

Base de datos II

Práctica para Checkpoint I

En esta práctica vamos a trabajar con la base de datos Emarket.

Arranquemos con el ejercicio.

¡Buena suerte!



Consigna

1. Necesitamos crear una función que dada la cantidad de unidades pedidas y el nombre de un producto, validemos que tengamos la cantidad de unidades en stock necesaria para cubrir el pedido.
2. Necesitamos crear una tabla de auditoría para la tabla de facturas. Crear algunos triggers para cada vez que insertamos registros en la tabla, para cada vez que modificamos registros y también para cuando eliminamos. La tabla tiene que tener un id propio, el id de la factura, la acción que se realizó, el responsable de la misma y la fecha y hora. La tabla se genera una sola vez y queda registrada en nuestra base de datos (no hace falta que esté en los triggers).
3. Crear un stored procedure SP_Genera_Factura que, dado un nombre de un producto, la cantidad del mismo, el nombre del cliente (compañía), el nombre del empleado, apellido del mismo y el nombre de la compañía de correo, genere el detalle de la factura y el encabezado de la misma. Nuestras facturas, solo van a tener 1 detalle y 1 registro de factura.
Tener en cuenta:
 - No validar lógicamente si no encontramos un empleado o un producto, etc. por su nombre.
 - No se puede generar una factura que supere la cantidad de unidades en stock (usar la función del punto 1).
 - Para las facturas, para todas las fechas usar el día y hora actual. El envío vía es el id de nuestro correo, el valor del transporte es gratis y el nombre es FULL. La dirección de la factura y de envío es la misma que la del cliente.
 - Para los detalles de facturas, no tenemos descuentos nunca y el precio unitario lo obtenemos del producto.



- Cuando guardemos las facturas, vamos a estar utilizando indirectamente el trigger del punto 2.
- No hace falta modificar el atributo UnidadesEnStock del registro del producto.

Además, agregar las transacciones y handlers que sean necesarios. Cuando estemos bajo un error, la transacción tiene que realizar un ROLLBACK.

4. Luego de un tiempo, nuestros clientes nos pidieron que el SP_Genera_Factura, debe ser más inteligente. Cuando intentamos generar una factura que supera el stock del producto indicado, insertar un registro en la tabla pedidoproveedor donde vamos a indicar el id del artículo que necesitamos, el id del proveedor, la fecha del pedido y la cantidad que necesitamos (la diferencia entre las unidades en stock y las que necesitamos para la factura). Una vez hecho esto, devolvemos un mensaje: 'No hay stock por el momento, pero ya se hizo el pedido a nuestro proveedor para contar con las unidades faltantes'.

Crear una tabla llamada pedidoproveedor con los datos necesarios antes de realizar la modificación del SP_Genera_Factura. La tabla se genera una sola vez y queda registrada en nuestra base de datos (no hace falta que esté en el SP).

5. Ahora vamos a generar un nuevo SP llamado SP_Genera_Factura_Varios_productos donde recibimos la misma cantidad de parámetros que SP_Genera_Factura, pero en vez de recibir el nombre de 1 producto, vamos a recibir los ids de varios productos en un VARCHAR.

Tener en cuenta:

- Usar un cursor para recorrer todos los registros de los productos. Tip para filtrar los productos por los ids en un VARCHAR: FIND_IN_SET.



- Debemos registrar 1 solo registro de factura y 1 registro por cada producto que nos devuelve nuestra query.