

```
create database Jamones
go
use Jamones
go

-- Tabla de empresas que nos suministran los jamones.
create table Empresa
(
    idEmpresa int primary key,
    nombre varchar (100)
)
-- Tabla de jamones
go
create table Jamon
( idJamon bigint identity primary key, -- va a ser nuestro identificador del
  jamon para nosotros.
  NumeroSerie varchar(30) not null, -- numero de serie del jamon
  idEmpresa int not null references Empresa(idEmpresa) on update cascade ,--
  empresa a la que compré el jamon
  peso int check (peso >0) , -- lo expresaremos en gramos para evitar parte
  decimal.
  precioKilo money , -- eso significa que el precio por gramo es
  precioKilo/1000.
  precioGramo as (precioKilo / 1000.) -- ¡¡COLUMNA CALCULADA!! Eso es
  compute by en otras bases de datos!!!!
)
go
create UNIQUE index AK_Jamon on Jamon (idEmpresa,NumeroSerie) -- hace que no
se repita el numero
-- de serie para
la misma empresa.
go
-- tabla de ventas.

create table Venta
(
    idVenta bigint identity primary key, -- identificador de la venta.
    fecha date default getdate(), -- la fecha venta , si no se dice nada , será la
    actual
    idJamon bigint not null references jamon(idJamon) on update cascade,
    pesoEnGramos int not null check (pesoEnGramos > 0)
)
GO

/* 1º
El sistema no puede dejar que se venda más peso de un jamón al peso
indicado en el jamón. (2,5 puntos)
*/

/*
Esto solo puede ocurrir cuando estamos insertando o modificando una venta,
luego hacemos un trigger asociado a la tabla venta.
*/
```

```
create trigger trTestPeso on venta for insert, update
as
-- 1º averiguar de qué jamón se trata:

declare @idJamon int
set @idJamon=(select idjamon from inserted)

-- 2º obtener el peso total del jamon que estamos vendiendo

declare @pesoJamonEnGramos int

set @pesoJamonEnGramos=(select peso from Jamon where idJamon=@idJamon)

-- 3º Calculamos el peso total de las ventas de ese jamon

declare @sumaVentas int

set @sumaVentas=(select SUM(@pesoJamonEnGramos) from Venta where
    idJamon=@idJamon)

-- comprobamos si nos hemos pasado

if @sumaVentas > @pesoJamonEnGramos
begin
    raiserror ('No puede vender más jamon que el que hay',16,1)
    rollback transaction
end
go

/* 2º
Una vista que nos dé la cantidad (peso) de jamón vendido cada día.
*/

create view vwVentasPorDia (fecha,"Cantidad vendida en gramos")
as
select fecha , SUM(pesoEngramos) from Venta
group by fecha

go

/* 3º
Un procedimiento almacenado al que le pasamos el nombre de la
empresa y nos da la cantidad de jamones que le hemos comprado.
*/

create procedure spCompradoAEmpresa @nombreEmpresa varchar(100)
as
declare @idEmpresa int -- 1º obtener el id de la empresa.

set @idEmpresa =(select idEmpresa from Empresa where nombre= @nombreEmpresa)
```

```
if @idEmpresa is null -- no existe
begin
    raiserror ('No existe esta empresa',16,1)
    return
end

-- sacar los datos pedidos

select COUNT(*) "Jamones comprados" from Jamon where idEmpresa=@idEmpresa
go

/* 4º
Un procedimiento almacenado al que le pasamos el número de serie de un jamón y ↗
nos da
cuanto dinero hemos conseguido vendiéndolo en paquetitos.
*** DETALLE FALTARÍA MANDAR TAMBIÉN EL IDEMPRESA o nombre de la empresa
, PERO NO SE PUSO EN EL EXAMEN**
*/

create procedure spGanancia @numSerieJamon varchar(30) -- , @idEmpresa int ↗
(no se pidio empresa)
as

--1º pasar el numero de serie del jamon a idJamon

declare @idJamon int

set @idJamon=(select idJamon from Jamon where NumeroSerie=@numSerieJamon
-- también haría falta poner " and idEmpresa = @idEmpresa ", pero ↗
no se pidio en el enunciado
)

if (@idJamon is null)
begin
    raiserror ('No existe este jamon',16,1)
    return
end

-- 2º vemos el total de las ventas = gramos vendido

declare @vendido int

set @vendido=(select SUM(pesoEnGramos) from Venta where idJamon=@idJamon)

-- 3º obtener cuanto vale gramo de este jamón.

declare @precioGramo money

set @precioGramo =(select precioGramo from Jamon where idJamon=@idJamon)
/* si no tenemos el campo calculado :
set @precioGramo =(select precioGramo from Jamon where ↗
```

```
        idJamon=@idJamon) / 1000.  
*/  
  
-- 4º obtemos el dinero conseguido:  
  
select ( @precioGramo*@vendido) "Dinero obtenido en su venta"  
  
go
```