✓ 47 10:02:50 delete from Producto where codigoProducto = 'P59'

Eliminar un producto
específico



CONSULTAS MULTITABLA

```
/** Consultas multitablas**/  
-- facturas con información del cliente y el usuario que realizó la venta  
select codigofactura, totalfactura, nombrecliente, nombreusuario from facturaventa join venta on codigoventafk = codigoventa  
join cliente on venta.docclientefk = cliente.doccliente join usuario on venta.docusuariofk = usuario.docusuario;
```

✓ 48 10:06:18 select codigofactura, totalfactura, nombrecliente, nombreusuario from facturaventa join venta on codigoventafk = codigoventa join cliente on venta.doc...

	codigofactura	totalfactura	nombrecliente	nombreusuario
▶	1	320000	PEDRO	Clara Lucía
	56	1500000	PEDRO	Clara Lucía
	2	45000	YEISON	Antonio José
	57	42000	YEISON	Antonio José
	3	1500000	YEISON	Antonio José
	59	1550000	YEISON	Antonio José
	4	275000	NELFY	Martín Felipe
	60	410000	NELFY	Martín Felipe

CONSULTAS MULTITABLA

```
-- detalles de venta incluyendo producto, cantidad y subtotal
select codigoventa, nombreproducto, cantidadprod, preciounitario, (cantidadprod * preciounitario) as 'subtotal'
from venta join facturaventa on venta.codigoventa = facturaventa.codigoventafk join detallesfactura on facturaventa.codigofactura = detallesfactura.codigofacturafk
join producto on detallesfactura.codigoproductofk = producto.codigoproducto;
```

✓

50 10:28:15

select codigoventa, nombreproducto, cantidadprod, preciounitario, (cantidadprod * preciounitario) as 'subtotal' from venta join facturaventa on venta.co...

	codigoventa	nombreproducto	cantidadprod	preciounitario	subtotal
▶	1	THOM GAFAS DE SOL迪特眼镜	5	3600	18000
	2	BOLSAS RAYBAN	12	3600	43200
	3	RELOJ CASIO MTP-V002	8	3600	28800
	4	RELOJ TIMEX WEEKENDER	17	3600	61200
	5	RELOJ SEIKO 5 SPORTS	24	3600	86400
	6	CAJA RELOJ LUJO	9	3600	32400
	7	CORREA CUERO NEGRO	15	4120	61800
	8	CORREA SILICONA AZUL	33	2780	91740
	9	RELOJ CITIZEN ECO-DRIVE	11	3490	38390

SUB-CONSULTAS

```
-- productos que nunca han sido vendidos  
select * from producto  
) where codigoproducto not in (  
    select codigoproductofk from detallesfactura
```

✓ 49 10:23:18 select * from producto where codigoproducto not in (select codigoproductofk from detallesfactura) LIMIT 0, 1000

	codigoProducto	nombreProducto	precioProducto	descripcionProducto	estadoProducto
✖	NULL	NULL	NULL	NULL	NULL

PROCEDIMIENTOS

```
-- Detalle de factura completa por código
delimiter //
create procedure detallefacturacompleta(in p_codigofactura int)
begin
    select codigofactura, totalfactura, fechaemision, estadopago,nombrecliente, nombreproducto, cantidadprod, preciounitario
    from facturaventa join venta on facturaventa.codigoventafk = venta.codigoventa
    join cliente on venta.docclientefk = cliente.doccliente
    join detallesfactura on facturaventa.codigofactura = detallesfactura.codigofacturafk
    join producto on detallesfactura.codigoproductofk = producto.codigoproducto
    where facturaventa.codigofactura = p_codigofactura;
end //
delimiter ;
select*from facturaventa;
call detallefacturacompleta(2);
```

	codigofactura	totalfactura	fechaemision	estadopago	nombrediente	nombreproducto	cantidadprod	preciounitario
▶	2	45000	2025-02-25	Pendiente	YEISON	BOLSAS RAYBAN	12	3600

PROCEDIMIENTOS

```
-- CAMBIAR DISPONIBILIDAD DE UN PRODUCTO
DELIMITER //
CREATE PROCEDURE cambiarDisponibilidadProducto(in p_codigoProducto varchar(20), p_estadoNuevo bool)
BEGIN

    update producto
    set estadoProducto = p_estadoNuevo
    where codigoProducto = p_codigoProducto;
END //
DELIMITER ;

call cambiarDisponibilidadProducto('P10', true);
```

✓	45	10:38:15	CREATE PROCEDURE cambiarDisponibilidadProducto(in p_codigoProducto varchar(20), p_estadoNuevo bool) BEGIN update producto set estadoPr...	0 row(s) affected
✓	46	10:38:18	call cambiarDisponibilidadProducto('P10', true)	0 row(s) affected

VISTAS

Mostar el valor total de las compras de un cliente

```
create view vista_totalcomprascliente as
select doccliente, nombrecliente, apellidocliente, sum(totalfactura) as 'total comprado'
from cliente join venta on cliente.doccliente = venta.docclientefk
join facturaventa on venta.codigoventa = facturaventa.codigoventafk
group by cliente.doccliente;
SELECT * FROM Vista_TotalComprasCliente;
```

✓ 56 10:35:54 create view vista_totalcomprascliente as select doccliente, nombrecliente, apellidocliente, sum(totalfactura) as 'total comprado' from cliente join venta o...

	doccliente	nombrecliente	apellidocliente	total comprado
▶	10280550	JOSE GERARDO	CARMONA	138000
	10289718	CARLOS ALBERTO	CARMONA	790000
	15325489	PEDRO	DIAZ TORREZ	1820000
	1002300530	YEISON	ESTUPIÑAN TORRES	3694000
	1002599645	NELFY	RODRIGUEZ	685000
	1010171307	LUIGY	HERNANDEZ	709000
	1010185799	JHONATAN	AVILA	2090000
	1010187182	YISETH	MAHECHA	138000

VISTAS

```
-- Productos con stock disponible
create view vista_productosconstock as
select codigoproducto, nombreproducto, cantidaddisponible from producto
join stock on producto.codigoproducto = stock.idproductofk where stock.cantidaddisponible > 0;
select * from vista_productosconstock;
```

✓ 58 10:38:56 create view vista_productosconstock as select codigoproducto, nombreproducto, cantidaddisponible from producto join stock on producto.codigoproducto = stock.idproductofk where stock.cantidaddisponible > 0;

	codigoproducto	nombreproducto	cantidaddisponible
▶	P1	THOM GAFAS DE SOL迪特眼镜	50
	P2	BOLSAS RAYBAN	50
	P3	RELOJ CASIO MTP-V002	30
	P4	RELOJ TIMEX WEEKENDER	25
	P5	RELOJ SEIKO 5 SPORTS	20
	P7	CORREA CUERO NEGRO	80
	P8	CORREA SILICONA AZUL	90
	P9	RELOJ CITIZEN ECO-DRIVE	15
	P10	RELOJ ORIENT CLASSIC	18

DISPARADORES

```
-- Disminuir stock cuando se inserta un detalle de factura
delimiter //
create trigger trg_disminuir_stock
after insert on detallesfactura
for each row
begin
    update stock
    set cantidaddisponible = cantidaddisponible - new.cantidadprod
    where idproductofk = new.codigoproductofk;
end //
delimiter ;
select*from detallesfactura;
insert into detallesfactura (codigofacturafk, codigoproductofk, cantidadprod, preciounitario)
values (1, 'P1', 2, 10.00);
```

```
create trigger trg_disminuir_stock after insert on detallesfactura for each row begin    update stock    set cantidaddisponible = cant
```

62 10:41:16 insert into detallesfactura (codigofacturafk, codigoproductofk, cantidadprod, preciounitario) values (1, 'P1', 2, 10.00)

	codigoStock	idProductoFK	cantidadDisponible	estadoStock	fEntradaProducto	fSalidaProducto
▶	1	P1	48	1	2025-04-01	2025-12-31
●	NULL	NULL	NULL	NULL	NULL	NULL

DISPARADORES

```
-- desactivar automáticamente una venta entregada
delimiter //
create trigger trg_autodesactivar_venta
before update on venta
for each row
) begin
)   if new.freparto < curdate() and new.estadoventa = true then
        set new.estadoventa = false;
-   end if;
- end //
delimiter ;
```

✓	68	10:50:55	create trigger trg_autodesactivar_venta before update on venta for each row begin if new.freparto < curdate() and new.estadoventa = true ther
✓	69	10:51:30	update venta set estadoventa = true where codigoventa = 3

```
select estadoventa from venta where codigoventa = 3;
```

	estadoventa
▶	0



Thank You