**Universidad del Valle de Puebla**

Administración de Tecnologías de la Información.

**Base de Datos I.**

Proyecto: Sistema de Base de datos para Tienda Proyecto Walmart

Ciclo: Primer Semestre

Profesor:

Dr. Daniel Pérez Rojas

Alumnos:

José Ángel Sánchez Pérez

Juan Luis Uriel Rodríguez Peralta

Contenido

[Definición del Problema 4](#_Toc473348143)

[El alcance de este proyecto 5](#_Toc473348144)

[Propuesta de Solución 6](#_Toc473348145)

[Diagrama ER 9](#_Toc473348146)

[Diagrama Relacional 10](#_Toc473348147)

[Normalización 11](#_Toc473348148)

[Descripción de la normalización 11](#_Toc473348149)

[Primera forma Normal 11](#_Toc473348150)

[La segunda forma Normal 12](#_Toc473348151)

[La tercera forma Normal 12](#_Toc473348152)

[Implementación 13](#_Toc473348153)

[Herramientas utilizadas para el desarrollo del Sistema 13](#_Toc473348154)

[Front End 13](#_Toc473348155)

[Materialize 13](#_Toc473348156)

[Angular 13](#_Toc473348157)

[BackEnd 13](#_Toc473348158)

[Php librería PDO .- 13](#_Toc473348159)

[Base de datos 13](#_Toc473348160)

[Consultas 13](#_Toc473348161)

[Descripción de uso 14](#_Toc473348162)

[Header 14](#_Toc473348163)

[Contenido 15](#_Toc473348164)

[Explicación de las "consultas interesantes" 19](#_Toc473348165)

[Consulta interesantes 1 19](#_Toc473348166)

[Consulta interesante 2 19](#_Toc473348167)

[Consulta interesante 3 21](#_Toc473348168)

[Consulta interesante 4 22](#_Toc473348169)

[Consulta interesante 5 23](#_Toc473348170)

[Consulta interesante 6 24](#_Toc473348171)

[Consulta interesante 7 25](#_Toc473348172)

[Consulta interesante 8 25](#_Toc473348173)

[Consulta interesante 9 26](#_Toc473348174)

[Consulta interesante 10 27](#_Toc473348175)

[Consulta interesante 11 28](#_Toc473348176)

[Consulta interesante 12 29](#_Toc473348177)

[Conclusiones del equipo 31](#_Toc473348178)

## Definición del Problema

Se pretende tener la solución de Base de datos con la capacidad de realizar un manejo de inventario para cada una de las tiendas las tiendas Walmart de la ciudad de Puebla, poder tener un manejo de cartera de clientes de todas las tiendas, tener una cartera de proveedores así como tener un control de las ventas y compras que realiza cada establecimiento de acuerdo a sus clientes y proveedores.

## El alcance de este proyecto

El alcance de este proyecto es el siguiente:

* Desarrollo e implementación de una Base de datos que cumpla con la solución al problema
* Generación de consultas importantes para el administrador o Gerente de la tienda puedan tomar decisiones importantes.
* Alta física de algunos catálogos de tiendas, departamentos y productos desde una aplicación desarrollada en PHP

## Propuesta de Solución

Llevar a cabo el proceso de **hacer inventario** es una tarea obligada para todo aquél que está al frente de una tienda o comercio, pues es clave para su buen funcionamiento. Un inventario actualizado ayuda a estar al tanto día a día de la **situación financiera de un negocio**, tener controlado el stock y conocer a ciencia cierta las necesidades de los clientes.

La tarea de **realizar el inventario** en cualquier comercio es conveniente llevarla a cabo de forma periódica. Aunque muchas empresas esperan al finalizar el ejercicio económico anual para hacerlo, no es aconsejable esperar tanto si nos interesa estar informados sobre la viabilidad económica de nuestro negocio en todo momento. Lo mejor es establecer la frecuencia de realizar el inventario en función de nuestro volumen de ventas y productos.

El inventario es exactamente un sistema de control que debe ser lo más preciso posible. La tarea consiste básicamente en gestionar el stock y llevar a cabo el **recuento de todas las existencias físicas reales**, es decir, de todas las unidades que tengamos de cada uno de los productos para después introducirlas en la aplicación informática que utilizamos para gestionar las ventas. Esto nos permitirá comparar los datos y comprobar si cuadran los números o existen desviaciones (faltan o sobran productos) que deben regularizarse.

El primero es contar con una clave única distinta para cada uno de los productos, así será mucho más fácil contabilizarlos y diferenciarlos entre ellos.

Confeccionar una lista tanto de los productos que más se hayan vendido como de los que no han tenido tanto éxito. Esto nos permitirá saber qué productos debemos tener en todo momento en la tienda, cuáles requieren una mayor promoción para ser vendidos, si es necesario ajustar los precios, etc.

Por todo ello, mantener el inventario actualizado es clave en el buen funcionamiento del negocio. Nos permitirá tomar las decisiones más acertadas, examinar con detalle las exigencias de nuestro clientes y ofrecerles aquellos productos o servicios que más demandan.

Por ello, es recomendable contar con una base de datos en la que podamos introducir todos la información que nos servirá para llevar el control de datos.

Después de un análisis de la información que se requiere almacenar a nivel Base de datos el equipo considera, que desarrollando un sistema de Base de datos son las siguientes entidades será suficiente para realizar el control que se desea tener:

1. Tiendas
2. Departamentos
3. Empleados
4. Proveedores
5. Clientes
6. Productos
7. Ventas Compras
8. Detalle Ventas
9. Detalle Compras

Con las entidades anteriores y una serie de consultas especificas se podrán generar estadísticas de precios, de compras, de ventas, de productos e inventarios.

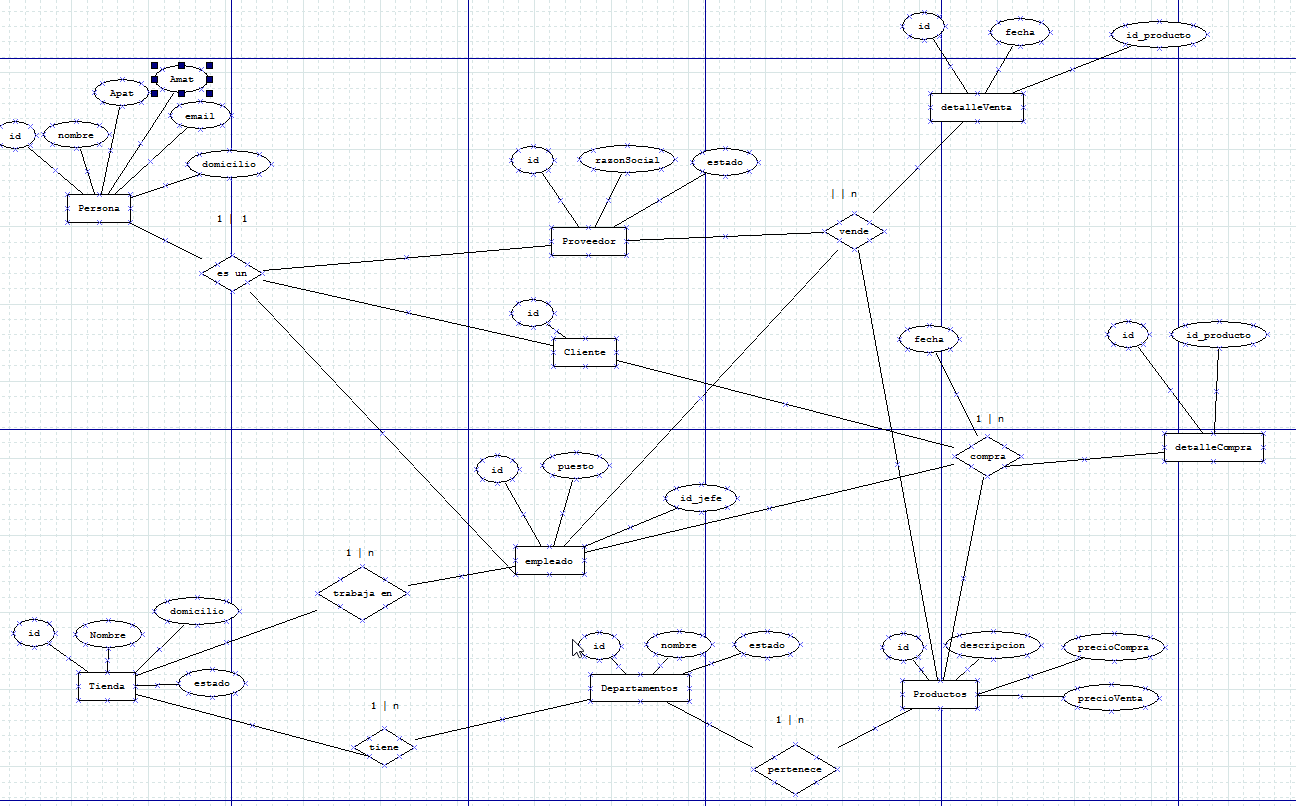
Se analiza el problema y se determina cuales serán las consultas que servirán de apoyo a los administradores o gerentes de cada tienda podrán tomar las decisiones necesarias para un correcto manejo de sus tiendas las cuales son:

* Consulta que indica con que departamentos cuenta cada sucursal
* Cuáles son los 10 productos que se compran más caro y su inventario
* Cuáles son los 20 productos a los que más ganancia se tiene por unidad, de acuerdo a cada tienda
* Consulta para saber en qué tienda y departamento se tienen los 20 productos de los que se obtiene menos ganancia
* Consulta para saber cuáles son los 20 productos de los que menos se tiene en el inventario
* Consulta para saber cuáles son los 20 productos de los que más consumo se tiene en el inventario
* Consulta pasa saber cuál es el activo de inversión que se tiene por tienda de acuerdo al inventario total
* Consulta para saber cuál será la ganancia total de cada tienda de acuerdo a su inventario para todos sus producto
* Consulta para saber cuáles son los empleados de cada tienda y el puesto ordenados por tienda y por puesto
* Consulta pasa saber los emails todos los gerentes de tiendas ordenados por nombre ascendentemente
* Consulta que nos sirve para saber que clientes existen en sistema y asi como saber hacen compra o no y en que tiendas así como sus emails
* Consulta que nos sirve para saber que proveedores están en sistema adicionalmente si venden en que tiendas y sus emails

Cabe destacar que estas son algunas consultas propuestas el usuario experimentado podría conectarse a la base de datos directamente y explotar la información de acuerdo a se vayan modificando sus necesidades de información.

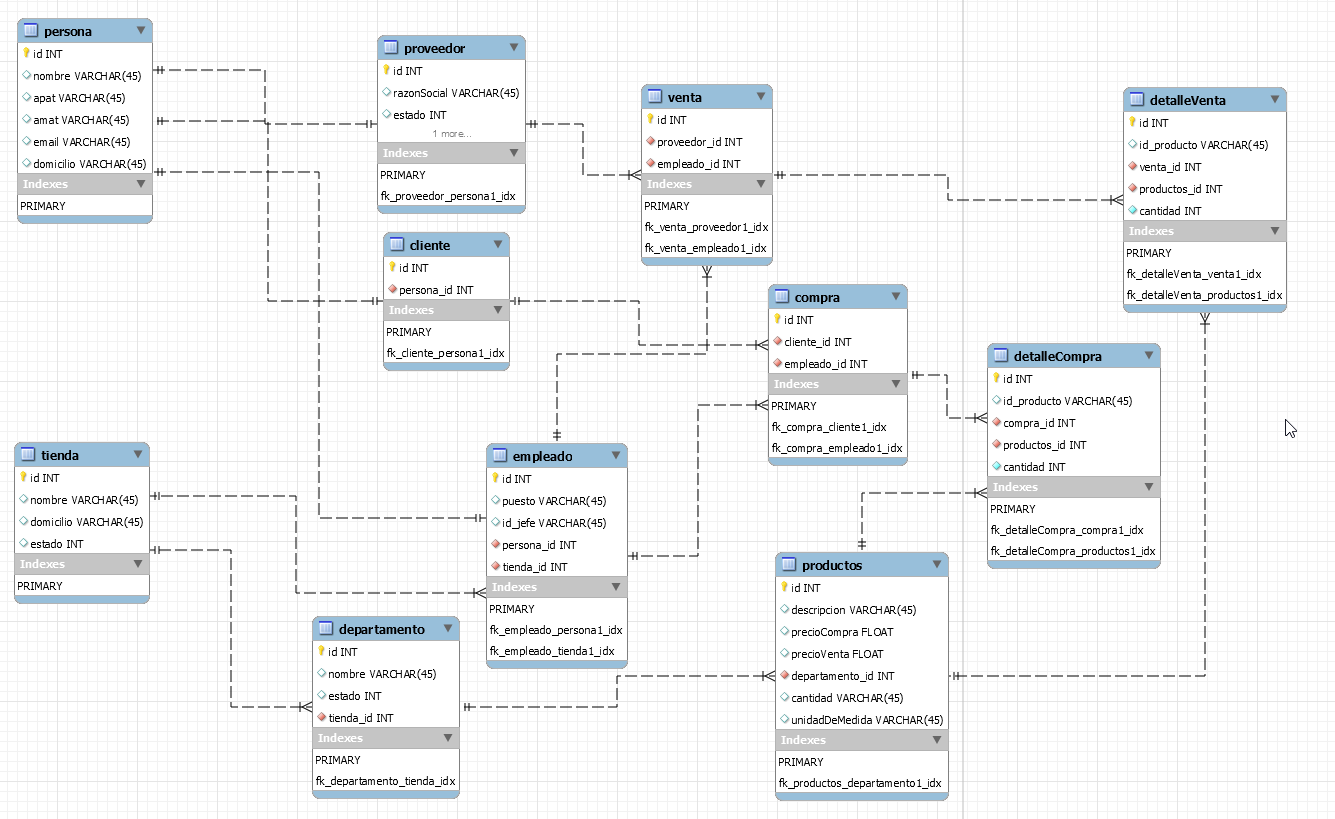
## Diagrama ER

A continuación se presenta el diagrama ER desarrollado para dar poder cumplir con la solución de almacenamiento de información que se requiere para cumplir con la solución propuesta.



## Diagrama Relacional

A continuación se presenta el diagrama Relacional para poder cumplir con la solución propuesta, es importante mostrar que existen relaciones entre las tablas de compras ventas, productos, detalles de venta y detalles de compra, con lo que se tendrá un control entre las ventas y el inventario en bodegas de cada producto con respecto a cada tienda.



## Normalización

Se valida que el modelo de base de datos sea un modelo normalizado, para esto se debe de garantizar que se cumple con las 3 formas normales.

## Descripción de la normalización

Normalización es el proceso de organizar los datos en una base de datos. Esto incluye la creación de tablas y el establecimiento de relaciones entre ellas según reglas diseñadas tanto para proteger los datos y como para hacer que la base de datos sea más flexible mediante la eliminación de redundancias y dependencias incoherentes.

Los datos redundantes desperdician espacio en disco y crean problemas de mantenimiento. Si es necesario cambiar datos que existen en más de un lugar, los datos deben cambiarse en exactamente la misma manera en todas las ubicaciones. Un cambio de dirección de es mucho más fácil de implementar si los datos se almacenan sólo en la tabla clientes y en ningún otro en la base de datos.

Hay unas cuantas reglas para la normalización de la base de datos. Cada regla se denomina "forma normal". Si se observa la primera regla, se dice que la base de datos está en la "primera forma normal". Si se observan las tres primeras reglas, se considera que la base de datos está en la "tercera forma normal". Aunque otros niveles de normalización son posibles, la tercera forma normal se considera el máximo nivel necesario para la mayoría de las aplicaciones.

### Primera forma Normal

* Eliminar grupos repetidos en tablas individuales.

Se valido que no existan grupos repetidos

* Crear una tabla independiente para cada conjunto de datos relacionados.

Se crearon tablas independientes para cada conjunto de datos

* Identificar cada conjunto de datos relacionados con una clave principal.

Se identifico cada conjunto de datos con una clave principal Irrepetible.

### La segunda forma Normal

* Crear tablas independientes para conjuntos de valores que se apliquen a varios registros.
* Se crean tablas independientes para conjuntos de datos tal es el caso de las compras y ventas, se crearon tablas especiales para su detalle de productos involucrados
* Relacione estas tablas con una clave externa.

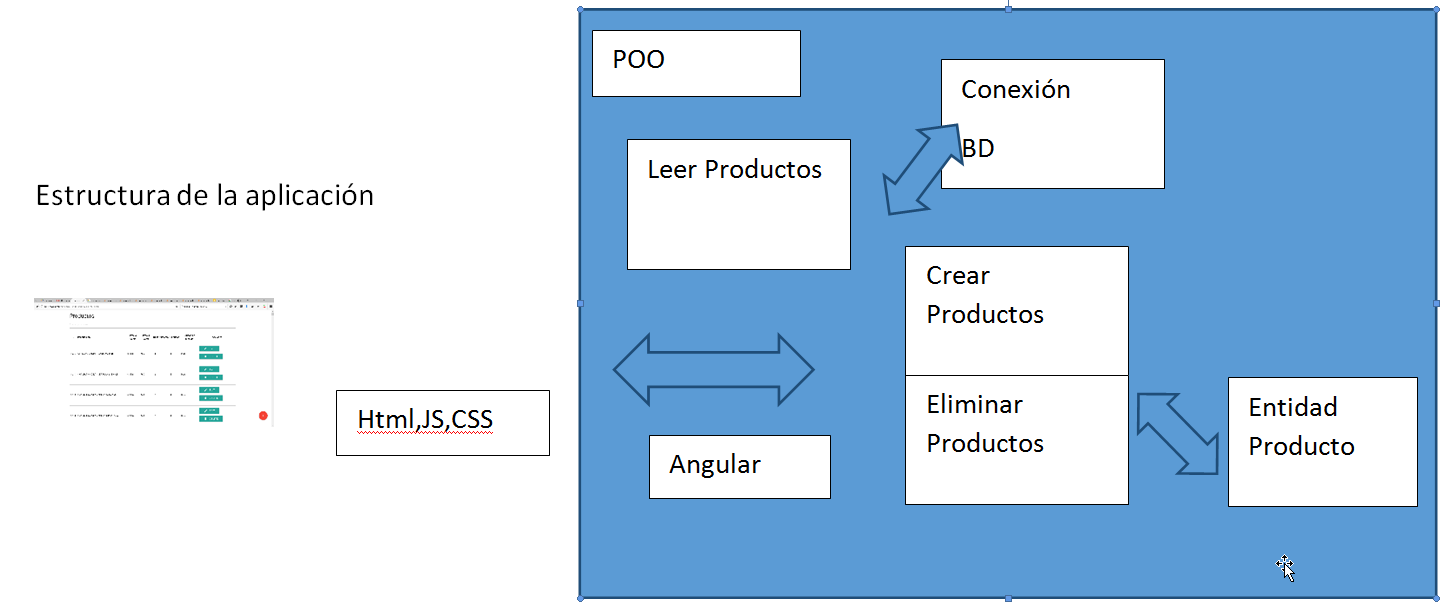
Se relacionaron los campos con llaves foráneas

### La tercera forma Normal

* Elimine los campos que no dependan de la clave.

Se valido que no existan campos que no dependan de la clave.

## Implementación



### Herramientas utilizadas para el desarrollo del Sistema

Front End .- Del lado del frontend se construyo una vista exclusivamente usando html5 , js y css , para mantener lo más puro posible un patrón de programación MVC

Materialize.- Framework CSS que permite cambiar la vista para el usuario.

Angular.- Framework de Javascript que se utiliza para la creación de paginas dinámicas evitando la recarga completa de la pagina.

BackEnd.- En el back end se definieron dos capas la parte de las entidades de los objetos que representan las tablas de las bases de datos y por otra parte se utilizaron pequeños archivos que sirvieron como controladores que respondieran a peticiones hechas por la aplicación

Php librería PDO .- La librería PDO de PHP permite un mejor manejo de la recuperación de la información. Esta librería permite que la programación sea bajo el paradigma POO

Base de datos: Se genero el esquema de base de datos en MYSQL, de acuerdo al diagrama Relacional mostrado anteriormente validando que con el cumplimiento de las primeras 3 formas normales

Consultas: Se desarrollaron los queries propuestos que servirán de de ayuda para analizar la información de la base de datos y que repercutirán en decisiones de tipo administrativo o gerencial

## Descripción de uso

Para accesar a la aplicación es necesario contar con conexión a internet e ingresar al sistema mediante la url:

http://www.legionx.com.mx/ventamati/

El sistema nos colocara en la pantalla principal.



La pantalla principal esta divida en 3 parte principales.

### Header



En esta parte se tiene acceso a los CRUD para Productos, Tiendas y Departamentos, se plantea que en esta sección en una fase complementaria se pueda contar con CRUD para todas las entidades.

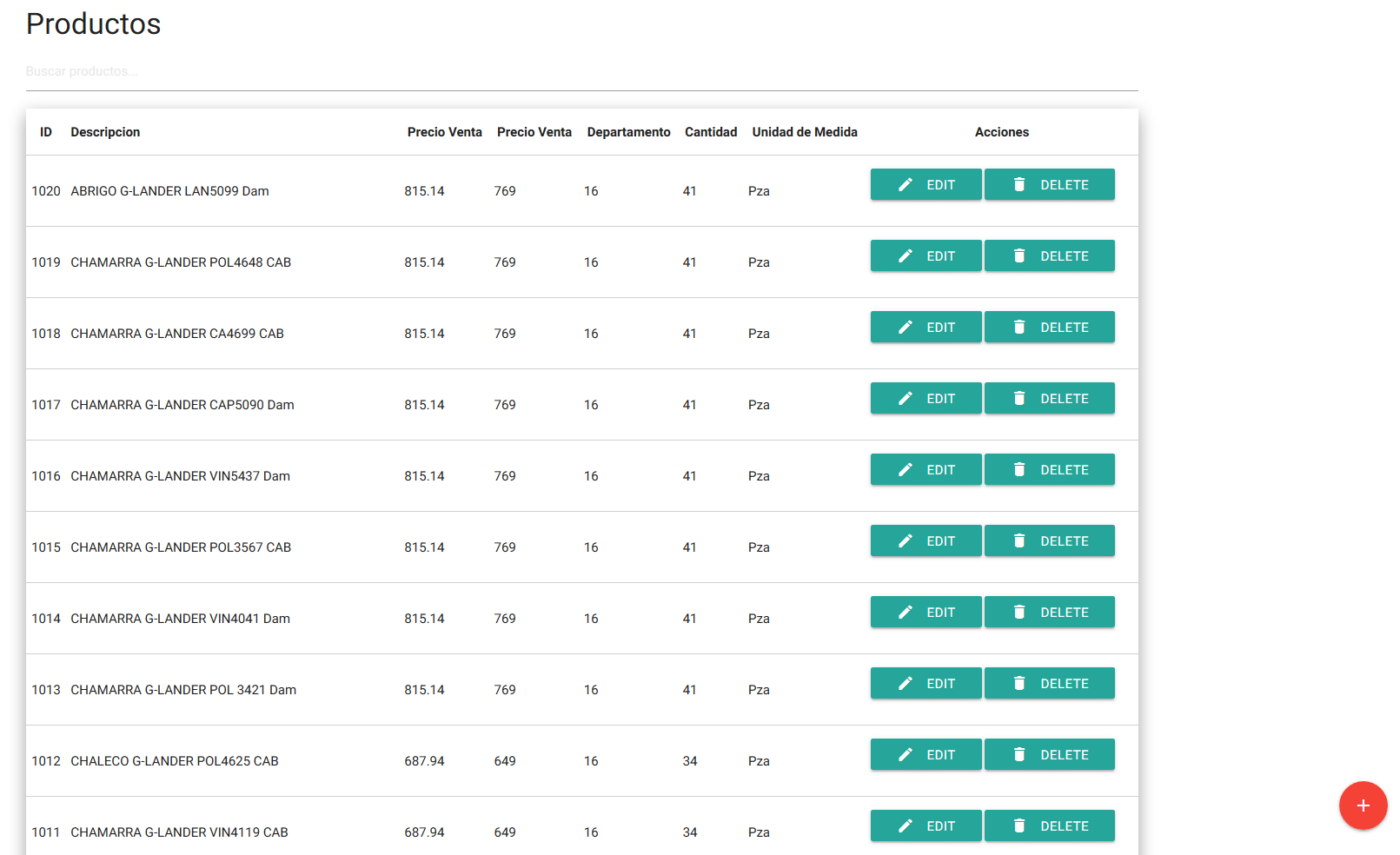
### Contenido



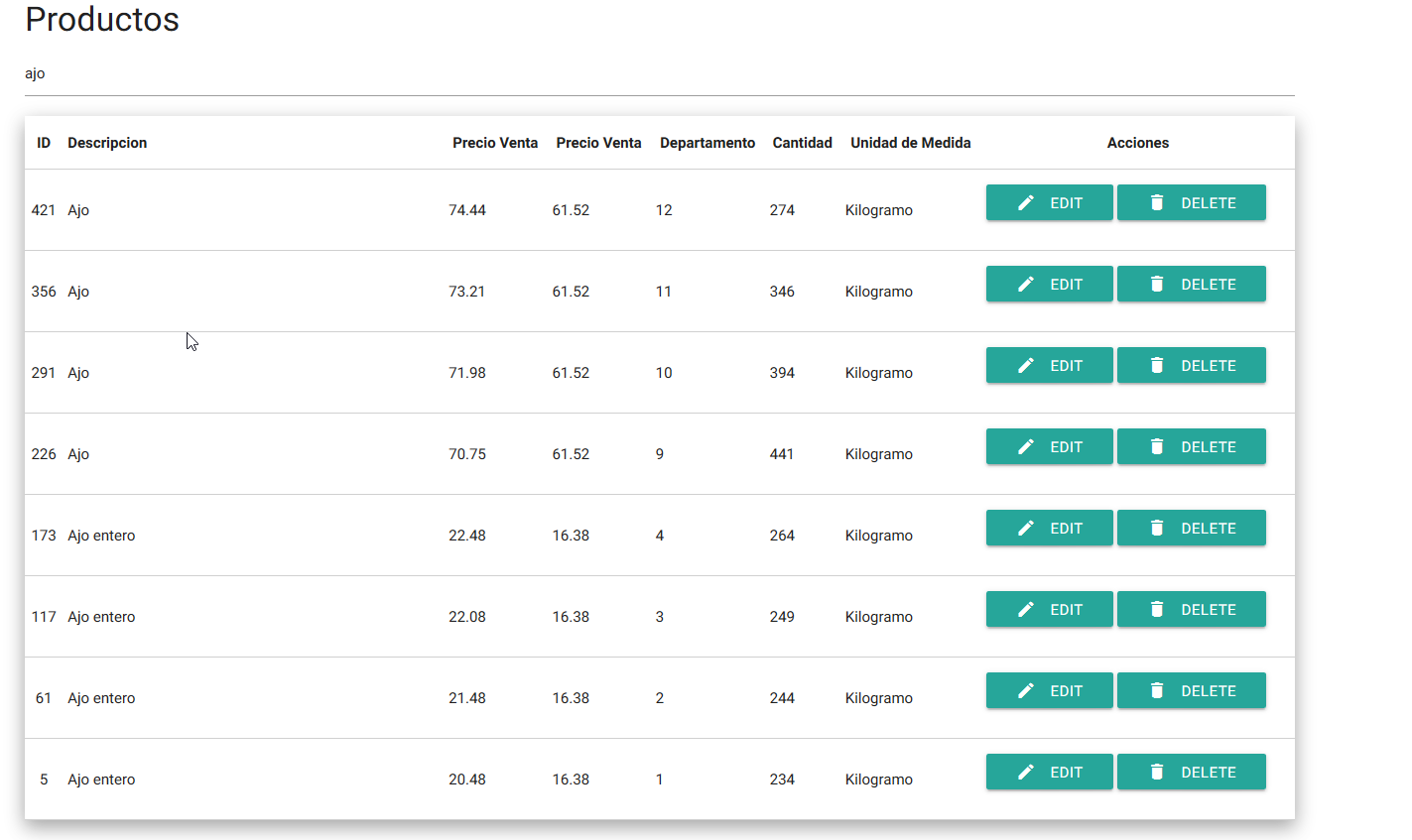
En esta área es donde se encuentra el área de trabajo de la aplicación, en esta parte se podrían colocar las capturas de productos, así como las funcionalidades de compras a proveedores y ventas a clientes, afectando de forma automática los inventarios de productos para cada Tienda Walmart.

Dentro de la parte de contenido al dar Click en el menú, Registro de producto, se tendrá acceso de forma automática a listado de productos capturados en el sistema.

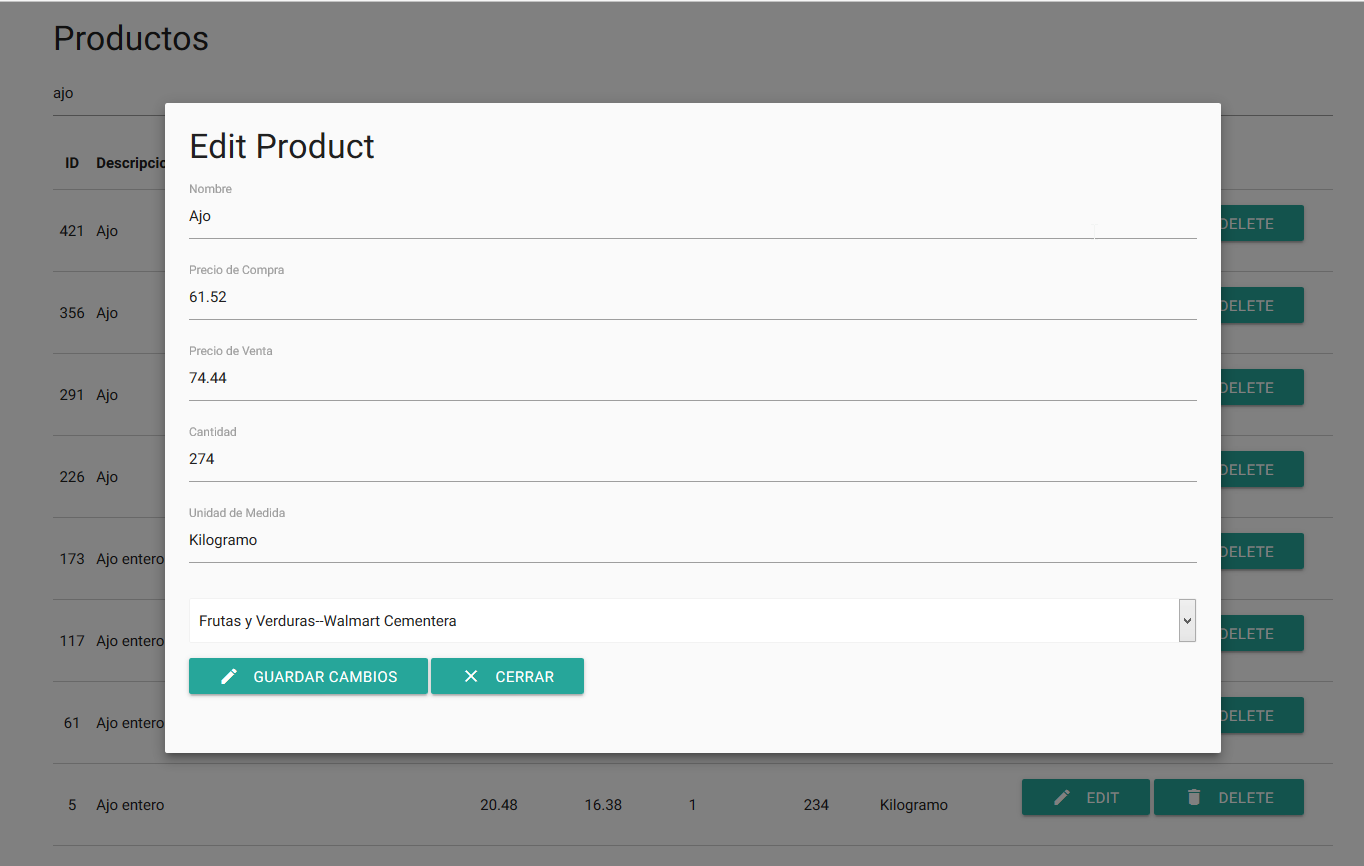
En los siguientes pasos se describe la forma en la que trabaja la aplicación en cualquiera de sus módulos tipo CRUD.



Como se pude ver en la imagen se cuenta con un área especifica para realizar búsquedas de productos.

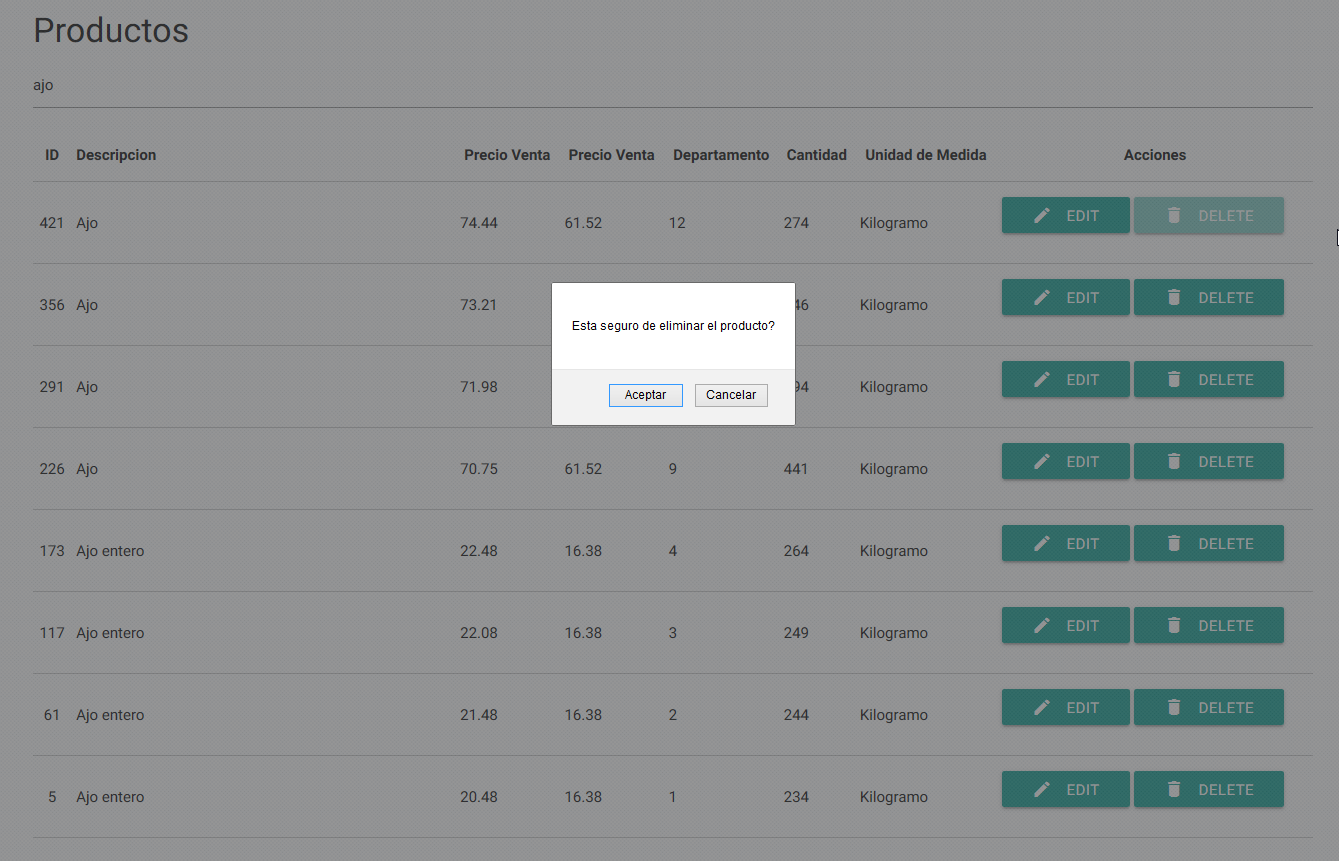


De ser requerido se podrá editar un registro para actualizar la información relevante del producto.



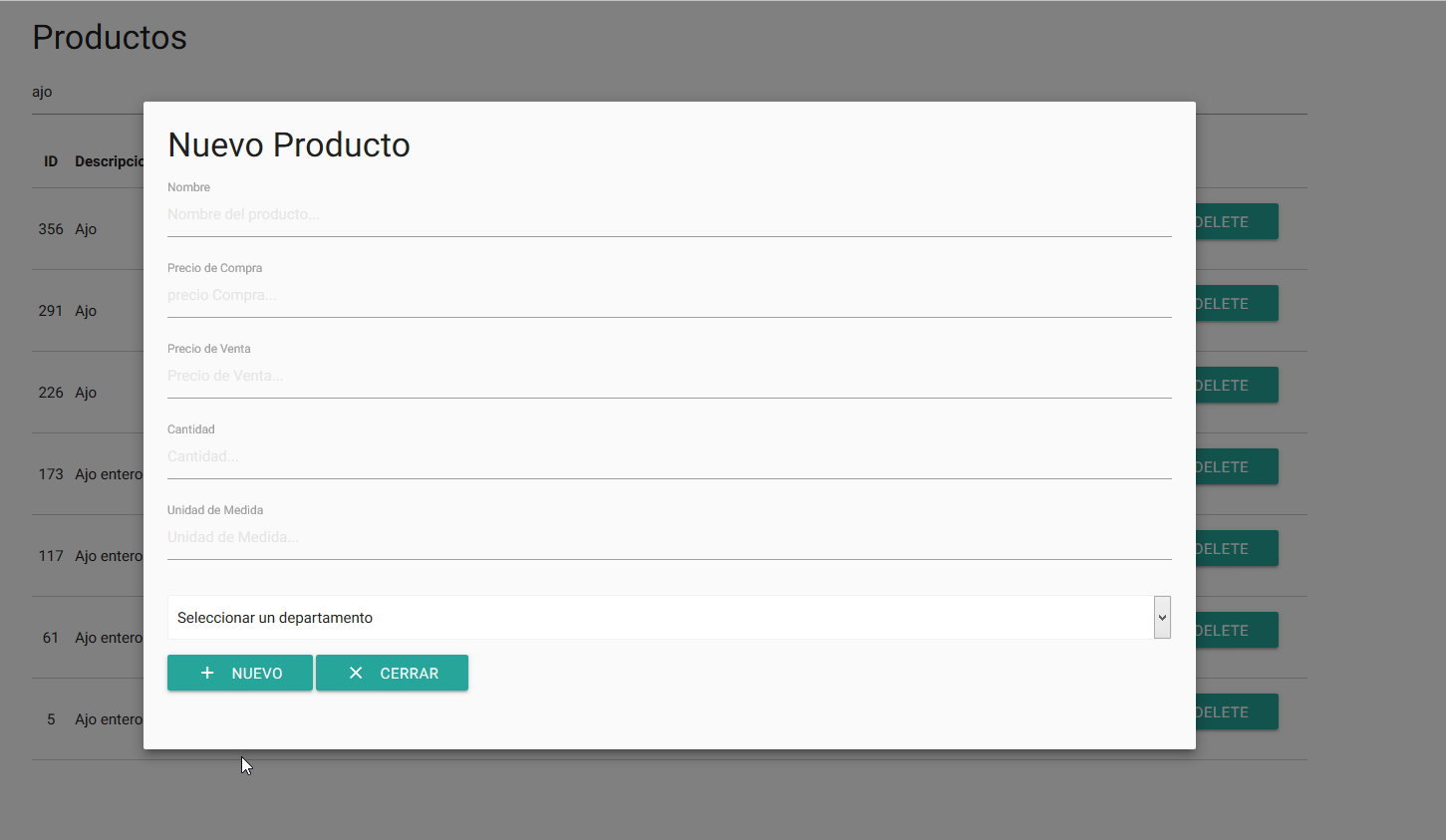
Al dar click en el botón Guardar cambios la información se almacenara de forma permanente en la BD.

En caso de ser requerido se cuenta con un botón delete, con el cual se podrá borrar el producto deseado.



El sistema como se puede ver en la imagen solicitara confirmación

En caso de contar con un producto nuevo se podrán agregar tantos productos como sean necesarios mediante el botón + .



Una vez que presionamos el botón NUEVO el registro quedara almacenado de forma permanente en el sistema.

En caso de que ya no sea requerido agregar el nuevo registro será suficiente con presionar el botón CERRAR en ese momento la pantalla de captura se cerrara y continuaremos trabajando de forma normal.

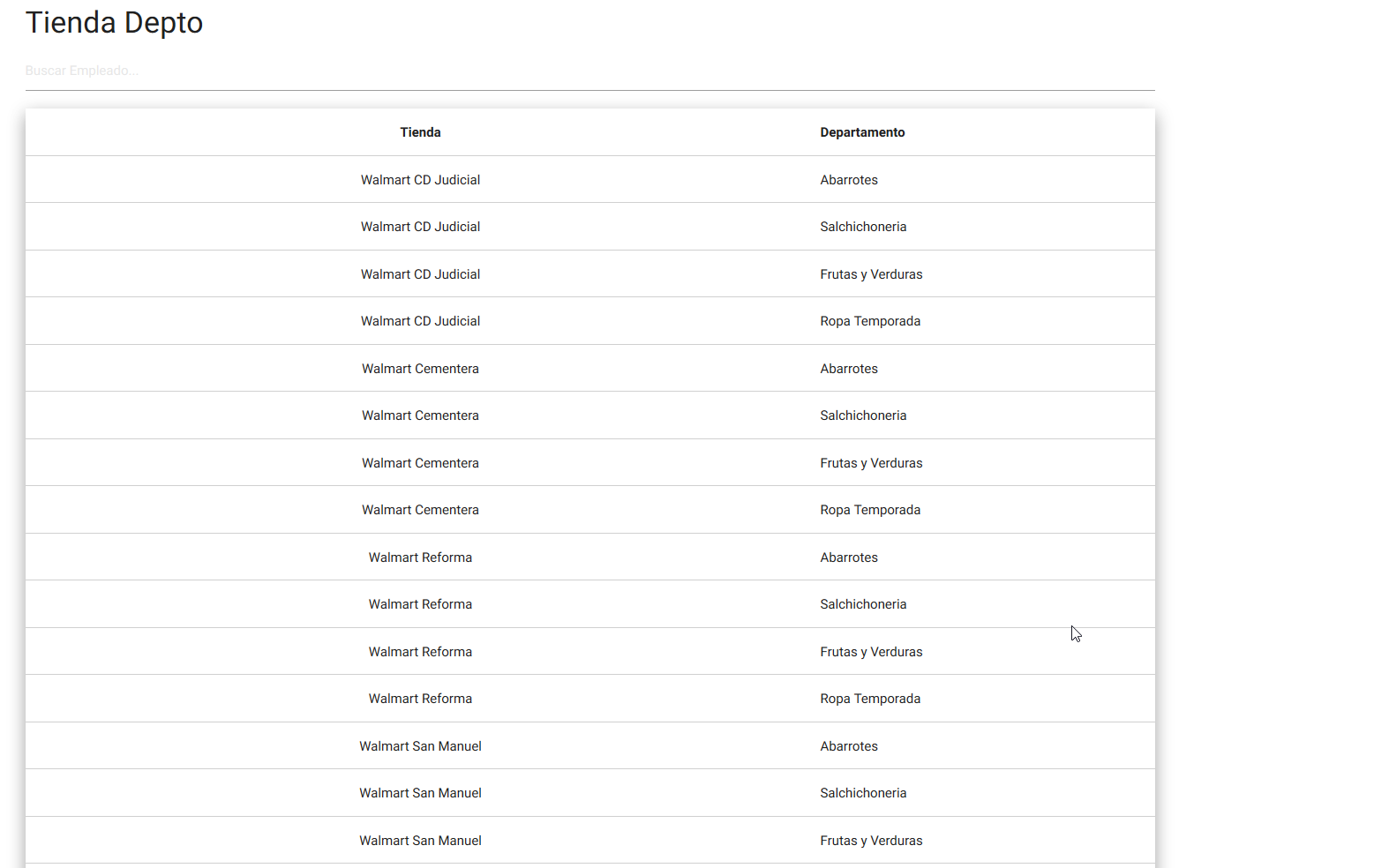
## Explicación de las "consultas interesantes"

## Consulta interesantes 1

Consulta que me dice con que departamentos cuenta cada sucursal

select t.nombre as nombreTienda, d.nombre nombreDepartamento from tienda t join departamento d on t.id = d.tienda\_id order by t.nombre;

Explicación: Como se pude ver se trata de un join entre las tablas relacionadas en este caso es la entidad tienda y departamento.



## Consulta interesante 2

Consulta para saber cuáles son los 10 productos que se tiene más inventario

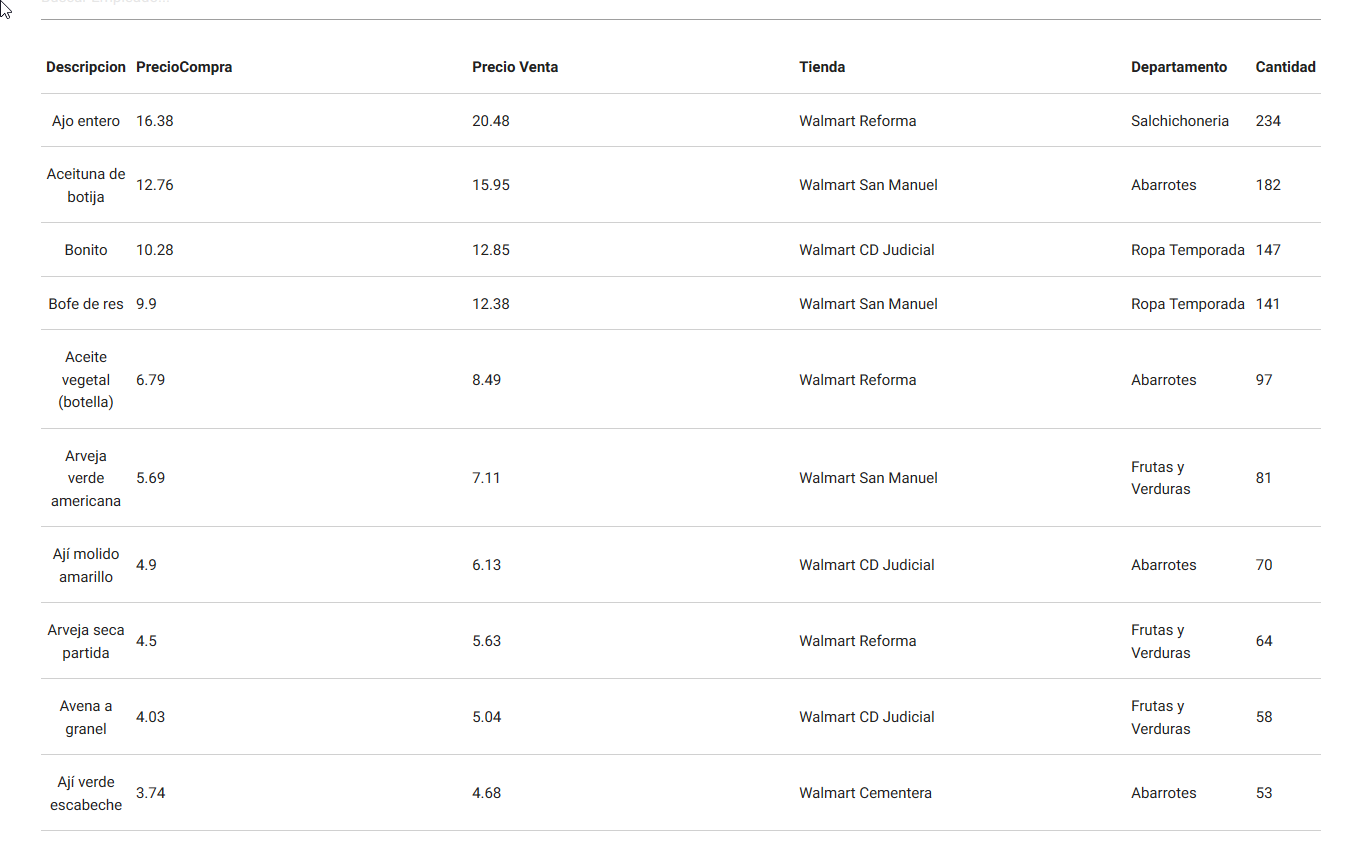
select p.descripcion, p.precioCompra,p.precioVenta, t.nombre tienda,d.nombre departamento, p.cantidad from productos p

join departamento d on d.id = p.id

join tienda t on t.id = d.tienda\_id

order by p.cantidad desc LIMIT 0, 10 ;

Explicación: Como se pude ver se trata de una consulta que involucra tiendas, departamentos y productos con el fin de saber cuáles son los productos más caros por tienda.



## Consulta interesante 3

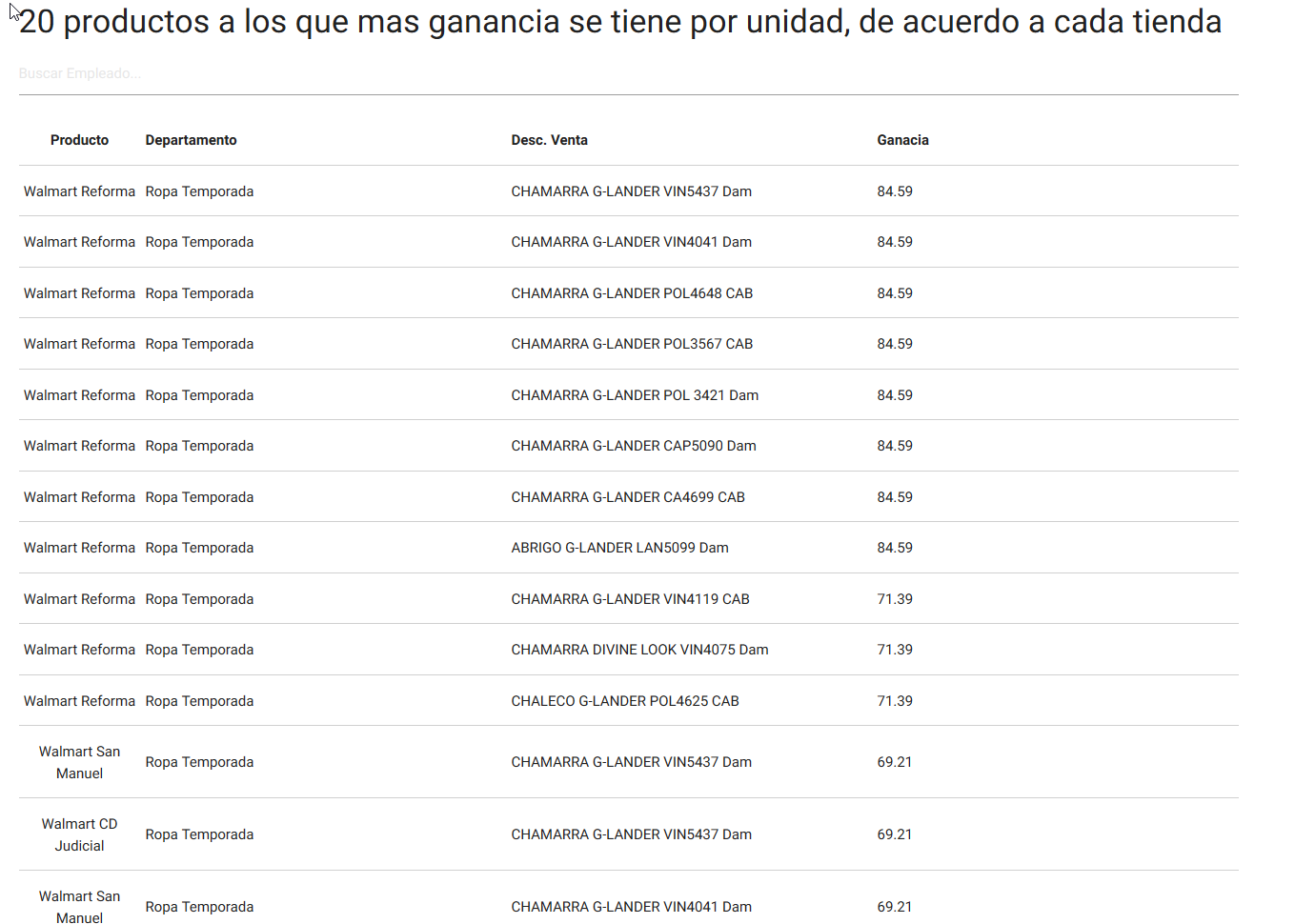
Consulta para saber que cuales son los 20 productos a los que mas ganancia se tiene por unidad, de acuerdo a cada tienda

select distinct t.nombre, d.nombre, p.descripcion, round(p.precioVenta-p.precioCompra,2) from productos p join departamento d on p.departamento\_id = d.id

join tienda t on t.id = d.tienda\_id

order by 4 desc, 3 desc limit 0,20;

**Explicación:** Como se pude ver se trata de una consulta que involucra tiendas, departamentos y productos se determina la ganancia mediante la formula precioVenta-p.precioCompra



## Consulta interesante 4

Consulta para saber en que tienda y departamento se tienen los 20 productos de los que se obtiene menos ganancia

select distinct t.nombre, d.nombre, p.descripcion, round(p.precioVenta-p.precioCompra,2) from productos p join departamento d on p.departamento\_id = d.id

join tienda t on t.id = d.tienda\_id

order by 4 asc, 3 desc limit 0,20;

Explicación: Como se pude ver se trata de una consulta que involucra tiendas, departamentos y productos se determina la ganancia mediante la formula precioVenta-p.precioCompra y se ordena de forma ascendente de modo que los primeros registros son los que generan menor ganancia



## Consulta interesante 5

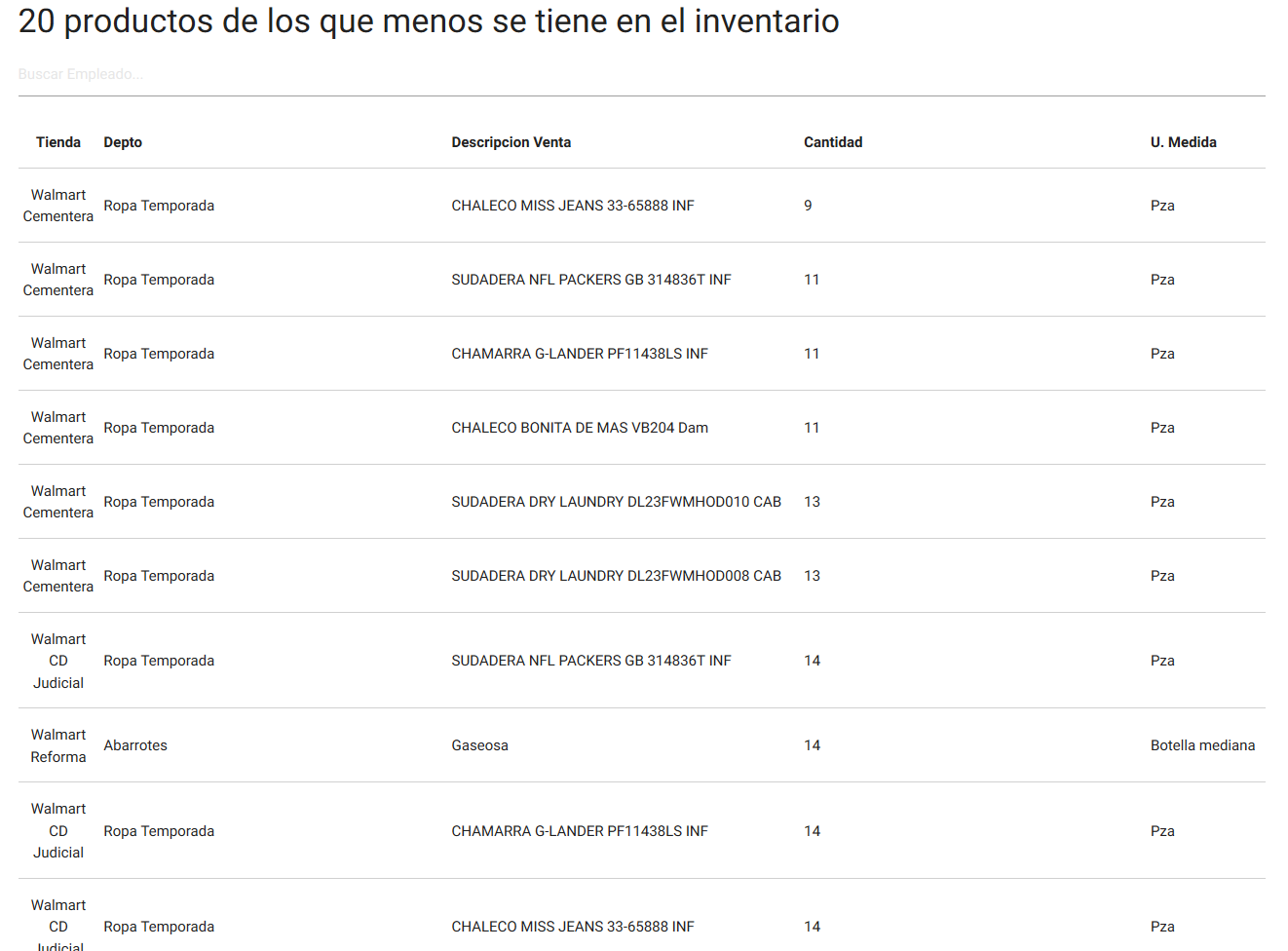
Consulta para saber cuales son los 20 productos de los que menos se tiene en el inventario

select distinct t.nombre, d.nombre, p.descripcion, p.cantidad , p.unidadDeMedida from productos p join departamento d on p.departamento\_id = d.id

join tienda t on t.id = d.tienda\_id

order by 4 asc, 3 desc limit 0,20;

Explicación: Como se pude ver se trata de una consulta que involucra tiendas, departamentos y productos se determina el inventario de acuerdo al campo cantidad, y se ordena de forma ascendente de modo que los primeros son los productos con menos inventario.



## Consulta interesante 6

query para saber cuáles son los 20 productos de los que mas se tiene en el inventario \*/

select distinct t.nombre, d.nombre, p.descripcion, p.cantidad , p.unidadDeMedida from productos p join departamento d on p.departamento\_id = d.id

join tienda t on t.id = d.tienda\_id

order by 4 desc, 3 desc limit 0,20;

Explicación: Como se pude ver se trata de una consulta que involucra tiendas, departamentos y productos se determina el inventario de acuerdo al campo cantidad, y se ordena de forma ascendente de modo que los primeros son los productos con menos inventario.

## Consulta interesante 7

Consulta pasa saber cuál es el activo de inversión que se tiene por tienda de acuerdo a una unidad de inventario de cada tienda

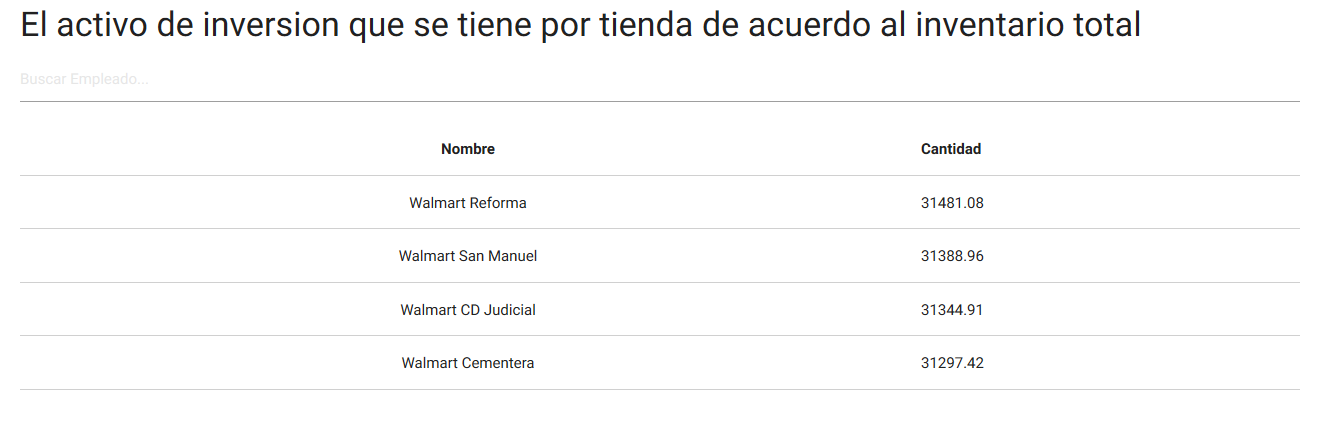
select distinct t.nombre, round(sum(p.precioCompra),2) from productos p join departamento d on p.departamento\_id = d.id

join tienda t on t.id = d.tienda\_id

group by 1

order by 2 desc;

Explicación: Como se pude ver se trata de una consulta que involucra tiendas, departamentos y productos se determina el total de productos y se sumariza cada uno de los costos de los productos para tener un gran total.



## Consulta interesante 8

Consulta para saber cuál será la ganancia total de artitulo tienda de acuerdo los productos de su inventario

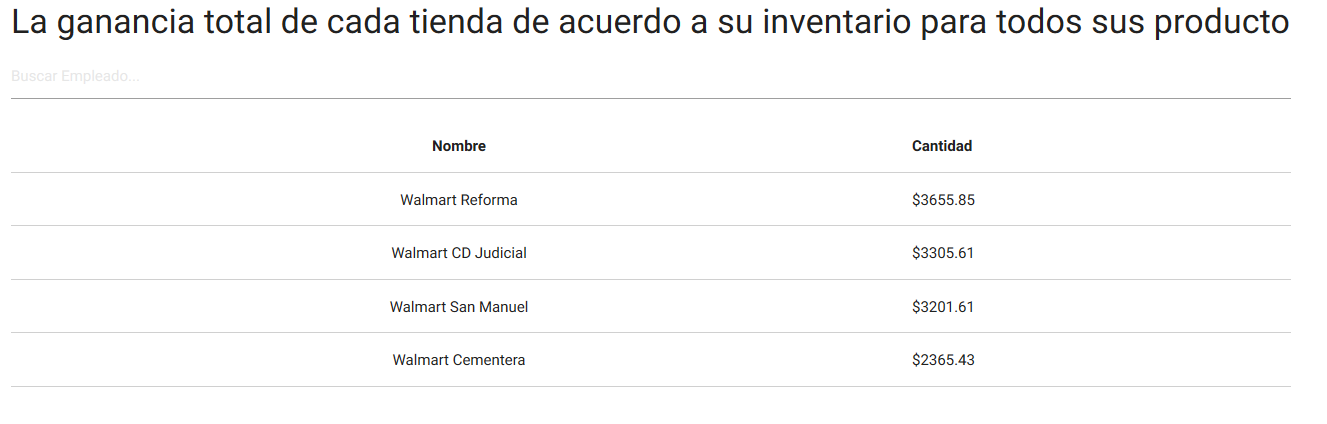
select distinct t.nombre, round(sum(p.precioVenta-p.precioCompra),2) from productos p join departamento d on p.departamento\_id = d.id

join tienda t on t.id = d.tienda\_id

group by 1

order by 2 desc;

Explicación: Como se pude ver se trata de una consulta que involucra tiendas, departamentos y productos se determina el total de productos y se sumariza la gananancia(p.precioVenta-p.precioCompra) cada uno de los costos de los productos para tener un gran total.



## Consulta interesante 9

Consulta para saber cuáles son los empleados de cada tienda y el puesto ordenados por tienda y por puesto

select distinct t.nombre Tienda, e.puesto, p.nombre, p.apat, p.amat from empleado e join persona p on e.persona\_id = p.id

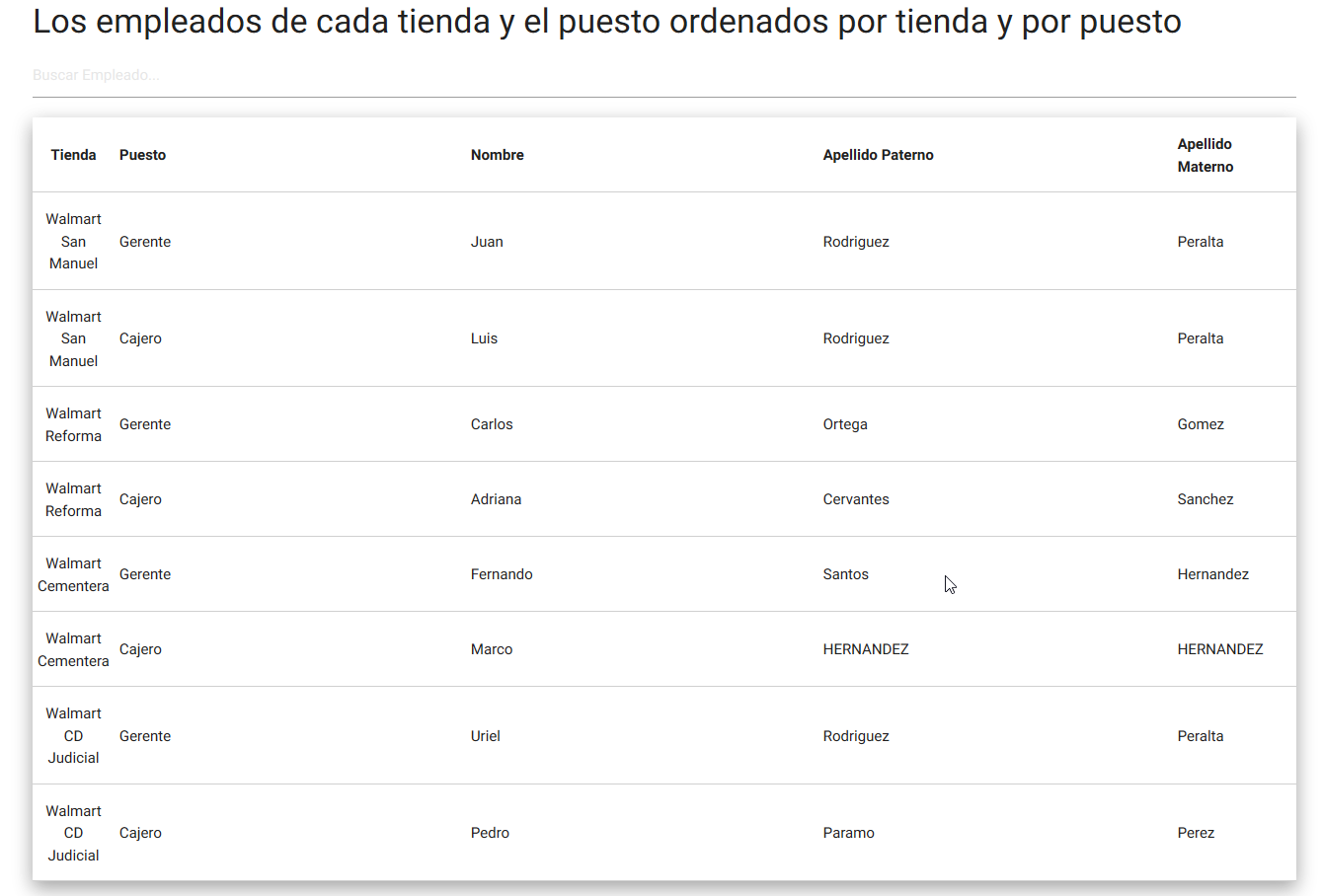
join tienda t on t.id = e.tienda\_id

group by 1,2

order by 1 desc, 2 desc;

p.precioVenta-p.precioCompra

Explicación: Como se pude ver se trata de una consulta que involucra empleados, personas y tiendas y se presentan



## Consulta interesante 10

Consulta pasa saber los emails todos los gerentes de tiendas ordenados por nombre ascendentemente

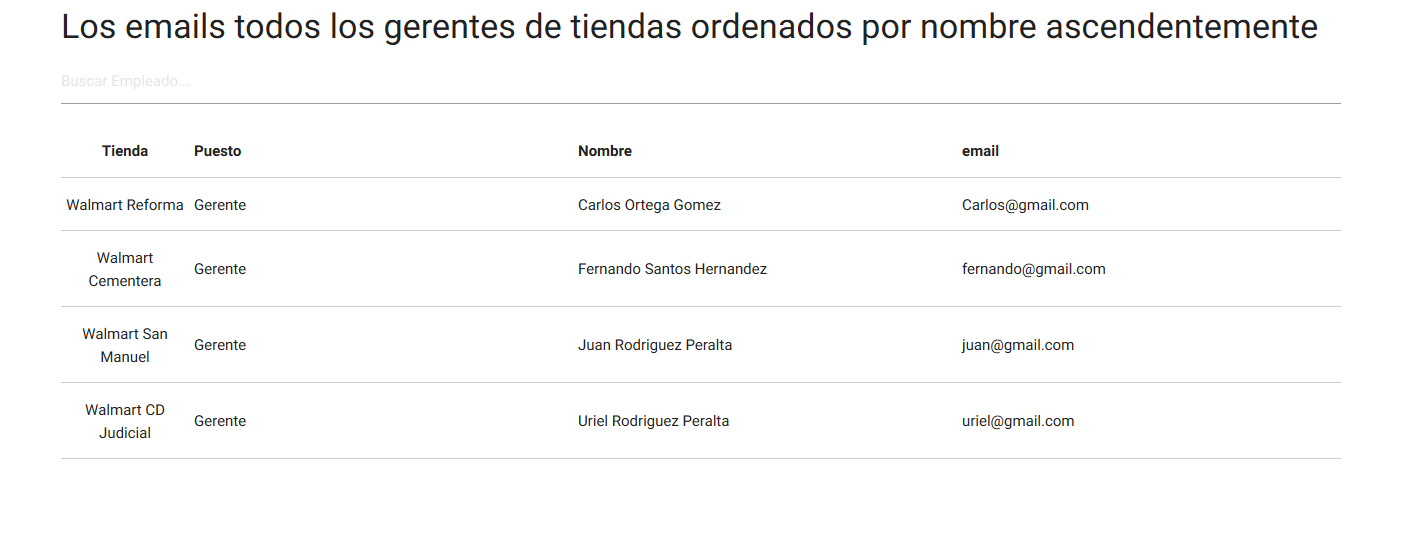
select distinct t.nombre Tienda, e.puesto, concat( p.nombre,' ', p.apat,' ', p.amat) nombre , p.email from empleado e join persona p on e.persona\_id = p.id

join tienda t on t.id = e.tienda\_id

where puesto= 'Gerente'

order by 3 asc;

Explicación: Como se pude ver se trata de una consulta que involucra empleados, personas y tiendas y se presentan filtrados por puesto= 'Gerente'



## Consulta interesante 11

Consulta que nos sirve para saber que clientes existen en sistema y así como saber en qué tiendas y hacen compras o no y sus emails

select distinct t.nombre Tienda, concat( p.nombre,' ', p.apat,' ', p.amat) nombreCliente , p.email emailCliente

from cliente c

left join persona p on c.persona\_id = p.id

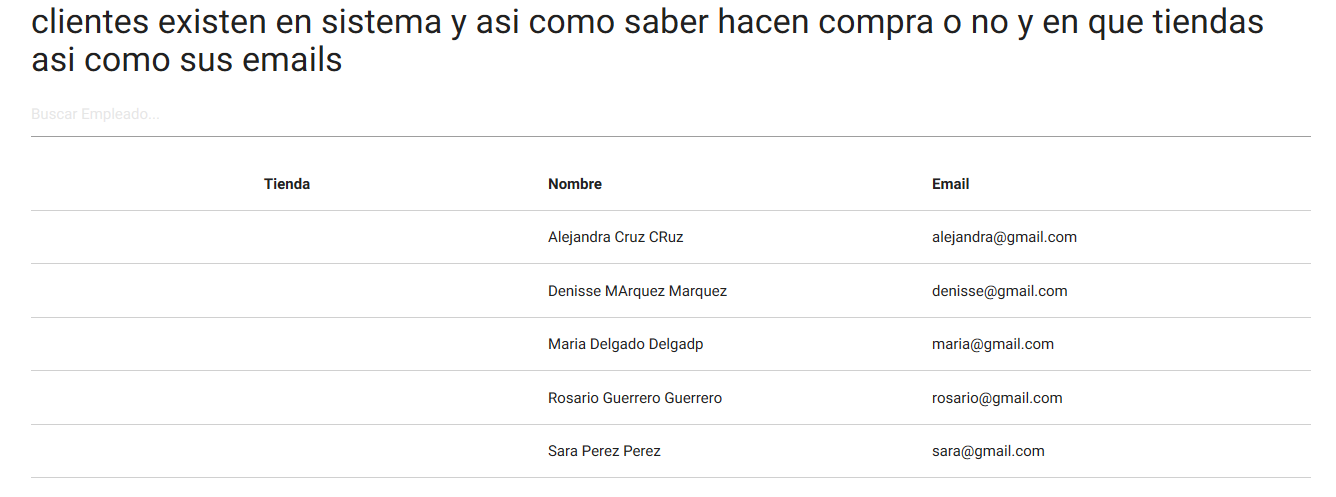
left join compra com on com.cliente\_id = c.id

left join empleado emp on empleado\_id = com.empleado\_id

left join tienda t on t.id = emp.tienda\_id

group by 2,1;

Explicación: Como se pude ver se trata de una consulta que involucra clientes, personas, compras empleados y tiendas se realizan los join entre las tablas correspondientes y se presenta la información nombre de tienda, nombre del cliente y email del cliente de cada una de las tablas involucradas



## Consulta interesante 12

Consulta que nos sirve para saber que proveedores están en sistema adicionalmente si venden en que tiendas y sus emails

select distinct t.nombre Tienda, concat( p.nombre,' ', p.apat,' ', p.amat) nombreProvedor , c.razonSocial, p.email emailProvedor

from proveedor c

left join persona p on c.persona\_id = p.id

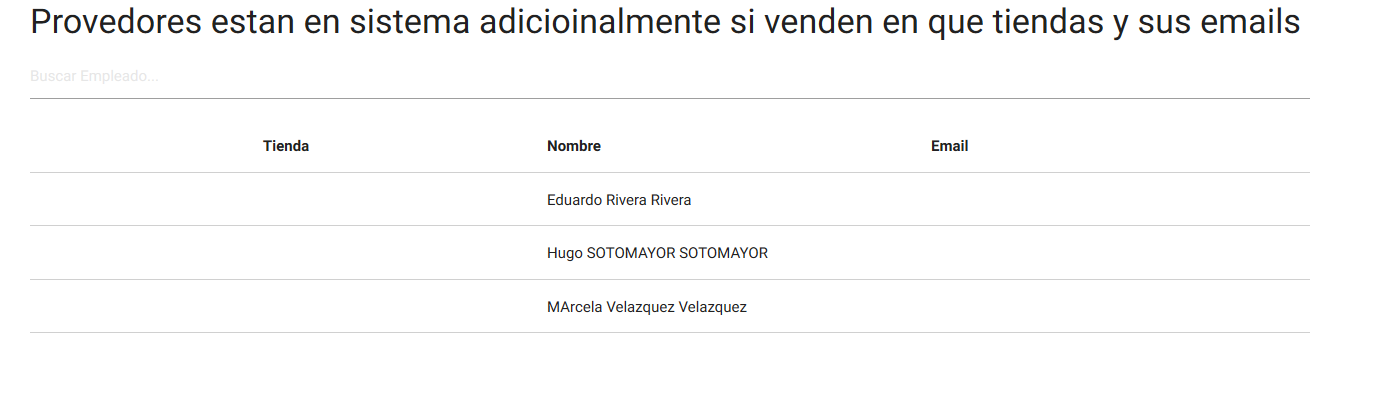
left join venta com on com.proveedor\_id = c.id

left join empleado emp on empleado\_id = com.empleado\_id

left join tienda t on t.id = emp.tienda\_id

group by 2,1;

Explicación: Como se pude ver se trata de una consulta que involucra proveedores, personas, compras empleados y tiendas se realizan Left joins entre las tablas correspondientes y se presenta la información nombre de tienda, nombre del nombreProvedor , razonSocial y emailProvedor del de cada una de las tablas involucradas quiero hacer notar aquí que el Left join nos ayuda a poder tener el despliegue de todos los proveedores aun que no participen en el comercio con las tiendas es decir están dados de alta en el sistema pero nunca han vendido anda a las tiendas.



## Conclusiones del equipo

Después de haber recopilado información para poblar la BD se obtuvo la conclusión de que el inventario es esencial en la creación y existencia de una empresa, ya que el inventario con ayuda de una buena Base de datos en ella podemos hacer un balance de los bienes que cuenta esta.

La carencia de un inventario en una empresa conllevaría a la quiebra de la empresa, debido a que habría una desorganización, a consecuencia de las carencias o excesos de bienes se tomarían malas decisiones en el manejo de la empresa.

Es decir podríamos tener un exceso de inventario de productos que los clientes no compran lo cual haría que se tenga inversión parada y podríamos tener pocos artículos a ninguno de los productos que los clientes consumen o comprar con mayor frecuencia en conclusión cada producto buscado por un cliente y no encontrado por problemas de inventario es perdida para la empresa.