Santiago Ramirez Arenas – Ejercicio C

**Reto BD Tienda Don Pepe**

**Se pide:**

* Indicar que ejercicio fue asignado
* Realizar el modelo E-R
* Realizar el modelo relacional
* Normalizar correctamente
* Escribir con sentencias SQL toda la definición de la base de datos.
* Escribir consultas que me permitan ver la información de cada tabla o de varias tablas (10).
* Generar de 4 a 6 vistas donde se evidencie lo más importante de cada ejercicio (haga una selección muy responsable de la información realmente importante según el contexto).
* Generar al menos 4 procedimientos almacenados.
* Generar al menos 4 triggers
* Poblar la base de datos (50 registros por tabla) utilizando una conexión desde Java.
* Al terminar el ejercicio responda ¿Está conforme con el resultado obtenido según el contexto o cree que hubiera obtenido un mejor resultado con una base de datos no relacional?
* documente muy bien su proceso (paso a paso) en un archivo PDF escriba todas las aclaraciones o especificaciones necesarias para realizar el ejercicio.

**CONTEXTO**

**Tienda Virtual Don pepe (Ejercicio C)**

Don pepe quiere que sus clientes puedan realizar compras desde sus casas. El junto a su esposa tienen una cantidad domiciliarios conocidos que se encargan de llevar los pedidos a los clientes.

A continuación, se muestra la conversación que se tuvo con don pepe:

* ¡Veee mijo! yo quiero que más gente me compre los producticos, cuando llega un vecino nuevo a la cuadra yo lo apunto en un cuadernito. ¿Entiendo don pepe, y no le gustaría que le comprarán por internet?
* Ehhh mijo pues no es mala idea y que hago con mi clientela?
* Pues don pepe hacemos un video tutorial para usar la aplicación, y le pedimos una información a sus clientes indicando sus datos personales (ID, cedula, Nombre, Dirección, Teléfono, email y password) a través de un formulario de registro. Una vez registrado podrá acceder a la realización de pedidos con su email y su password.
* ¡ eeeee yo no te creo! Asi de fácil? ¿Como motilando calvos?
* Don pepe ojala fuera así de sencillo déjeme le cuento mejor, Los productos que oferta el supermercado deben estar divididos en diversas categorías. Los datos necesarios para cada categoría son: nombre de la categoría, condiciones de almacenamiento (frío, congelado, seco) y observaciones. También debemos detallar la información de los productos (nombre, marca, origen, dimensiones (volumen y peso), una fotografía, la categoría y unidades disponibles).¡no mijo eso me va salir muy caro con tanto detalle!
* don pepe todo lo contrario va aumentar mucho sus ganancias Espéreme le cuento algo mas, la aplicación permitirá visualizar un listado de productos ordenado por categoría, permitiendo seleccionar los productos que desee comprar mediante una caja de texto donde se indicará el número de unidades seleccionadas. La aplicación llevará la cuenta (cesta de la compra) de los productos que el cliente ha ido seleccionando. La aplicación permitirá también efectuar un pedido con todos los productos que lleve almacenados en su cesta de la compra. Los datos del pedido son: código del pedido, fecha del pedido, cliente, dirección de entrega, productos pedidos, importe total del pedido y datos de pago (número de tarjeta y fecha de caducidad)®.

Para poder generar un pedido se deberán dar dos situaciones:

* El cliente deberá pertenecer a una zona (Código Postal) donde existan domiciliarios. Un domiciliario se identifica mediante un nombre, número de matrícula de la furgoneta y zona donde reparte.
* Debe haber unidades suficientes por cada producto para satisfacer las demandas de cada pedido.

Una vez generado el pedido se mostrará al usuario una página con los datos de su pedido, se restarán del stock las unidades pedidas y se emitirá una nota de entrega a los responsables de almacén para que sirvan ese pedido.

**Paso a paso de la solución**

**Paso 1:** Lectura del contexto y reconocimiento de las entidades y sus atributos según la historia de usuario narrada

**Cliente**

Inicialmente se identifica la entidad **Cliente** cuyos atributos serán**:**

* Un **identificador único** de cada cliente en el momento en el que se registra
* **Nombre** que se va a componer de un **Nombre** y un **Apellido**
* **Cedula Cliente**
* **Dirección** que se va a componer de un **barrio y un numero de casa**
* Un **teléfono** como atributo multivaluado por si se desean ingresar múltiples teléfonos de un mismo usuario
* Por último como credenciales de acceso un **Email** y **Password**

**Domiciliario**

Se identificó la entidad **domiciliario** al cual le será asignado un pedido según el código postal al que pertenece, se reconocieron los siguientes atributos

* **Nombre** que estará compuesto por **Nombre y Apellido ,**
* **Identificador único** que le asignará la aplicación al momento de su registro
* Por último, un **numero de matricula**

**Proveedor**

Se identificó la entidad **proveedor** quien será la encargada de proveer los productos de la **Tienda Don Pepe,** se reconocieron los siguientes atributos**.**

* **Identificador único de proveedor**
* **Nombre del proveedor** (bien puede ser el nombre de la empresa o razón social)
* **Teléfono proveedor**

**Producto**

Se identificó la entidad **producto,** la cual tendrá la información de los productos que ofrece la **tienda Don Pepe,** se reconocieron los siguientes atributos**.**

* **Fotografía** para identificar el producto visualmente
* **Unidades disponibles:** Esta es la cantidad de unidades que se encuentra en el stock
* **Observaciones**: Para dar información que se considere relevante como la Fecha de caducidad
* **Dimensiones** que se compone del **volumen** y el **peso** del producto
* **Marca:** La marca del producto en cuestión
* **Nombre**: El nombre exacto del producto
* **Origen**: el lugar de procedencia, es decir donde fue fabricado

**Categoría del producto**

Se identificó la entidad **categoría producto,** la cual contiene **el tipo de almacenamiento,** y una **descripción del almacenamiento,** se reconocieron los siguientes atributos**.**

* **Tipo de almacenamiento como una conexión a la tabla Tipo Almacenamiento:** este contiene la llave foránea del tipo de almacenamiento, bien sea Frio, a temperatura ambiente, seco, congelado , etc.
* **Descripción:** La descripción más detallada de cómo debe ser su almacenamiento
* **Identificador único categoría producto**

**Tipo de almacenamiento producto**

* **Id Almacenamiento:** Este es el Id del tipo de almacenamiento que requiere el producto
* **Tipo de almacenamiento:** Este es el tipo de almacenamiento

Aclaración: Esto se hace por si se desea agregar un tipo de almacenamiento más adelante y pensando también en el momento en que se van a agregar el tipo de almacenamiento

**Pedido**

Se identificó la entidad **Pedido,** la cual va a contener información importante de cada compra, la cual contiene los siguientes atributos.

* **Fecha de pedido:** Fecha en la cual se realizan los pedidos
* **Importe total pedido:** Este es el valor total del pedido
* **Productos pedidos:** Estos son los productos pedidos
* **Datos de pago:** Está compuesto por **Fecha de caducidad y Numero de tarjeta**
* **Código Postal Cliente:** Código postal del cliente para realizar la verificación
* **Código Postal Domiciliario:** Código postal del domiciliario para realizar la verificación
* **Verificación Realización Pedido:** Este atributo tiene como valor un True o un False como resultado de una comparación entre El Código Postal Cliente y Código Postal Domiciliario, si son iguales retorna un True y se realiza el pedido y si retorna un False se cancela el pedido. (Esta lógica no fue implementada).
* **Dirección del pedido**

**Paso 2:** Relaciones entre entidades

Ahora realizamos las relaciones entre las entidades teniendo en cuenta el orden de los procedimientos.

* Un Proveedor **proveer** muchos productos, mientras que un producto puede ser provisto por un solo proveedor. Relación (1,N).
* Un producto **tiene** una categoría, mientras que una categoría puede tener muchos productos. Relación (1,N).
* Un tipo de almacenamiento puede **tener** muchas categorías del Producto, mientras que una categoría del producto puede tener un solo Tipo de almacenamiento. Relación (1,N).
* Un Producto puede ser **contenido** en muchos pedidos, mientras que un pedido puede contener muchos productos. Relación (1,N).
* Un cliente puede **generar** muchos pedidos, mientras que un pedido puede ser generado por un único cliente. (Si tenemos en cuenta que un pedido le va a pertenecer a un único cliente por cada proceso). Relación (1,N).
* Un domiciliario puede ser **asignado** a un muchos pedidos , mientras que un pedido solo se puede asignar a un único domiciliario (1,N)

**Imagen 1:** Modelo entidad relación de la tienda Don Pepe

Diagrama, Esquemático

Descripción generada automáticamente

**Paso 3**: Creación del modelo relacional

* Ahora se crean las tablas en el modelo relacional, y se asignan las llaves foráneas en las tablas que tienen una cardinalidad (1,N), asignando la llave primaria de la entidad cuyo Cardinalidad es 1, como llave foránea a la tabla cuya cardinalidad es N.
* Después se crea la tabla para la relación (N,M) que va a contener las llaves primarias de las dos entidades que se relacionan , en este caso especifico de las tablas Producto y pedido

**Imagen 2:** Modelo Relacional de la tienda Don Pepe

Diagrama

Descripción generada automáticamente

**Paso 4**

Para este paso se crea el código SQL que genere la base de datos anterior

**CREATE** **DATABASE** IF **NOT** **EXISTS** TiendaDonPepe;

USE TiendaDonPepe;

-- Tabla cliente

**CREATE** **TABLE** cliente (

IdCliente VARCHAR(**30**) **PRIMARY** **KEY**,

Password VARCHAR(**30**),

Email VARCHAR(**30**),

NombreCliente VARCHAR(**30**),

ApellidoCliente VARCHAR(**30**),

ClienteCedula VARCHAR(**30**),

Barrio VARCHAR(**30**),

NumeroCasa VARCHAR(**30**)

);

-- Tabla TelefonoClIENTE

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

IdTelefono INT **PRIMARY** **KEY**,

FKIdCliente VARCHAR(**30**),

NumeroTelefono VARCHAR(**30**),

**CONSTRAINT** FK\_TelefonoCliente\_Cliente **FOREIGN** **KEY** (FKIdCliente)

**REFERENCES** Cliente (IdCliente)

);

-- Tabla Domiciliario

**CREATE** **TABLE** Domiciliario (

IdDomiciliario VARCHAR(**30**) **PRIMARY** **KEY**,

NumeroMatricula VARCHAR(**30**),

NombreDomiciliario VARCHAR(**30**),

ApellidoDomiciliario VARCHAR(**30**)

);

-- Tabla Proveedor

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

IdProveedor VARCHAR(**30**) **PRIMARY** **KEY**,

NombreProveedor VARCHAR(**30**),

TelefonoProveedor VARCHAR(**30**)

);

-- Tabla Tipo Almacenamiento

**CREATE** **TABLE** TipoAlmacenamiento (

IdTipoAlmacenamiento VARCHAR(**30**) **PRIMARY** **KEY**,

TipoDeAlmacenaminento VARCHAR(**30**)

);

-- Tabla Categoria Producto

**CREATE** **TABLE** CategoriaProducto (

IdCategoriaProducto VARCHAR(**30**) **PRIMARY** **KEY**,

Recomendaciones VARCHAR(**30**),

FKIdTipoAlmacenamiento VARCHAR(**30**),

**CONSTRAINT** FKCategoriaProductoTipoAlmacenamiento **FOREIGN** **KEY** (FKIdTipoAlmacenamiento)

**REFERENCES** TipoAlmacenamiento (IdTipoAlmacenamiento)

);

-- Tabla Pedido

**CREATE** **TABLE** Pedido (

IdPedido VARCHAR(**30**) **PRIMARY** **KEY**,

FKIdCliente VARCHAR(**30**),

FKIdDomiciliario VARCHAR(**30**),

FechaPedido VARCHAR(**30**),

ImporteTotalPedido VARCHAR(**30**),

VerificacionRealizacionPedido BOOLEAN,

CodigoPostalCliente VARCHAR(**30**),

CodigoPostalDomiciliario VARCHAR(**30**),

ProductosPedidos VARCHAR(**30**),

**CONSTRAINT** FKPedidoCliente **FOREIGN** **KEY** (FKIdCliente)

**REFERENCES** Cliente (IdCliente),

**CONSTRAINT** FKPedidoDomiciliario **FOREIGN** **KEY** (FKIdDomiciliario)

**REFERENCES** Domiciliario (IdDomiciliario)

);

-- Tabla Producto

**CREATE** **TABLE** Producto (

IdProducto VARCHAR(**30**) **PRIMARY** **KEY**,

FKIdCategoriaProducto VARCHAR(**30**),

FKIdProveedor VARCHAR(**30**),

FKIdPedido VARCHAR(**30**),

Fotografia VARCHAR(**30**),

UnidadesDisponibles INT,

Observaciones VARCHAR(**30**),

Volumen VARCHAR(**30**),

Peso VARCHAR(**30**),

Nombre VARCHAR(**30**),

**CONSTRAINT** FKProductoCategoriaProducto **FOREIGN** **KEY** (FKIdCategoriaProducto)

**REFERENCES** CategoriaProducto (IdCategoriaProducto),

**CONSTRAINT** FKProductoProveedor **FOREIGN** **KEY** (FKIdProveedor)

**REFERENCES** Proveedor (IdProveedor),

**CONSTRAINT** FKProductoPedido **FOREIGN** **KEY** (FKIdPedido)

**REFERENCES** Pedido (IdPedido)

);

-- Tabla Contener

**CREATE** **TABLE** Contener (

FKIdContenerPedidoProducto VARCHAR(**30**),

FKIdContenerProductoPedido VARCHAR(**30**),

**PRIMARY** **KEY** (FKIdContenerPedidoProducto,FKIdContenerProductoPedido),

**foreign** **key**(FKIdContenerPedidoProducto) **references** Pedido(IdPedido),

**foreign** **key**(FKIdContenerProductoPedido) **references** Producto(IdProducto)

);

Aplicando ingeniería inversa tenemos el siguiente diagrama.

**Imagen 3:** Modelo Relacional de la tienda Don Pepe en WorkBench

Diagrama

Descripción generada automáticamente con confianza baja

**Consultas SQL**

Para este ejemplo los datos fueron insertados manualmente

-- Mostrar la cantidad total de unidades disponibles de cada producto:

**SELECT** p.Nombre **AS** NombreProducto, **SUM**(p.UnidadesDisponibles) **AS** TotalUnidades

**FROM** Producto p

**GROUP** **BY** p.Nombre;

**Imagen 3:** Mostrar la cantidad total de unidades disponibles de cada producto:

Tabla

Descripción generada automáticamente

-- Mostrar todos los pedidos realizados por un cliente específico:

**SELECT** p.IdPedido, p.FechaPedido, p.ImporteTotalPedido, d.NombreDomiciliario, p.VerificacionRealizacionPedido

**FROM** Pedido p

**INNER** **JOIN** Domiciliario d **ON** p.FKIdDomiciliario = d.IdDomiciliario

**WHERE** p.FKIdCliente = '1';

Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente**Imagen 4:** Mostrar todos los pedidos realizados por un cliente específico

----- Mostrar el número total de pedidos realizados por cada cliente:

**SELECT** **c**.NombreCliente, **COUNT**(p.IdPedido) **AS** TotalPedidos

**FROM** Cliente **c**

**LEFT** **JOIN** Pedido p **ON** **c**.IdCliente = p.FKIdCliente

**GROUP** **BY** **c**.NombreCliente;

**Imagen 5:** Mostrar el número total de pedidos realizados por cada cliente

Tabla

Descripción generada automáticamente

-- Mostrar la lista de todos los domiciliarios con su número de matrícula y el número de pedidos que han entregado:

**SELECT** d.NombreDomiciliario, d.NumeroMatricula, **COUNT**(p.IdPedido) **AS** TotalPedidos

**FROM** Domiciliario d

**LEFT** **JOIN** Pedido p **ON** d.IdDomiciliario = p.FKIdDomiciliario

**GROUP** **BY** d.NombreDomiciliario, d.NumeroMatricula;

**Imagen 6:** Mostrar la lista de todos los domiciliarios con su número de matrícula y el número de pedidos que han entregado

Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente

-- Mostrar la cantidad total de productos suministrados por cada proveedor:

**SELECT** pr.NombreProveedor, **COUNT**(p.IdProducto) **AS** TotalProductos

**FROM** Proveedor pr

**LEFT** **JOIN** Producto p **ON** pr.IdProveedor = p.FKIdProveedor

**GROUP** **BY** pr.NombreProveedor;

**Imagen 7:** Mostrar la cantidad total de productos suministrados por cada proveedor

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

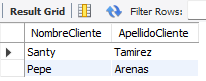
-- Mostrar el nombre y el apellido de los clientes que han realizado algún pedido:

**SELECT** NombreCliente, ApellidoCliente

**FROM** Cliente

**WHERE** IdCliente **IN** (**SELECT** FKIdCliente **FROM** Pedido)

**Imagen 8:** Mostrar el nombre y el apellido de los clientes que han realizado algún pedido:



-- Mostrar la cantidad de pedidos realizados por cada domiciliario, ordenados de forma descendente:

**SELECT** d.NombreDomiciliario, d.ApellidoDomiciliario, **COUNT**(\*) **AS** CantidadPedidos

**FROM** Domiciliario d

**INNER** **JOIN** Pedido pe **ON** d.IdDomiciliario = pe.FKIdDomiciliario

**GROUP** **BY** d.IdDomiciliario

**ORDER** **BY** CantidadPedidos **DESC**

**Imagen 9:** Mostrar la cantidad de pedidos realizados por cada domiciliario, ordenados de forma descendente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

-- Mostrar el nombre y el importe total de los pedidos realizados por el cliente con IdCliente = '1', ordenados por fecha de forma descendente:

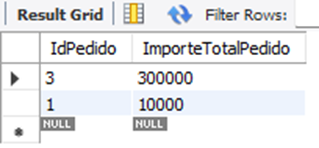
**SELECT** pe.IdPedido, pe.ImporteTotalPedido

**FROM** Pedido pe

**WHERE** pe.FKIdCliente = '1'

**ORDER** **BY** pe.fechaPedido **DESC**;

**Imagen 10:** Mostrar el nombre y el importe total de los pedidos realizados por el cliente con IdCliente = '1', ordenados por fecha de forma descendente



-- Mostrar el nombre y el teléfono del proveedor del producto con IdProducto = '1':

**SELECT** p.NombreProveedor, p.TelefonoProveedor

**FROM** Proveedor p

**INNER** **JOIN** Producto pr **ON** p.IdProveedor = pr.FKIdProveedor

**WHERE** pr.IdProducto = '1'

**Imagen 11:** Mostrar el nombre y el teléfono del proveedor del producto con IdProducto = '1':

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Vistas importantes**

* **Primer vista:** Esta vista es importante para tener un historial de compra que tiene cada cliente y ofrecer posibles rebajas según la inversión de los clientes en el negocio(Según la fidelidad del cliente)

-- -- Vista que muestra Vista que muestra el nombre y el importe total de todos los pedidos realizados por los clientes:

**CREATE** **VIEW** ImporteTotalPedidos **AS**

**SELECT** pe.FKIdCliente, **c**.NombreCliente, **c**.ApellidoCliente, **SUM**(pe.ImporteTotalPedido) **AS** ImporteTotal

**FROM** Pedido pe

**INNER** **JOIN** Cliente **c** **ON** pe.FKIdCliente = **c**.IdCliente

**GROUP** **BY** pe.FKIdCliente, **c**.NombreCliente, **c**.ApellidoCliente

**Imagen 12:** Vista que muestra Vista que muestra el nombre y el importe total de todos los pedidos realizados por los clientes:

Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

* **Segunda vista:** Vista que nos muestra el total de pedidos por el código Postal, esto nos puede ayudar a saber que tantos pedidos realizan los clientes de una determinada área de la ciudad, y dar posibles rebajas o planear mejores rutas de acceso desde la tienda hasta el área en la que se piden los pedidos

-- Vista que nos muestra el total de pedidos por el codigo Postal

**CREATE** **VIEW** PedidosPorCodigoPostal **AS**

**SELECT** pe.CodigoPostalCliente, **COUNT**(pe.IdPedido) **AS** NumeroPedidos

**FROM** Pedido pe

**GROUP** **BY** pe.CodigoPostalCliente;

**Imagen 13:** Vista que muestra el nombre de los productos y el número de pedidos en los que han sido incluidos

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* **Tercera vista: Esta vista nos puede ayudar**

-- Vista que muestra el numero total de pedidos por domiciliario:

**CREATE** **VIEW** PedidosPorDomiciliario **AS**

**SELECT** pe.FKIdDomiciliario, **COUNT**(pe.IdPedido) **AS** NumeroPedidos

**FROM** Pedido pe

**GROUP** **BY** pe.FKIdDomiciliario