**TEMA 1:** Bases de Datos

Grupo: **1ºSI** Curso: 24/25

Fecha: 15/10/24

**Método para pasar a tablas un modelo E/R**

**Paso 1: ENTIDADES**

Cada entidad del modelo corresponderá a una tabla y los atributos de la entidad serán campos de la tabla. Hay que tener en cuenta sobre todo la Clave Principal (Primary Key) y también todas las indicaciones o anotaciones del diseñador que pueda haber en el modelo de campos únicos o no nulos o restricciones concretas para algunos campos.

Por ejemplo, en el siguiente modelo:

Diagrama

Descripción generada automáticamente

Tendré las siguientes tablas:

CLIENTE

COCHE

REVISION

Con sus correspondientes campos.

**Paso 2: RELACIONES 1:N (o 1:1)**

Para implementar las relaciones 1:N en la Base de Datos lo único que tengo que hacer es seguir la siguiente regla:

* Cojo la Clave Principal (PK) de la tabla con cardinalidad “1” y la paso a la tabla con cardinalidad “N”. Es decir, realmente estoy duplicando un campo en ambas tablas, el Cod\_cliente del CLIENTE por ejemplo, en el siguiente modelo, estará también según esta regla en la tabla COCHE, pero es que ESA ES LA ÚNICA MANERA DE PONER EN CONTACTO AMBAS TABLAS (de relacionarlas).

Diagrama

Descripción generada automáticamente

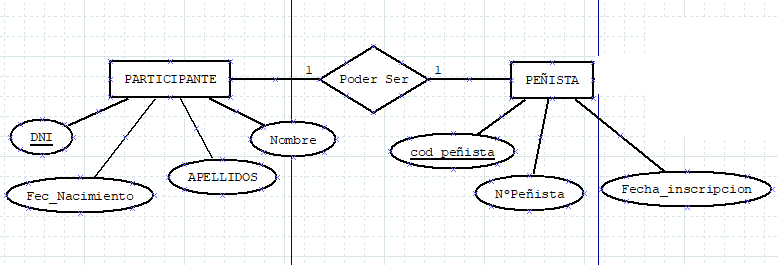
El campo NIF en la tabla COCHE lo puedo llamar como yo quiera “cliente”, “NIF\_cliente”, “cod\_cliente”, “dueño\_del\_coche” . . . da igual siempre y cuando represente bien lo que contiene. Por supuesto DEBE SER DEL MISMO TIPO Y TAMAÑO que la Clave Principal de la tabla de cardinalidad 1 CLIENTE.

Una vez creado en la tabla COCHE tendré que exigir INTEGRIDAD REFERENCIAL para que no puedan meter un NIF de un cliente que no exista.

A partir de aquí, el NIF que se crea en la tabla de COCHE se llamará CLAVE AJENA (FOREIGN KEY), porque en realidad es una clave que NO pertenece a esta tabla sino a la de cardinalidad “1”, en nuestro caso CLIENTE.

**Paso 2.1: RELACIONES 1:1**

Es un caso concreto de las relaciones 1:N, en este caso hay que elegir una de las dos tablas y pasar su PK a la otra tabla, donde será una FK. En el caso que vimos de las (No)Fiestas del Pilar teníamos una relación de este estilo:

****

Funcionará en cualquier caso, tanto si paso el cod\_peñista a la tabla PARTICIPANTE como si paso el DNI a la tabla PEÑISTA, pero sólo hay que hacer una de las dos opciones, no las dos. En todos los casos hay uno que es más claro que el otro, más intuitivo y que seguramente me permite ahorrar espacio. En este caso en concreto si paso el cod\_peñista, TODOS los Participantes tendrán ese código, sean o no peñistas, de modo que los NO peñistas lo tendrán en blanco, pero “molestando” porque es un campo que existe pero no utilizo. De modo que en este ejemplo sería mejor pasar el DNI de Participante a peñista porque todos los peñistas serán participantes, por lo tanto tendrán DNI pero no todos los participantes serán peñista. Por supuesto en la relación volveré a exigir INTEGRIDAD REFERENCIAL.

¡¡¡¡ EL CAMPO EXTRA (FK) DEBERÁ SER MARCADO COMO ÚNICO !!!!

**Paso3: RELACIONES N:M**

Para implementar las relaciones N:M en la Base de Datos lo único que tengo que hacer es seguir la siguiente regla:

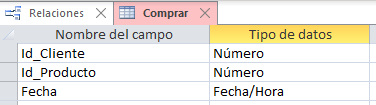
* Cojo la Clave Principal (PK) de una de las tablas, por ejemplo la de la izquierda y **creo una tabla NUEVA** que contenga esa clave como campo de la nueva tabla (todavía no sé si será clave principal o no). Es decir, realmente estoy duplicando un campo en ambas tablas para poder hacer el enlace. En el ejemplo siguiente, crearé la tabla “Comprar” y le pondré un campo que será el id\_Producto del PRODUCTO. Acto seguido haré lo mismo con la otra tabla, cogeré su Clave Principal y la añadiré a la nueva tabla como otro campo de la misma. En el ejemplo cogeré el id\_Cliente y lo pondré como campo de la tabla “Comprar”. Además la nueva tabla (Comprar) tendrá los campos que esa relación N:M tenga en el modelo E/R, en nuestro caso tendrá el campo “Fecha\_Compra”.

**Diagrama

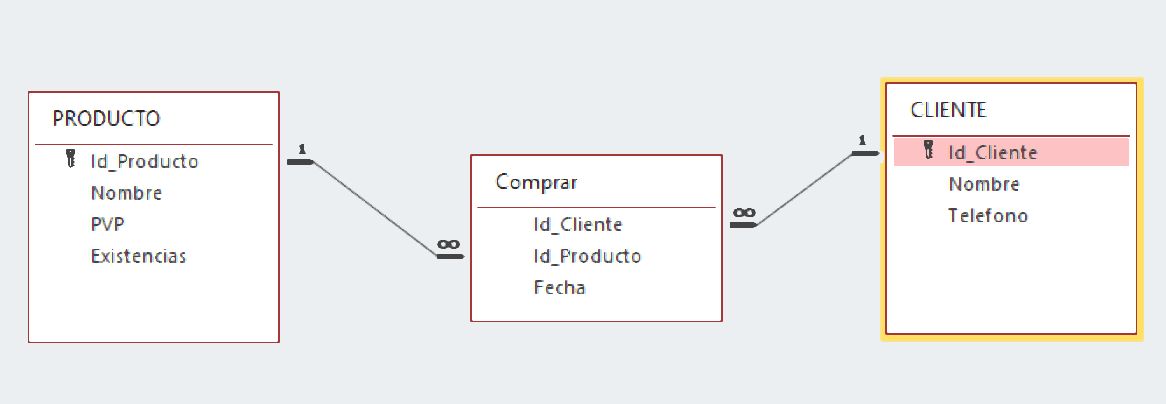
Descripción generada automáticamente**

Los campos de la nueva tabla que son claves primaria de las otras dos, se denominan Claves Ajenas (FK).

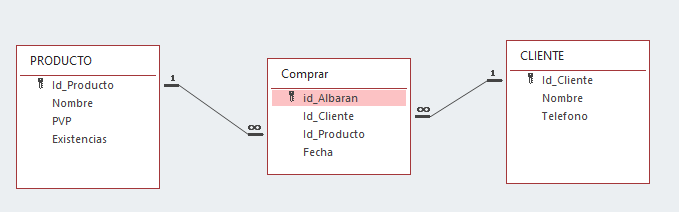
De este modo tendría una tabla así:



Crearé las relaciones entre la tabla intermedia y las dos entidades que pone en contacto. En el ejemplo, relacionaré el id\_Cliente de Comprar con el id\_Cliente de CLIENTE y pediré INTEGRIDAD REFERENCIAL.



La Clave Principal de la tabla no es una regla fija, depende del modelo y del caso en concreto, en este caso en que los artículos son genéricos me interesa que el cliente los pueda comprar cuantas más veces mejor, de modo que NO ES BUENA IDEA EN ESTE CASO poner como clave principal a las dos claves ajenas heredadas de las tablas, y CREARÉ UNA CLAVE PRINCIPAL adicional, por ejemplo un id\_compra, o un id\_ticket, o un id\_albaran.



**Caso concreto de relaciones N:M: Relaciones reflexivas.**