

EVALUACION	Parcial 1	GRUPO	N3A	FECHA	05/05/2022
MATERIA	Base de Datos 2				
CARRERA	Analista Programador/Analista en Tecnologías de la Información				
CONDICIONES	Puntos 30 Sin material Duración 2 horas				

Utilizando la base de datos *NorthWind* vista en clase (se adjunta modelo de datos) realizar los siguientes ejercicios:

1. Se tiene una nueva tabla *ProductFailures* que tiene como objetivo almacenar las roturas de producto que se van dando. La tabla tiene la siguiente estructura:

ProductFailure (**FailureId**, *FailureTimestamp*, *ProductId*, *Units*, *UnclaimableAmount*, *UserName*)

Se sabe que el identificador de la tabla es un número autoincremental, *FailureTimestamp* almacena fecha y hora en la que ocurrió la rotura, *ProductId* hace referencia al Identificador de Producto de la tabla *Products*, *Units* es la cantidad de unidades que se rompieron, *UnclaimableAmount* es la cantidad de dinero en U\$S que no se va a poder recuperar pues nadie reembolsa ese monto que se registra con precisión 2 decimales, *UserName* es un alfanumérico de largo máximo 30 caracteres que almacena el nombre de usuario que registró la rotura de producto.

Se debe considerar:

- Username es el único campo de esta tabla que puede contener valores nulos.
- Units es un entero mayor que cero
- UnclaimableAmount es mayor o igual que cero
- La combinación de los campos *FailureTimestamp* y *ProductId* no se puede repetir

Se pide:

- Definir mediante DDL el script de creación de dicha tabla incluyendo todas las restricciones relacionales y de reglas de negocio planteadas
- Crear el/los índices necesarios en esta nueva tabla según los criterios vistos en clase

2. Se quiere realizar un incremento masivo de un 3% del precio de producto para los productos frágiles, que son los que tuvieron roturas con monto irrecuperable (UnclaimableAmount mayor que cero)

Esto consta de 2 pasos que debe resolver:

- 2.1. En la siguiente tabla "ProductChanges" que ya está definida, se debe crear una tupla por cada producto que se le va a modificar el precio (para registrar el precio actual del Producto previo a la modificación)

```
Create table ProductChanges (PChangeld int identity(1,1) not null,  
PChangeTimestamp datetime not null,  
ProductId int not null,  
UnitPrice decimal(10,4) not null,  
UserName varchar(30) not null,  
constraint PKProductChanges Primary Key (PChangeld),  
constraint FKProducto Foreign Key (ProductId) references Products(ProductId)  
)
```

Considerar para esta carga:

El campo PChangeTimestamp se debe cargar con la fecha-hora del momento

El campo Username se carga con el valor fijo 'EncargadoRoturas'

Tip: puede utilizar la función getdate() que retorna un valor de tipo datetime con la fecha y hora actual

- 2.2. En el segundo paso, se debe actualizar efectivamente el precio de producto(UnitPrice)
3. Resolver mediante SQL los siguientes ejercicios:
- a. Listar nombre de producto de los productos que tienen Categoría = 1 (CategoryID) y fueron enviados al País "Germany"
Consideración: listar cada producto una sola vez
- b. Para cada Proveedor (SupplierId) listar la cantidad total de unidades producto (Quantity) que vendió el proveedor considerando las ordenes del año 1997 (OrderDate)
Se debe listar solamente los Proveedores que vendieron más de 100 unidades

(Cada ejercicio vale 10 puntos)

