

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

AÑO:	2019	PERIODO:	SEGUNDO TÉRMINO
MATERIA:	SISTEMA DE BASES DE DATOS I	PROFESORES:	GUSTAVO CALI, VANESSA CEDEÑO, JORGE MAGALLANES, FRANK MALO, ANGEL LOPEZ
EVALUACIÓN:	SEGUNDA	FECHA:	28/01/2020

NOMBRE: _____ **MATRÍCULA:** _____

Sección 1 (10%) Conceptos

Responda verdadero o falso (10 puntos):

Los STORE PROCEDURES pueden recibir y devolver información; para ello se emplean parámetros, de entrada y salida, respectivamente.	
En un árbol B+, toda la data se guarda en las hojas. Los nodos internos sólo contienen claves y punteros. Todas las hojas se encuentran en el mismo nivel, que corresponde al más bajo.	
Con LOCK TABLES se pueden bloquear varias tablas en un solo comando si las escribimos separadas por comas.	
Una sentencia SELECT normal bloquea filas de una tabla.	
Las sentencias INSERT, UPDATE o DELETE realiza un bloqueo ROW EXCLUSIVE de las filas afectadas por el WHERE.	
Cuando se cumple una condición establecida al realizar una operación de inserción (INSERT), actualización (UPDATE) o borrado (DELETE), para que un TRIGGER se ejecute el usuario debe manualmente ejecutar las sentencias de SQL.	
Para ejecutar una vista (VIEW), el usuario aparte de conocer su nombre debe conocer los nombres de las tablas que fueron usadas para crear la vista.	
Algunas características importantes para realizar un PLAN DE COSTOS son la cantidad de tuplas por cada operación y, el tiempo que le toma al query arrojar el resultado.	
Para optimizar un query, un PLAN DE EJECUCIÓN nos ayuda a visualizar el costo y tuplas para cada operación.	
Para que un usuario tenga total acceso de todos los recursos, objetos y datos de todas las bases de datos, solo es necesario autenticar su identidad al conectarse al RDBMS.	

Sección 2 (30%) Queries

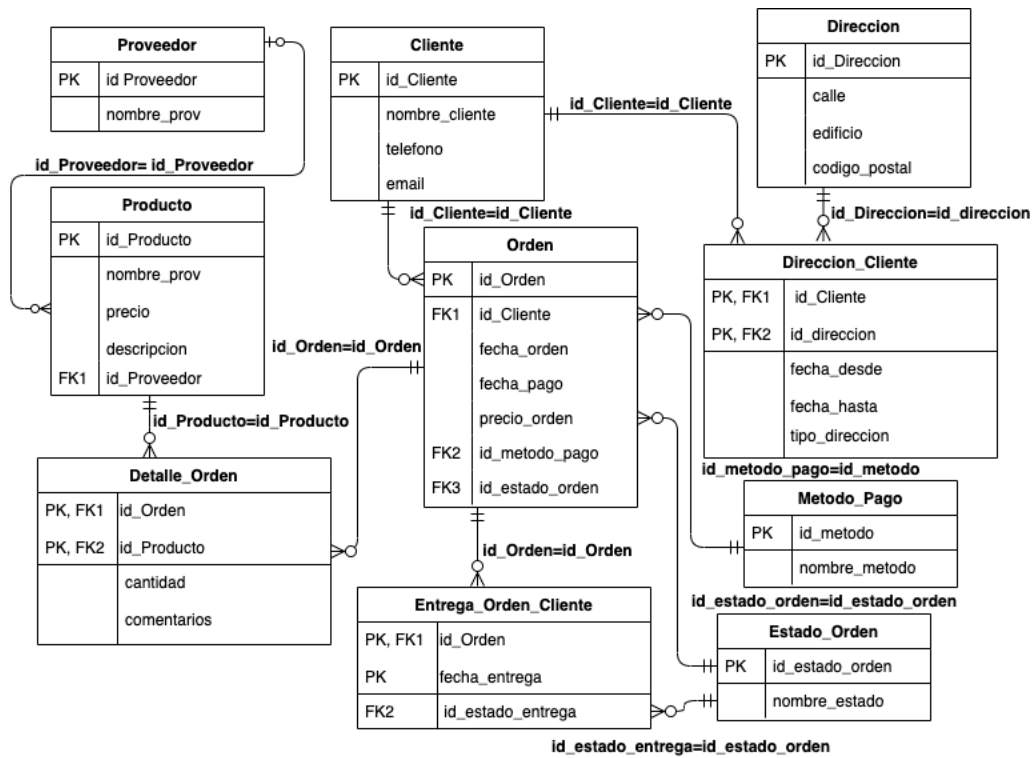


Figura 1: Esquema relacional de la base de datos de una tienda con servicio de entrega db_store.

Utilice el esquema de base de datos **db_store** y escriba los queries necesarios que permitan:

- (10 puntos) Mostrar el listado de órdenes “NO ENTREGADA” por cliente. Incluir el nombre del cliente, y ordenar por el número de órdenes en orden descendente. Muestre solamente los clientes con más de 4 órdenes sin entregar. (Nota: En la tabla Estado_Orden existe un registro con nombre_estado “NO ENTREGADA”)
- (10 puntos) Determinar el producto con más unidades vendidas en la tienda en el año 2019.
- (10 puntos) Mostrar las órdenes que hayan sido entregadas a los dos días de la fecha de creación de la orden. (Nota: En la tabla Estado_Orden existe un registro con nombre_estado “ENTREGADA”)

Sección 3 (20%) Queries Avanzados

Utilice el esquema de base de datos **db_store** y escriba los queries necesarios que permitan:

A. (10 puntos) Se ha ingresado un nuevo estado de orden 'SIN ENTREGAR 10 D' con id 23, el cual corresponde a las ordenes que llevan más de 10 días sin entregar desde la fecha de creación de la orden. Para actualizar las ordenes que se encuentran en la base, se ha pedido implementar un procedimiento almacenado de nombre 'sp_actualizar_Estado' que actualiza el id_estado_orden en la tabla Orden de las ordenes sin entregar y que tienen más de 10 días desde la fecha de creación de la orden. El procedimiento se ejecutará cada día a las 00:00 AM. (Nota: En la tabla Estado_Orden existe un registro con id=4 de nombre_estado "SIN ENTREGAR")

B. (10 puntos) Desarrolle un trigger que cada vez que se actualice la tabla Direccion, se guarde un historial de las direcciones en la tabla Direccion_historial que tiene las mismas columnas de la tabla Direccion con un campo adicional fecha_registro que es la fecha actual.

Sección 4 (40%) Optimización de Queries, DCL y Transacciones.

Dado el esquema de base de datos **db_store** para el siguiente query:

```
SELECT c.nombre_cliente
FROM Cliente c, Direccion_Cliente dc, Direccion d
WHERE c.id_Cliente = dc.id_Cliente AND dc.id_direccion = d.id_Direccion
      AND c.nombre_cliente like 'A%'
      AND dc.fecha_desde BETWEEN '10/15/2019 06:00:00' AND '12/15/2019 11:00:00'
      AND d.codigo_postal = 'EC00911'
```

A. (10 puntos) Diseñe el plan físico optimizado del query descrito:

B. (10 puntos) Genere la estimación de costos para el query descrito asumiendo que existen los siguientes datos:

- 5.000 tuplas en Cliente
- 10.000 tuplas en Direccion
- 25.000 tuplas en Direccion_Cliente
- 1500 clientes con nombres que inician con la letra A
- 400 tuplas en Direccion con fecha_desde en el rango entre '10/15/2019 06:00:00' y '12/15/2019 11:00:00'
- 1200 tuplas con codigo_postal EC00911

C. (10 puntos) En base de datos db_store, ingrese un nuevo usuario con User: USU1 y Password: QWERTY123. Y que cumpla con el siguiente nivel de seguridad.

Objeto	Select	Insert	Delete	Update	Ejecución
Proveedor (Tabla)					
Producto (Tabla)					
Cliente (Tabla)	x				
Orden (Tabla)		x	x		
Detalle Orden (Tabla)		x	x		
Entrega Orden Cliente (Tabla)				x	
Estado Orden (Tabla)				x	
Metodo Pago (Tabla)	x	x	x		
Direccion (Tabla)					
Direccion Cliente (Tabla)					
sp actualizar Estado					x

D. (10 puntos) Para la siguiente tabla Empleados, considere las siguientes transacciones y escriba el output de los Select en los cuadros blancos de la columna Resultado:

Id Empleado	Nombre	Apellido	Salario
1	Carlos	Banda	2340
2	Juan	Lino	1295
3	Ana	Vera	2345
4	Cecilia	Celi	2340

apellido	salario
Smith	9999
Joseph	1234

Ejemplo de output de Select para incluir en la columna Resultado: →

	USUARIO 1	USUARIO 2	Resultado
1	start transaction;		
2	select apellido, salario from Empleados where apellido in ('Banda','Vera','Jaramillo');		
3	UPDATE Empleados SET salario=3000 WHERE apellido = 'Banda'		
4		start transaction;	
5		select apellido, salario from Empleados where apellido in ('Banda','Vera','Jaramillo');	
6		UPDATE Empleados SET salario=4000 WHERE apellido = 'Vera'	
7	INSERT into Empleados (id_Empleado, nombre, apellido, salario) VALUES (5,'Luis','Jaramillo',640);		
8		select apellido, salario from Empleados where apellido in ('Banda','Vera','Jaramillo');	
9		UPDATE Empleados SET salario=2700 WHERE apellido = 'Banda';	
10	Commit;		
11		select apellido, salario from Empleados where apellido in ('Banda','Vera','Jaramillo');	
12		Commit;	
13	select apellido, salario from Empleados where apellido in ('Banda','Vera','Jaramillo');		