

Diagrama entidad relación:

Índice del contenido

- ¿Qué es un diagrama de entidad relación o ERD?
- Partes de un diagrama entidad relación
- ¿Cómo hacer un diagrama entidad relación?
- Ejemplo práctico
- Análisis
- Atributos y claves
- Etapas del diseño de una base de datos + ejemplo
- Etapa del diseño conceptual + Caso práctico
- Identificar Entidades:
- Identificar Atributos de las Entidades:
- Identificar las Relaciones entre entidades
- Identificar restricciones:
- Crear el Diagrama entidad relación:

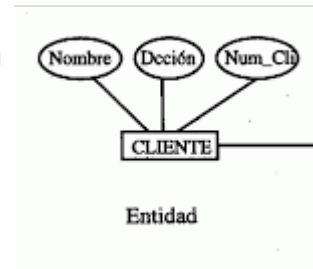
Llegado el momento de pensar en diseñar nuestra propia base de datos relacional recordemos que debemos seguir un proceso compuesto por las **3 etapas de diseño** y finalmente “elegir un Software o gestor de bases de datos relacionales” para exportarlo y crear nuestro sistema de bases de datos relacionales. Dónde a la hora de diseñar la misma **habremos establecido en el modelo numerosas reglas, relaciones y requisitos para conservar la integridad de los datos e información que esta va a almacenar.**

Recomiendo!

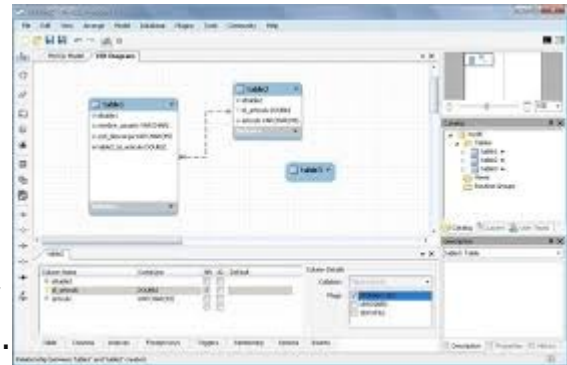
Que complementes bases de datos relacionales con videos de Youtube y mucha lectura, ya que no es un tema tan llano y sencillo como pueda parecer y a veces solemos dejar pasar muchos conceptos importantes de la teoría a la hora de la práctica. Sin más, a ello.

Diseñar una base de datos consta de 3 etapas:

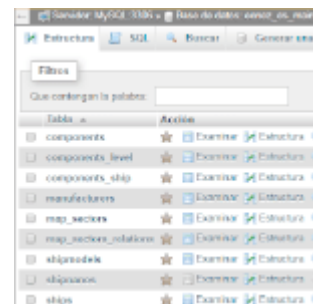
- **Diseño o modelo Conceptual:** Aquí es donde comprendemos y estudiamos el planteamiento del problema o necesidad que va a solucionar nuestra base de datos y realizamos un diagrama llamado "Diagrama entidad relación" que responde a un "boceto preliminar" de lo que podría ser la estructura de la base de datos.



- **Diseño o modelo Lógico:** Una vez que hemos desarrollado este "**Diagrama de entidad relación**" es momento de convertirlo en un "modelo lógico" en dónde las entidades y relaciones que hemos establecido en el diagrama pasan a convertirse en tablas y se definen los tipos y restricciones de cada atributo de las mismas. Se elabora así finalmente un modelo completo de cómo será nuestra base de datos, pero recordemos que efectivamente sigue siendo un "modelo" y aún no está definida.



- **Diseño o modelo Físico:** En esta etapa vamos culminando nuestra base de datos exportando este modelo a un **Gestor de bases de datos relacionales**. En esta etapa y en la anterior se puede hacer uso de determinados "softwares" que facilitan el trabajo de creación de modelos y nos permiten exportarlos directamente a nuestro gestor de bases de datos relacionales. En esta etapa se definen los nombres, tipos, propiedades de los campos y sus restricciones, así como las propiedades de las tablas y su creación, entre otras cosas, dando lugar finalmente a nuestra base de datos relacional.



Vamos a seguir estas tres etapas con un ejemplo práctico sencillo, pero primero "la teoría teórica" dijo la tía.

¿Qué es un diagrama de entidad relación o ERD?.

Un diagrama de entidad relación es un diagrama clásico o modelo similar a un diagrama de flujos el cual hemos visto anteriormente en programación en python, pero con la diferencia de que

“

En este post aprenderás a diseñar bases de datos y exportarlas a MySQL para luego trabajar con ellas desde tu aplicación en Python..

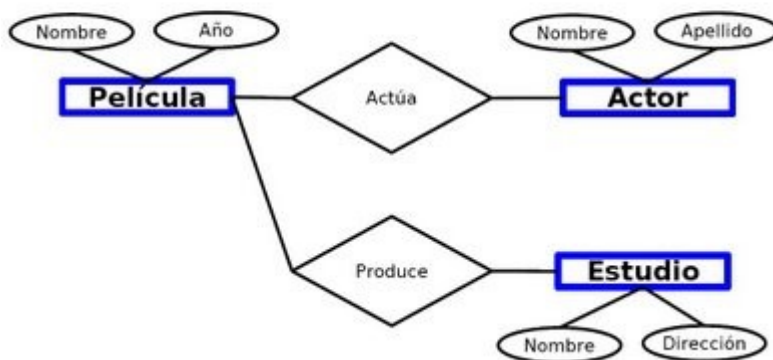
PYTHONES.NET

ilustra cómo las entidades, personas, objetos, datos se relacionan entre sí dentro del sistema.

A menudo los **ERD** (Entidad Relation Diagram) se utilizan para DISEÑAR y DEPURAR las bases de datos relacionales.

Un diagrama entidad relación puede verse así:

Entidades



Partes de un diagrama entidad relación:

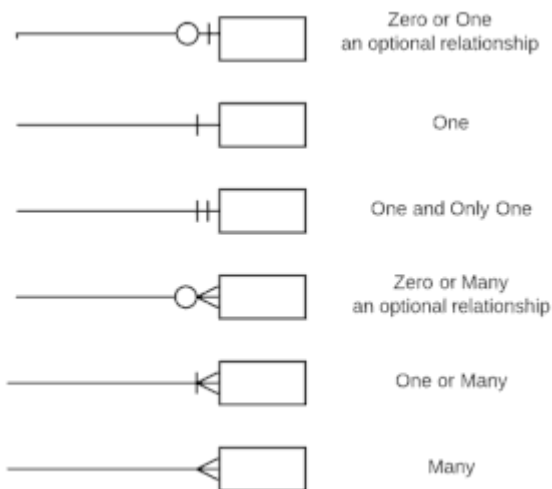
- **Entidades:** Una entidad (en este caso) es una unidad de una base de datos que contiene información. Esta unidad es la representación de un objeto, persona, empresa del mundo real y posee ciertos **Atributos** que la diferencian de las demás.
- **Tipos de entidades:** el tipo de entidad está dado por un grupo de objetos o cosas que se pueden definir, por ejemplo el tipo de entidad "Estudiantes" es un grupo de varias entidades que son el individuo en sí (alumno) por ejemplo.
- **Conjunto de entidades:** un conjunto de entidades son entidades agrupadas en un momento determinado en base a un criterio específico. Por ejemplo podría ser un conjunto de entidades "Alumnos desaprobados" dado por las entidades Alumnos que no cumplen un criterio de aprobación.
- **Categorías de entidades** (Fuertes, Débiles o asociativas): Las entidades se clasifican en estas 3 categorías mencionadas anteriormente basándonos en que una entidad es **Fuerte** cuando se puede definir una clave en base a sus propios atributos, si los atributos no son suficientes para definir la entidad y esta depende directamente de otra decimos que estamos ante una entidad **Débil**. Una entidad asociativa será aquella que establezca una relación entre las entidades dentro de un conjunto de ellas.

- **Clave:** Como vimos en el modelo relacional una [clave](#) es aquel atributo que nos permite diferenciar a una entidad de otra inequívocamente dentro de un conjunto de entidades. Normalmente los atributos que forman una clave se suelen encontrar subrayados. Así existen diferentes tipos de claves:



- **Clave primaria:** definida por el diseñador de la base de datos permite identificar inequívocamente a la entidad del conjunto.
- **Superclave:** Es una clave definida por un conjunto de atributos que permiten diferenciar a esta entidad de las demás.
- **Clave candidata:** es aquella en la que no existen dos valores para los mismos atributos dentro de la base de datos, lo que significa que en conjunto, es una superclave pero tratando de establecer los atributos mínimos.
- **Clave foránea:** Es aquella que permite identificar inequívocamente una relación entre una o más de las entidades.
- **Atributos:** Los atributos (en este caso) son las propiedades que poseen las Entidades, como bien podría ser por ejemplo en el caso de una Persona; Nombre, Edad, Dirección, Teléfono, etc. Existen también diferentes tipos de atributos.
- **Atributos Simples:** Los atributos simples son aquellos que tienen un solo valor para la entidad.
- **Atributos Compuestos:** Son aquellos que pueden subdividirse en "sub-Atributos" por ejemplo el atributo Teléfono puede subdividirse a su vez en Celular, o Fijo.
- **Atributos Derivados:** Son aquellos que derivan o se calculan a partir de otro/s atributos, por ejemplo "Edad" normalmente se puede calcular a partir de la fecha de nacimiento de la persona.

- **Relaciones:** En una base de datos Relacional, existen Relaciones entre las Entidades que describen cierta dependencia o asociación entre las mismas.



- **Cardinalidad:** Expresada en diagramas entidad-relación.

Por ejemplo la entidad Habitación estará relacionada con la Entidad Huésped. Así mismo estas relaciones constan de una propiedad llamada "**Cardinalidad**" que especifica el número de entidades con la que puede estar relacionada una entidad dada, podemos decir por ejemplo que una sola Habitación puede estar relacionada con diversos huéspedes que se hayan alojado en ella, por lo que podríamos asegurar que la entidad Habitación tiene una relación de cardinalidad "uno a varios" con la entidad Huésped.

Relación de cardinalidad "**uno a uno**".

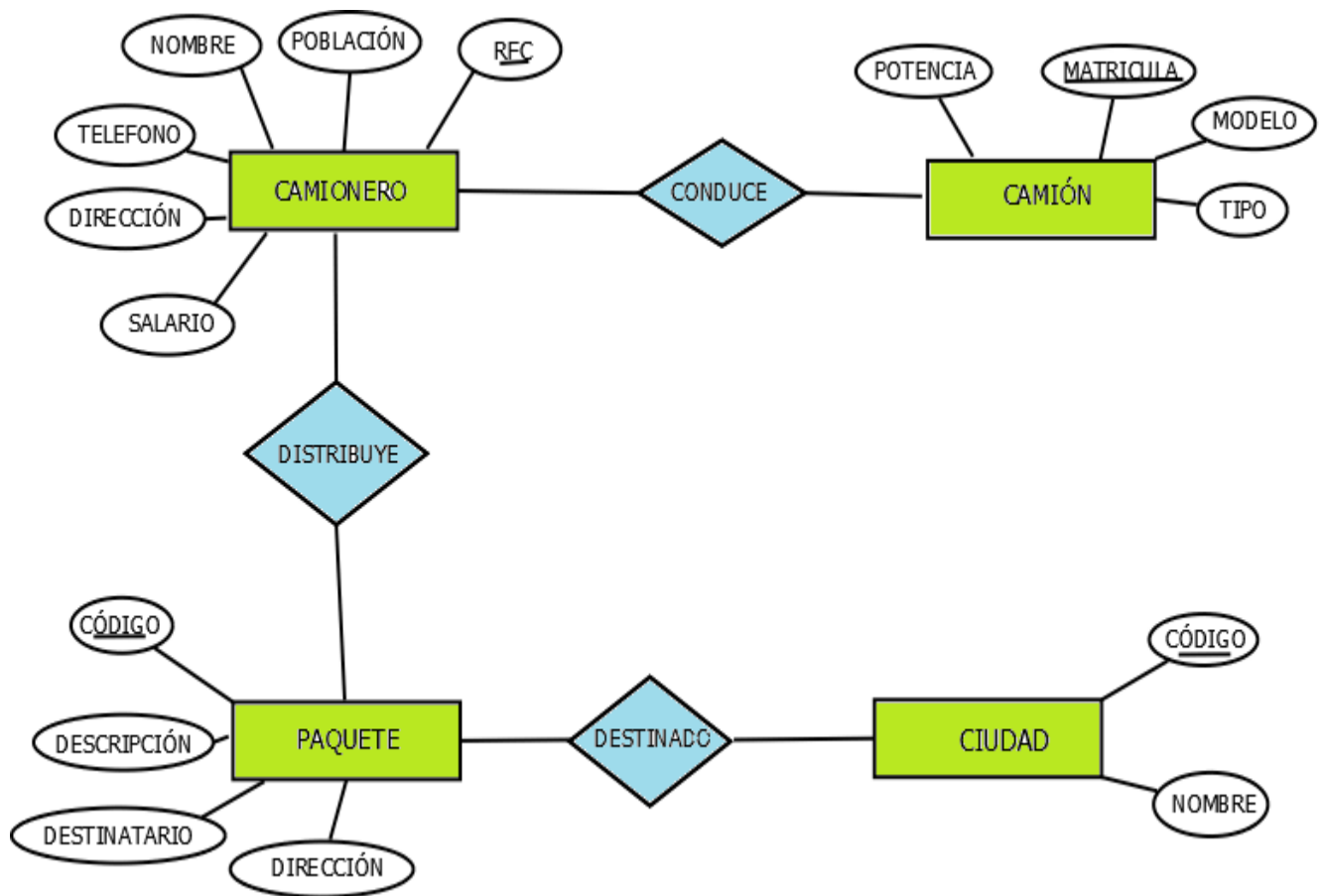
Relación de cardinalidad "**uno a varios**".

Relación de cardinalidad "**varios a uno**".

Relación de cardinalidad "**varios a varios**".

Supongamos por un momento que tenemos que **diseñar una base de datos para una empresa de transporte**.

Un diagrama de entidad-relación sencillo, básico sólo para expresar una idea y organizar mejor la estructura inicial haciendo hincapié en que se trate de la etapa de **Diseño Conceptual** podría verse así:



!!Atención!!:

El diagrama anterior no expresa "cardinalidad" en las relaciones. Este diagrama como vemos cuenta con un conjunto de símbolos geométricos para expresar la interconexión de entidades, relaciones, atributos, etc. Facilitando al diseñador de la base de datos relacional plasmar y comprender mejor esa estructura para luego finalmente exportarla a un sistema de gestión de bases de datos y comenzar a utilizarla para sus fines.

¿Cómo hacer un diagrama entidad relación?.

El modelo de datos entidad-relación está basado en una percepción del mundo real que consta de una colección de objetos básicos, llamados entidades, y de relaciones entre esos objetos amorfos.

Como te dije, el diagrama de entidad relación puro especifica las entidades, atributos, claves y relaciones de forma gráfica donde cada figura geométrica tiene un significado y permite comprender mejor su estructura.

Fijate:




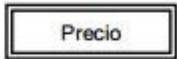







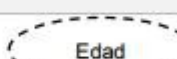
Símbolo	Significado	Ejemplo
	Entidad Fuerte	
	Entidad Débil	
	Atributo	
	Relación	
	Atributo multivaluado	
	Atributo Derivado	

Tabla 1. Símbolos básicos para diagramas de entidad-relación.

Elaboración propia.

Así utilizando estas figuras o símbolos podemos comenzar a organizar información dándole una estructura y estableciendo relaciones entre los datos que serán almacenados siguiendo la misma.

Más allá o no de que luego creemos una base de datos, hablamos de **organizar información importante de forma que nos resulte más fácil y rápido acceder a ella, modificarla y comprender sus relaciones de forma que nos permita subdividir un problema mayor en pequeños problemas de más fácil solución.**

Lo primero siempre es tener en cuenta los requerimientos de la base de datos y la planificación conceptual del problema para comenzar a crear nuestro diagrama..

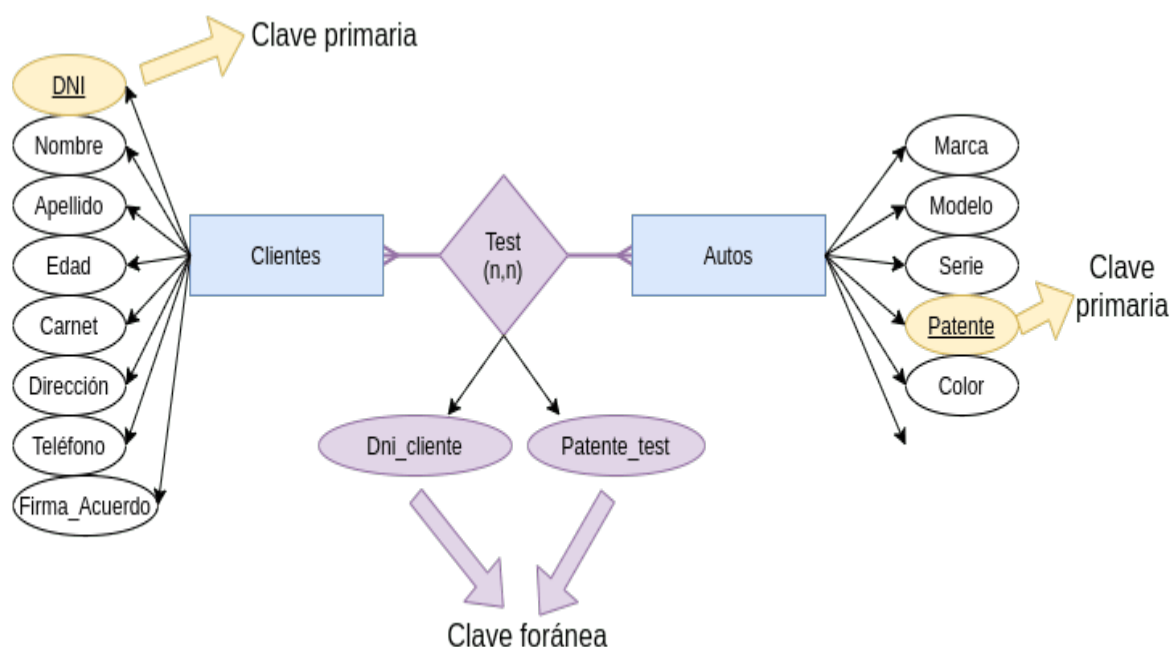
Un ejemplo práctico sencillo!..

En un concesionario queremos llevar un registro de las personas que prueban determinados coches para asegurarnos de saber quién ha sido el responsable en caso de que se descubra algún desperfecto o daño. Para ello debemos exigir un registro a los potenciales clientes y registrar el coche o automóvil que van a probar.



Análisis

Si tuvieras que representar esta relación en un diagrama de entidad relación reconocerías como entidades "**Cientes**" y "**Autos**". Como atributos podríamos colocar para "**Cientes**" (Dni, Nombre, Apellido, Edad, Carnet, Dirección, Teléfono, Si firmó el acuerdo de responsabilidad por los daños, etc). Para "**Autos**" (Marca, Modelo, Serie, Patente, Color, etc.) y para establecer la cardinalidad de la relación (que es una entidad asociativa) "Test", evidentemente hablamos de "**varios a varios**". Ya que varios clientes pueden probar tantos coches como quieran.. ¿Pero cómo registramos que una persona y su cónyuge han probado determinado coche por ejemplo?. Pues en este caso por tratarse de una relación "**varios a varios**" podríamos posteriormente "en la etapa de **Diseño Lógico**" también crear una Tabla para la entidad asociativa "Test". Pero mientras tanto podemos establecer en nuestro diagrama entidad relación las claves foráneas (««« enlace) así:



Atributos y claves:

Recordemos que en el caso de las claves primarias de las que ya hablamos anteriormente en "El modelo Relacional" estas deben ser atributos que identifiquen inequívocamente a cada entidad dentro del tipo de entidad. Por lo que jamás la clave primaria de Personas, podría ser Nombre, ni Nombre y Apellido como clave primaria compuesta. Porque un día puede llegar alguien que se llame igual a otra persona de la lista y estaríamos en un problema, y si.. Puede pasar solo fíjate en facebook cuantas personas se llaman igual!. Pero bien sabemos, nadie tendrá nunca el mismo **DNI** o número de identificación..

También es importante recordar que una clave es foránea porque referencia a otra primaria, en este caso en las claves foráneas de (Test) una almacenará el **DNI** del cliente y la otra la **Patente** del coche que probó. Y al tener estas claves en dicha entidad "Test" nos estará señalando que al momento de crear un modelo Lógico debemos tenerla en cuenta como una **"tabla intermedia adicional"**.

¿Cuándo no es necesario crear una tabla adicional a partir de una relación?.

Pues cuando se trata de una relación de uno a uno donde podemos incluir la **clave foránea** dentro de alguna de las entidades relacionadas que referencie la clave primaria de la otra entidad.

Etapas del diseño de una base de datos:

Para diseñar correctamente una base de datos **existen tres etapas que es conveniente respetar para controlar la complejidad dividiendo un problema mayor en problemas pequeños de fácil solución**. Vamos a aprender estas tres etapas haciendo uso de un ejemplo práctico teniendo en cuenta que la realización de una nueva base de datos normalmente nace para resolver un problema o cubrir la necesidad de una mejor organización de los datos de determinada institución o particular.

Caso práctico: El profesor Marcelo le ha pedido a sus alumnos desarrollar una base de datos que permita conocer las materias en las que están inscritos los alumnos y los profesores que las imparten. Vamos, algo bien sencillito como para practicar. Conozcamos y apliquemos las 3 etapas del diseño de una base de datos...



Etapas del diseño conceptual + Caso práctico:

El diseño conceptual parte de las necesidades y requisitos o restricciones del usuario que nos expone en este caso unas determinadas descripciones de la base de datos requerida para satisfacer sus necesidades. Básicamente el diseño conceptual se trata de describir vagamente el contenido de la base de datos y no la estructura de almacenamiento..

Identificar Entidades:

Es importante en esta etapa identificar las **Entidades** que van a participar de nuestra base de datos, en este caso supongamos por ejemplo que identificamos sólo 4 entidades y nos interesamos por hacer foco en ellas y no en su organización todavía: **Alumnos, Materias y Profesores**.

Identificar Atributos de las Entidades:

Una vez focalizados en las entidades es momento de pensar en los **Atributos** de cada entidad teniendo en cuenta la especificación del problema. En este caso podrían ser:

Profesores: (*Dni*, Nombre, Apellido, Dirección, Teléfono, Edad)

Alumnos: (*Dni*, Nombre, Apellido, Dirección, Teléfono, Edad)

Materias: (*ID*, Nombre)

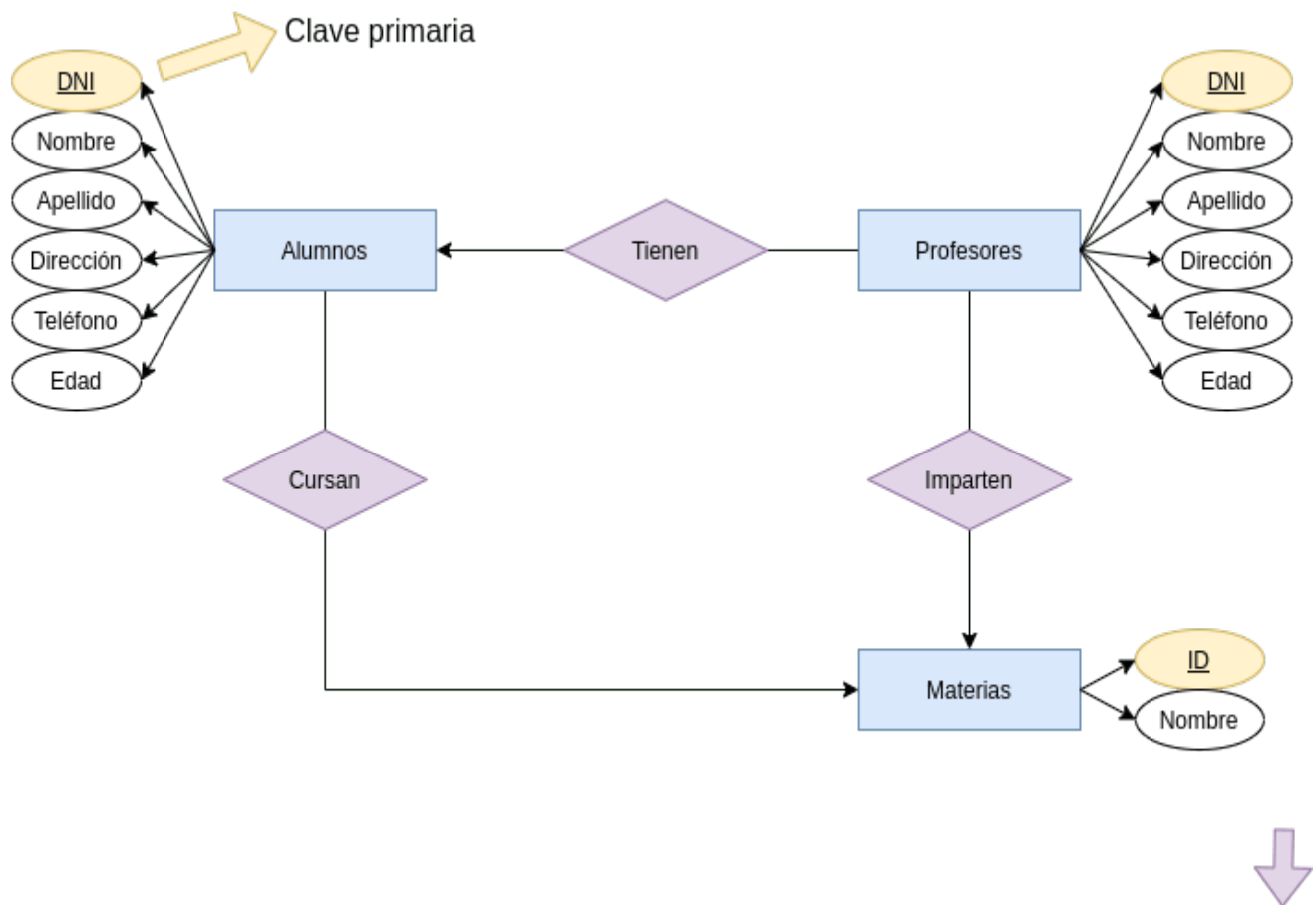
Yo aquí ya incluí los atributos que serán claves primarias y los demás atributos. Pero aún no he definido relaciones ni claves foráneas en este paso.

Y tú podrías preguntarte ¿Qué pasaría si yo lo organizo de una forma diferente y coloco las materias dentro de Alumnos y Profesores por ejemplo?. Pues nada, **más tarde aplicando reglas de normalización terminarás por adecuar las entidades en tablas aparte, debido a que será casi obligatorio respetando las formas normales**.

Debido a que podemos intuir que se repetirán los datos en el campo de las materias en ambas tablas y terminarás agrupando varios datos en un mismo campo, **algo no muy recomendable según la primera forma normal**.

Identificar las Relaciones entre entidades:

Por tercer paso tenemos que identificar las **Relaciones** entre las entidades y si tienen atributos asociados entre ellas. Así que colocamos en nuestro diagrama Entidad y atributos y luego procedemos a relacionarlos. Por ejemplo en este caso tenemos que cada **Alumno** cursa varias **Materias** y cada **Materia** tiene un **Profesor**.



Identifiqué y añadí relaciones pero aún no establecí la cardinalidad ni llaves foráneas.

Identificar restricciones:

- Aquí por un lado tenemos que identificar las restricciones de clave primaria y de cardinalidad de las relaciones. Recordemos que debemos establecer un atributo como **clave primaria** que nos permita identificar inequívocamente cada grupo de datos, fila o tupla de la base de datos. Para ello en este caso podemos utilizar el atributo **DNI** tanto de Profesores como de Alumnos. Y en el caso de Materias y Calificaciones añadimos un atributo **ID** (Número de identificación único) como clave primaria para cada materia y calificación añadida. Recordemos también la importancia de saber elegir los atributos que serán claves primarias, estos atributos elegidos deben ser de carácter único y jamás deberían poder repetirse (podría ser el caso del DNI por ejemplo, pero jamás el nombre o apellido). Ante la duda es mejor utilizar o crear un Atributo **ID** como identificador de clave primaria.
- Las restricciones de cardinalidad de las relaciones las analizamos como las siguientes:

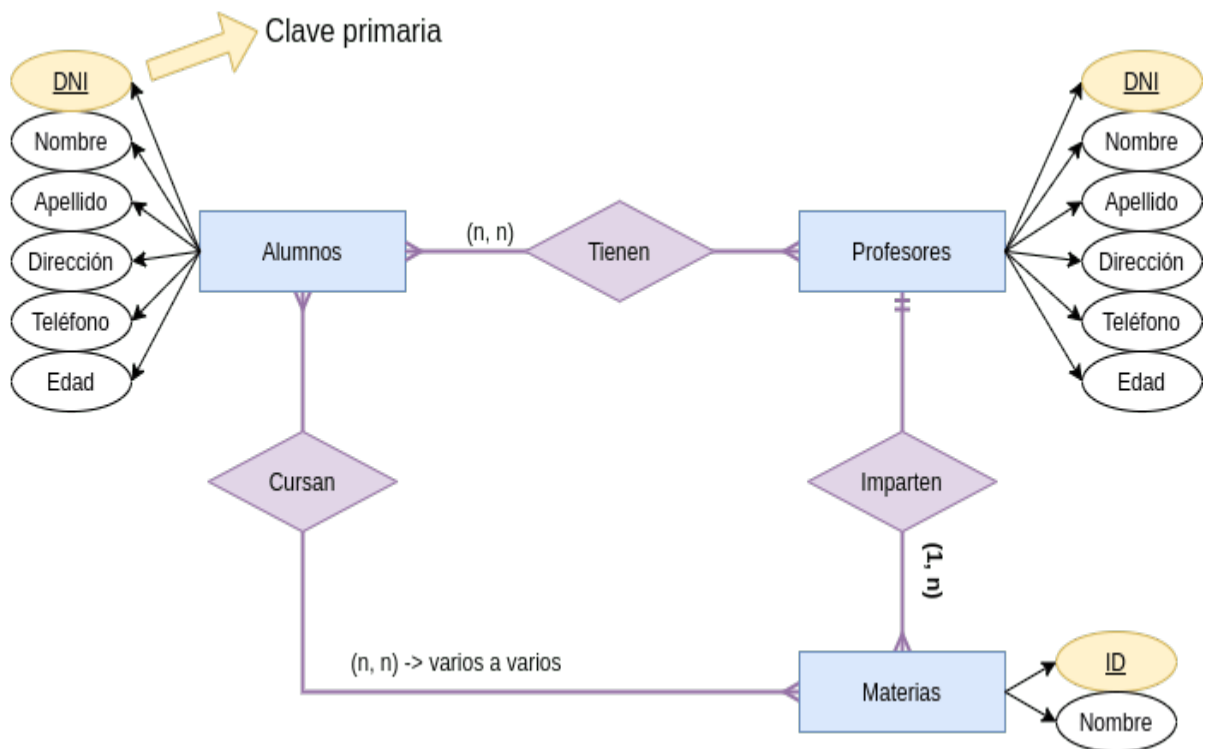
- Un **Alumno** cursa una o varias **Materias** y una materia tiene varios **Alumnos**. (Relación varios a varios).
- Un **Profesor** puede impartir varias **Materias**. Pero una **Materia** puede ser impartida por un solo **Profesor** (Relación uno a varios).
- Un **Profesor** puede tener varios **Alumnos** y un **Alumno** puede tener más de un **Profesor**. (Relación varios a varios).

Relaciones y cardinalidad.

Al existir varias relaciones de cardinalidad “**varios a varios**” podemos incluir una tabla aparte con estas relaciones estableciendo las claves foráneas correspondientes para indicar dicha relación. Y en el caso de las relaciones “**uno a varios**” podemos incluir una clave foránea en la entidad que tiene la relación de varios. Este es el caso de la relación entre **Materias** y **Profesores**, incluiremos el dni (clave primaria) del profesor que dicta cada materia en **Materias** y no al revés para evitar un grupo de varios datos en un mismo campo (1NF – Primera forma normal). Ha llegado el momento de hacer un diagrama entidad relación completo.

Crear el Diagrama entidad relación:

Llegados a este punto podemos crear un boceto de nuestro diagrama entidad relación para comenzar a tener una idea clara de cómo estará estructurada nuestra base de datos en cuanto a entidades y la cardinalidad de sus relaciones. Para ello recordemos debemos tener en cuenta Entidades, Atributos de clave primaria, los demás Atributos y establecer las Relaciones en forma de líneas que unifican las Entidades y manifiestan la cardinalidad de las mismas.



En este caso podemos observar que **Alumnos, Profesores y Materias** han sido designadas como **Entidades** y hemos designado los correspondientes atributos antes mencionados.

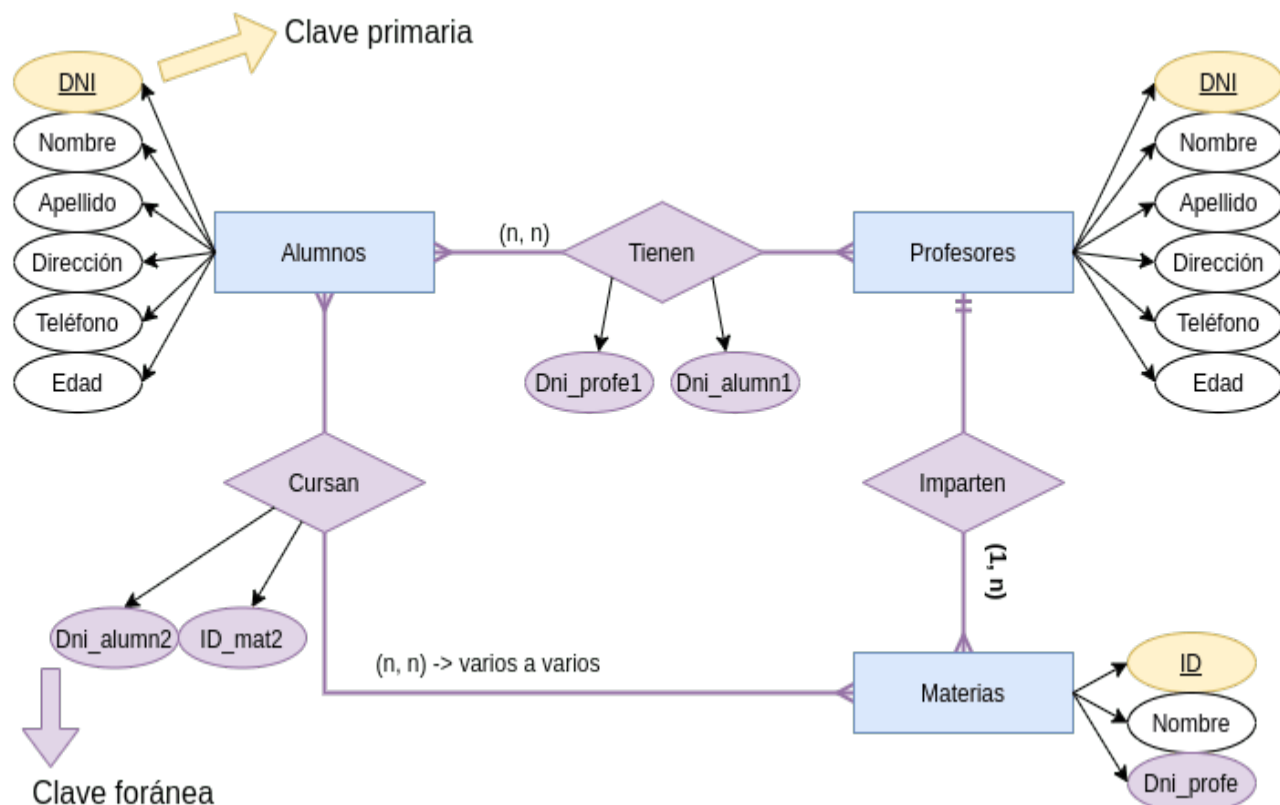
Además hemos designado la cardinalidad correspondiente a cada relación. Pero aún falta **especificar las restricciones y crear las tablas aparte para las relaciones que sean necesarias estableciendo ya las claves foráneas**.

Recordemos que, para el caso de las relaciones "Tienen" y "Cursan" podemos crear tablas aparte con las claves foráneas para identificar las relaciones, pero en el caso de la relación "Imparten" al ser de una a varias podemos crear una clave foránea en **Materias** a la que llamaremos (Dni_profe), para identificar el profesor que dicta cada materia.

Y en el caso de "**Tienen**" crearemos una tabla donde especificaremos para la primera las claves foráneas (Dni_profe**1**) y (Dni_alumn**1**) para especificar qué profesor tiene que alumnos.

Y en el caso de "**Cursan**" creamos una tabla donde especificaremos las claves foráneas ID_mat**2** y Dni_alumn**2** para expresar que alumno cursa qué materia.

Recordemos que es necesario que ninguna clave foránea se llame igual que otra ni que una clave privada aún sea que se encuentre en otra tabla o entidad diferente.



Como ves es fácil identificar que valor almacenará cada clave foránea ya gracias a nombre, esto nos facilita la comprensión en un futuro. Por ejemplo es obvio que Dni_alumn2 almacenará el DNI del alumno relacionado con ID_mat2 que almacenará los ID de la materia correspondiente al alumno. Y al final haciendo uso de estas claves podremos filtrar y ver el conjunto de entidades para determinado alumno o determinada materia mediante una consulta..