

## Práctica de T-SQL

**Según el modelo dado resuelva:**

1. Hacer una función que dado un artículo y un deposito devuelva un string que indique el estado del depósito según el artículo. Si la cantidad almacenada es menor al límite retornar “OCUPACION DEL DEPOSITO XX %” siendo XX el % de ocupación. Si la cantidad almacenada es mayor o igual al límite retornar “DEPOSITO COMPLETO”.
2. Realizar una función que dado un artículo y una fecha, retorne el stock que existía a esa fecha
3. Cree el/los objetos de base de datos necesarios para corregir la tabla empleado en caso que sea necesario. Se sabe que debería existir un único gerente general (debería ser el único empleado sin jefe). Si detecta que hay más de un empleado sin jefe deberá elegir entre ellos el gerente general, el cual será seleccionado por mayor salario. Si hay más de uno se seleccionara el de mayor antigüedad en la empresa. Al finalizar la ejecución del objeto la tabla deberá cumplir con la regla de un único empleado sin jefe (el gerente general) y deberá retornar la cantidad de empleados que había sin jefe antes de la ejecución.
4. Cree el/los objetos de base de datos necesarios para actualizar la columna de empleado empl\_comision con la sumatoria del total de lo vendido por ese empleado a lo largo del último año. Se deberá retornar el código del vendedor que más vendió (en monto) a lo largo del último año.
5. Realizar un procedimiento que complete con los datos existentes en el modelo provisto la tabla de hechos denominada Fact\_table tiene la siguiente definición:

```
Create table Fact_table  
( anio char(4),  
  mes char(2),  
  familia char(3),  
  rubro char(4),  
  zona char(3),  
  cliente char(6),  
  producto char(8),  
  cantidad decimal(12,2),  
  monto decimal(12,2)  
)
```

```
Alter table Fact_table
```

```
Add constraint primary key(anio,mes,familia,rubro,zona,cliente,producto)
```

6. Realizar un procedimiento que si en alguna factura se facturaron componentes que conforman un combo determinado (o sea que juntos componen otro producto de mayor nivel), en cuyo caso deberá reemplazar las filas correspondientes a dichos productos por una sola fila con el producto que componen con la cantidad de dicho producto que corresponda.
7. Hacer un procedimiento que dadas dos fechas complete la tabla Ventas. Debe insertar una línea por cada artículo con los movimientos de stock realizados entre esas fechas. La tabla se encuentra creada y vacía.

## VENTAS

Código	Detalle	Cant. Mov.	Precio de Venta	Renglón	Ganancia
Código del articulo	Detalle del articulo	Cantidad de movimientos de ventas (Item factura)	Precio promedio de venta	Nro. de línea de la tabla	Precio de Venta – Cantidad * Costo Actual

8. Realizar un procedimiento que complete la tabla Diferencias de precios, para los productos facturados que tengan composición y en los cuales el precio de facturación sea diferente al precio del cálculo de los precios unitarios por cantidad de sus componentes, se aclara que un producto que compone a otro, también puede estar compuesto por otros y así sucesivamente, la tabla se debe crear y está formada por las siguientes columnas:

## DIFERENCIAS

Código	Detalle	Cantidad	Precio_generado	Precio_facturado
Código del articulo	Detalle del articulo	Cantidad de productos que conforman el combo	Precio que se compone a través de sus componentes	Precio del producto

9. Crear el/los objetos de base de datos que ante alguna modificación de un ítem de factura de un artículo con composición realice el movimiento de sus correspondientes componentes.
10. Crear el/los objetos de base de datos que ante el intento de borrar un artículo verifique que no exista stock y si es así lo borre en caso contrario que emita un mensaje de error.

11. Cree el/los objetos de base de datos necesarios para que dado un código de empleado se retorne la cantidad de empleados que este tiene a su cargo (directa o indirectamente). Solo contar aquellos empleados (directos o indirectos) que sean errores que su jefe directo.
12. Cree el/los objetos de base de datos necesarios para que nunca un producto pueda ser compuesto por sí mismo. Se sabe que en la actualidad dicha regla se cumple y que la base de datos es accedida por n aplicaciones de diferentes tipos y tecnologías. No se conoce la cantidad de niveles de composición existentes.
13. Cree el/los objetos de base de datos necesarios para implantar la siguiente regla “Ningún jefe puede tener un salario mayor al 20% de la suma de los salarios de sus empleados totales (directos + indirectos)”. Se sabe que en la actualidad dicha regla se cumple y que la base de datos es accedida por n aplicaciones de diferentes tipos y tecnologías
14. Agregar el/los objetos necesarios para que si un cliente compra un producto compuesto a un precio menor que la suma de los precios de sus componentes que imprima la fecha, que cliente, que productos y a qué precio se realizó la compra. No se deberá permitir que dicho precio sea menor a la mitad de la suma de los componentes.
15. Cree el/los objetos de base de datos necesarios para que el objeto principal reciba un producto como parametro y retorne el precio del mismo.  
Se debe prever que el precio de los productos compuestos sera la sumatoria de los componentes del mismo multiplicado por sus respectivas cantidades. No se conocen los niveles de anidamiento posibles de los productos. Se asegura que nunca un producto esta compuesto por si mismo a ningun nivel. El objeto principal debe poder ser utilizado como filtro en el where de una sentencia select.
16. Desarrolle el/los elementos de base de datos necesarios para que ante una venta automaticamente se descuenten del stock los articulos vendidos. Se descontaran del deposito que mas producto posea y se supone que el stock se almacena tanto de productos simples como compuestos (si se acaba el stock de los compuestos no se arman combos)  
En caso que no alcance el stock de un deposito se descontara del siguiente y asi hasta agotar los depositos posibles. En ultima instancia se dejara stock negativo en el ultimo deposito que se desconto.

17. Sabiendo que el punto de reposición del stock es la menor cantidad de ese objeto que se debe almacenar en el depósito y que el stock máximo es la máxima cantidad de ese producto en ese depósito, cree el/los objetos de base de datos necesarios para que dicha regla de negocio se cumpla automáticamente. No se conoce la forma de acceso a los datos ni el procedimiento por el cual se incrementa o descuenta stock
18. Sabiendo que el límite de crédito de un cliente es el monto máximo que se le puede facturar mensualmente, cree el/los objetos de base de datos necesarios para que dicha regla de negocio se cumpla automáticamente. No se conoce la forma de acceso a los datos ni el procedimiento por el cual se emiten las facturas
19. Cree el/los objetos de base de datos necesarios para que se cumpla la siguiente regla de negocio automáticamente “Ningún jefe puede tener menos de 5 años de antigüedad y tampoco puede tener más del 50% del personal a su cargo (contando directos e indirectos) a excepción del gerente general”. Se sabe que en la actualidad la regla se cumple y existe un único gerente general.
20. Crear el/los objeto/s necesarios para mantener actualizadas las comisiones del vendedor.  
El cálculo de la comisión está dado por el 5% de la venta total efectuada por ese vendedor en ese mes, más un 3% adicional en caso de que ese vendedor haya vendido por lo menos 50 productos distintos en el mes.
21. Desarrolle el/los elementos de base de datos necesarios para que se cumpla automáticamente la regla de que en una factura no puede contener productos de diferentes familias. En caso de que esto ocurra no debe grabarse esa factura y debe emitirse un error en pantalla.
22. Se requiere recategorizar los rubros de productos, de forma tal que ningún rubro tenga más de 20 productos asignados, si un rubro tiene más de 20 productos asignados se deberán distribuir en otros rubros que no tengan más de 20 productos y si no entran se deberá crear un nuevo rubro en la misma familia con la descripción “RUBRO REASIGNADO”, cree el/los objetos de base de datos necesarios para que dicha regla de negocio quede implementada.
23. Desarrolle el/los elementos de base de datos necesarios para que ante una venta automáticamente se controle que en una misma factura no puedan venderse más de dos productos con composición. Si esto ocurre deberá rechazarse la factura.

