



Enunciado:

ANEXO I - DESARROLLO

Para Esquema E1:

A. TRIGGERS Y SERVICIOS

1. Se le agregara columna cantidad_salas (cantidad de salas de convención) y cantidad_oficinas (cantidad de oficinas regulares) a la tabla CLIENTE. Es necesario mantener actualizadas las columnas cantidad_salas y cantidad_oficinas con la cantidad de salas de convención y oficinas regulares que cada cliente tiene alquiladas por tiempo indeterminado (es decir que aún no tienen fecha de fin). Se debe realizar con triggers FOR STATEMENT.
2. Utilizando 2 vistas V_OFICINA_REGULAR y V_SALA_CONVENCION que contienen todos los datos de las oficinas regulares o de las salas de convención respectivamente, construir los triggers INSTEAD OF necesarios para mantener actualizadas las tablas de OFICINA, OFICINA_REG y SALA_CONVENCION de manera de respetar el diseño de datos de la jerarquía.

B. VISTAS

1. Construya una vista V_CLIENTES_COMP que contenga las oficinas que han sido alquiladas por todos los clientes.
2. Construya una vista V_OFICINAS_REG que liste para cada oficina su identificador, su tipo, su superficie, su monto de alquiler y la cantidad promedio de escritorios por superficie.

Nota: El cambio de nombres del archivo de creación ya fue realizado y se consultara en la clase si cumple los requerimientos.

A TENER EN CUENTA:

A. SERVICIOS

Implemente lo siguiente con el recurso que crea más conveniente (triggers, procedimientos y/o vistas). Si necesita para este ítem crear tablas adicionales al esquema creado en el script de creación (GXX_Creacion.sql) incorpórelas en el script **GXX_Cambios.sql**.

B. DEFINICIÓN DE VISTAS

Escriba la sentencia SQL para crear las vistas (En lo posible actualizables en Postgresql) e incorpórelas en el script **GXX_Cambios.sql**. En cada caso comente y ejemplifique en el script si es actualizable o no, indicando la/s causas/s.



Consideraciones Generales: (TODO DEBE TENER GR01)

El prefijo de grupo **<GRXX>** se usará en tablas, vistas, procedimientos almacenados, triggers, índices y constraints. XX se deberá reemplazar por el número de grupo asignado por la cátedra en cada caso. La nomenclatura a seguir es:

1. Tablas/Vistas: **<GRXX>_<nombre_tabla o vista>**
2. Índices: **IDX_<GRXX>_<nombre_indice>**
3. Constraints
 - **Clave Primaria: PK_<GRXX>_<nnnn>**
donde: nnnn = el nombre de la tabla para la que la constraint se construye (Por ejemplo: PK_GRXX_Beneficiario)
 - **Clave Extranjera: FK_<GRXX>_<pppp>_<cccc>**
donde pppp = el nombre de la tabla referenciante u original, cccc = el nombre de la tabla referenciada.
 - **Unique: UQ_<GRXX>_<nnnn>_<cccc>**
donde: nnnn = el nombre de la tabla para la que la constraint se construye, cccc = el nombre del campo interviniente.
4. Procedimientos, Triggers y Funciones
 - **Triggers: TR_<GRXX>_<tbl>_<nnnn>**
siendo <nnnn> el nombre que considere adecuado para el trigger y <tbl> la tabla a la cual está asociada el trigger.
 - **Procedimientos: PR_<GRXX>_<nnnn>**
siendo <nnnn> el nombre que considere adecuado para el procedimiento.
 - **Funciones: FN_<GRXX>_<nnnn>**
siendo <nnnn> el nombre que considere adecuado para la función.
 - **Funciones de triggers: TRFN_<GRXX>_<nnnn>**
siendo <nnnn> el nombre que considere adecuado para la función de trigger.

RECORDAR: (no borrar nada si algo no anda, dejarlo comentado y documentado)

al momento de realizar los script que:

Aquellos **insert, update o delete** que provocan fallos de por alguna restricción, es decir los que son de test de restricciones deben colocarse como comentarios **/* */**.

Recuerde colocarle **CASCADE** a las sentencias drop table si realiza el borrado de las tablas en cualquier orden, sino tiene que respetar el orden dado por las foreign key.

Una función que es llamada por un trigger debe estar dentro del script siempre antes del create trigger que la invoca.

Recuerde que aquellas sentencias que Postgresql no soporta deben ir comentariadas **/* */**