

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

AÑO:	2019	PERIODO:	PRIMER TÉRMINO
MATERIA:	SISTEMA DE BASES DE DATOS I	PROFESORES:	GUSTAVO CALI, VANESSA CEDEÑO, JORGE MAGALLANES, FRANK MALO, ANGEL LOPEZ
EVALUACIÓN:	SEGUNDA	FECHA:	27/08/2019

NOMBRE: _____ **MATRÍCULA:** _____

Sección 1(20%) Conceptos

Responda verdadero o falso (10 puntos):

Las operaciones de tipo <i>join</i> y <i>producto cartesiano</i> son de las más costosas (acceso I/O) en las bases de datos	
Las vistas almacenan los datos en estructuras temporales	
Los triggers se ejecutan en el momento en que son invocados, en cambio los procedimientos se ejecutan automáticamente cuando sucede un evento de modificación de datos	
Los triggers se crean asociándolos a tablas mientras que los procedimientos almacenados no se asocian a tablas	
Para finalizar una transacción se utiliza el comando ROLLBACK siempre y cuando la transacción se encuentre en el estado de Error	
El bloqueo de tablas con el comando LOCK TABLES puede ser usado por un usuario para bloquear una tabla para otro usuario específico.	
SQL provee mecanismos de seguridad ilimitados por lo que no se debe confiar en el software para implementar un framework de seguridad más robusto.	
La Encriptación provee una capa adicional de seguridad, protegiendo los datos de personas no autorizadas. Aún si se obtiene acceso a la base de datos, no será fácil descifrar datos encriptados.	
En bases de datos una serie de instrucciones es considerada como una transacción si posee solamente las características de Atomicidad, Consistencia y Durabilidad.	
En un proceso de optimización de consultas las operaciones de unión de tablas (<i>join</i>) deberían tener preferencia (ejecutarse antes) sobre las de selección y proyección	

Responda cual es el comando o concepto asociado a cada sentencia (10 puntos):

Tipos de índices que agrupa los datos en el disco de forma ordenada	
Comando para restaurar los cambios hechos en una transacción	
Privilegio que necesita un usuario para poder llamar a un procedimiento almacenado (store procedure)	
Comando para asignar privilegios/permisos sobre estructuras de la base de datos	
Comando para forzar (explícitamente) la actualización de los privilegios asignados	

Sección 2 (30%) Queries

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

- A. (10 puntos) Mostrar el listado de libros y el número de reseñas que cada uno tiene. Incluir el nombre del libro, el nombre de su categoría, y ordenar por el número de reseñas en orden descendente.
- B. (10 puntos) Determinar el nombre del libro, la edición del libro, y el promedio del rating, del libro con el mayor promedio de rating en las reseñas. Tenga en consideración solamente los libros con más de 5 reseñas.
- C. (10 puntos) Mostrar 10 usuarios que hayan registrado un bookmark pero que nunca hayan escrito una reseña.

Sección 3 (20%): Queries Avanzados

Utilice el esquema de base de datos **db_books (Figura 1)** y escriba los queries necesarios que permitan:

A) La tabla *books* tiene un campo sensible (*book_price*), sobre el cual se ha pedido implementar un mecanismo de control que consiste en validar cualquier modificación del precio teniendo en cuenta lo siguiente: el precio actual no puede disminuir por debajo de un 5%; en caso de que el nuevo precio es inferior se mantiene el precio actual.

B) La siguiente VISTA representa un reporte mensual (anio, mes) del número de bookmarks por ciudad de usuario y categoría del libro, sin embargo contiene errores. Reescriba la vista para que pueda ser ejecutada sin problemas.

Nota: La función YEAR devuelve el año de una fecha y la función MONTH devuelve el mes de una fecha.

```

ADD VIEW v_monthly_bookmarks
BEGIN
    SELECT YEAR(ub.date), MONTH(ub.date), u.city, c.category_name, count(*) AS total
    FROM users u, books b, bookmarks ub, categories c
    WHERE
        u.username = b.username
        b.id_category = c.id_category
    ORDER BY YEAR(date), MONTH(date), city, category_name
END;
```

Sección 4 (30%): Optimización de Queries, Transacciones y DCL.

a) (10 puntos) Suponga que existen 3 operaciones que se realizan a la tabla “books” de la base de datos “db_books” (Figura 1)

Query 1	SELECT AVG(book_price) - AVG(book_original_price) FROM books WHERE subject like '%Magic%';
Query 2	SELECT book_name,author,book_price FROM books WHERE date BETWEEN '2019-01-01' AND '2019-01-31' and book_status ='SOLD' ;
Query 3	UPDATE books SET date = now() WHERE book_status = 'AVAILABLE'

Para cada escenario, indique una recomendación de optimización y cómo se debería implementar:

Escenario	Frecuencia de Uso	Cantidad de Tuplas de la tabla Books	Respuesta
a.	Query1) 10% Query2) 80% Query3) 10%	100.000	
b.	Query1) 10% Query2) 5% Query3) 85%	100.000	

b) (10 puntos) Consideren las siguientes transacciones: Cuales son los registros resultantes en T al término de las acciones de los usuarios.

	USER 1	USER 2	USER 3
1	create table T (id int, desc char(1));		
2	start transaction;		
3	insert into T values (1, 'A');	start transaction;	start transaction;
4	insert into T values (2, 'B');	select * from T;	select * from T;
5	insert into T values (3, 'C');	insert into T values (2, 'B');	insert into T values(5, 'F')
6	insert into T values (4, 'D');	select count(*) from T;	select count(*) from T;
7	delete from T where id=2;	delete from T where id=3;	insert into T values (2, 'G')
8	commit ;	select count(*) from T;	select count(*) from T;
9		rollback;	commit;
10			

c) (10 puntos) Diseñe el plan físico optimizado del siguiente query y escriba el query optimizado final en SQL:

T1(idA, nombre, codigo)

T2(idB, idA)

T3(idC, idB, descripcion, valor)

SELECT idC, nombre

FROM T1, T2, T3

WHERE T1.idA = T2.idA AND T2.idB = T3.idB

AND T1.codigo= 'EC10345' and T1.nombre like 'A%'

AND T2.categoria = 'GENERICO'

AND T3.descripcion like '%Test%' AND T3.valor < 10

Figura 1: Esquema relacional de la base de datos db_books

