**Qhantuperú**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

SCRIPT COMPLETO: QhatuPeru - Estructura + Datos (completo)

- Crea la base de datos QhatuPeru

- Crea 9 tablas: Roles, Usuarios, Clientes, Proveedores,

Categorias, Productos, Ventas, DetalleVentas, Pagos

- Inserta datos: Roles (5), Proveedores (20), Categorías (8),

100 Usuarios, 100 Clientes, 100 Productos, 100 Ventas,

DetalleVentas (varios items por venta), 100 Pagos.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- 0) (Opcional) Si ejecutas en SSMS/Azure Data Studio:

IF EXISTS(SELECT \* FROM sys.databases WHERE name = 'QhatuPeru')

BEGIN

DROP DATABASE QhatuPeru;

END

GO

CREATE DATABASE QhatuPeru;

GO

USE QhatuPeru;

GO

-- 1) Crear tablas (orden pensado para respetar FK)

CREATE TABLE Roles (

IdRol INT IDENTITY(1,1) PRIMARY KEY,

NombreRol NVARCHAR(50) NOT NULL,

Descripcion NVARCHAR(200)

);

CREATE TABLE Usuarios (

IdUsuario INT IDENTITY(1,1) PRIMARY KEY,

Nombre NVARCHAR(100) NOT NULL,

Apellido NVARCHAR(100),

Correo NVARCHAR(150) UNIQUE,

Contraseña NVARCHAR(200),

IdRol INT NULL,

FechaRegistro DATETIME DEFAULT GETDATE(),

CONSTRAINT FK\_Usuarios\_Roles FOREIGN KEY (IdRol) REFERENCES Roles(IdRol)

);

CREATE TABLE Clientes (

IdCliente INT IDENTITY(1,1) PRIMARY KEY,

Nombre NVARCHAR(100) NOT NULL,

Apellido NVARCHAR(100),

DNI NVARCHAR(15),

Telefono NVARCHAR(20),

Direccion NVARCHAR(300),

Correo NVARCHAR(150)

);

CREATE TABLE Proveedores (

IdProveedor INT IDENTITY(1,1) PRIMARY KEY,

RazonSocial NVARCHAR(150),

RUC NVARCHAR(20),

Telefono NVARCHAR(20),

Direccion NVARCHAR(300),

Correo NVARCHAR(150)

);

CREATE TABLE Categorias (

IdCategoria INT IDENTITY(1,1) PRIMARY KEY,

Nombre NVARCHAR(100),

Descripcion NVARCHAR(200)

);

CREATE TABLE Productos (

IdProducto INT IDENTITY(1,1) PRIMARY KEY,

Nombre NVARCHAR(200),

Precio DECIMAL(10,2),

Stock INT,

IdCategoria INT NULL,

IdProveedor INT NULL,

CONSTRAINT FK\_Productos\_Categorias FOREIGN KEY (IdCategoria) REFERENCES Categorias(IdCategoria),

CONSTRAINT FK\_Productos\_Proveedores FOREIGN KEY (IdProveedor) REFERENCES Proveedores(IdProveedor)

);

CREATE TABLE Ventas (

IdVenta INT IDENTITY(1,1) PRIMARY KEY,

IdCliente INT NULL,

IdUsuario INT NULL,

FechaVenta DATETIME DEFAULT GETDATE(),

Total DECIMAL(12,2),

CONSTRAINT FK\_Ventas\_Clientes FOREIGN KEY (IdCliente) REFERENCES Clientes(IdCliente),

CONSTRAINT FK\_Ventas\_Usuarios FOREIGN KEY (IdUsuario) REFERENCES Usuarios(IdUsuario)

);

CREATE TABLE DetalleVentas (

IdDetalle INT IDENTITY(1,1) PRIMARY KEY,

IdVenta INT NOT NULL,

IdProducto INT NOT NULL,

Cantidad INT,

Subtotal DECIMAL(12,2),

CONSTRAINT FK\_DetalleVentas\_Ventas FOREIGN KEY (IdVenta) REFERENCES Ventas(IdVenta),

CONSTRAINT FK\_DetalleVentas\_Productos FOREIGN KEY (IdProducto) REFERENCES Productos(IdProducto)

);

CREATE TABLE Pagos (

IdPago INT IDENTITY(1,1) PRIMARY KEY,

IdVenta INT NOT NULL,

MetodoPago NVARCHAR(50),

Monto DECIMAL(12,2),

FechaPago DATETIME DEFAULT GETDATE(),

CONSTRAINT FK\_Pagos\_Ventas FOREIGN KEY (IdVenta) REFERENCES Ventas(IdVenta)

);

GO

-- 2) Insertar Roles (necesario antes de Usuarios)

INSERT INTO Roles (NombreRol, Descripcion)

VALUES

('Administrador', 'Control total del sistema'),

('Vendedor', 'Gestiona ventas y clientes'),

('Almacenero', 'Gestiona inventario y productos'),

('Contabilidad', 'Reportes y pagos'),

('Soporte', 'Atiende incidencias y usuarios');

GO

-- 3) Insertar Proveedores (20)

INSERT INTO Proveedores (RazonSocial, RUC, Telefono, Direccion, Correo)

VALUES

('Distribuciones Andinas SAC', '20100011112', '987654321', 'Av. Industrial 123, Lima', 'andinas@proveedor.com'),

('Importaciones del Sur SA', '20100011113', '987654322', 'Jr. Comercio 45, Arequipa', 'importsur@proveedor.com'),

('Alimentos Peru S.A.', '20100011114', '987654323', 'C. Central 67, Cusco', 'alimentos@proveedor.com'),

('Bebes y Mas EIRL', '20100011115', '987654324', 'Av. La Marina 789, Lima', 'bebes@proveedor.com'),

('Logistica Norte S.A.', '20100011116', '987654325', 'MZ B Lt 3, Piura', 'lognorte@proveedor.com'),

('Productos Amazonicos S.A.', '20100011117', '987654326', 'Iquitos 455', 'amazonicos@proveedor.com'),

('La Cafeteria S.R.L.', '20100011118', '987654327', 'Cuzco 98', 'cafe@proveedor.com'),

('Texto & Co', '20100011119', '987654328', 'Jr. Letras 10', 'texto@proveedor.com'),

('Farmacia Central', '20100011120', '987654329', 'Av. Salud 22', 'farmacen@proveedor.com'),

('Electro Sur', '20100011121', '987654330', 'Av. Tecnología 77', 'electrosur@proveedor.com'),

('La Granja Peruana', '20100011122', '987654331', 'Sector Agro 12', 'granja@proveedor.com'),

('Lácteos Andinos', '20100011123', '987654332', 'Av. Leche 9', 'lacteos@proveedor.com'),

('Panificadora Lima', '20100011124', '987654333', 'C. Pan 55', 'pan@proveedor.com'),

('Hogar y Cocina S.A.', '20100011125', '987654334', 'Jr. Hogar 4', 'hogar@proveedor.com'),

('Frutas del Valle', '20100011126', '987654335', 'Av. Fruta 80', 'frutas@proveedor.com'),

('Materiales Casa', '20100011127', '987654336', 'C. Materiales 2', 'material@proveedor.com'),

('Tinta y Papel', '20100011128', '987654337', 'Jr. Oficina 3', 'tintapapel@proveedor.com'),

('Bazar Express', '20100011129', '987654338', 'Plaza Bazar 1', 'bazar@proveedor.com'),

('Tecnología Total', '20100011130', '987654339', 'Av. Tech 200', 'tecnologia@proveedor.com'),

('Super Mayorista', '20100011131', '987654340', 'Av. Comercio 500', 'mayorista@proveedor.com');

GO

-- 4) Insertar Categorías (8)

INSERT INTO Categorias (Nombre, Descripcion)

VALUES

('Abarrotes', 'Alimentos y productos básicos'),

('Bebidas', 'Bebidas alcohólicas y no alcohólicas'),

('Limpieza', 'Productos de limpieza y hogar'),

('Lacteos', 'Leche, yogurt, quesos'),

('Snack', 'Snacks y galletas'),

('Electrónica', 'Pequeños aparatos y accesorios'),

('Ropa', 'Prendas y accesorios'),

('Hogar', 'Utensilios y menaje');

GO

-- 5) Insertar 100 Usuarios con nombres/peruanos realistas (correos unicos)

;WITH nums AS (

SELECT 1 AS n

UNION ALL

SELECT n + 1 FROM nums WHERE n < 100

)

INSERT INTO Usuarios (Nombre, Apellido, Correo, Contraseña, IdRol, FechaRegistro)

SELECT

FN.Nombre,

LN.Apellido,

LOWER(CONCAT(FN.Nombre, '.', LN.Apellido, n, '@qhatu.com')),

CONCAT('Qhatu', RIGHT('000' + CAST(n AS VARCHAR(3)), 3), '!'),

((n - 1) % 5) + 1, -- asigna roles 1..5 en ciclo

DATEADD(DAY, - (ABS(CHECKSUM(NEWID())) % 365), GETDATE())

FROM nums

CROSS APPLY (

SELECT TOP(1) Nombre FROM (VALUES

('Juan'),('José'),('Luis'),('Carlos'),('Miguel'),

('David'),('Jorge'),('Diego'),('Andrés'),('Fernando'),

('Marco'),('Raúl'),('Pablo'),('Sergio'),('Alberto'),

('Ricardo'),('Victor'),('Eduardo'),('Daniel'),('Rafael'),

('Sebastián'),('Hugo'),('Emilio'),('Esteban'),('Iván'),

('Marcos'),('Óscar'),('Gonzalo'),('Agustín'),('Nelson')

) AS F(Nombre)

ORDER BY NEWID()

) FN

CROSS APPLY (

SELECT TOP(1) Apellido FROM (VALUES

('Pérez'),('García'),('López'),('Ramírez'),('Torres'),

('Quispe'),('Huamán'),('Flores'),('Salazar'),('Rojas'),

('Vargas'),('Castillo'),('Mendoza'),('Soto'),('Delgado'),

('Herrera'),('Cruz'),('Ortega'),('Reyes'),('Ibáñez'),

('Silva'),('Paredes'),('Vega'),('Morales'),('Navarro'),

('Benítez'),('Gil'),('Ruiz'),('Campos'),('Domínguez')

) AS L(Apellido)

ORDER BY NEWID()

) LN

OPTION (MAXRECURSION 0);

GO

-- 6) Insertar 100 Clientes (nombres, DNI, teléfono, dirección, correo)

;WITH nums AS (

SELECT 1 AS n

UNION ALL

SELECT n + 1 FROM nums WHERE n < 100

)

INSERT INTO Clientes (Nombre, Apellido, DNI, Telefono, Direccion, Correo)

SELECT

FN.Nombre,

LN.Apellido,

RIGHT(CONCAT('', 10000000 + (ABS(CHECKSUM(NEWID())) % 90000000)), 8),

CONCAT('9', CAST(10000000 + (ABS(CHECKSUM(NEWID())) % 90000000) AS VARCHAR(8))),

CONCAT('Av. ', ST.Calle, ' N° ', CAST(1 + (ABS(CHECKSUM(NEWID())) % 999) AS VARCHAR(4)), ', ', CT.Ciudad),

LOWER(CONCAT(FN.Nombre, '.', LN.Apellido, n, '@gmail.com'))

FROM nums

CROSS APPLY (

SELECT TOP(1) Nombre FROM (VALUES

('Juan'),('José'),('Luis'),('Carlos'),('Miguel'),

('Paola'),('María'),('Lucia'),('Ana'),('Rosa'),

('Claudia'),('Elena'),('Carla'),('Andrea'),('Karla')

) AS F(Nombre)

ORDER BY NEWID()

) FN

CROSS APPLY (

SELECT TOP(1) Apellido FROM (VALUES

('Pérez'),('García'),('López'),('Ramírez'),('Torres'),

('Quispe'),('Huamán'),('Flores'),('Salazar'),('Rojas'),

('Vargas'),('Castillo'),('Mendoza'),('Soto'),('Delgado')

) AS L(Apellido)

ORDER BY NEWID()

) LN

CROSS APPLY (

SELECT TOP(1) Calle FROM (VALUES

('Arequipa'),('Tacna'),('Cusco'),('Lima'),('Cajamarca'),

('Ayacucho'),('Ancash'),('Moquegua'),('Piura'),('Puno')

) AS S(Calle)

ORDER BY NEWID()

) ST

CROSS APPLY (

SELECT TOP(1) Ciudad FROM (VALUES

('Lima'),('Cusco'),('Arequipa'),('Trujillo'),('Chiclayo'),

('Puno'),('Iquitos'),('Huancayo'),('Piura'),('Tacna')

) AS C(Ciudad)

ORDER BY NEWID()

) CT

OPTION (MAXRECURSION 0);

GO

-- 7) Insertar 100 Productos (asociados a categorías y proveedores)

;WITH nums AS (

SELECT 1 AS n

UNION ALL

SELECT n + 1 FROM nums WHERE n < 100

)

INSERT INTO Productos (Nombre, Precio, Stock, IdCategoria, IdProveedor)

SELECT

CONCAT(ProdBase.NombreBase, ' ', n),

CAST(ROUND((ABS(CHECKSUM(NEWID())) % 2000) / 10.0 + 5.0, 2) AS DECIMAL(10,2)),

(ABS(CHECKSUM(NEWID())) % 200) + 1,

((ABS(CHECKSUM(NEWID())) % (SELECT COUNT(\*) FROM Categorias)) + 1),

((ABS(CHECKSUM(NEWID())) % (SELECT COUNT(\*) FROM Proveedores)) + 1)

FROM nums

CROSS APPLY (

SELECT TOP(1) NombreBase FROM (VALUES

('Arroz'),('Azúcar'),('Aceite'),('Galleta'),('Yogurt'),

('Jugo'),('Café'),('Té'),('Jabón'),('Detergente'),

('Leche'),('Queso'),('Pan'),('Camisa'),('Pantalón'),

('Bufanda'),('Linterna'),('Cargador'),('Auriculares'),('Sartén')

) AS ProdBase(NombreBase)

ORDER BY NEWID()

) ProdBase

OPTION (MAXRECURSION 0);

GO

-- 8) Insertar 100 Ventas (cliente y usuario aleatorio)

;WITH nums AS (

SELECT 1 AS n

UNION ALL

SELECT n + 1 FROM nums WHERE n < 100

)

INSERT INTO Ventas (IdCliente, IdUsuario, FechaVenta, Total)

SELECT

((ABS(CHECKSUM(NEWID())) % (SELECT COUNT(\*) FROM Clientes)) + 1),

((ABS(CHECKSUM(NEWID())) % (SELECT COUNT(\*) FROM Usuarios)) + 1),

DATEADD(DAY, - (ABS(CHECKSUM(NEWID())) % 365), GETDATE()),

0.00 -- total se actualizará luego sumando detalle

FROM nums

OPTION (MAXRECURSION 0);

GO

-- 9) Generar DetalleVentas para cada venta: entre 1 y 5 items aleatorios

DECLARE @v INT = 1;

DECLARE @maxVentas INT = (SELECT COUNT(\*) FROM Ventas);

WHILE @v <= @maxVentas

BEGIN

DECLARE @items INT = (ABS(CHECKSUM(NEWID())) % 5) + 1; -- 1..5

DECLARE @j INT = 1;

WHILE @j <= @items

BEGIN

DECLARE @prodId INT = ((ABS(CHECKSUM(NEWID())) % (SELECT COUNT(\*) FROM Productos)) + 1);

DECLARE @cantidad INT = ((ABS(CHECKSUM(NEWID())) % 5) + 1);

DECLARE @precio DECIMAL(10,2) = (SELECT Precio FROM Productos WHERE IdProducto = @prodId);

DECLARE @subtotal DECIMAL(12,2) = ROUND(@precio \* @cantidad,2);

INSERT INTO DetalleVentas (IdVenta, IdProducto, Cantidad, Subtotal)

VALUES (@v, @prodId, @cantidad, @subtotal);

SET @j = @j + 1;

END

-- Actualizar total de la venta sumando sus detalle

UPDATE Ventas

SET Total = ISNULL((SELECT SUM(Subtotal) FROM DetalleVentas WHERE IdVenta = @v), 0)

WHERE IdVenta = @v;

SET @v = @v + 1;

END

GO

-- 10) Insertar Pagos: para simplificar, insertar 1 pago por venta (mismo monto del total)

INSERT INTO Pagos (IdVenta, MetodoPago, Monto, FechaPago)

SELECT IdVenta,

CASE WHEN (ABS(CHECKSUM(NEWID())) % 3) = 0 THEN 'Efectivo'

WHEN (ABS(CHECKSUM(NEWID())) % 3) = 1 THEN 'Tarjeta'

ELSE 'Transferencia' END AS MetodoPago,

Total,

DATEADD(HOUR, (ABS(CHECKSUM(NEWID())) % 24), FechaVenta)

FROM Ventas;

GO

-- 11) Verificaciones rápidas

SELECT 'Roles' as Tabla, COUNT(\*) as Total FROM Roles;

SELECT 'Usuarios' as Tabla, COUNT(\*) as Total FROM Usuarios;

SELECT 'Clientes' as Tabla, COUNT(\*) as Total FROM Clientes;

SELECT 'Proveedores' as Tabla, COUNT(\*) as Total FROM Proveedores;

SELECT 'Categorias' as Tabla, COUNT(\*) as Total FROM Categorias;

SELECT 'Productos' as Tabla, COUNT(\*) as Total FROM Productos;

SELECT 'Ventas' as Tabla, COUNT(\*) as Total FROM Ventas;

SELECT 'DetalleVentas' as Tabla, COUNT(\*) as Total FROM DetalleVentas;

SELECT 'Pagos' as Tabla, COUNT(\*) as Total FROM Pagos;

GO

-- 12) Ejemplos (ver primeras filas)

SELECT TOP 10 \* FROM Usuarios ORDER BY IdUsuario;

SELECT TOP 10 \* FROM Clientes ORDER BY IdCliente;

SELECT TOP 10 \* FROM Productos ORDER BY IdProducto;

SELECT TOP 10 \* FROM Ventas ORDER BY IdVenta;

SELECT TOP 20 \* FROM DetalleVentas ORDER BY IdDetalle;

GO