

# Proyecto Final

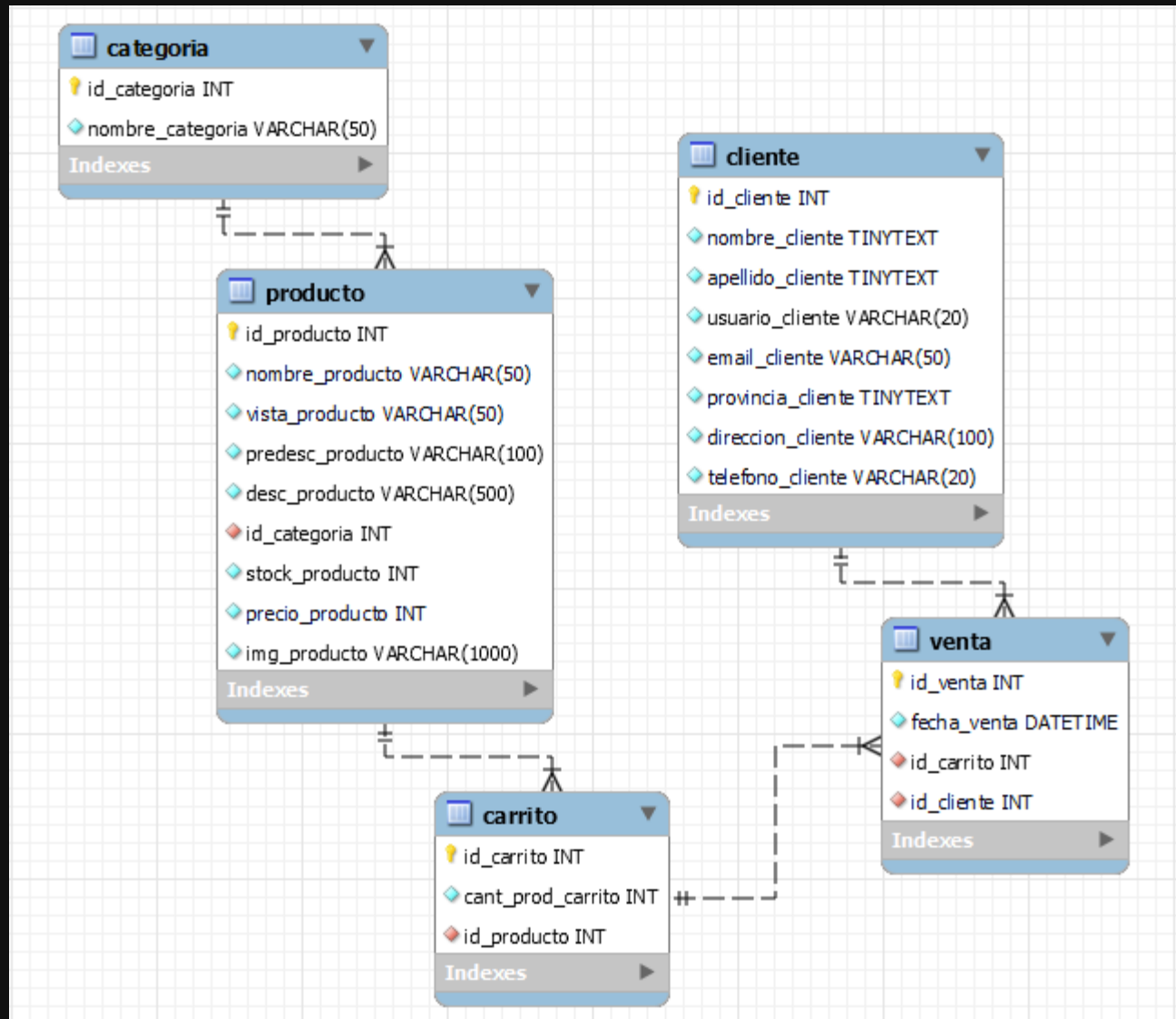
Backup y Restauración – SQL – Comisión 34955

# Temática:

## Administración de inventarios de RedStore (ecommerce)

Para esta temática repliqué mi lista de stock NoSQL a esta nueva BBDD SQL donde busco tener el control y ordenamiento de toda la información de los productos para que todos los procesos de comunicación internos y externos, ventas y distribución sean un éxito.

Cuento con detalle de la siguiente información: clientes, facturas de compra con fecha y tiempo de emisión, ventas, productos, precios, cantidades y categorías de los mismos.



NOMBRE DEL CAMPO	DESCRIPCION	TIPO DE DATO	LONGITUD	LLAVE
ID_CATEGORIA	Nº DE IDENTIFICACION DE LA CATEGORIA	INT		PK
NOMBRE_CATEGORIA	NOMBRE DE LA CATEGORIA	VARCHAR	50	

NOMBRE DEL CAMPO	DESCRIPCION	TIPO DE DATO	LONGITUD	LLAVE
ID_PRODUCTO	Nº DE IDENTIFICACION DEL PRODUCTO	INT		PK
NOMBRE_PRODUCTO	NOMBRE DEL PRODUCTO	VARCHAR	50	
VISTA_PRODUCTO	NOMBRE DE VISTA DEL PRODUCTO	VARCHAR	50	
PREDESC_PRODUCTO	PRE-DESCRIPCION DEL PRODUCTO	VARCHAR	100	
DESC_PRODUCTO	DESCRIPCION DEL PRODUCTO	VARCHAR	700	
ID_CATEGORIA	Nº DE IDENTIFICACION DE LA CATEGORIA	INT		FK
STOCK_PRODUCTO	CANTIDAD DISPONIBLE DEL PRODUCTO	INT		
PRECIO_PRODUCTO	PRECIO DEL PRODUCTO POR UNIDAD	INT		
IMG_PRODUCTO	DIRECCION URL DE LA IMAGEN DEL PRODUCTO	VARCHAR	1000	

NOMBRE DEL CAMPO	DESCRIPCION	TIPO DE DATO	LONGITUD	LLAVE
ID_CARRITO	Nº DE IDENTIFICACION DEL CARRITO	INT		PK
CANT_PROD_CARRITO	CANTIDAD DE PRODUCTOS EN EL CARRITO	INT		
ID_PRODUCTO	Nº DE IDENTIFICACION DEL PRODUCTO	INT		FK

NOMBRE DEL CAMPO	DESCRIPCION	TIPO DE DATO	LONGITUD	LLAVE
ID_CLIENTE	Nº DE IDENTIFICACION DEL CLIENTE	INT		PK
NOMBRE_CLIENTE	NOMBRE DEL CLIENTE	TINYTEXT	50	
APELLIDO_CLIENTE	APELLIDO DEL CLIENTE	TINYTEXT	50	
USUARIO_CLIENTE	USUARIO DEL CLIENTE	VARCHAR	20	
EMAIL_CLIENTE	CORREO ELECTRONICO DEL CLIENTE	VARCHAR	50	
PROVINCIA_CLIENTE	UBICACIÓN DEL CLIENTE (MACRO)	TINYTEXT	50	
DIRECCION_CLIENTE	CALLE, NUMERACION, BARRIO (MICRO)	VARCHAR	100	
TELEFONO_CLIENTE	TELEFONO CELULAR DEL CLIENTE	VARCHAR	20	



# TABLA VENTA

NOMBRE DEL CAMPO	DESCRIPCION	TIPO DE DATO	LONGITUD	LLAVE
ID_VENTA	Nº DE IDENTIFICACION DE LA VENTA	INT		PK
FECHA_VENTA	FECHA Y HORA DE LA VENTA	DATETIME		
ID_CARRITO	Nº DE IDENTIFICACION DEL CARRITO	INT		FK
ID_CLIENTE	Nº DE IDENTIFICACION DEL CLIENTE	INT		FK

# Creación de base de datos y tablas en SQL

A través de la utilización de la sentencia **CREATE** del sublenguaje **DDL** se crearon las tablas definiendo para cada una sus campos, tipos de datos, clave primaria y claves foráneas que las relacionan según el diagrama E-R.

# Inserción de datos

Para la inserción de los datos de la base se optó por la utilización de la sentencia **INSERT** de sublenguaje **DML**. A su vez para la incorporación de los demás datos a la tabla '**producto**' se procedió a incorporarlos a través de importación mediante un archivo **CSV**, detallando los pasos aplicados a continuación.



## 1. Verificar contenido

Preparar los datos a importar para que coincidan con el formato de los tipos de datos establecidos en la creación de la tabla.

## 2. Asistente de importación

Dentro de MySQL Workbench, en el schema '**redstore**' y en la tabla de '**producto**' selecciono Table Data Import (asistente de importación) el cual nos asistirá en el proceso.

## 3. Importar contenido

Selecciono el archivo **CSV** con los datos de los demás productos y los incorporo a la tabla ya existente. Se puede crear una tabla nueva y/o sobrescribir los datos existentes con el asistente.

## 4. Finalizar la importación

Verificamos que coincidan los datos en los campos correspondientes y finalizamos la importación. Por último, validamos el contenido importado.

## Vista 1

productos\_sin\_stock: compuesta por la tabla **producto**, visualiza de una forma más resumida y rápida los productos con cantidad 0(cero) en stock.

## Vista 2

clientes\_mendocinos: cuenta la cantidad de clientes agrupados por provincia en donde el valor sea 'Mendoza'.

## Vista 3

productos\_auriculares: la misma muestra por listado nombre\_categoria, nombre\_producto y predesc\_producto de cada producto perteneciente a la categoría **Auriculares**.

## Vista 4

compras\_de\_mario: visualiza de forma práctica todo el historial de compras del cliente con nombre 'Mario'.

## Vista 5

producto\_mas\_caro: visualiza rápidamente dentro de la lista de productos, el más caro de todos.

# Creación de funciones

Función **cantidad\_productos\_según**: la misma facilita la visualización del retorno de los productos según la palabra ingresada por el usuario que la consulte.

Función **precio\_mas\_iva**: la misma facilita la visualización del precio final de un producto según el id del mismo (30% IVA).

# Creación de stored procedure

**ordenar\_productos\_según**: el mismo me permitirá ordenar la tabla **producto** según el parámetro elegido, para ello se le asigna a cada campo de la tabla **producto** un número entero para ser llamado al momento de elegir el primer parámetro (campo) y a su vez con el segundo parámetro (orden) podemos indicar si ese orden lo queremos de manera ASC = 1 o DESC = 2. Este procedimiento facilitará la visualización y la posibilidad de hacer consultas rápidas de la tabla principal.

**crear\_nuevo\_producto**: el mismo me permitirá el ingreso de datos de productos de una manera más práctica en la tabla **producto**. Cuando se la llama se ingresan los parámetros nombre, vista, predescripción, descripción, categoría, stock, precio y crea un nuevo registro en la tabla **producto**.

# Creación de triggers

Dentro de la base redstore se crearon dos tablas de tipo LOG con el objetivo de reunir información de las modificaciones sufridas en las tablas '**producto**' y '**cliente**'.

Cada trigger registra el usuario, fecha y hora de cada nueva modificación en dichas tablas.

# SCRIPTS SQL/ RECURSOS



TABLAS



INSERCIONES



DATOS



VISTAS



FUNCIONES



STORED  
PROCEDURE



TRIGGERS



DCL



TCL



DUMP