**Calidad en el desarrollo de sw**

**Ejercicio 1**

La escuela secundaria “Benito Juárez”, tiene la necesidad de la implementación de un sistema que permita realizar la creación de cursos, los cuales se componen de un profesor, una materia, un aula y de muchos alumnos a los cuales se les impartirá dicho curso.

Realiza una aplicación que permita realizar lo que la escuela secundaria esta solicitando.

Utiliza tu clase Conexión y las clase lógicas, genera e implementa los métodos que se te piden.

1. Método para cargar el comboBox matricula de la vista Frame GenerarCurso
2. Método para consultar el Alumno según su matricula
3. Método para cargar el comboBox codigo profesor de la vista Frame GenerarCurso
4. Método para consultar el profesor según su codProfesor
5. Método para cargar el comboBox codMateria de la vista Frame GenerarCurso
6. Método para consultar la materia según su codMateria
7. Método que permita insertar un nuevo curso
8. Método para insertar los alumos al curso
9. Método que muestre los alumnos agregados al curso

create database dbEscuela;

use dbEscuela;

create table alumno

(

matricula varchar(6) not null primary key,

Nombre varchar(50) not null

);

INSERT INTO alumno VALUES('141023','Aldo Mendez Cortes');

INSERT INTO alumno VALUES('141044','Susana Aguliar Díaz');

INSERT INTO alumno VALUES('141055','Alejandro Rodriguez Pérez');

INSERT INTO alumno VALUES('141034','Martha Cervantes Ochoa');

INSERT INTO alumno VALUES('141001','Gabriela González Orozco');

INSERT INTO alumno VALUES('141002','Jesús Aviles Ortiz');

INSERT INTO alumno VALUES('141003','Moises Zuñiga Mares');

INSERT INTO alumno VALUES('141004','Arturo Aguilar Gómez');

create table profesor

(

codProfesor varchar(6) not null primary key,

nombre varchar(50) not null,

carrera varchar(100) not null,

telefono varchar(15) not null

);

INSERT INTO profesor VALUES('202401','Mariana Villaseñor Ortega','Informática','33121110');

INSERT INTO profesor VALUES('203402','Roberto Gómez Hernández', 'Biología','3311589871');

INSERT INTO profesor VALUES('201403','Bruno Ortega López','Industrial','33124567');

INSERT INTO profesor VALUES('205404','Edgardo Castillo Suarez','Ciencias sociales','33445562');

INSERT INTO profesor VALUES('203405','Alejandra Sahagún Arce','Administrativa', '331154345');

create table materia

(

codMateria varchar(6) not null primary key,

nombre varchar(50) not null,

cargaHoraria int not null

);

INSERT INTO materia VALUES('mat141','Matemáticas',6);

INSERT INTO materia VALUES('esp132','Español',5);

INSERT INTO materia VALUES('bio122','Biología',4);

INSERT INTO materia VALUES('civ149','Civismo',3);

INSERT INTO materia VALUES('ing135','Inglés',6);

create table curso

(

codigoCurso varchar (6) not null primary key,

fecha\_inicio date,

noAula int not null,

codProfesor varchar(6) not null,

codMateria varchar(6) not null,

foreign key (codProfesor) references profesor(codProfesor),

foreign key (codMateria) references materia(codMateria)

);

create table curso\_alumnos

(

codigoCurso varchar (6) not null,

matricula varchar(6) not null,

foreign key (codigoCurso) references curso(codigoCurso),

foreign key (matricula) references alumno(matricula)

);

**Ejercicio 2**

La zapatería “Pies cómodos” desea ampliar su aplicación con un módulo de consultas que permitan realizar lo siguiente:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Consultas

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(1) ¿cuántos colores distintos se ofrecen?

(2) ¿en qué meses se ha vendido el modelo \_\_\_\_\_?

(3) ¿cuantos pares se han vendido por cada mes?

(4) ¿en qué mes se vendieron menos pares de zapatos?

(5) ¿qué modelo no se ha vendido?

(6) ¿qué modelo se ha vendido por lo menos una vez todos los meses del año?

(7) ¿Cuál es el segundo modelo mejor vendido? ( solo usar lo visto en clase )

(8) ¿en qué mes del año se registran nuestras mejores ventas ( más cantidad de dinero recaudado por ventas ) ?

(9) ¿en qué mes del año se registran mayor cantidad de ventas ( más pares de zapatos vendidos ) ?

(10) Del listado de modelos que más se venden por mes, ¿cuáles son los modelos que se repiten en más de 2 meses?

create database dbzapateria;

use dbzapateria;

create table tmodelosZapatos

(

codModelo varchar(5) not null primary key,

existencia int,

costo decimal(5,2),

estado bit not null

);

insert into tmodelosZapatos values ('zap01',150,230.00,1);

create table tzapatos

(

modelo varchar (5) not null references tmodelosZapatos(codModelo) ,

color varchar (10) not null,

talla decimal(5,2) not null,

precio decimal(5,2),

existencia int not null default 10,

primary key ( modelo, color, talla )

);

create table tVentas

(

folio int not null primary key auto\_increment,

fecha date ,

total decimal(5,2)

);

create table tdetVentas

(

folio int not null references tventas(folio),

modelo varchar(5) not null ,

color varchar(10) not null ,

talla decimal (5,2) not null ,

cantidad int,

precio decimal(5,2),

subtotal decimal(8,2),

foreign key (modelo, color, talla) references tzapatos(modelo, color, talla)

);

**Ejercicio 3**

La tienda de ropa “Rosy S.A.” quiere que se le desarrolle una aplicación para llevar la gestión de ventas. Los productos de la empresa es ropa, de los cuales se conoce su codProd, descripcion, costo, precio y stock.

Estos productos son vendidos a clientes, los datos que se requieren son codCliente, Nombre domicilio y teléfono.

Todas las ventas realizadas se registran con un folio, fecha, codCliente y el total de la venta, así como el detalle de la venta de los productos requiere folio, codProd, cantidad y subtotal, el subtotal se calcula la cantidad por el precio del producto.

Realizar un programa en java que permita:

1. Llevar a cabo las ventas de los productos.
   1. La aplicación permitirá seleccionar un código de cliente desde un comboBox, este se cargará de la tabla clientes, al seleccionar el código del cliente mostrará los datos en la pantalla.
   2. El folio se mostrará en automático cada que se desea una nueva venta.
   3. Para agregar los productos a la venta estos se seleccionaran de un comboBox, este se cargará de la tabla productos, al seleccionar el código del producto mostrará los datos en la pantalla.
   4. al dar clic en el botón agregar, se agregará al jtable,
   5. ira calculando el total de la venta y mostrará este dato en la pantalla.
   6. Para finalizar la venta, deberá presionar el botón “Finalizar venta”, esto permitirá guardar la venta tanto en la tabla ventas, como en la tabla ventasProductos.
2. Mostrar las siguientes consultas:

Clientes

1. Lista de clientes (Quitar filtro)
2. Filtro de clientes según su código, nombre, teléfono

Productos

1. Lista de productos (Quitar filtro), filtro de productos según su código, descripción
2. Producto más vendido
3. Mostrar el producto con menos stock
4. Mostrar el producto con mayor stock

Ventas

* 1. Detalle de la venta según el folio
  2. Ventas según el mes seleccionado
  3. Ventas según el cliente seleccionado
  4. Total de ventas

create database dbtienda;

use dbtienda;

create table productos

(

codProd varchar (6) not null primary key,

descripcion varchar (50) not null,

costo decimal (5,2) not null,

precio decimal(5,2) not null,

stock int not null

);

create table clientes

(

codCliente varchar(6) not null primary key,

Nombre varchar(50) not null,

docicilio varchar(100) not null,

telefono varchar(15) not null

);

create table ventas

(

folio int not null primary key auto\_increment,

fecha date ,

codCliente varchar(6) not null,

total decimal(5,2),

foreign key (codCliente) references clientes(codCliente)

);

create table ventasProductos

(

folio int not null,

codProd varchar(6) not null,

cantidad int not null,

subtotal decimal(5,2),

foreign key (codProd) references productos(codProd)

);

/\*Registros productos\*/

insert into productos values ('pro001','pantalon mezclilla',200.00,280,20);

insert into productos values ('pro002','Blusa algodón',150.00,210,34);

insert into productos values ('pro003','pantalon vestir',300.00,420,45);

insert into productos values ('pro004','Camisa rayada',100.00,140,67);

insert into productos values ('pro005','Short playero',120.00,280,0);

insert into productos values ('pro006','Playera roja',80.00,280,98);

insert into productos values ('pro007','Blusa casual',560.00,280,10);

insert into productos values ('pro008','pantalon casual',163.00,280,5);

insert into productos values ('pro009','Vestido floreado',420.00,280,25);

insert into productos values ('pro010','Falda negra',180.00,280,70);

insert into productos values ('pro011','Playera tipo polo',90.00,280,56);

/\*Registro clientes\*/

insert into clientes values ('c00001','Pedro machetes','Juan Alvarez 578','33-33-33-33-33');

insert into clientes values ('c00002','María Benitez','Guirnalda 632','33-23-92-43-33');

insert into clientes values ('c00003','Alejandro Rodriguez','Sto. Tomas 1391','36-53-12-49-36');

insert into clientes values ('c00004','Sara Arce','Rita Pérez 921','36-38-12-45-24');

insert into clientes values ('c00005','Ricardo Hernández','Pedro Negrete 1386','37-83-53-42-10');

/\*Resgistro Ventas\*/

insert into ventas values (0,'2014-01-23','c00001',490.00);

insert into ventas values (0,'2014-02-14','c00001',560.00);

insert into ventas values (0,'2014-03-06','c00003',400.00);

/\*Resgistro Ventas\*/

insert into ventasProductos values (1,'pro001',1,280.00);

insert into ventasProductos values (1,'pro002',1,210.00);

insert into ventasProductos values (2,'pro003',1,420.00);

insert into ventasProductos values (2,'pro004',1,120.00);

insert into ventasProductos values (3,'pro005',1,420.00);

**SUERTE**