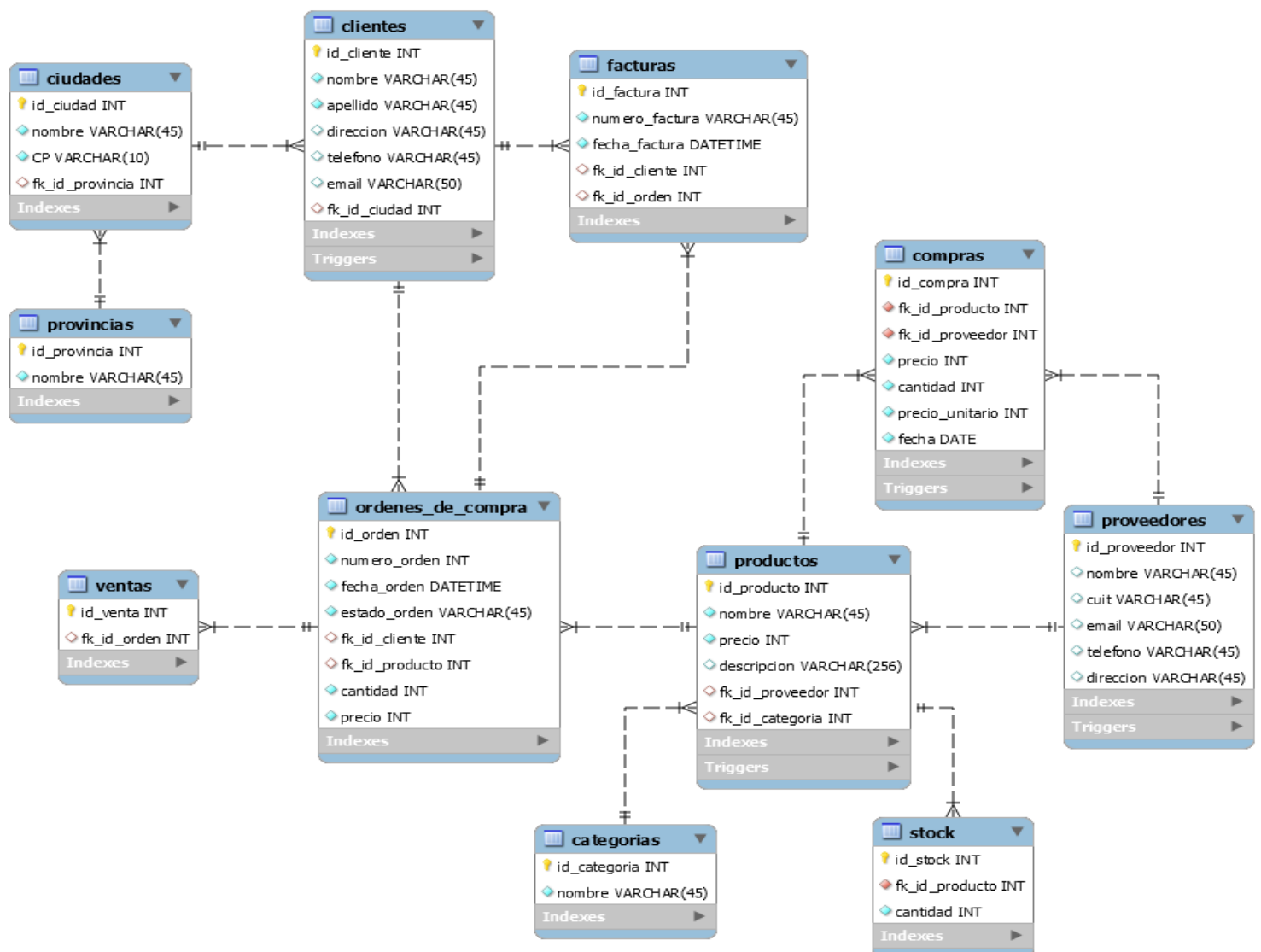


## BASE DE DATOS “TIENDA ONLINE DE PRODUCTOS DE INFORMÁTICA”

La base de datos desarrollada a continuación hace referencia a una tienda online que comercializa productos de hardware y software para PC, su principal punto de venta es la tienda online.

Como principales tablas encontramos Clientes, Ventas, Productos, Órdenes de compra, Facturas, Proveedores, Categorías, Stock, Provincias, Ciudades, Compras.

### DIAGRAMA ENTIDAD RELACIÓN:



## **TABLAS:**

**Tabla PRODUCTOS**

**Almacena la descripción, precio, stock de los productos**

<i>KEY</i>	<i>COLUMN</i>	<i>TYPE</i>	<i>NOT NULL</i>	<i>UNIQUE</i>	<i>NOTES</i>
<i>PK</i>	id_producto	INT	TRUE	TRUE	id del producto
	nombre	VARCHAR	TRUE		nombre del producto
	precio	INT	TRUE		precio del producto
	descripción	VARCHAR	TRUE		describe el producto
<i>FK</i>	id_proveedor	INT	TRUE		clave foránea, indica el proveedor
<i>FK</i>	id_categoria	INT	TRUE		clave foránea, indica la categoría a la cual pertenece un producto

**Tabla ORDENES\_DE\_COMPRA**

**Detalla los productos, clientes, fecha y facturas**

<i>KEY</i>	<i>COLUMN</i>	<i>TYPE</i>	<i>NOT NULL</i>	<i>UNIQUE</i>	<i>NOTES</i>
<i>PK</i>	id_orden	INT	TRUE	TRUE	id de la orden
	numero_orden	INT	TRUE	TRUE	nº interno de orden de compra
	fecha_orden	DATE	TRUE		indica fecha del pedido
	estado_orden	VARCHAR	TRUE		indica el estado en que se encuentra la orden (en preparación, lista para enviar, enviada, cancelada)
	cantidad	INT	TRUE		indica la cantidad de unidades compradas
	precio	INT	TRUE		indica el precio pagado
<i>FK</i>	id_cliente	INT	TRUE	TRUE	clave foránea, indica el cliente que realizo la orden
<i>FK</i>	id_factura	INT	TRUE	TRUE	clave foránea, indica la factura de referencia

### Tabla CLIENTES

Contiene los datos de los clientes de la tienda

KEY	COLUMN	TYPE	NOT NULL	UNIQUE	NOTES
PK	id_cliente	INT	TRUE	TRUE	id del cliente
	nombre	VARCHAR	TRUE		nombre del cliente
	apellido	VARCHAR	TRUE		apellido del cliente
	direccion	VARCHAR			dirección del cliente
	telefono	VARCHAR			teléfono del cliente
	email	VARCHAR	TRUE		email del cliente
FK	id_ciudad	INT	TRUE	TRUE	id de la ciudad

### Tabla VENTAS

Detalla las ventas

KEY	COLUMN	TYPE	NOT NULL	UNIQUE	NOTES
PK	id_venta	INT	TRUE	TRUE	id de la venta
FK	id_orden	INT	TRUE	TRUE	clave foránea, indica nº orden de compra que contiene todos los datos de la venta

### Tabla PROVEEDORES

Contiene información sobre los proveedores de la tienda

KEY	COLUMN	TYPE	NOT NULL	UNIQUE	NOTES
PK	id_proveedor	INT	TRUE	TRUE	id del proveedor
	nombre	VARCHAR	TRUE		nombre del proveedor
	cuit	INT	TRUE		cuit del proveedor
	email	VARCHAR	TRUE		email del proveedor
	telefono	INT	TRUE		teléfono del proveedor
	direccion	VARCHAR	TRUE		domicilio del proveedor

### Tabla FACTURAS

Detalla las facturas emitidas por la tienda

KEY	COLUMN	TYPE	NOT NULL	UNIQUE	NOTES
PK	id_factura	INT	TRUE	TRUE	id de la factura
	número	INT	TRUE	TRUE	nº de factura
	fecha	DATE	TRUE		fecha de la factura
FK	id_orden	INT	TRUE		clave foránea, indica el nº de orden asociada a la factura
FK	id_cliente	INT	TRUE		clave foránea, indica el id del cliente asociado a la factura

### Tabla CATEGORIAS

Indica el nombre de cada categoría de productos para su clasificación o filtrado

KEY	COLUMN	TYPE	NOT NULL	UNIQUE	NOTES
PK	id_categoria	INT	TRUE	TRUE	id de la categoría
	nombre	VARCHAR	TRUE		indica la categoría del producto

### Tabla STOCK

Indica el stock disponible de cada producto

KEY	COLUMN	TYPE	NOT NULL	UNIQUE	NOTES
PK	id_stock	INT	TRUE	TRUE	id stock
FK	id_producto	INT	TRUE	TRUE	clave foránea, indica el id del producto
	cantidad	INT	TRUE		indica la cantidad de unidades

### Tabla PROVINCIAS

Detalla las provincias para los domicilios de los clientes

KEY	COLUMN	TYPE	NOT NULL	UNIQUE	NOTES
PK	id_provincia	INT	TRUE	TRUE	id de la provincia
	nombre	VARCHAR	TRUE	TRUE	nombre de la ciudad

### Tabla CIUDADES

Detalla las ciudades para los domicilios de los clientes

KEY	COLUMN	TYPE	NOT NULL	UNIQUE	NOTES
PK	id_ciudad	INT	TRUE	TRUE	id de la ciudad
	nombre	VARCHAR	TRUE	TRUE	nombre de la ciudad
	CP	VARCHAR	TRUE	TRUE	código postal
FK	id_provincia	INT	TRUE		clave foranea, indica el id de la provincia a la que pertenece la ciudad

### Tabla COMPRAS

Detalla las compras de insumos realizada por la tienda

KEY	COLUMN	TYPE	NOT NULL	UNIQUE	NOTES
PK	id_compra	INT	TRUE	TRUE	id de la compra
FK	id_producto	INT	TRUE	TRUE	clave foránea, indica el id del producto
FK	id_proveedor	INT	TRUE	TRUE	clave foránea, indica el id del proveedor
	precio	INT	TRUE		precio pagado
	cantidad	INT	TRUE		indica la cantidad de unidades compradas
	Precio_unitario	INT	TRUE		precio unitario del producto
	fecha	DATE	TRUE		fecha de la compra

## VISTAS:

### 1) **"productos\_por\_orden"**

Esta vista permite a través de un JOIN, unir las tablas "ordenes\_de\_compra" y "productos" para poder observar que productos incluye cada orden.

### 2) **"productos\_por\_categoria"**

Permite ver el total de productos disponibles por cada categoría de producto existente. Utiliza la función COUNT y realiza un JOIN con las tablas "productos" y "categorías".

### 3) **"cantidad\_ordenes\_por\_estado"**

Indica el total de ordenes que hay de acuerdo al estado que pueden tomar las mismas. Utiliza la función GROUP BY y COUNT para lograrlo.

### 4) **"compras\_por\_proveedor"**

Vista que indica el total de compras realizadas a cada proveedor, ordenadas de forma descendente. Utiliza subconsultas dentro de un INNER JOIN, GROUP BY y la función DESC para ordenar los resultados.

### 5) **"compras\_por\_cliente"**

Vista que indica el total de compras realizadas por cada cliente, ordenadas de forma descendente. Utiliza subconsultas dentro de un INNER JOIN, GROUP BY y la función DESC para ordenar los resultados.

### 6) **"prom\_por\_estado\_orden"**

Permite ver el promedio de ventas de acuerdo al estado en que se encuentra la orden de compra. Utiliza la función AVG y GROUP BY.

## FUNCIONES:

### 1) "**monto\_orden**"

Indica el total de la orden de compra, ingresando como parámetro el n° de orden. Realiza una multiplicación de la columna cantidad y precio de la tabla "ordenes\_de\_compra".

### 2) "**beneficio\_por\_producto**"

Calcula el beneficio neto que deja cada producto, tomando como parámetros el costo y el precio de venta.

### 3) "**nombre\_cliente**"

Permite obtener el nombre y apellido del cliente ingresando el n° de id. Utiliza la tabla "clientes" y concatena las columnas nombre y apellido para formar el resultado.

### 4) "**clientes\_por\_ciudad**"

Permite obtener el total de clientes por la ciudad que se indique, esto sirve para realizar, por ejemplo, promociones en las ciudades con mayor cantidad de clientes.

## STORED PROCEDURES:

### 1) **"sp\_generar\_proveedor"**

Permite la creación de un nuevo proveedor ingresando los datos del mismo a la tabla "proveedores". Utiliza un Stored Procedure llamado "sp\_ultimo\_proveedor" que indica el id del ultimo proveedor de la tabla, para utilizar este número y generar el nuevo registro. El beneficio que aporta es que con un simple CALL y los datos del nuevo proveedor se puede agregar fácilmente un nuevo registro en la tabla "proveedores".

### 2) **"sp\_ordenar\_productos"**

Ordena la tabla "productos" de forma descendente o ascendente de acuerdo a la columna que se indique como parámetro. Utiliza la función CONCAT para armar la sentencia que permite ejecutar el query y obtener el resultado deseado.

### 3) **"sp\_generar\_cliente"**

Permite generar un nuevo cliente en la tabla cliente una vez que se ingresan los datos correspondientes al mismo. Se basa en un Stored Procedure llamado "sp\_ultimo\_cliente" para generar el nuevo registro.



## TRIGGERS:

### 1) **"bu\_proveedores"**

Trigger de tipo BEFORE UPDATE que permite registrar en una tabla de log los cambios que se realicen sobre algún registro de un proveedor, indicando además la fecha y el usuario que lo realizó. Requiere primero generar la tabla para registrar los cambios, llamada "log\_proveedores".

### 2) **"ad\_proveedores"**

Trigger AFTER DELETE, cuando se elimina algún proveedor o dato de estos, este trigger registra en la tabla "log\_proveedores" los datos borrados junto con el usuario y la fecha.

### 3) **"ai\_compras"**

Trigger AFTER INSERT en tabla "compras", registra los nuevos cambios en la tabla "log\_compras" para mantener un registro de la fecha y el usuario que ingresa datos a una tabla importante como es compras.

### 4) **"bd\_cliente"**

Trigger BEFORE DELETE, se encarga de emitir un mensaje de alerta cuando se intenta eliminar los datos o un registro de la tabla "clientes", manteniendo así un nivel adicional de seguridad para la base.

### 5) **"bi\_productos"**

Trigger BEFORE INSERT, verifica que al momento de ingresar el precio de un producto este no sea menor a 0, evitando los problemas que esto conlleva para el negocio.

### 6) **"bi\_clientes"**

Trigger BEFORE INSERT en tabla "clientes", realiza un control del email ingresado para el cliente, verificando que tenga la estructura adecuada de un email.