Leticia Costan

12-12-2024

PROYECTO 1. Diseño de BBDD relacional para una tienda online

### Proyecto 1. Diseñar una base de datos relacional para una tienda online.

**La escuela Bojo Bushido de Tronchavacas de Arriba necesita ayuda. Cada mes tiene que mandar un pedido a Japón con material necesario para su alumnado. Se está complicando la cuestión porque cada vez hay más alumnos. Necesitan la ayuda para hacer una base de datos para una tienda online propia de la escuela.**

**El desafío es crear un esquema de base de datos relacional parar la escuela, en la que el alumnado pueda indicar sus necesidades de material. Después habrá que diseñar el esquema, aplicando la normalización para reducir la redundancia de datos y evitar las anomalías de inserción, actualización y eliminación. Diseñar las tablas considerando las mejores prácticas de definición de claves primarias y foráneas para establecer relaciones entre las entidades. Realizar un diagrama de entidad-relación para visualizar las entidades, atributos y relaciones. Por último, habrá que escribir las instrucciones SQL para crear la base de datos, las tablas y las relaciones entre ellas, incluyendo restricciones de integridad como claves primarias, claves foráneas y cualquier otra restricción que se considere necesaria.**

**Luego habrá que subir al portfolio un documento con el código .sql que contenga las instrucciones SQL necesarias parar crear la base de datos, las tablas y establecer las relaciones; dicho código deberá poder ejecutarse en un SGBD y que cree la estructura de la base de datos diseñada, junto con el diagrama E/R.**

**Entidades:**

* **Productos**
  + **ID de producto**
  + **Nombre producto**
  + **Descripción producto**
  + **Precio producto**
  + **Stock**
* **Clientes**
* **Pedidos**
* **Detalles de pedidos**

#### Guía de evaluación

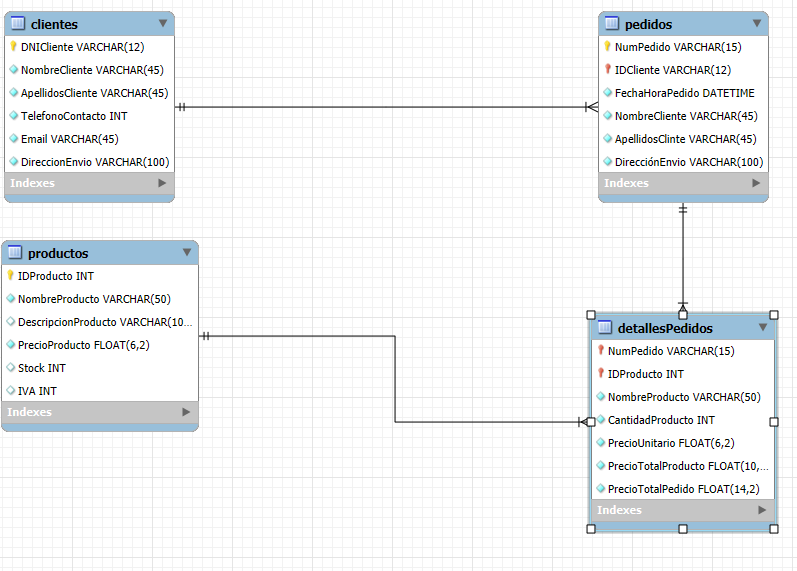
|  |  |  |  |
| --- | --- | --- | --- |
|  | Superado con honores | Superado | No superado |
| Diseño del esquema | **Diseñar el esquema, aplicando conceptos y normalización y diseñar las tablas considerando las mejores prácticas de definición de claves primarias y foráneas.** | Diseñar el esquema cumpliendo los requisitos esenciales. | No ser capaz de diseñar el esquema. |
| Establecimiento de relaciones | **Establecer las relaciones entre las entidades y realizar un diagrama entidad-relación, permitiendo entender mejor la estructura de la base de datos y facilitar la implementación.** | Establecer las relaciones entre las entidades, aunque no se haya realizado el diagrama entidad-relación | No ser capaz de establecer las relaciones |
| Instrucciones SQL | **Escribir las instrucciones SQL para crear la base de datos, las tablas y las relaciones entre ellas, incluyendo las restricciones de integridad, como claves primarias, claves foráneas u otras.** | Escribir las instrucciones SQL para crear la base de datos, las tablas y las relaciones entre ellas, de manera esencial. | No ser capaz de escribir las instrucciones SQL. |
| Documento .sql | **Subir un documento .sql al portfolio. Haberlo probado que se puede ejecutar en un sistema gestor de bases de datos y sus resultados son óptimos para el uso de la tienda.** | Subir un documento .sql al portfolio. Haberlo probado que se pueda ejecutar en un sistema gestor de bases de datos. Su utilidad para la tienda online es suficiente. | No ser capaz de crear el documento .sql |

### Entidades y atributos

* Productos
  + ID producto
  + Nombre producto
  + Descripción producto
  + Precio producto
  + Stock
  + IVA
* Clientes
  + Nombre
  + Apellidos
  + DNI
  + Teléfono
  + Email
  + Dirección\*
* Pedidos
  + Número pedido
  + Fecha y hora del pedido
  + Nombre cliente\*
  + Apellidos cliente\*
  + Dirección envío\*
* Detalles del pedido
  + Numero pedido
  + Nombre producto\*
  + Cantidad productos
  + Precio unitario\*
  + Precio total de producto
  + Precio total pedido
  + Gastos envío

\* Productos añadidos a través de una consulta.

### Diagrama E/R



### Normalización

* **Clientes:** 
  + DNICliente VARCHAR(12)
  + NombreCliente VARCHAR(45)
  + ApellidosClientes VARCHAR (45)
  + TelefonoContacto INT pe
  + Email VARCHAR (45)
  + DirecciónEnvio VARCHAR (100)
  + PRIMARY KEY: DNICliente
  + UNIQUE: DNICliente
  + NOT NULL: DNICliente, NombreCliente, ApellidosCliente, TelefonoContacto
* **Productos:** 
  + IDProducto INT
  + NombreProducto VARCHAR(50)
  + DescripciónProducto VARCHAR(100)
  + PrecioProducto DOUBLE (6,2)
  + Stock INT(3)
  + IVA INT(2)
  + PRIMARY KEY: IDProducto
  + UNIQUE: IDProducto
  + NOT NULL: IDProducto, NombreProducto, PrecioProducto, Stock, IVA
* **Pedido:**
  + NumPedido VARCHAR(15)
  + FechaHoraPedido DATETIME
  + NombreCliente VARCHAR(45)
  + ApellidosClientes VARCHAR (45)
  + DireccionEnvio VARCHAR(100)
  + IDCliente VARCHAR(12)
  + PRIMARY KEY: NumPedido
  + FOREIGN KEY: IDCliente REFERENCES clientes (DNIClientes)
  + UNIQUE:NumPedido
  + NOT NULL: NumPedido, FechaHoraPedido, NombreCliente, DireccionEnvio, IDCliente
* **DetallesPedido:** 
  + NumPedido VARCHAR(15)
  + IDProducto INT
  + NombreProducto VARCHAR(50)
  + CantidadProductos INT(3)
  + PrecioUnitario DOUBLE(6,2)
  + PrecioTotalProducto DOUBLE(10,2)
  + PrecioTotalPedido DOUBLE(14,2)
  + PRIMARY KEY: NumPedido, IDProducto
  + FOREIGN KEY:
    - IDProducto REFERENCES IDProducto (productos)
    - NumPedido REFERENCIES NumPedido (DetallesPedido)
  + UNIQUE:
    - IDProducto
    - NumPedido
  + NOT NULL:
    - NumPedido
    - IDProducto

### Código SQL para crear tablas

/\*Eliminacion de la base de datos en caso de que exista y creacion de la misma\*/

drop database if exists proyecto1;

create database proyecto1;

use proyecto1; /\*Se establece por defecto la bd que se debe de usar\*/

/\*Creacion de las tablas\*/

create table clientes (

DNICliente varchar(12) unique not null,

NombreCliente varchar(45) not null,

ApellidosCliente varchar(45) not null,

TelefonoContacto int not null,

Email varchar(45) not null,

DireccionEnvio varchar(100) not null,

primary key (DNICliente)

);

create table productos (

IDProducto int unique not null,

NombreProducto varchar(50) not null,

DescripcionProducto varchar(100),

PrecioProducto float(6,2) not null,

Stock int,

IVA int default 21, /\*Establecemos el valor del IVA por defecto a 21\*/

primary key (IDProducto)

);

create table pedidos(

NumPedido varchar(15) unique not null,

IDCliente varchar(12) not null,

FechaHoraPedido datetime not null,

NombreCliente varchar(45) not null,

ApellidosClinte varchar(45) not null,

DirecciónEnvio varchar(100) not null,

primary key (NumPedido, IDCliente),

foreign key (IDCliente) references clientes(DNICliente)

);

create table detallesPedidos(

NumPedido varchar(15) unique not null,

IDProducto int unique not null,

NombreProducto varchar(50) not null,

CantidadProducto int not null,

PrecioUnitario float(6,2) not null,

PrecioTotalProducto float(10,2) not null,

PrecioTotalPedido float(14, 2) not null,

primary key (NumPedido, IDProducto),

foreign key (NumPedido) references pedidos(NumPedido),

foreign key (IDProducto) references productos(IDProducto)

);