**Universidad Autónoma de Zacatecas “J. Jesús González Ortega”**

Unidad Académica de Ingeniería Eléctrica

Programa Académico de Ingeniería de Software.

**Nombre de la materia:** Sistema de base de datos I

**Nombre de la carrera:** Ingeniería de Software

**Título:** Proyecto final: Sistemas de base de datos I

**Nombre del alumno:** Jesús Manuel Juárez Pasillas

**Nombre del docente:** Gustavo Zepeda Valles

**Tema:** Tienda en línea.

**Fecha:** 29/05/2021

Tabla de contenido

[**Actividad 1:** 3](#_Toc73284590)

[Introducción: 3](#_Toc73284591)

[Descripción: 3](#_Toc73284592)

[Justificación: 3](#_Toc73284593)

[**Actividad 2:** 4](#_Toc73284594)

[Requerimientos de negocio. 4](#_Toc73284595)

[**Actividad 3:** 5](#_Toc73284596)

[Diagrama Entidad-Relación. 5](#_Toc73284597)

[**Actividad 5:** 7](#_Toc73284598)

[Sentencias SQL–DLL para la creación de las tablas y las relaciones entre las mismas. 7](#_Toc73284599)

[**Actividad 6:** 11](#_Toc73284600)

[Sentencias SQL–DML para insertar por lo menos 5 registros por cada tabla. 11](#_Toc73284601)

[**Actividad 7:** 15](#_Toc73284602)

[Sentencias SQL–DML para modificar 10 datos de la Base de Datos. 15](#_Toc73284603)

[**Actividad 8:** 16](#_Toc73284604)

[Sentencias SQL–DML para eliminar 10 datos de la Base de Datos. 16](#_Toc73284605)

[**Actividad 9:** 17](#_Toc73284606)

[Redactar 25 enunciados para consultar datos y elaborar las sentencias SQL–DML necesarias para dar respuesta a dichos enunciados. 17](#_Toc73284607)

# **Actividad 1:**

## Introducción:

Para este proyecto se tomará algún tema de interés (en este caso una tienda de productos diversos en línea), dado el tema se sacarán los respectivos requerimientos con los que se irán creado el modelo entidad-relación y con este crear el modelo relacional (mínimo 5 registros por tabla) y hacer el esquema correspondiente. Revisar que el modelo cumpla con la 3 forma normal y si no es así hacer los ajustes necesarios. Luego de eso se crearán las sentencias SQL-DLL con las que se crearán las tablas y las relaciones que esta tendría. Luego de esto hacer las sentencias SQL-DML para insertar datos en las tablas, luego otras sentencias con las que se modifiquen 10 datos de la base de datos y por último redactar 25 enunciados para consultar los datos y desarrollar las sentencias SQL-DML para cada enunciado.

## Descripción:

El tema del proyecto que se estará evaluando será el de una tienda en línea la cual tiene una gran variedad de productos los cuales son necesarios administrar, estos productos serán almacenados en los diferentes estados del país. Esta tienda también contara con un sistema para que los usuarios se puedan identificar además de almacenar datos importantes de los usuarios.

El sistema solo estará disponible para personas residentes del país de México.

## Justificación:

Una tienda la cual vende una gran variedad de productos a decidió abrir un sitio web para poner a la venta sus productos en dicho sitio y poder ofrecer un servicio a domicilio. Con esto planea aumentar sus ganancias, además de permitir a sus compradores hacerlo desde cualquier lugar y que los productos le lleguen a su casa.

# **Actividad 2:**

## Requerimientos de negocio.

El negocio conocido como “Bodega soriana” requiere de un sitio web con el cual poder ofrecer sus productos a sus usuarios para que las compras que estos hagan las puedan hacer desde cualquier lugar y los productos se lleven directo a una dirección dada por el usuario.

Para lograr esto se necesita saber la información de los productos que se están vendiendo, así como información del cliente que esta comprando dicho producto. Como los productos se estarán almacenando en bodegas en diferentes puntos del país es necesario saber información de dichas bodegas y que producto esta almacenado en que bodega.

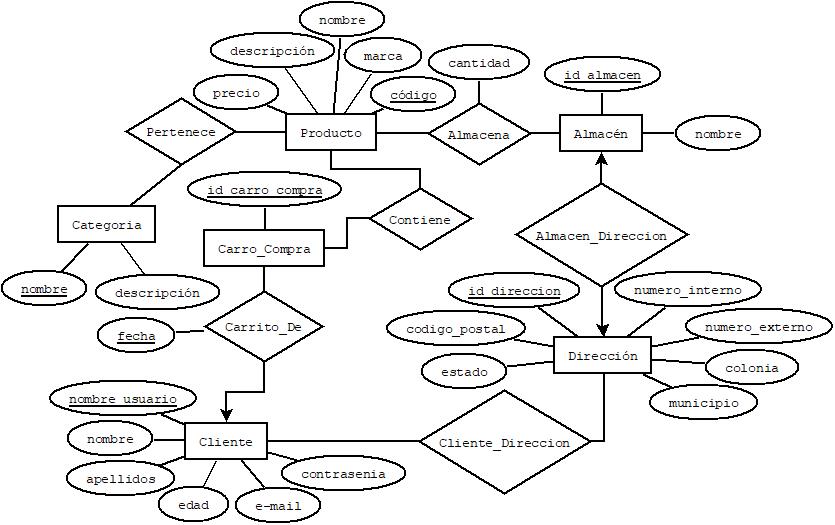
Los clientes se tienen que registrar con un nombre de usuario y una contraseña, además de esto se tendrá que poner su nombre completo y la dirección a la cual van a llegar los productos.

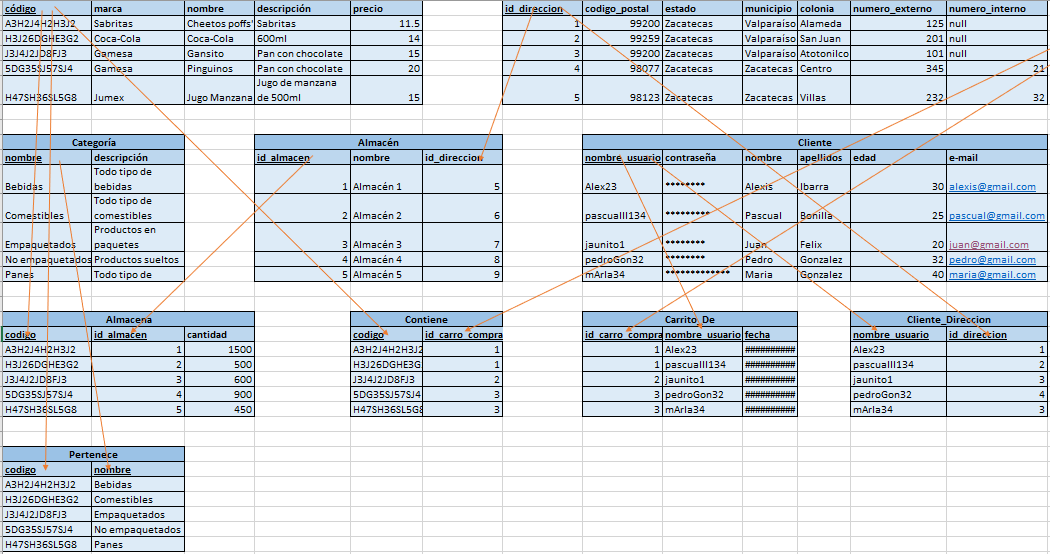
La información que se tendrá de los productos será la marca, nombre y precio del producto, además del tipo de producto que es para poder ordenarlos por categoría. También se requiere saber la cantidad del mismo producto almacenada. Los productos pueden estar almacenados en diferentes almacenes.

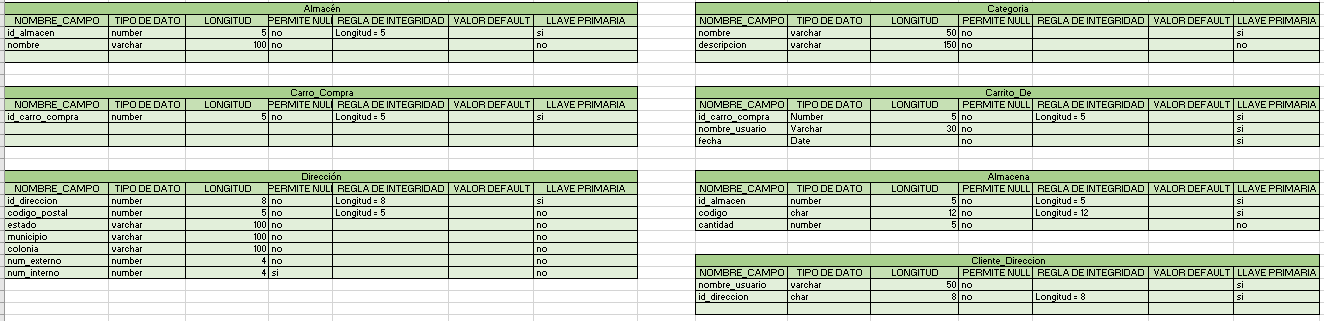
Los almacenes se requiere guardar la información de la dirección en la que se encuentra.

# **Actividad 3:**

## Diagrama Entidad-Relación.









# **Actividad 5:**

## Sentencias SQL–DLL para la creación de las tablas y las relaciones entre las mismas.

**TABLA DIRECCIONES**

CREATE TABLE DIRECCIONES(

ID\_DIRECCION NUMBER(8) CONSTRAINT DIRECCION\_PK PRIMARY KEY,

CODIGO\_POSTAL NUMBER(5) CONSTRAINT DIR\_COD\_POS\_NN NOT NULL,

ESTADO VARCHAR(100) CONSTRAINT DIR\_EST\_NN NOT NULL,

MUNICIPIO VARCHAR(100) CONSTRAINT DIR\_MUN\_NN NOT NULL,

COLONIA VARCHAR(100) CONSTRAINT DIR\_COL\_NN NOT NULL,

NUM\_EXTERNO NUMBER(4) CONSTRAINT DIR\_NUM\_EXT\_NN NOT NULL,

NUM\_INTERNO NUMBER(4)

);

**TABLA CLIENTES**

CREATE TABLE CLIENTES(

NOMBRE\_USUARIO VARCHAR(50) CONSTRAINT CLIENTE\_PK PRIMARY KEY,

CONTRASENIA VARCHAR(50) CONSTRAINT CLI\_CON\_NN NOT NULL,

NOMBRE VARCHAR(50) CONSTRAINT CLI\_NOM\_NN NOT NULL,

APELLIDO VARCHAR(100) CONSTRAINT CLI\_APE\_NN NOT NULL,

EDAD NUMBER(2),

EMAIL VARCHAR(150) CONSTRAINT CLI\_EM\_NN NOT NULL

);

**TABLA CLIENTE\_DIRECCIONES**

CREATE TABLE CLIENTE\_DIRECCIONES(

NOMBRE\_USUARIO VARCHAR(50),

ID\_DIRECCION NUMBER(8),

CONSTRAINT CLI\_DIR\_PK PRIMARY KEY (NOMBRE\_USUARIO,ID\_DIRECCION),

CONSTRAINT CLI\_CLI\_DIR\_FK FOREIGN KEY (NOMBRE\_USUARIO) REFERENCES CLIENTES(NOMBRE\_USUARIO),

CONSTRAINT DIR\_CLI\_DIR\_FK FOREIGN KEY (ID\_DIRECCION) REFERENCES DIRECCIONES(ID\_DIRECCION)

);

**TABLA ALMACENES**

CREATE TABLE ALMACENES(

ID\_ALMACEN NUMBER(5) CONSTRAINT ALMACENES\_PK PRIMARY KEY,

NOMBRE VARCHAR(100) CONSTRAINT ALM\_NOM\_NN NOT NULL,

ID\_DIRECCION NUMBER(8) CONSTRAINT ALM\_DIR\_NN NOT NULL,

CONSTRAINT DIR\_ALM\_FK FOREIGN KEY (ID\_DIRECCION) REFERENCES DIRECCIONES(ID\_DIRECCION)

);

**TABLA PRODUCTOS**

CREATE TABLE PRODUCTOS(

CODIGO CHAR(12) CONSTRAINT PRODUCTO\_PK PRIMARY KEY,

MARCA VARCHAR(50) CONSTRAINT PRO\_MAR\_NN NOT NULL,

NOMBRE VARCHAR(100) CONSTRAINT PRO\_NOM\_NN NOT NULL,

DESCRIPCION VARCHAR(150),

PRECIO NUMBER(7,2) CONSTRAINT PRO\_PRE\_NN NOT NULL

);

**TABLA ALMACENA**

CREATE TABLE ALMACENA(

ID\_ALMACEN NUMBER(5),

CODIGO CHAR(12),

CANTIDAD NUMBER(5) CONSTRAINT ALM\_CAN\_NN NOT NULL,

CONSTRAINT ALMACENA\_PK PRIMARY KEY (ID\_ALMACEN,CODIGO),

CONSTRAINT ALM\_ALM\_FK FOREIGN KEY (ID\_ALMACEN) REFERENCES ALMACENES(ID\_ALMACEN),

CONSTRAINT PRO\_ALM\_FK FOREIGN KEY (CODIGO) REFERENCES PRODUCTOS(CODIGO)

);

**TABLA CATEGORIAS**

CREATE TABLE CATEGORIAS(

NOMBRE VARCHAR(50) CONSTRAINT CATEGORIA\_PK PRIMARY KEY,

DESCRIPCION VARCHAR(150) CONSTRAINT CAT\_DES\_NN NOT NULL

);

**TABLA PERTENECE**

CREATE TABLE PERTENECE(

NOMBRE VARCHAR(50),

CODIGO CHAR(12),

CONSTRAINT PERTENECE\_PK PRIMARY KEY (NOMBRE,CODIGO),

CONSTRAINT CAT\_PER\_FK FOREIGN KEY (NOMBRE) REFERENCES CATEGORIAS(NOMBRE),

CONSTRAINT PRO\_PER\_FK FOREIGN KEY (CODIGO) REFERENCES PRODUCTOS(CODIGO)

);

**TABLA CARROS\_COMPRAS**

CREATE TABLE CARROS\_COMPRAS(

ID\_CARRO\_COMPRA NUMBER(5) CONSTRAINT CARRO\_COMPRA\_PK PRIMARY KEY

);

**TABLA CONTIENE**

CREATE TABLE CONTIENE(

CODIGO CHAR(12),

ID\_CARRO\_COMPRA NUMBER(5),

CONSTRAINT CONTIENE\_PK PRIMARY KEY (CODIGO,ID\_CARRO\_COMPRA),

CONSTRAINT PRO\_CON\_FK FOREIGN KEY (CODIGO) REFERENCES PRODUCTOS(CODIGO),

CONSTRAINT CARR\_COM\_CON\_FK FOREIGN KEY (ID\_CARRO\_COMPRA) REFERENCES CARROS\_COMPRAS(ID\_CARRO\_COMPRA)

);

**TABLA CARRITO\_DE**

CREATE TABLE CARRITO\_DE(

NOMBRE\_USUARIO VARCHAR(50),

ID\_CARRO\_COMPRA NUMBER(5),

FECHA DATE,

CONSTRAINT CARRITO\_DE\_PK PRIMARY KEY (NOMBRE\_USUARIO,ID\_CARRO\_COMPRA,FECHA),

CONSTRAINT CLI\_CARR\_FK FOREIGN KEY (NOMBRE\_USUARIO) REFERENCES CLIENTES(NOMBRE\_USUARIO),

CONSTRAINT CARR\_COM\_CARR\_FK FOREIGN KEY (ID\_CARRO\_COMPRA) REFERENCES CARROS\_COMPRAS(ID\_CARRO\_COMPRA)

);

# **Actividad 6:**

## Sentencias SQL–DML para insertar por lo menos 5 registros por cada tabla.

**TABLA DIRECCIONES**

INSERT INTO DIRECCIONES VALUES(1,99200,'ZACATECAS','VALPARAÍSO','ALAMEDA',125,NULL);

INSERT INTO DIRECCIONES VALUES(2,99259,'ZACATECAS','VALPARAÍSO','SAN JUAN',201,NULL);

INSERT INTO DIRECCIONES VALUES(3,99200,'ZACATECAS','VALPARAÍSO','ATOTONILCO',101,NULL);

INSERT INTO DIRECCIONES VALUES(4,98077,'ZACATECAS','ZACATECAS','CENTRO',345,21);

INSERT INTO DIRECCIONES VALUES(5,98123,'ZACATECAS','ZACATECAS','VILLAS',232,32);

INSERT INTO DIRECCIONES VALUES(6,99200,'ZACATECAS','VALPARAÍSO','VILLAS DEL SOL',111,NULL);

INSERT INTO DIRECCIONES VALUES(7,99200,'ZACATECAS','VALPARAISO','FLORES MAGON',194,23);

INSERT INTO DIRECCIONES VALUES(8,98123,'ZACATECAS','ZACATECAS','BENITO JUAREZ',169,NULL);

INSERT INTO DIRECCIONES VALUES(9,98099,'ZACATECAS','ZACATECAS','ZAPATA',147,NULL);

INSERT INTO DIRECCIONES VALUES(10,99200,'ZACATECAS','VALPARAISO','ZAPATA',200,NULL);

**TABLA CLIENTES**

INSERT INTO CLIENTES VALUES('ALEX23','32XELA','ALEXIS','IBARRA',30,'ALEXIS@GMAIL.COM');

INSERT INTO CLIENTES VALUES('pascualll134','ISDB782YHE','Pascual','Bonilla',25,'pascual@gmail.com');

INSERT INTO CLIENTES VALUES('jaunito1','IUKGF75UFU','Juan','Felix',20,'juan@gmail.com');

INSERT INTO CLIENTES VALUES('pedroGon32','57UTF75TUFTV','Pedro','Gonzalez',32,'pedro@gmail.com');

INSERT INTO CLIENTES VALUES('mArIa34','75TF7FNB86','Maria','Gonzalez',40,'maria@gmail.com');

INSERT INTO CLIENTES VALUES('FDKJBV','AOKVFI2','FERNANDO','ESCALANTE',20,'FERESQ@GMAIL.COM');

**TABLA CLIENTE\_DIRECCIONES**

INSERT INTO CLIENTE\_DIRECCIONES VALUES('ALEX23',1);

INSERT INTO CLIENTE\_DIRECCIONES VALUES('pascualll134',2);

INSERT INTO CLIENTE\_DIRECCIONES VALUES('jaunito1',3);

INSERT INTO CLIENTE\_DIRECCIONES VALUES('pedroGon32',4);

INSERT INTO CLIENTE\_DIRECCIONES VALUES('mArIa34',3);

**TABLA ALMACENES**

INSERT INTO ALMACENES VALUES(1,'ALMACEN 1',5);

INSERT INTO ALMACENES VALUES(2,'ALMACEN 2',6);

INSERT INTO ALMACENES VALUES(3,'ALMACEN 3',7);

INSERT INTO ALMACENES VALUES(4,'ALMACEN 4',8);

INSERT INTO ALMACENES VALUES(5,'ALMACEN 5',9);

**TABLA PRODUCTOS**

INSERT INTO PRODUCTOS VALUES('A3H2J4H2H3J2','Sabritas','Cheetos poffs','Sabritas',11.5);

INSERT INTO PRODUCTOS VALUES('H3J26DGHE3G2','Coca-Cola','Coca-Cola','600ml',14);

INSERT INTO PRODUCTOS VALUES('J3J4J2JD8FJ3','Gamesa','Gansito','Pan con chocolate',15);

INSERT INTO PRODUCTOS VALUES('5DG35SJ57SJ4','Gamesa','Pinguinos','Pan con chocolate',20);

INSERT INTO PRODUCTOS VALUES('H47SH36SL5G8','Jumex','Jugo Manzana','Jugo de manzana de 500ml',15);

INSERT INTO PRODUCTOS VALUES('V83GVB36GBIE','ELECTROLIT','SUERO COCO','SUERO DE COCO',20);

**TABLA ALMACENA**

INSERT INTO ALMACENA VALUES(1,'A3H2J4H2H3J2',1500);

INSERT INTO ALMACENA VALUES(2,'H3J26DGHE3G2',500);

INSERT INTO ALMACENA VALUES(3,'J3J4J2JD8FJ3',600);

INSERT INTO ALMACENA VALUES(4,'5DG35SJ57SJ4',900);

INSERT INTO ALMACENA VALUES(5,'H47SH36SL5G8',450);

**TABLA CATEGORIAS**

INSERT INTO CATEGORIAS VALUES('Bebidas','Todo tipo de bebidas');

INSERT INTO CATEGORIAS VALUES('Comestibles','Todo tipo de comestibles');

INSERT INTO CATEGORIAS VALUES('Empaquetados','Productos en paquetes');

INSERT INTO CATEGORIAS VALUES('No empaquetados','Productos sueltos');

INSERT INTO CATEGORIAS VALUES('Panes','Todo tipo de panes');

INSERT INTO CATEGORIAS VALUES('FRUTAS Y VERDURAS','TODO TIPO DE FRUTAS Y VERDURAS');

**TABLA PERTENECE**

INSERT INTO PERTENECE VALUES('Bebidas','A3H2J4H2H3J2');

INSERT INTO PERTENECE VALUES('Comestibles','H3J26DGHE3G2');

INSERT INTO PERTENECE VALUES('Empaquetados','J3J4J2JD8FJ3');

INSERT INTO PERTENECE VALUES('No empaquetados','5DG35SJ57SJ4');

INSERT INTO PERTENECE VALUES('Panes','H47SH36SL5G8');

**TABLA CARROS\_COMPRAS**

INSERT INTO CARROS\_COMPRAS VALUES(1);

INSERT INTO CARROS\_COMPRAS VALUES(2);

INSERT INTO CARROS\_COMPRAS VALUES(3);

INSERT INTO CARROS\_COMPRAS VALUES(4);

INSERT INTO CARROS\_COMPRAS VALUES(5);

**TABLA CONTIENE**

INSERT INTO CONTIENE VALUES('A3H2J4H2H3J2',1);

INSERT INTO CONTIENE VALUES('H3J26DGHE3G2',1);

INSERT INTO CONTIENE VALUES('J3J4J2JD8FJ3',2);

INSERT INTO CONTIENE VALUES('5DG35SJ57SJ4',3);

INSERT INTO CONTIENE VALUES('H47SH36SL5G8',3);

**TABLA CARRITO\_DE**

INSERT INTO CARRITO\_DE VALUES('Alex23',1,'24/05/2021');

INSERT INTO CARRITO\_DE VALUES('pascualll134',1,'24/05/2021');

INSERT INTO CARRITO\_DE VALUES('jaunito1',2,'25/05/2021');

INSERT INTO CARRITO\_DE VALUES('pedroGon32',3,'26/05/2021');

INSERT INTO CARRITO\_DE VALUES('mArIa34',3,'26/05/2021');

# **Actividad 7:**

## Sentencias SQL–DML para modificar 10 datos de la Base de Datos.

1. **MODIFICAR EL NOMBRE DEL “ALMACEN 1” A “ALMACEN 6”**

UPDATE ALMACENES SET NOMBRE ='ALMACEN 6' WHERE NOMBRE='ALMACEN 1';

1. **MODIFICAR EL NOMBRE DEL CLIENTE mArIa34**

UPDATE CLIENTES SET NOMBRE='Fernanda' WHERE NOMBRE\_USUARIO = 'mArIa34';

1. **MODIFICAR EL NOMBRE DEL PRODUCTO CON EL CÓDIGO 5DG35SJ57SJ4**

UPDATE PRODUCTOS SET NOMBRE='Doritos' WHERE CODIGO='A3H2J4H2H3J2';

1. **MODIFICAR LA EDAD DEL USUARIO QUE SE LLAMA Juan**

UPDATE CLIENTES SET EDAD=21 WHERE NOMBRE = 'Juan';

1. **MODIFICAR EL NUMERO INTERNO DE LA DIRECCIÓN = 1**

UPDATE DIRECCIONES SET NUM\_INTERNO=11 WHERE ID\_DIRECCION=1;

1. **MODIFICAR LA CANTIDAD DE PRODUCTO ALMACENADO CON NOMBRE "Coca-Cola"**

UPDATE ALMACENA SET CANTIDAD = 599 WHERE CODIGO = (SELECT CODIGO FROM PRODUCTOS WHERE NOMBRE = 'Coca-Cola');

1. **MODIFICAR LA DESCRIPCION DE LA CATEGORIA CON EL NOMBRE "Comestibles"**

UPDATE CATEGORIAS SET DESCRIPCION = 'Todo tipo de frituras' WHERE NOMBRE = 'Comestibles';

1. **MODIFICAR LA CONTRASEÑA DEL CLIENTE CON EL NOMBRE "Pedro" Y APELLIDOS "Gonzalez"**

UPDATE CLIENTES SET CONTRASENIA = 'NVU83IGC8WI29' WHERE NOMBRE = 'Pedro' AND APELLIDO = 'Gonzalez';

1. **MODIFICAR LA DIRECCIÓN DEL ALMACÉN CON NOMBRE "ALMACEN 3"**

UPDATE ALMACENES SET ID\_DIRECCION = 4 WHERE NOMBRE = 'ALMACEN 3';

1. **MODIFICAR EL MUNICIPIO, COLONIA, NUM\_EXTERNO Y NUM\_INTERNO DEL MUNICIPIO CON MUNICIPIO = "VALPARAÍSO", COLONIA = "ATOTONILCO", NUM\_EXTERNO = 101**

UPDATE DIRECCIONES SET MUNICIPIO = 'JEREZ',COLONIA = 'CENTRO',NUM\_EXTERNO = 222,NUM\_INTERNO = 24 WHERE MUNICIPIO = 'VALPARAÍSO' AND COLONIA = 'ATOTONILCO' AND NUM\_EXTERNO = 101;

# **Actividad 8:**

## Sentencias SQL–DML para eliminar 10 datos de la Base de Datos.

1. **ELIMINAR EL PRODUCTO CON NOMBRE "Coca-Cola" DEL LOS ALMACENES**

DELETE FROM ALMACENA WHERE CODIGO = (SELECT CODIGO FROM PRODUCTOS WHERE NOMBRE = 'Coca-Cola');

1. **ELIMINAR CATEGORIAS DEL PRODUCTO "Doritos"**

DELETE FROM PERTENECE WHERE CODIGO = (SELECT CODIGO FROM PRODUCTOS WHERE NOMBRE = 'Doritos');

1. **ELIMINAR LA LA RELACION QUE EXISTE ENTRE LOS PRODUCTOS Y LA CATEGORIA "Panes"**

DELETE FROM PERTENECE WHERE NOMBRE = (SELECT NOMBRE FROM CATEGORIAS WHERE NOMBRE = 'Panes');

1. **ELIMINAR LA RELACION DE LOS ALMACENES QUE TENGAN COMO NOMBRE "ALMACEN 3"**

DELETE FROM ALMACENA WHERE ID\_ALMACEN = (SELECT ID\_ALMACEN FROM ALMACENES WHERE NOMBRE = 'ALMACEN 3');

1. **ELIMINAR EL ALMACEN 2**

DELETE FROM ALMACENES WHERE NOMBRE = 'ALMACEN 2';

1. **ELIMINAR LA DIRECCION CON LA COLONIA "ZAPATA" Y EL NUMERO EXTERNO 200**

DELETE FROM DIRECCIONES WHERE COLONIA = 'ZAPATA' AND NUM\_EXTERNO = 200;

1. **ELIMINAR EL CLIENTE CON EL NOMBRE FERNANDO Y EL APELLIDO ESCALANTE**

DELETE FROM CLIENTES WHERE NOMBRE = 'FERNANDO' AND APELLIDO = 'ESCALANTE';

1. **ELIMINAR EL PRODUCTO DE LA MARCA "ELECTROLIT" Y CON EL NOMBRE "SUERO COCO"**

DELETE FROM PRODUCTOS WHERE MARCA = 'ELECTROLIT' AND NOMBRE = 'SUERO COCO';

1. **ELIMINAR LA CATEGORIA DE "FRUTAS Y VERDURAS"**

DELETE FROM CATEGORIAS WHERE NOMBRE = 'FRUTAS Y VERDURAS';

1. **ELIMINAR LA RELACION ENTRE EL CARRO 1 Y LOS CLIENTES**

DELETE FROM CARRITO\_DE WHERE ID\_CARRO\_COMPRA = 1;

# **Actividad 9:**

## Redactar 25 enunciados para consultar datos y elaborar las sentencias SQL–DML necesarias para dar respuesta a dichos enunciados.

1. **CONSULTAR LAS DIRECCIONES DE TODOS LOS ALMACENES**

SELECT AL.\*,D.\* FROM ALMACENES AL JOIN DIRECCIONES D ON AL.ID\_DIRECCION = D.ID\_DIRECCION;

1. **CONSULTAR EL NOMBRE DEL CLIENTE QUE A COMPRADO "Gansito"**

SELECT DISTINCT CL.\* FROM CLIENTES CL JOIN CARRITO\_DE CD ON CL.NOMBRE\_USUARIO = CD.NOMBRE\_USUARIO

JOIN CARROS\_COMPRAS CC ON CD.ID\_CARRO\_COMPRA = CC.ID\_CARRO\_COMPRA

JOIN CONTIENE CO ON CC.ID\_CARRO\_COMPRA = CO.ID\_CARRO\_COMPRA

JOIN PRODUCTOS P ON CO.CODIGO = (SELECT CODIGO FROM PRODUCTOS WHERE NOMBRE = 'Gansito');

1. **CONSULTAR LOS CLIENTES QUE TENGAN COMO DIRECCIÓN EL CÓDIGO POSTAL 99200**

SELECT CL.\* FROM CLIENTES CL JOIN CLIENTE\_DIRECCIONES CD ON CL.NOMBRE\_USUARIO = CD.NOMBRE\_USUARIO

JOIN DIRECCIONES D ON CD.ID\_DIRECCION = D.ID\_DIRECCION

AND D.ID\_DIRECCION IN (SELECT ID\_DIRECCION FROM DIRECCIONES WHERE CODIGO\_POSTAL = 99200);

1. **CONSULTAR LA DIRECCIÓN DEL ALMACEN 4**

SELECT \* FROM DIRECCIONES WHERE ID\_DIRECCION =

(SELECT ID\_DIRECCION FROM ALMACENES WHERE NOMBRE = 'ALMACEN 4');

1. **CONSULTAR EL NOMBRE DE USUARIO, NOMBRE Y APELLIDO DEL CLIENTE CON EL E-MAIL "pascual@gmail.com"**

SELECT NOMBRE\_USUARIO,NOMBRE,APELLIDO FROM CLIENTES WHERE EMAIL = 'pascual@gmail.com';

1. **CONSULTAR EL ESTADO DE LA DIRECCIÓN DEL CLIENTE CON NOMBRE "Pascual" Y APELLIDO "Bonilla"**

SELECT DISTINCT CL.NOMBRE||' '||APELLIDO AS NOMBRE,D.ESTADO FROM DIRECCIONES D

JOIN CLIENTE\_DIRECCIONES CD ON D.ID\_DIRECCION = CD.ID\_DIRECCION

JOIN CLIENTES CL ON CL.NOMBRE\_USUARIO = (SELECT NOMBRE\_USUARIO

FROM CLIENTES WHERE NOMBRE = 'Pascual' AND APELLIDO = 'Bonilla');

1. **CONSULTAR LA DESCRIPCIÓN DE LA CATEGORÍA A LA QUE PERTENECE EL PRODUCTO "Gansito"**

SELECT DISTINCT P.NOMBRE AS PRODUCTO, CA.DESCRIPCION AS DESCRIPCION\_CATEGORIA FROM CATEGORIAS CA JOIN PERTENECE PE ON CA.NOMBRE = PE.NOMBRE

JOIN PRODUCTOS P ON PE.CODIGO = P.CODIGO AND P.CODIGO = (SELECT CODIGO FROM PRODUCTOS WHERE NOMBRE = 'Gansito');

1. **CONSULTAR EL PRECIO DE LOS PRODUCTOS COMPRADOS POR EL CLIENTE CON NOMBRE DE USUARIO "jaunito1"**

SELECT CL.NOMBRE AS CLIENTE,P.NOMBRE,P.PRECIO FROM PRODUCTOS P JOIN CONTIENE CO ON P.CODIGO = CO.CODIGO

JOIN CARROS\_COMPRAS CC ON CO.ID\_CARRO\_COMPRA = CC.ID\_CARRO\_COMPRA

JOIN CARRITO\_DE CD ON CC.ID\_CARRO\_COMPRA = CD.ID\_CARRO\_COMPRA

JOIN CLIENTES CL ON CD.NOMBRE\_USUARIO = CL.NOMBRE\_USUARIO AND CL.NOMBRE\_USUARIO = 'jaunito1';

1. **CONSULTAR LOS PRODUCTOS ALMACENADOS EN "ALMACEN 4"**

SELECT AL.NOMBRE AS ALMACEN, P.NOMBRE AS PRODUCTO FROM ALMACENES AL

JOIN ALMACENA ALM ON AL.ID\_ALMACEN = ALM.ID\_ALMACEN JOIN

PRODUCTOS P ON ALM.CODIGO = P.CODIGO AND AL.NOMBRE = 'ALMACEN 4';

1. **CONSULTAR LAS DIRECCIONES DE TODOS LOS CLIENTES**

SELECT CL.NOMBRE AS CLIENTE,D.\* FROM DIRECCIONES D JOIN CLIENTE\_DIRECCIONES CD ON D.ID\_DIRECCION = CD.ID\_DIRECCION

JOIN CLIENTES CL ON CD.NOMBRE\_USUARIO = CL.NOMBRE\_USUARIO;

1. **CONSULTAR LOS PRODUCTOS QUE PERTENECEN A LA CATEGORÍA "Comestibles"**

SELECT P.\* FROM PRODUCTOS P JOIN PERTENECE PE ON P.CODIGO = PE.CODIGO

JOIN CATEGORIAS CA ON PE.NOMBRE = CA.NOMBRE AND CA.NOMBRE = 'Comestibles';

1. **CONSULTAR LOS DATOS DEL CLIENTE CON LA DIRECCIÓN (MUNICIPIO = VALPARAÍSO, COLONIA = ALAMEDA, NUM\_EXTERNO = 125**

SELECT CL.\* FROM CLIENTES CL JOIN CLIENTE\_DIRECCIONES CD ON CL.NOMBRE\_USUARIO = CD.NOMBRE\_USUARIO

JOIN DIRECCIONES D ON CD.ID\_DIRECCION = D.ID\_DIRECCION AND D.ID\_DIRECCION =

(SELECT ID\_DIRECCION FROM DIRECCIONES WHERE MUNICIPIO = 'VALPARAÍSO' AND COLONIA = 'ALAMEDA' AND NUM\_EXTERNO = 125);

1. **CONSULTAR EL NOMBRE DEL ALMACEN EN EL QUE ESTA GUARDADO EL PRODUCTO "Pinguinos"**

SELECT AL.NOMBRE FROM ALMACENES AL JOIN ALMACENA ALM ON AL.ID\_ALMACEN = ALM.ID\_ALMACEN

JOIN PRODUCTOS P ON ALM.CODIGO = P.CODIGO AND P.CODIGO = (SELECT CODIGO

FROM PRODUCTOS WHERE NOMBRE = 'Pinguinos');

1. **CONSULTAR EL MUNICIPIO DE DONDE ESTÁN LOS ALMACENES**

SELECT D.MUNICIPIO FROM DIRECCIONES D JOIN ALMACENES AL ON D.ID\_DIRECCION = AL.ID\_DIRECCION;

1. **CONSULTAR EL NOMBRE Y APELLIDO DE LOS CLIENTES QUE NO HAN COMPRADO NINGÚN PRODUCTO**

SELECT NOMBRE||' '||APELLIDO AS CLIENTE FROM CLIENTES

WHERE NOMBRE\_USUARIO NOT IN(SELECT NOMBRE\_USUARIO FROM CARRITO\_DE);

1. **CONSULTAR LOS CLIENTES QUE NO TIENEN DIRECCIÓN**

SELECT NOMBRE||' '||APELLIDO AS CLIENTE FROM CLIENTES WHERE NOMBRE\_USUARIO

NOT IN(SELECT NOMBRE\_USUARIO FROM CLIENTE\_DIRECCIONES);

1. **CONSULTAR EL TOTAL DE DIRECCIONES QUE TIENE CADA CLIENTE**

SELECT DISTINCT CL.NOMBRE||' '||CL.APELLIDO AS CLIENTE,CD.TOTAL AS TOTAL\_DIRECCIONES

FROM CLIENTES CL,(SELECT COUNT(\*) AS TOTAL FROM CLIENTE\_DIRECCIONES GROUP BY NOMBRE\_USUARIO)CD;

1. **CONSULTAR EL PRODUCTO MAS CARO**

SELECT \* FROM PRODUCTOS WHERE PRECIO = (SELECT MAX(PRECIO) FROM PRODUCTOS);

1. **CONSULTAR EL PRODUCTO MAS VARATO**

SELECT \* FROM PRODUCTOS WHERE PRECIO = (SELECT MIN(PRECIO) FROM PRODUCTOS);

1. **CONSULTAR EL CLIENTE CON MAYOR EDAD**

SELECT \* FROM CLIENTES WHERE EDAD = (SELECT MAX(EDAD) FROM CLIENTES);

1. **CONSULTAR EL CLIENTE CON MENOR EDAD**

SELECT \* FROM CLIENTES WHERE EDAD = (SELECT MIN(EDAD) FROM CLIENTES);

1. **CONSULTAR EL TOTAL DE VENTAS DE CADA PRODUCTO**

SELECT P.NOMBRE AS PRODUCTO,CO.TOTAL AS TOTAL\_VENTAS FROM PRODUCTOS P,(SELECT CODIGO,COUNT(\*) AS TOTAL

FROM CONTIENE GROUP BY CODIGO)CO WHERE P.CODIGO = CO.CODIGO;

1. **CONSULTAR LA DIRECCIÓN EN LA QUE ESTA ALMACENADO EL PRODUCTO "Jugo Manzana"**

SELECT P.NOMBRE AS PRODUCTO,D.\* FROM DIRECCIONES D JOIN ALMACENES AL ON D.ID\_DIRECCION = AL.ID\_DIRECCION

JOIN ALMACENA ALM ON AL.ID\_ALMACEN = ALM.ID\_ALMACEN

JOIN PRODUCTOS P ON ALM.CODIGO = P.CODIGO AND P.CODIGO = (SELECT CODIGO FROM PRODUCTOS WHERE NOMBRE = 'Jugo Manzana');

1. **CONSULTAR LOS DATOS GENERALES DE LOS PRODUCTOS Y SU CATEGORÍA**

SELECT P.\*,CA.NOMBRE AS CATEGORIA FROM PRODUCTOS P

JOIN PERTENECE PE ON P.CODIGO = PE.CODIGO JOIN CATEGORIAS CA ON PE.NOMBRE = CA.NOMBRE;

1. **CONSULTAR LA DIRECCIÓN DONDE ESTA ALMACENADO EL PRODUCTO CON LA CATEGORÍA "Comestibles"**

SELECT CA.NOMBRE AS CATEGORIA,D.\* FROM DIRECCIONES D JOIN ALMACENES AL ON D.ID\_DIRECCION = AL.ID\_DIRECCION

JOIN ALMACENA ALM ON AL.ID\_ALMACEN = ALM.ID\_ALMACEN

JOIN PRODUCTOS P ON ALM.CODIGO = P.CODIGO

JOIN PERTENECE PE ON P.CODIGO = PE.CODIGO

JOIN CATEGORIAS CA ON PE.NOMBRE = CA.NOMBRE AND CA.NOMBRE = (SELECT NOMBRE FROM CATEGORIAS WHERE NOMBRE = 'No empaquetados');