

Curso SQL: Proyecto Final Datos para todos

Autor: Lucas Andrés Márquez

Tutor: Sergio Occhipinti

Profesor: César Aracena

CoderHouse - Comisión 34965

Contenido.

- Descripción de la temática de los Datos.
- Diagrama Entidad - Relación
- Listado de Tablas
- Código SQL para la creación del schema e_commerce_eeuu

Descripción de la temática de los Datos.

Base de datos sobre E-Commerce de Estados Unidos

Para mi proyecto final escogí un dataset de kaggle sobre los e-commerce de Estados Unidos en el 2020, utilizándolo para explorar las funcionalidades del lenguaje SQL y sublenguajes DML, DDL, DLC y TCL.

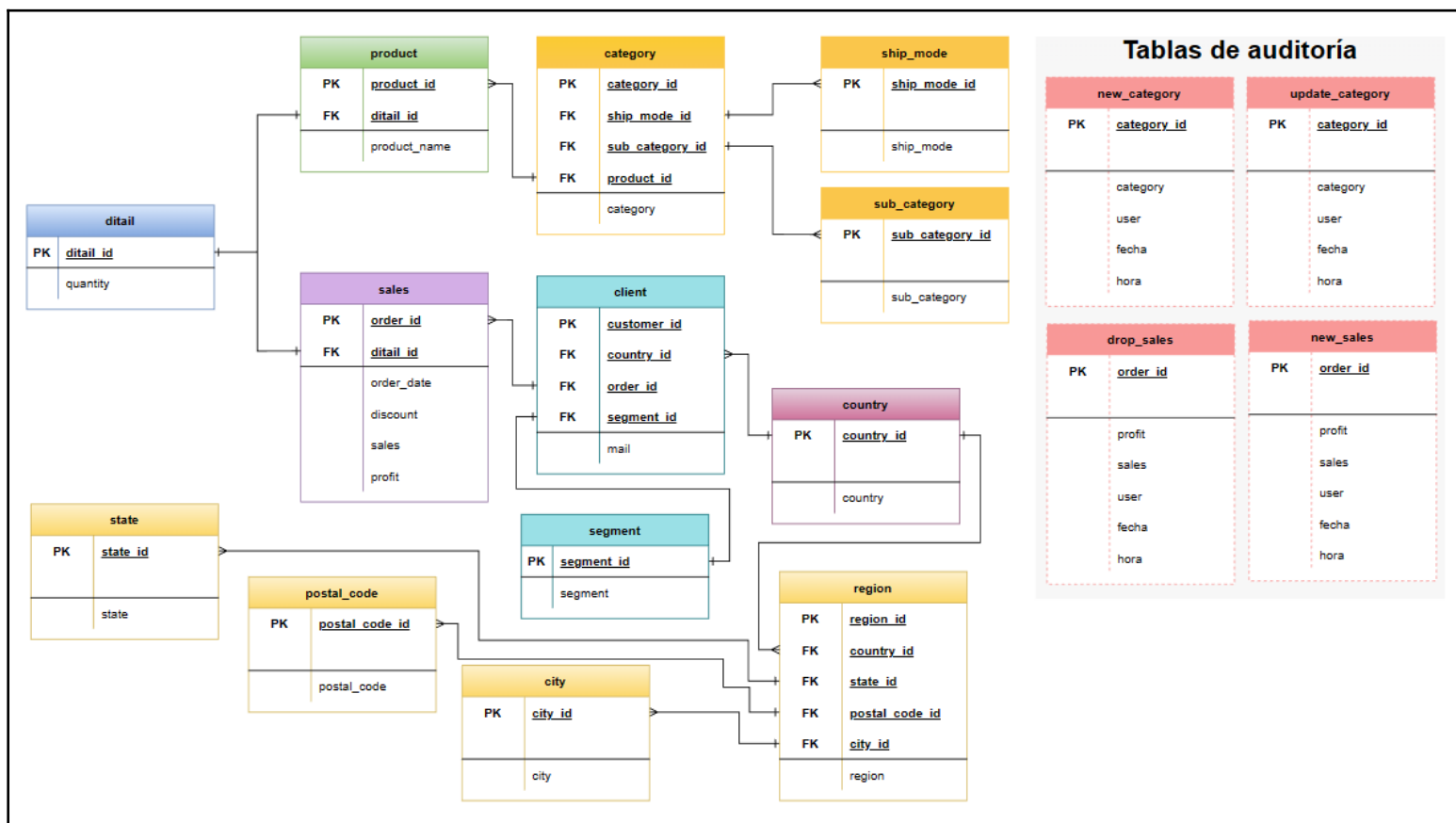
En la misma se generarán diferentes tipos de tablas, realizando su estructuración y relación con otras tablas para poder realizar consultas sobre los diferentes datos cargados. Las tablas serán: “client”, “segment”, “region”, “postal_code”, “country”, “city”, “state”, “sales”, “product”, “category”, “ship_mode”, “sub_category”, “ditail”.

Objetivos:

Realizar un seguimiento detallado de la mercadería, generando un control de egreso de las mismas para evaluar por región, estado y ciudad las ventas en Estados Unidos y cuales son sus ingresos.

Diagrama Entidad - Relación

A continuación se presenta el diagrama de entidad relación creado. Además se deja el [hiperlink de acceso al Diagrama Entidad - Relación](#) para una visualización más detallada.



Listado de Tablas

Listado de Tablas

Tabla	client						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS DE CADA CLIENTE						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	customer_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario de cada cliente
FK	order_id	INT	-	NOT NULL	-	-	id de cada region dentro de los Estados Unidos
FK	segment_id	INT	-	NOT NULL	-	-	id sobre que tipo de cliente es
FK	country_id	INT	-	NOT NULL	-	-	id de cada país
-	mail	VARCHAR	40		-	-	mail del cliente
Tabla	segment						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS SOBRE QUE TIPO DE CLIENTE ES						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	segment_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario sobre que tipo de cliente es
-	segment	VARCHAR	40	NOT NULL	-	-	que tipo de cliente es
Tabla	region						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS LOGISTICOS						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	region_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario de cada region
FK	country_id	INT	-	NOT NULL	-	-	id país de la venta
FK	postal_code	INT	-	NOT NULL	-	-	id codigo postal de la region
FK	city_id	INT	-	NOT NULL	-	-	id sobre la ciudad de la venta
FK	state_id	INT	-	NOT NULL	-	-	id del estado donde se realizo la venta
-	region	VARCHAR	40	-	-	-	region dentro de los Estados Unidos
Tabla	postal_code						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS DE LOS CODIGOS POSTALES						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	postal_code	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario del codigo postal de la region
-	postal_code	INT	-	NOT NULL	-	-	codigo postal de la region
Tabla	country						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS SOBRE EL PAÍS DONDE SE REALIZO LA VENTA						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	country_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario de cada país
-	country	VARCHAR	40				país de la venta

Tabla	city						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS SOBRE LA CIUDAD DONDE SE REALIZO LA VENTA						
KEY	COLUMN	TYPE	LENGTH	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	city_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario sobre la ciudad de la venta
-	city	VARCHAR	40	-	-	-	ciudad de la venta
Tabla	state						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS SOBRE EL ESTADO DONDE SE REALIZO LA VENTA						
KEY	COLUMN	TYPE	LENGTH	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	state_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario del estado donde se realizo la venta
-	state	VARCHAR	40	-	-	-	estado donde se realizo la venta
Tabla	sales						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS DE LAS VENTAS						
KEY	COLUMN	TYPE	LENGTH	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	order_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario de cada pedido
FK	detail_id	INT	-	NOT NULL	-	-	id de cada cliente
-	order_date	DATE	-	-	-	-	fecha de la venta
-	discount	DECIMAL	4,2	-	-	-	descuento realizado sobre la compra
-	sales	DECIMAL	11,3	-	-	-	numero de la venta realizada
-	profit	DECIMAL	11,2	-	-	-	saldo positivo o negativo luego de la venta
Tabla	product						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS NOMBRES DE LOS PRODUCTOS						
KEY	COLUMN	TYPE	LENGTH	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	product_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario de cada producto
FK	detail_id	INT	-	NOT NULL	-	-	id primario de cada categoria
-	product_name	VARCHAR	100	NOT NULL	-	-	nombre del producto
Tabla	category						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS SOBRE QUE CATEGORIA ES						
KEY	COLUMN	TYPE	LENGTH	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	category_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario de cada categoria
FK	product_id	INT	-	NOT NULL	-	-	id de cada producto
FK	ship_mode_id	INT	-	NOT NULL	-	-	id de la clase de envio
FK	sub_category	INT	-	NOT NULL	-	-	id de la subdivision de la categoria
-	category	VARCHAR	40	-	-	-	categoria del producto

Tabla	new_category						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS SOBRE EL INSERT QUE SE GENERE EN LA TABLA CATEGORY						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	category_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario de cada categoria
-	category	VARCHAR	40	-	-	-	categoria del producto
-	user	VARCHAR	40	NOT NULL	-	-	usuario que genero la accion
-	fecha	DATE	-	NOT NULL	-	-	fecha del proceso
-	hora	TIME	-	NOT NULL	-	-	hora del proceso
Tabla	update_category						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS SOBRE EL UPDATE QUE SE GENERE EN LA TABLA CATEGORY						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	category_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario de cada categoria
-	category	VARCHAR	40	-	-	-	categoria del producto
-	user	VARCHAR	40	NOT NULL	-	-	usuario que genero la accion
-	fecha	DATE	-	NOT NULL	-	-	fecha del proceso
-	hora	TIME	-	NOT NULL	-	-	hora del proceso
Tabla	drop_sales						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS DE LAS VENTAS CUANDO HAY UN DELETE EN SALES						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	order_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario de cada pedido
-	sales	DECIMAL	11,3	-	-	-	numero de la venta realizada
-	profit	DECIMAL	11,2	-	-	-	saldo positivo o negativo luego de la venta
-	user	VARCHAR	40	NOT NULL	-	-	usuario que genero la accion
-	fecha	DATE	-	NOT NULL	-	-	fecha del proceso
-	hora	TIME	-	NOT NULL	-	-	hora del proceso
Tabla	drop_sales						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS DE LAS VENTAS CUANDO HAY UN INSERT EN SALES						
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	order_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario de cada pedido
-	sales	DECIMAL	11,3	-	-	-	numero de la venta realizada
-	profit	DECIMAL	11,2	-	-	-	saldo positivo o negativo luego de la venta
-	user	VARCHAR	40	NOT NULL	-	-	usuario que genero la accion
-	fecha	DATE	-	NOT NULL	-	-	fecha del proceso
-	hora	TIME	-	NOT NULL	-	-	hora del proceso

Tabla	ship_mode						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS SOBRE LAS CATEGORIAS DE ENVIO						
KEY	COLUMN	TYPE	LENGTH	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	ship_mode_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario de la clase de envío
-	ship_mode	VARCHAR	40	-	-	-	la clase de valor por la que se pago el envío
Tabla	sub_category						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS SOBRE LA SUBDIVISION DE LA CATEGORIA						
KEY	COLUMN	TYPE	LENGTH	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	sub_category_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario de la subdivision de la categoria
-	sub_category	VARCHAR	40	-	-	-	la subdivision de la categoria
Tabla	detail						
Descripción	TABLA DIMENSIONAL DONDE SE GUARDAN LOS DATOS SOBRE LA CANTIDAD DE PRODUCTOS VENDIDOS						
KEY	COLUMN	TYPE	LENGTH	NOT NULL	UNIQUE	DEFAULT	NOTES
PK	detail_id	INT	-	NOT NULL	-	AUTO_INCREMENT	id primario para cada transaccion realizada
-	quantity	INT	-	NOT NULL	-	-	cantidad de producto vendido

Listado de Vistas (Views), Funciones, Stored Procedure y Triggers.

Se muestran las vistas, funciones, stored procedures y triggers que forman parte de la base de datos a utilizar en el sistema de gestión del drugstore.

Vistas:

- **client_sales_view:** se crea una view que muestra el mail, order_date, discount, sales, profit y id de los clientes donde el mail sea "@hotmail". La composición está dada por las tablas: client y sales.
- **region_segment_detail_view:** se crea una view que muestra cuántos clientes hay en cada segmento agrupado por región. La composición está dada por las tablas: client, region, segment y detail.
- **city_sales_detail_product_view:** se crea una view donde se visualiza city, sales, quantity, product_name ordenado por city de forma descendiente. La composición está dada por las tablas: city, client, sales, detail y product.
- **category_ship_mode_sub_category_product_detail_sales_view:** se crea una view donde se visualiza category, ship mode, subcategory, product name, quantity y toda la tabla de sales donde quantity sea mayor a dos y ship_mode sea "First Class". La composición está dada por las tablas: category, ship_mode, sub_category, product, detail y sales.
- **product_sales_city_postal_code_view:** se crea una view donde se visualiza city, postal_code y la suma de profit agrupada por el código postal y ordenada por ciudad. La composición está dada por las tablas: sales, product, city y postal_code.

Funciones:

- **func_segment_level:** se crea la función 'segment_level' para separar la segmentación según la cantidad en platinum, gold o silver. Dependiendo cada sector, será distinto cuando pase a formar parte de una u otra categoría. Está compuesta por las tablas: segment y quantity.
- **func_discount:** se crea la función 'discount' por medio del cual se pretende pasar un monto de venta y que devuelva el descuento que se debería hacer. Está compuesta por la tabla: sales.

Store Procedures:

- **sp_concat_location:** se crea un sp 'concat_location' que devuelve una concatenación de customer_id, city, state, region, postal_code separado por " - ". Se compone por las tablas: client, city, state, region y postal_code.
- **sp_order_date_client:** se crea un sp 'sp_order_date_client' que pide una mecha de entrada y otra de salida devolviendo todos los datos de la tabla sales que estén entre esos parámetros. Se compone por la tabla: sales.

Triggers:

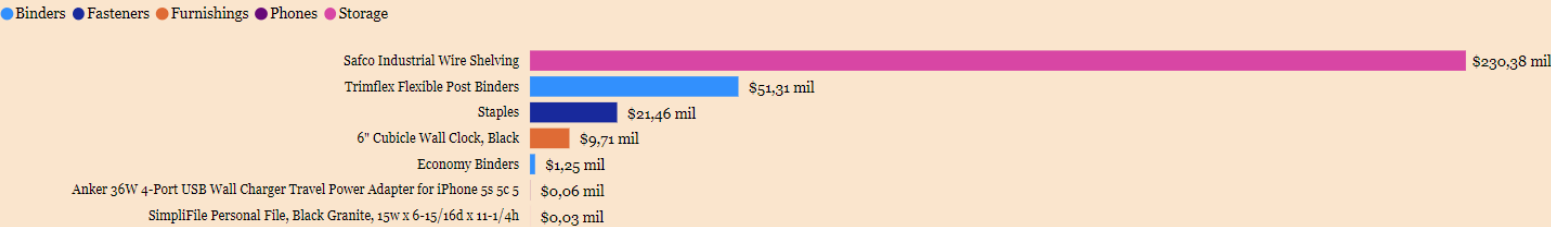
- **tr_add_new_sales:** este trigger nos permite auditar la tabla sales. Luego de ingresar nuevos valores se genera un registro de order_id, sales, profit, hora, fecha y el usuario que realizó el procedimiento mediante el uso de la sentencia SESSION_USER(). Para esto se creó una tabla llamada new_sales.
- **tr_delete_sales:** este trigger nos permite auditar la tabla sales. Antes de borrar la fila se registran los valores de order_id, sales, profit, hora, fecha y el usuario que realizó el procedimiento mediante el uso de la sentencia SESSION_USER(). Para esto se creó una tabla llamada drop_sales.
- **tr_update_category:** este trigger nos permite auditar la tabla category. Antes de borrar la fila se registran los valores, tanto viejos como nuevos, de category_id y category. También se registra la hora, fecha y el usuario que realizó el procedimiento mediante el uso de la sentencia SESSION_USER(). Para esto se creó una tabla llamada update_category.
- **tr_add_new_category:** este trigger nos permite auditar la tabla category. Luego de ingresar nuevos valores se genera un registro de category_id, category, hora, fecha y el usuario que realizó el procedimiento mediante el uso de la sentencia SESSION_USER(). Para esto se creó una tabla llamada new_category.

Gráficos de informes generados por views

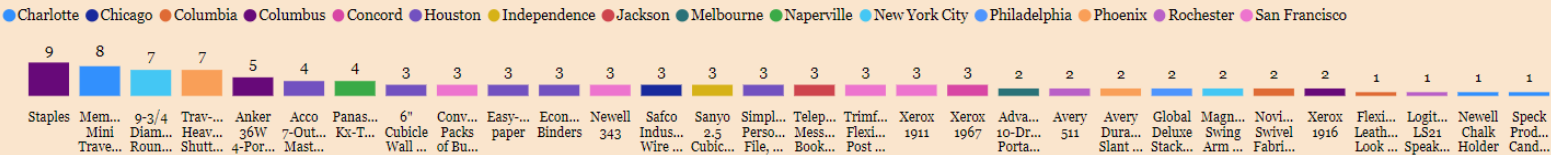
Ubicación de ganancias por código postal



Ventas de productos por subcategorias



Cantidad de productos vendidos por ciudad



Fechas de ventas a clientes

customer_id	Año	Trimestre	Mes	Día	sales
20	2025	Trim. 4	diciembre	20	\$41,96
14	2006	Trim. 4	noviembre	20	\$5.682
15	2009	Trim. 4	noviembre	20	\$96,53
21	2005	Trim. 4	noviembre	20	\$2.388
27	2018	Trim. 3	septiembre	20	\$10,86
4	2010	Trim. 3	septiembre	20	\$147,168
5	2019	Trim. 3	septiembre	20	\$95.616
16	2017	Trim. 2	junio	20	\$51.312
1	2015	Trim. 2	abril	20	\$15.552
30	2021	Trim. 2	abril	20	\$97.264

Código SQL para la creación del schema e_commerce_eeuu

[Link a e_commerce_eeuu - schema](#)

[Link a e_commerce_eeuu - data](#)

Tecnologías utilizadas

- MySQL Server 8.0.30 + MySQL Workbench 8.0.30.

Gestor de base de datos y herramienta visual de diseño de base de datos.

- Diagrams.net V14.9.0.tar.gz

Software para la creación de flowcharts.

- Microsoft Power BI 2.112.283.0

Herramienta utilizada para la generación de visualizaciones.