

Tarea para BD02.

Detalles de la tarea de esta unidad.

Enunciado.

Realiza los ejercicios en la Hoja de trabajo de `SQLDeveloper` o desde `SQLPlus` conectando con tu usuario, creado en la primera tarea, y comprueba que las sentencias son correctas antes de incluirlas en el script. Una vez hayas hecho todas para probar el script, desde `SQLPlus`, ejecuta `start nombrescript.sql` o `@nombrescript.sql` eliminando previamente todas las tablas creadas para poder crearlas de nuevo.

EJERCICIO 1:

Vamos a crear las tablas para una tienda virtual que distribuye productos agrupados en familias en varias tiendas.

Realiza un script llamado `creatienda.sql` que implemente los ejercicios descritos a continuación.

Precede cada una de las sentencias `SQL` de los ejercicios con un comentario que incluya el enunciado del ejercicio correspondiente. Recuerda que los comentarios van precedidos del símbolo `--` al inicio de la línea.

Con las sentencias `DDL` de `SQL` crea la tablas especificadas a continuación aplicando las restricciones (`constraints`) pedidas. Se debe cumplir la integridad referencial.

TABLA FAMILIA: Contiene las familias a las que pertenecen los productos, como por ejemplo ordenadores, impresoras, etc.

Nombre Columna	Descripción	Tipo dato	Restricciones
Codfamilia	Código que distingue una familia de otra	Numérico de 3 dígitos	Clave primaria.
Denofamilia	Denominación de la familia	Alfanumérico de 50 caracteres	No puede haber dos familias con la misma denominación. Debe tener contenido.

TABLA PRODUCTO => contendrá información general sobre los productos que distribuye la empresa a las tiendas.

Nombre Columna	Descripción	Tipo dato	Restricciones
Codproducto	Código que distingue un producto de otro	Numérico de 5 dígitos	Clave primaria.
Denoproducto	Denominación del producto	Alfanumérico de 20 caracteres	Debe tener contenido.
Descripcion	Descripción del producto	Alfanumérico de 100 caracteres	
PrecioBase	Precio base del producto	Numérico de 8 dígitos dos de ellos decimales	Mayor que 0. Debe tener contenido.
PorcReposición	Porcentaje de reposición aplicado a ese producto. Se utilizará para aplicar a las unidades mínimas y obtener el número total de unidades a reponer cuando el stock esté bajo mínimo	Numérico de 3 dígitos	Mayor que 0

UnidadesMinimas	Unidades mínimas recomendables en almacen	Numérico de 4 dígitos	Mayor que 0. Debe tener contenido.
Codfamilia	Código de la familia a la que pertenece el producto	Numérico de 3 dígitos	Clave ajena, referencia a Codfamilia de la tabla FAMILIA. Debe tener contenido.

TABLA TIENDA=> contendrá información básica sobre las tiendas que distribuyen los productos.

Nombre Columna	Descripción	Tipo dato	Restricciones
Codtienda	Código que distingue una tienda de otra.	Numérico de 3 dígitos	Clave primaria.
Denotienda	Denominación o nombre de la tienda.	Alfanumérico de 20 caracteres	Debe tener contenido.
Telefono	Teléfono de la tienda	Alfanumérico de 11 caracteres	
CodigoPostal	Codigo Postal donde se ubica la tienda	Alfanumérico de 5 caracteres	Debe tener contenido.
Provincia	Provincia donde se ubica la tienda	Alfanumérico de 5 caracteres	Debe tener contenido.

TABLA STOCK => Contendrá para cada tienda el número de unidades disponibles de cada producto. La **clave primaria** está formada por la concatenación de los campos Codtienda y Codproducto.

Nombre Columna	Descripción	Tipo dato	Restricciones	
Codtienda	Código de la tienda.	Numérico de 3 dígitos	Clave primaria: (Codtienda,Codproducto) Permite que un producto pueda aparecer en varias tiendas, y que en una tienda puedan haber varios productos.	Clave ajena, referencia a Codtienda de la tabla tienda. Debe tener contenido.
Codproducto	Código del producto	Numérico de 5 dígitos		Clave ajena, referencia a Codproducto de la tabla PRODUCTO. Debe tener contenido.
Unidades	Unidades de ese producto en esa tienda	Numérico de 6 dígitos.	Mayor o igual a 0. Debe tener contenido.	

EJERCICIO 2:

A) Modificar las tablas creadas en el ejercicio anterior siguiendo las indicaciones. Los ejercicios se incluirán en un script llamado `ModificaTienda.sql`. Cada uno de ellos, como en el ejercicio anterior, irá precedido de un comentario con el enunciado.

- Añadir a la tabla STOCK
 - Una columna de tipo fecha llamada `FechaUltimaEntrada` que por defecto tome el valor de la fecha actual.
 - Una columna llamada `Beneficio` que contendrá el tipo de porcentaje de beneficio que esa tienda aplica en ese producto. Se debe controlar que el valor que almacene sea 1,2, 3, 4 o 5.
- En la tabla PRODUCTO
 - Eliminar de la tabla producto la columna `Descripción`.
 - Añadir una columna llamada `perecedero` que únicamente acepte los valores: S o N.
 - Modificar el tamaño de la columna `Denoproducto` a 50.
- En la tabla FAMILIA
 - Añadir una columna llamada `IVA`, que represente el porcentaje de IVA y únicamente pueda contener los valores 21,10,ó 4.
- En la tabla tienda
 - La empresa desea restringir el número de tiendas con las que trabaja, de forma que no pueda haber más de una tienda en una misma zona (la zona se identifica por el código postal). Definir mediante DDL las restricciones necesarias para que se cumpla en el campo correspondiente..

B) Renombra la tabla STOCK por PRODXTIENDAS.

C) Elimina la tabla FAMILIA y su contenido si lo tuviera.

D) Crea un usuario llamado `C##INVITADO` siguiendo los pasos de la unidad 1 y dale todos los privilegios sobre la tabla PRODUCTO.

E) Retira los permisos de modificar la estructura de la tabla y borrar contenido de la tabla PRODUCTO al usuario anterior.

EJERCICIO 3:

SQLDeveloper permite obtener el diagrama del modelo entidad relación a partir de las tablas ya creadas con la información contenida en el Diccionario de Datos. Una vez tengas realizados los ejercicios 1 y 2 genera el diagrama entidad relación y expórtalo en formato PNG.

En este enlace tienes los pasos a seguir

[SQL Developer: Obtener Entidad/Relacion](#)