

**IPN, ESCOM, ING. EN SISTEMAS COMPUTACIONALES**

**MATERIA: DISTRIBUTED DATA BASES**

**GRUPO: 3CM10**

**PROFESOR: HERNÁNDEZ CONTRERAS EULER**

**NOMBRE: SALDAÑA AGUILAR GABRIELA**

**NO. DE PRÁCTICA: 2**

**PRACTICA: CONSULTAS DE SELECCIÓN**

**FECHA DE ENTREGA: 22/02/2016**



INDICE

MARCO TEÓRICO………………………… 3

DESARROLLO……………………………….. 4

CONCLUSIÓN Y BIBLIOGRAFÍA….. 11

MARCO TEORICO

SELECT

Las consultas de selección se utilizan para indicar al motor de datos que devuelva información de las bases de datos, esta información es devuelta en forma de conjunto de registros que se pueden almacenar en un objeto recordset

Este conjunto de registros puede ser modificable.

Consultas básicas

La sintaxis básica de una consulta de selección es la siguiente:

SELECT   
   Campos   
FROM   
   Tabla

DEVOLVER LITERALES

En determinadas ocasiones nos puede interesar incluir una columna con un texto fijo en una consulta de selección:

SELECT   
   Empleados.Nombre, 'Tarifa semanal: ', Empleados.TarifaHora \* 40   
FROM   
   Empleados   
WHERE   
   Empleados.Cargo = 'Electricista'

ORDENAR LOS REGISTROS

Adicionalmente se puede especificar el orden en que se desean recuperar los registros de las tablas mediante la cláusula ORDER BY Lista de Campos. En donde Lista de campos representa los campos a ordenar. Ejemplo:

SELECT   
   CodigoPostal, Nombre, Telefono   
FROM   
   Clientes   
ORDER BY   
   Nombre

Incluso se puede especificar el orden de los registros: ascendente mediante la cláusula (ASC - se toma este valor por defecto) ó descendente (DESC)

ALTER TABLE

statement is used to add, delete, or modify columns in an existing table.

SYNTAXIS

To add a column in a table, use the following syntax:

ALTER TABLE table\_name  
ADD column\_name datatype

To delete a column in a table, use the following syntax (notice that some database systems don't allow deleting a column):

ALTER TABLE table\_name  
DROP COLUMN column\_name

To change the data type of a column in a table, use the following syntax:

ALTER TABLE table\_name  
ALTER COLUMN column\_name datatype

## CREATE TABLE

The CREATE TABLE statement is used to create a table in a database.

Tables are organized into rows and columns; and each table must have a name.

### SQL CREATE TABLE Syntax

CREATE TABLE *table\_name*  
(  
*column\_name1 data\_type*(*size*),  
*column\_name2 data\_type*(*size*),  
*column\_name3 data\_type*(*size*),  
....  
);

The column\_name parameters specify the names of the columns of the table. The data\_type parameter specifies what type of data the column can hold (e.g. varchar, integer, decimal, date, etc.).

The size parameter specifies the maximum length of the column of the table.

INSTRUCCIONES

PRACTICA 2:/

source C:/Users/alumno/Desktop/elektra.sql

1\*Mostrar el nombre completo del cliente y su monto de crédito

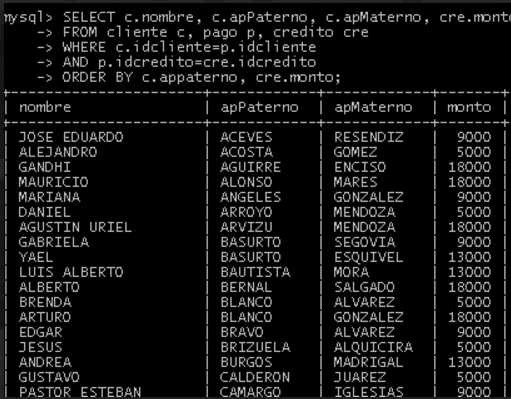
SELECT c.nombre, c.apPaterno, c.apMaterno, cre.monto

FROM cliente c, pago p, credito cre

WHERE c.idcliente=p.idcliente

AND p.idcredito=cre.idcredito

ORDER BY c.appaterno, cre.monto;//el primero no lleva alias porque solo existe un appaterno



2\*Mostrar el nombre del producto , su marca y su precioUnitario de aquellos productos

que pertenecen a la categoria transporte.

SELECT p.nombre, p.marca, p.precioUnitario

FROM producto p, subcategoria sub, categoria c

WHERE p.idsubcategoria=sub.idsubcategoria

AND sub.idcategoria=c.idcategoria

AND c.nombre like "%transpo%"

ORDER BY p.preciounitario;

SELECT p.nombre, p.marca, p.precioUnitario

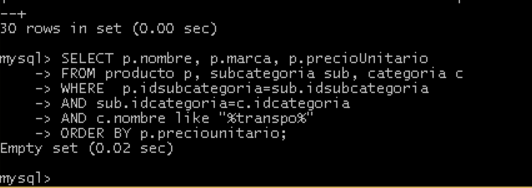
FROM producto p, subcategoria sub, categoria c

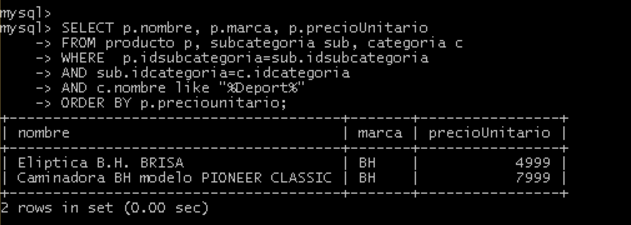
WHERE p.idsubcategoria=sub.idsubcategoria

AND sub.idcategoria=c.idcategoria

AND c.nombre like "%Deport%"

ORDER BY p.preciounitario;





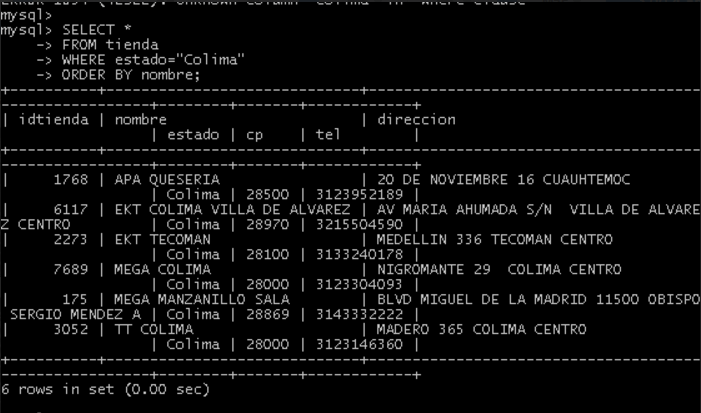
3\*Mostrar los datos correspondientes de las sucursales existentes en el estado de colima

SELECT \*

FROM tienda

WHERE estado="Colima"

ORDER BY nombre;



4\*Que clientes hicieron pagos el día 10 de marzo del 2010?

Mostrar su nombre, apellido paterno y el monto de crédito correspondiente.

SELECT c.nombre, c.appaterno, cre.monto

FROM pago p, cliente c, credito cre

WHERE c.idcliente=p.idcliente

AND p.idcredito=cre.idcredito

AND p.fechapago like "%2010-03-10%"

ORDER BY cre.monto, appaterno;



5\*Modificar la definición de la relación cliente para almacenar correos

Hay que crear la relación de correo para poder almacenar un atributo multivalor

, PRIMERO ANTES DE BORRAR HAY QUE HACER UNA COPIA DE LOS CORREOS, la foreign key sera la de la tabla

sobre la que estamos extrayendo. y se hace una unión de llave primaria

A)

CREATE table correo(

idcliente int not null,

email varchar(50) not null,

primary key(idcliente,email),

foreign key (idcliente) references cliente(idcliente)

on delete cascade on update cascade

);

B) LLENAR DE DATOS LA RELACION

INSERT INTO correo

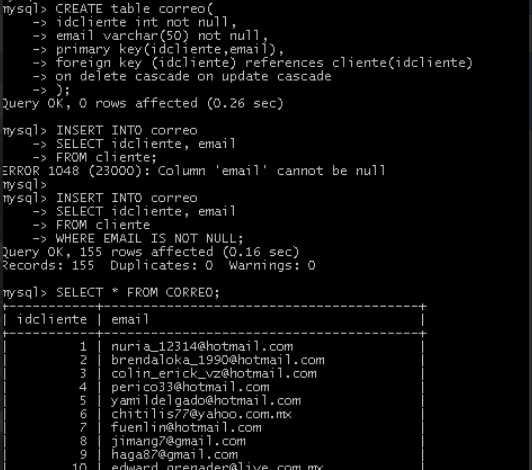
SELECT idcliente, email

FROM cliente

WHERE EMAIL IS NOT NULL;

C) ELIMINAR EL CAMPO EMAIL DE LA RELACION CLIENTE

alter table CLIENTE DROP COLUMN EMAIL;



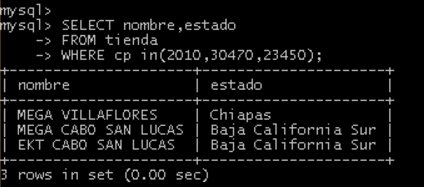
6\*Mostrar el nombre de la sucursal, así como el estado en que se ubican las sucursales que están

en las sig cp... 20010,30470,23450

SELECT nombre,estado

FROM tienda

WHERE cp in(2010,30470,23450);



7\*Mostrar el nombre de los productos, marca y subcategoría

de aquellos productos que son suministrados en la sucursal ekt tecate

SELECT p.nombre, p.marca, sub.nombre

FROM producto p, subcategoria sub, tiendaproducto tp, tienda t

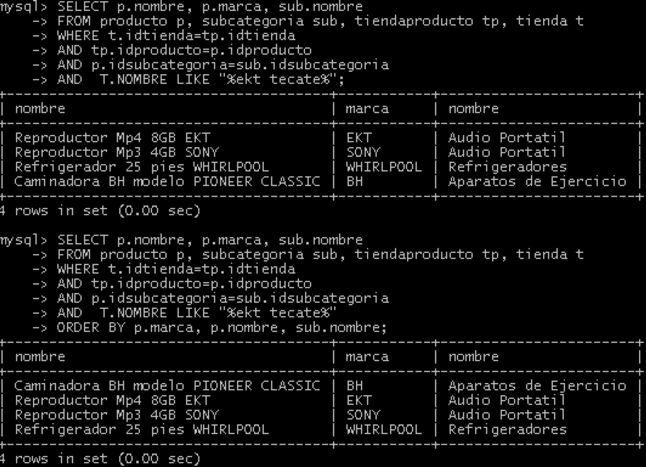
WHERE t.idtienda=tp.idtienda

AND tp.idproducto=p.idproducto

AND p.idsubcategoria=sub.idsubcategoria

AND T.NOMBRE LIKE "%ekt tecate%"

ORDER BY p.marca, p.nombre desc, sub.nombre;



REALIZAR LAS SIGUINETES MODIFICACIONES

1\* renombrar el campo nombre en la relacion producto y llmarle producto..

ALTER TABLE producto

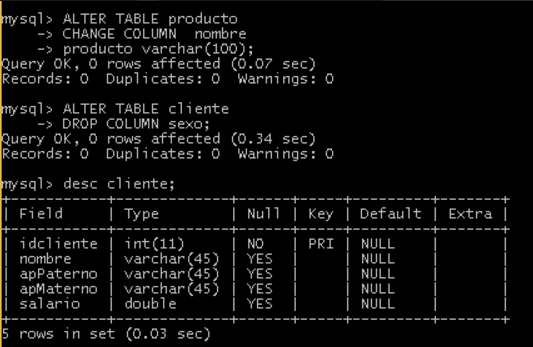
CHANGE COLUMN nombre// renombrar un campo o cambiar el y tamaño o tipo de datos

producto varchar(100);

2\*Eliminar el sexo en cliente...

ALTER TABLE cliente

DROP COLUMN sexo;



3\*Agregar una relación para almacenar infromacion de los empleados

para las sucursales existentes asociar dicha relación con tienda o sucursal

empleado(idempleado PK, nombre, salario, email, edad,dir, idtienda Fk);

CREATE TABLE empleado(

idempelado int not null primary key,

nombre varchar(80),

salario double,

email varchar(50),

edad int,

dir varchar(300),

idtienda int,

foreign key(idtienda) references tienda(idtienda)

on delete cascade on update cascade

);



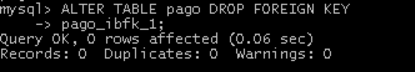
4\*Eliminar la asociacion existente entre la relacion pago y la relacion credito

show create table pago;//ver los constraints

//para eliminar la llave foranea de una tabla aun queda el campo en la tabla

ALTER TABLE pago DROP FOREIGN KEY

pago\_ibfk\_1;

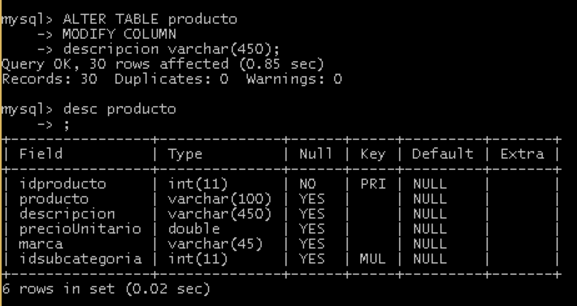


5\*Aumentar el numero de caracterers sobre el campo descripcion en la relacion producto, almacenar 450 caracteres

ALTER TABLE producto

MODIFY COLUMN

descripcion varchar(450);



CONCLUCIÓN

Se pudo dar repaso a temas antes vistos en bases de datos, en esta práctica especialmente se tomó el repaso de cómo hacer consultas de selección básicas, se retomaron ciertos conocimientos como :

\*Interpretar un E-R para poder sacar las consultas

\*Atributos de reunión

\*Ordenamiento

\*Wild Card

\*Alias

\*Condiciones WHERE

\*Cargar un script de una BD

Entre otros. El repaso fue de mucha ayuda porque nos servirá tanto para prácticas posteriores como para el examen, también porque es conveniente tratar de generar las consultas lo mejor posible en el menor tiempo.

Se retomaron comandos como lo son el ALTER TABLE, CREATE TABLE entre otras operaciones que se realizan para modificar campos específicos de una tabla o para crear una tabla en si como en el caso que nos piden generar una nueva relación, o cuando es necesario normalizar una ya existente.

BIBLIOGRAFÍA

<http://www.desarrolloweb.com/articulos/2257.php>

https://dev.mysql.com/doc/refman/5.5/en/select.html