

Gestión de Datos - Práctica de T-SQL Adicional

1. Escriba una función que dados los parámetros código de fabricante y periodo retorne la diferencia de tiempo (en el periodo indicado S:semanas, D: días) entre el promedio de tiempo de embarque de las ordenes que están compuestas solo por productos que él fabrica y el tiempo que el proveedor demora en la entrega de dichos productos. Si alguno de los parámetros es incoherente se retornara el valor -1
2. Cree una nueva tabla prod_mismo_pedido (stock_num1,manu_code1,stock_num2,manu_code2,cantidad). Una vez creada la tabla cree un objeto de base de datos que al ser ejecutado con un parámetro cant_min int, complete la tabla con la lista de productos que hayan sido vendidos juntos en al menos cant_min ocasiones. Se considera que han sido vendidos juntos si se encuentran en la misma orden. El campo cantidad debe ser completado con la cantidad que veces que se vendieron juntos. El ejercicio debe ser resuelto con y sin la utilización de cursores
3. Cree una nueva tabla Cust_extrm (cust_num, total_orders, total_paid, position). Una vez creada la tabla cree un objeto de base de datos que al ser ejecutado con un parámetro n int, complete la tabla con los n clientes que más órdenes tengan relacionadas y con los n clientes que menos ordenes tengan relacionadas. Para ambos casos se deberá completar cust_num con el código de cliente , total orders con la cantidad de ordenes asociadas a él, total_paid con la suma de todo lo que haya abonado por las ordenes y position con '+' en caso que sea del grupo de los que más ordenes tiene y '-' si pertenece al grupo de los que menos ordenes tiene. En caso que no existan n clientes la tabla se completara con la totalidad de los clientes tant en el grupo + como en el grupo -. Si existe más de 1 cliente con la misma cantidad de órdenes se considerara el customer_num para definir cual se inserta preponderando siempre al de menor número El ejercicio debe ser resuelto con y sin la utilización de cursores
4. Cree un objeto de base de datos que permita recodificar los fabricantes sin alterar la estructura de datos y sin perder información en ningún caso. Defina los parámetros necesarios del objeto y las validaciones que considere necesarias.
5. Cree el/los objetos de base de datos necesarios para que se revise (y modifique los datos de ser necesario) el cumplimiento de una nueva regla de negocio que dice "Por cada 2 unidades que ordene un cliente que correspondan a un mismo fabricante se aplicará un descuento del 2% acumulativo (2 unidades 2%, 4 unidades 4%, etc. Hasta un tope de 20% máximo) en las ordenes del mes siguiente de dicho cliente que tengan al menos 1 producto de ese fabricante"
6. Cree el/los objetos de base de datos necesarios para que automáticamente se cumpla la siguiente regla de negocio sin alterar la estructura de las tablas. La regla a implementar es "Las fechas de la tabla orders deben si o si cumplir con la siguiente regla order_date< ship_date < paid_date. En caso que alguna de las fechas sea null solo se realizaran las verificaciones posibles". No se conoce como se accede a la tabla orders ni como se manipulan los datos de la misma
7. Cree una tabla y el/los objetos de base de datos necesarios para almacenar una auditoria detallada de la tabla Stock. En dicha tabla se deben almacenar todos los datos posibles para poder identificar como se llevo al estado de cada uno de los registros. ". No se conoce como se accede a la tabla stock ni como se manipulan los datos de la misma
8. Altere la tabla call_type agregando una columna counter y cree el/los elementos de base de datos necesarios para que en dicha columna se vaya almacenando automáticamente la cantidad de llamados de cada tipo que se realizan.
9. Cree el/los objetos de base de datos necesarios para que automáticamente se cumpla la siguiente regla de negocio sin alterar la estructura de las tablas. La regla a implementar es "Ninguna orden con backlog=y puede tener más de 10 items". No se conoce como se accede a las tablas orders e items ni como se manipulan los datos de las misma
10. Cree el/los objetos de base de datos necesarios para que ante un intento de borrado de un estado en vez de ejecutarse el borrado físico se modifique su descripción por "Estado baja". No se conoce como se accede a la tabla State ni como se manipulan los datos de la misma