

Grado en Ingeniería de Software



# Laboratorio de Bases de Datos y Sistemas Distribuidos

Curso 2023-2024



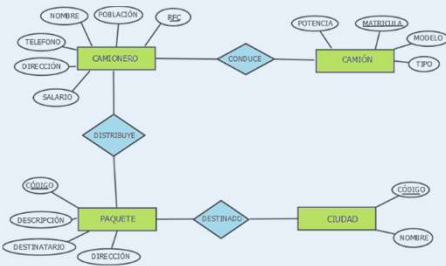
# Tema 2

Conceptos Generales  
Modelo E/R y Esquema de una BD

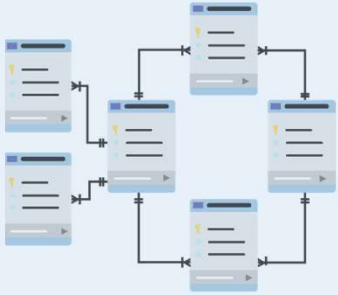
## ÍNDICE

- Fases del Diseño de una BD relacional
  - Diseño Conceptual (modelo ER)
  - Diseño Lógico (diagrama Relacional)
  - Diseño Físico (implementación)
- Modelo Entidad/Relación
  - Entidades
  - Relaciones
  - Atributos
  - Cardinalidad
- Diagrama Relacional
  - Transformación del Modelo ER en el diagrama Relacional
  - Normalización
  - Claves
- Conceptos de la implementación física
- Herramientas de diseño ER
- Herramientas gráficas para interfaz con Gestores de Base de Datos

## Modelo E/R (diseño conceptual)



## Diagrama relacional (diseño lógico)



## Esquema de BD (diseño físico)



Diseñar una BD consiste en definir la estructura de los datos que debe tener un sistema de información determinado. Tres fases:

- En la fase de **diseño conceptual** se describe la información que debe contener la BD y las relaciones entre los datos que almacena. Es hacer una representación del mundo real en un diagrama conceptual → **Modelo E/R**
- El **diseño lógico** parte del resultado del diseño conceptual y da como resultado la estructura de la base de datos, o esquema de la BD, es decir, en el caso de bases de datos relacionales el diseño lógico **define el modelo relacional**: las tablas que existirán, sus atributos, las relaciones entre ellas, su normalización, etc...
- El **diseño físico** parte del lógico y da como resultado una descripción de la implementación de una BD en memoria: las estructuras de almacenamiento y los métodos utilizados para tener un acceso eficiente a los datos. El objetivo es conseguir una mayor eficiencia, y se tienen en cuenta aspectos concretos del Sistema de Gestión de Base de Datos sobre el que se vaya a implementar.

**El modelo E/R** parte de una situación real a partir de la cual se definen entidades, que están compuestas por atributos y relaciones entre dichas entidades

## Entidades y Atributos

Una Entidad es un objeto del mundo real sobre el que queremos almacenar información, puede ser algo abstracto (Ej: estudiante, curso, matricula, etc)

Las entidades están compuestas de **atributos** que son los datos que definen el objeto (nombre, matricula, dirección,...). De entre los atributos habrá uno, o un conjunto de ellos, que no se repite; a este atributo o conjunto de atributos se le llama **clave de la entidad**. En toda entidad siempre hay al menos una clave que en el peor de los casos estará formada por todos los atributos de la entidad. Ya que puede haber varias claves y necesitamos elegir una, lo haremos atendiendo a estas normas:

- Que sea única.

- Que se tenga pleno conocimiento de ella

- Que sea mínima, ya que será muy utilizada por el gestor de base de datos.

## Relaciones

Una relación es la asociación entre entidades, sin existencia propia en el mundo real que estamos modelando, pero necesaria para reflejar las interacciones existentes entre entidades.

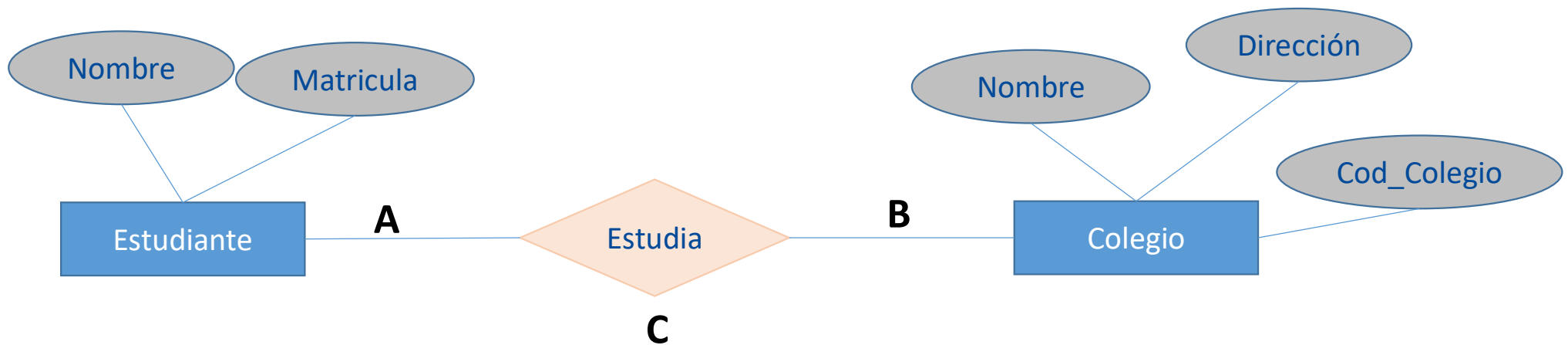
Las relaciones pueden ser de tres tipos, según su **cardinalidad**:

- Relaciones 1:1: las ocurrencias de las entidades que intervienen en la relación se asocian una a una
- Relaciones 1:n o n:1: Una ocurrencia de una entidad está asociada con muchas ocurrencias (n) de otra entidad, o viceversa
- Relaciones n:n: Cada ocurrencia, en cualquiera de las dos entidades de la relación, puede estar asociada con muchas ocurrencias (n) de la otra entidad y viceversa

### **CARDINALIDAD**

Es el número de ocurrencias de una entidad que se pueden asociar a una ocurrencia de otra entidad a través de una relación

## Representación gráfica de entidades, relaciones y atributos Modelo Entidad/Relación



**“Estudia”** es la relación entre “Estudiante” y “Colegio”:

- Los estudiantes “estudian” en colegios
- En los colegios “estudian” estudiantes

ENTIDAD



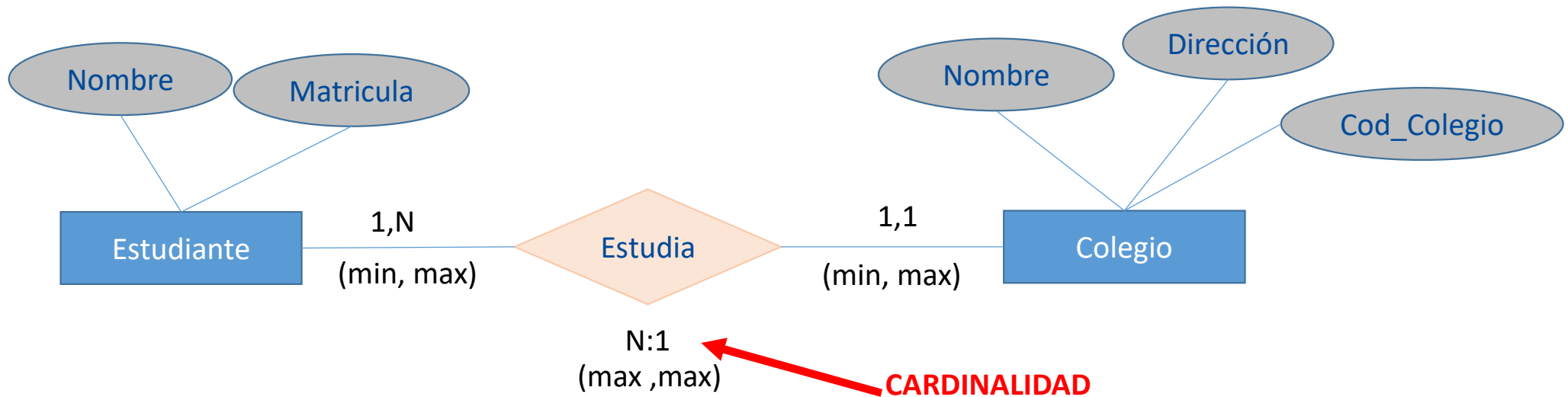
RELACIÓN



ATRIBUTOS



## Añadiendo la cardinalidad a la representación gráfica



### Cardinalidad de la relación "estudia":

- Un "estudiante "estudia" **en 1** colegio como mínimo y **en 1** como máximo
- En un colegio "estudian" **1** estudiante como mínimo o muchos (**N**) estudiantes como máximo





## Paso del Modelo Conceptual (E/R) al Modelo Lógico (Diagrama Relacional de la BD)

- En una BD relaciona todos los datos se almacenan en forma de **tablas**, tanto las **entidades** como los datos necesarios para representar una **relación**. La tabla es además la unidad de almacenamiento principal.
- Las **tablas** están compuestas por **filas (o registros) y columnas (o atributos)**. Cada uno de los registros tiene la información sobre una entidad concreta (considerados una unidad/ocurrencia).
- Cada tabla debe poseer una **clave primaria**, esto es, un identificador único de cada registro compuesto por una o más columnas.
- Para **establecer una relación entre dos tablas** es necesario **incluir**, en forma de columna, en **una de ellas la clave primaria de la otra**. A esta columna se le llama **clave externa**.
- Las relaciones de cardinalidad n:n se representarán en el esquema relacional de BD **como una nueva tabla**.

## ENTIDADES

Para cada **entidad** del modelo se creará una **tabla del diagrama** con tantos campos como atributos tenga la entidad.

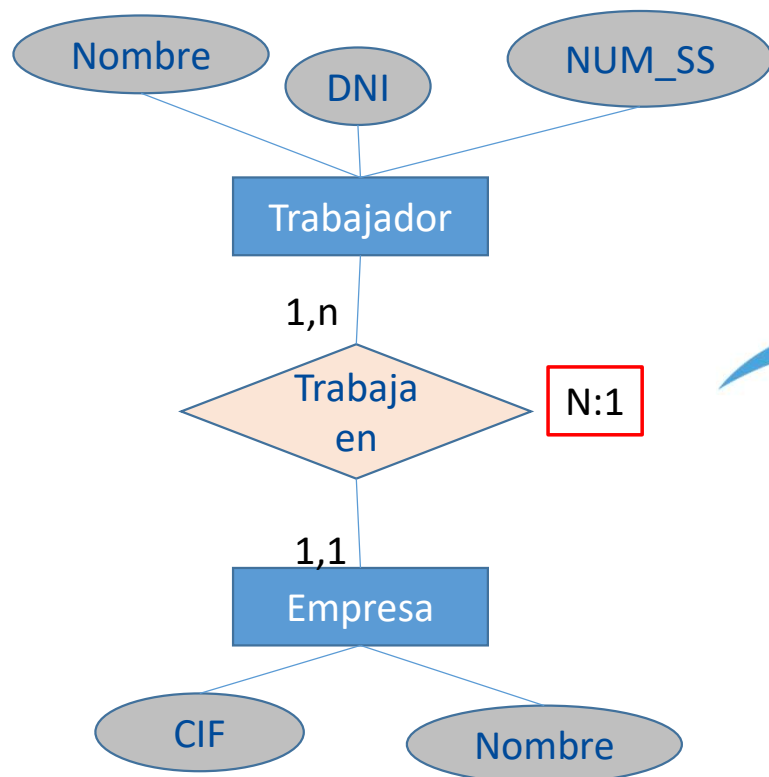


Tabla 'TRABAJADOR

DNI	NUM_SS	nombre-apellidos	...
11111111	XXXXXXXXXX	Fulano de tal	...
22222222	YYYYYYYYYY	Mengano de cual	...
.....	.....	.....	.....

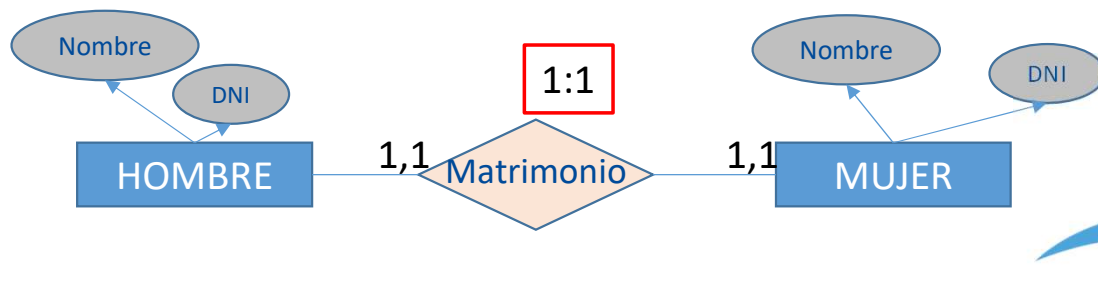
Tabla EMPRESA

CIF	Nombre	...
AA-XXX-BB	Empresa 1	...
CC-YYY-DD	Empresa 2	...
.....	.....	.....

## RELACIONES 1-1

Se transforman incluyendo en una de las dos tablas la clave primaria de la otra tabla con la que se está relacionada.

Ese nuevo campo que se incluye en la tabla recibe el nombre de **clave ajena**.



Donde el campo **DNI-ESPOSO** es clave ajena de la tabla **HOMBRE**.

El campo DNI-ESPOSO puede tomar o bien un valor nulo, en el caso de aquellas mujeres que no estén casadas, o bien el valor de alguno de los DNI de la tabla HOMBRE. En el caso de las mujeres casadas, ese DNI o **la clave ajena**, **no se deberá repetir en ningún otro registro** de la tabla MUJER (debe definirse como “UNIQUE”).

Tabla **HOMBRE**

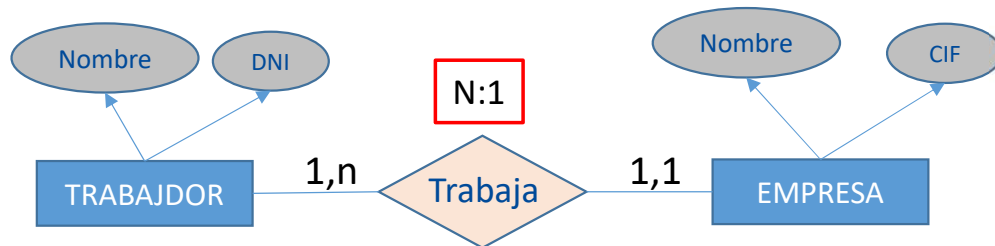
DNI	Nombre	...
11111111	...	...
22222222	...	...
...	...	...

Tabla **MUJER**

DNI	Nombre	...	DNI-ESPOSO
33333333	...	...	11111111
44444444	...	...	(nulo)
...	...	...	...

## RELACIONES 1:N o N:1

Se representan como las relaciones 1-1, es decir, incluyendo la clave primaria de una de las tablas en la otra, pero en este caso no es indiferente donde se coloque la clave ajena, esta **debe estar obligatoriamente en la tabla del 'mucho' (n)**. Además, en este caso sí se permitirá que haya valores repetidos en dicho campo.



El campo **CIF** de la tabla TRABAJDOR es clave ajena y hace referencia a la clave primaria "CIF" de la tabla EMPRESA.

El campo **CIF** puede tomar valores repetidos ya que varios trabajadores pueden trabajar en la misma empresa.

Tabla **EMPRESA**

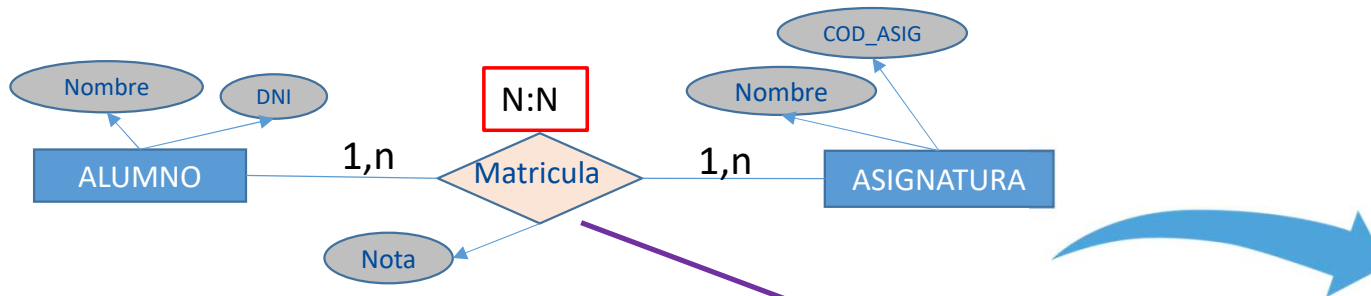
CIF	Nombre	...
XX-111-AAA	...	...
YY-222-BBB	...	...
...	...	...

Tabla **TRABAJDOR**

DNI	Nombre	...	CIF
11111111			XX-111-AAA
22222222			YY-222-BBB
33333333	...	...	XX-111-AAA
44444444	...	...	YY-222-BBB
...	...	...	...

## RELACIONES N:N

Se transforman en tablas, es decir, se creará una nueva tabla para representar dicha relación. Esta nueva tabla tendrá dos claves ajenas, que serán las claves primarias de las tablas que relaciona. Su propia clave estará formada por la unión de ambas claves ajenas.



En la tabla **MATRÍCULA** es donde se refleja la relación. La clave de dicha tabla está formada por los campos DNI y COD-ASIGNATURA ; y cada uno de ellos es clave ajena, el primero de ALUMNO y el segundo de ASIGNATURA.

La tabla **MATRÍCULA** puede tener más campos además de los que son clave ajena, por ej un campo **NOTA**.

Tabla **ALUMNO**

DNI	Nombre	...
11111111	...	...
22222222	...	...
...	...	...

Tabla **ASIGNATURA**

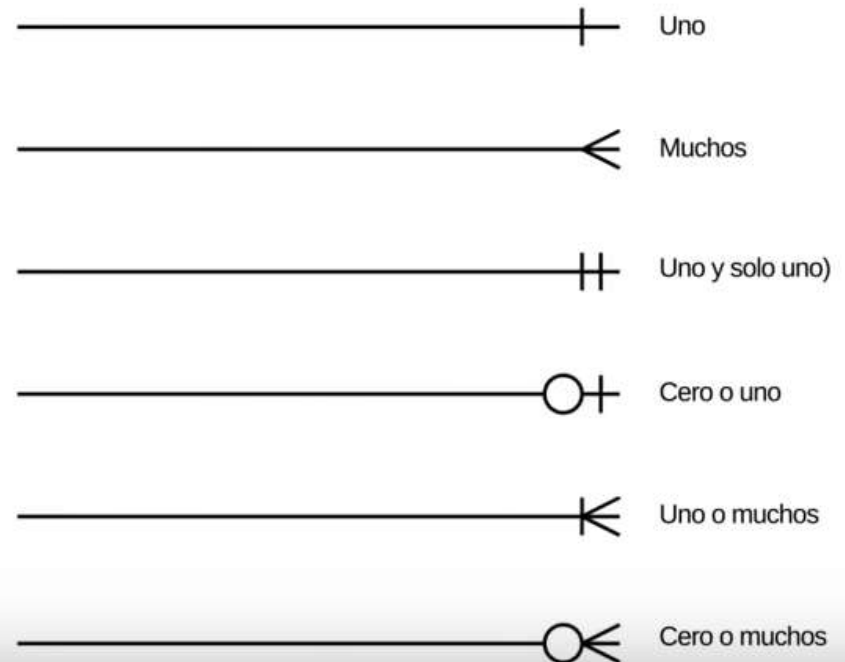
COD ASIG	Nombre	...
01	...	...
02	...	...
...	...	...

Tabla **MATRÍCULA**

DNI	COD ASIG	NOTA
11111111	01	7,5
11111111	02	6,25
22222222	01	5,5
22222222	02	9

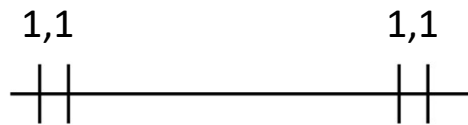
## Representación gráfica de la “cardinalidad” en el diagrama Relacional

### Cardinalidad



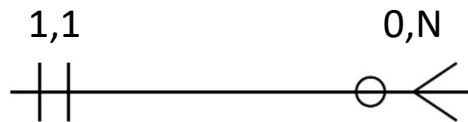
# Diagrama Relacional: Cardinalidad

CARDINALIDAD: (max, max)



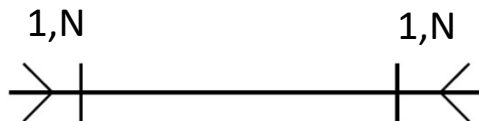
UNO a UNO obligatorio  
1:1

Ej.: un empleado en una empresa tiene **uno y solo un puesto**, un puesto pertenece a un solo empleado.



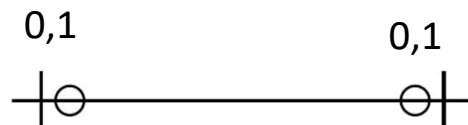
uno a MUCHOS  
1:n

Ej.: un profesor en la Fac. de Ingeniería puede dar **cero** o varias clases y una clase está dada **por uno y solo** un profesor



MUCHOS a MUCHOS  
n:n

Ej.: una asignatura puede pertenecer **a una o varias** carreras, una carrera está formada **por una o varias** asignaturas



UNO a UNO opcional

Ej.: Un paciente puede estar asignado **a una habitación o a ninguna**. En una habitación puede haber **cero o un paciente**

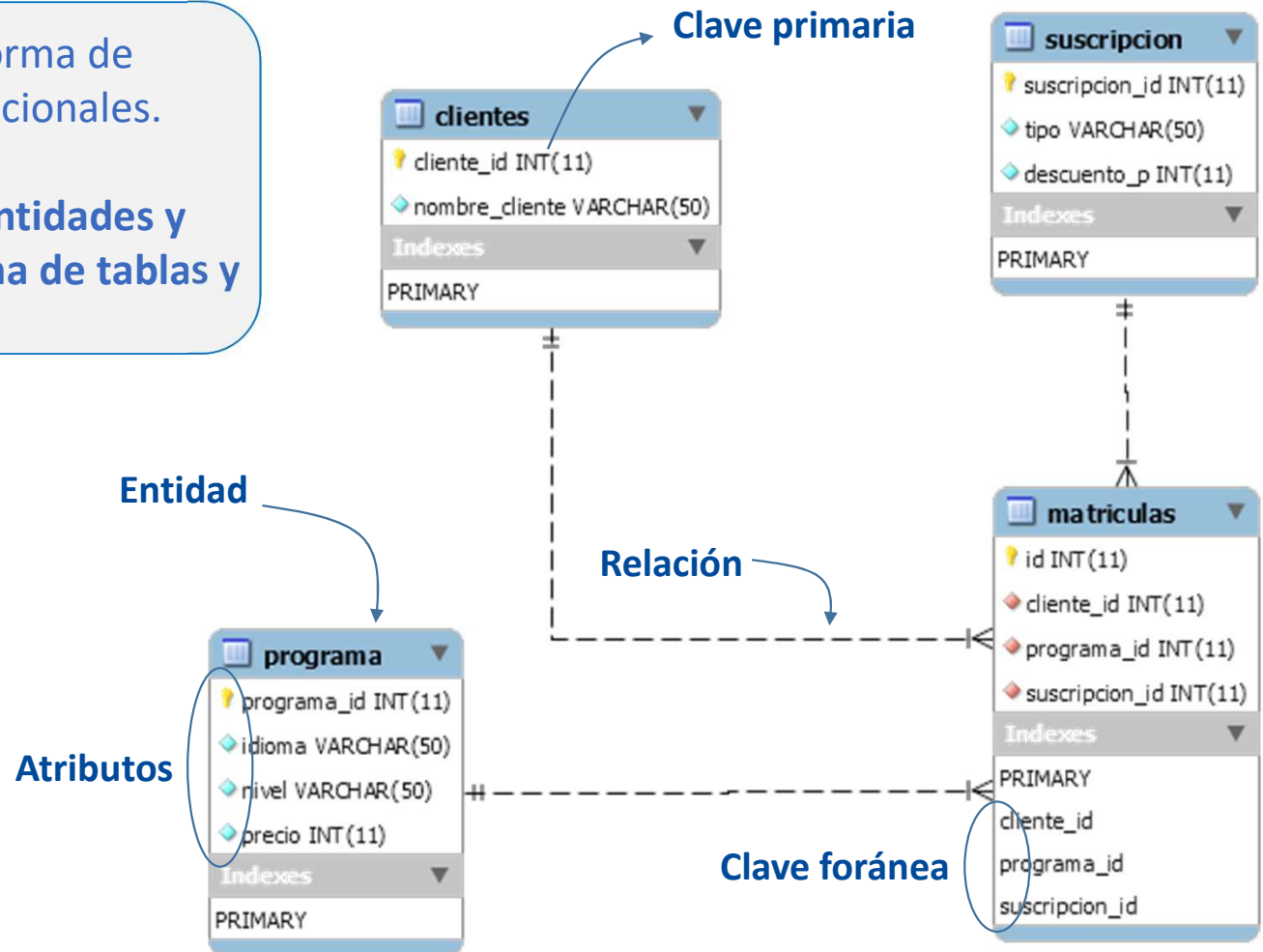
# Diagrama Relacional: representación

El **diagrama relacional**, es la mejor forma de representar la estructura de BBDD relacionales.

En el siguiente diagrama vemos como **entidades y relaciones** están ya representadas en forma de **tablas y atributos**

Elementos del modelo:

- **Entidad**
- **Atributos**
- **Relación**
- **Claves**





## BBDD: academia\_idiomas

clientes

cliente_id	nombre_cliente
1	Pedro
2	Aurelia
3	Federico

suscripción

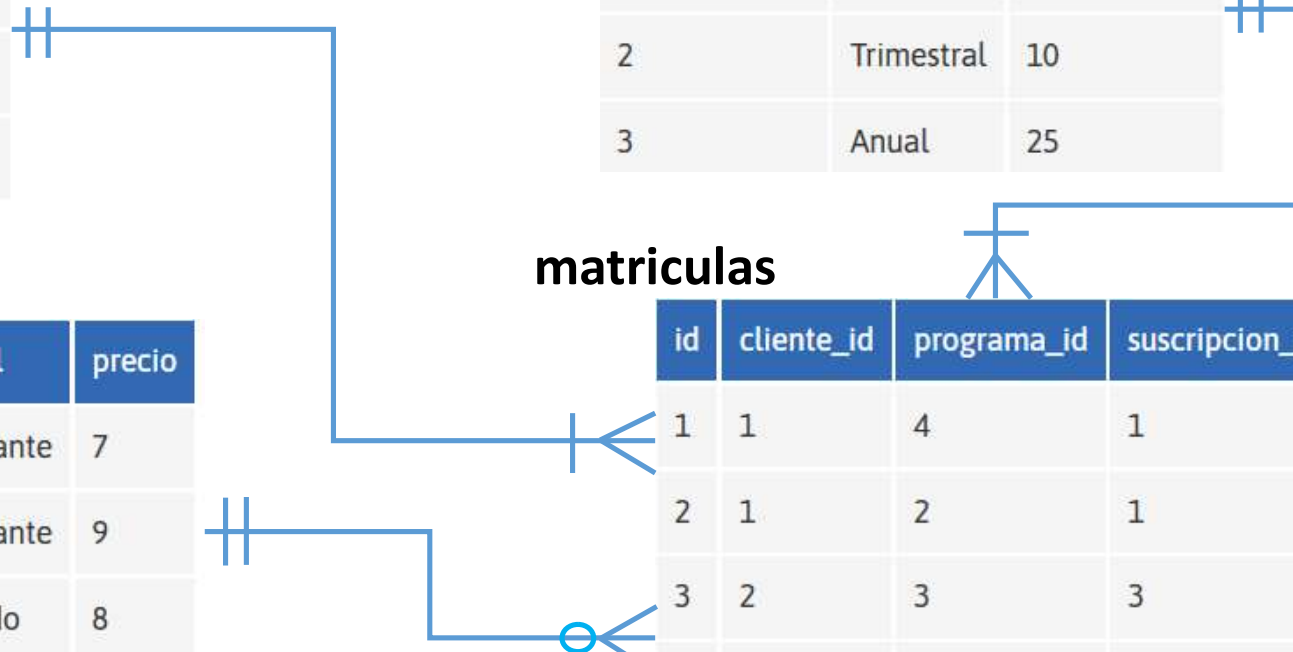
suscripcion_id	tipo	descuento_%
1	Mensual	0
2	Trimestral	10
3	Anual	25

programas




programa_id	idioma	nivel	precio
1	alemán	principiante	7
2	chino	principiante	9
3	francés	avanzado	8
4	inglés	intermedio	7

matriculas

id	cliente_id	programa_id	suscripcion_id
1	1	4	1
2	1	2	1
3	2	3	3
4	3	4	2



Una vez creadas las tablas tenemos que verificar y asegurar la consistencia de sus datos además de garantizar la eficiencia de su manipulación.  
Este proceso se denomina **NORMALIZACIÓN**.

-  **Redundancia:** la información se repite innecesariamente en muchas tuplas. En la relación siguiente, length y filmType.
-  **Anomalías de actualización:** cuando al cambiar la información en una tupla se descuida el actualizarla en otra.  
Si en la relación encontramos que el length de StarWars es 125, podríamos cambiarlo para la primer tupla y olvidar actualizar las demás.
-  **Anomalías de eliminación:** si un conjunto de valores llegan a estar vacíos y se llega a perder información relacionada como un efecto de la eliminación.  
Si eliminamos al actor Emilio Estevez, perdemos también la tupla de la película MightyDucks.

title	year	length	filmType	studioName	starName
Star Wars	1977	124	color	Fox	Carrie Fisher
Star Wars	1977	124	color	Fox	Mark Hamill
Star Wars	1977	124	color	Fox	Harrison Ford
Mighty Ducks	1991	104	color	Disney	Emilio Estevez
Wayne's World	1992	95	color	Paramount	Dana Carvey
Wayne's World	1992	95	color	Paramount	Mike Meyers

Una clave es un conjunto de atributos suficiente para distinguir las ocurrencias de una entidad (tabla) entre sí.

Los valores de los atributos deben ser tales que permitan identificar unívocamente cada ocurrencia de una entidad. En otras palabras, no se permite que ningún par de ocurrencias de una entidad tengan exactamente los mismos valores de sus atributos.

## Superclave

Es un conjunto de uno o más atributos que, tomados colectivamente, permiten identificar de forma única una ocurrencia dentro del conjunto de ocurrencias de una entidad.

En nuestro ejemplo de la academia de idiomas, el atributo *id-cliente* de la entidad cliente es suficiente para distinguir una ocurrencia cliente de las otras. Así, *id-cliente* es una superclave.

Análogamente, la combinación de *nombre-cliente* e *id-cliente* es una superclave del conjunto de ocurrencias de la entidad cliente. Por lo que una superclave puede contener atributos innecesarios.



# Algo más sobre las claves....



## Clave Candidata

Es una clave primaria alternativa utilizada para identificar unívocamente a una y sólo una presencia (ocurrencia) de cada entidad

El atributo *id-cliente* es un atributo que permite identificar unívocamente las ocurrencias de la entidad clientes. Supongamos que la combinación {*nombre-cliente* y *calle-cliente*} también es suficiente para distinguir entre los miembros de la entidad cliente. Entonces, los conjuntos {*id-cliente*} y {*nombre-cliente*, *calle-cliente*} son **claves candidatas**. Sin embargo, aunque los atributos *id-cliente* y *nombre-cliente* juntos puedan distinguir entidades cliente (superclave), su combinación no forma una clave candidata, ya que el atributo *id-cliente* por sí solo es una clave candidata.

## Clave Primaria

Es una clave candidata elegida por el diseñador de la base de datos como elemento principal para identificar las entidades dentro de un conjunto de entidades. **No puede tener valores repetidos ni NULOS.**

La designación de una clave representa una restricción en el desarrollo del mundo real que se modela. La clave primaria se debería elegir de manera que sus atributos nunca, o muy raramente, cambien.

## Clave Foránea o Externa

Representa un atributo que a su vez es una clave primaria en otra tabla, pero que está incluida entre los atributos de una tabla para permitir la relación

## Índices

El índice de una BD es una estructura de datos que mejora la velocidad de las operaciones, por medio de un identificador único de cada una de las filas de una tabla, permitiendo el acceso directo y rápido a los registros de dicha tabla.

El índice tiene un funcionamiento similar al índice de un libro, guardando parejas de elementos: el elemento que se desea indexar y su posición en la base de datos. Los índices pueden ser creados usando una o más columnas

Sin índice, se debe recorrer secuencialmente toda la tabla para encontrar un registro.

## Tablespace

Un tablespace es una unidad lógica de almacenamiento que almacena la información contenida en las tablas de la BD. Un tablespace se compone de, al menos, un datafile. Un datafile (fichero de disco) que solo puede pertenecer a un tablespace.

Cuando se crea una tabla o índice, se crean en un tablespace determinado, es decir, los tablespace son estructuras donde se almacenan los objetos del esquema de la base de datos, tales como tablas, índices, y se pueden repartir en varios ficheros.

## Usuarios/Roles de una Base de Datos

### ➤ ADMINISTRADOR (DBA)

La administración de los recursos de una BD (los datos y el sw relacionado del SGDM) es responsabilidad del administrador de la BD. El DBA es responsable de autorizar el acceso a la BD, coordinar y vigilar su utilización y adaptar los recursos de software y hardware que sean necesarios.

### ➤ DISEÑADORES

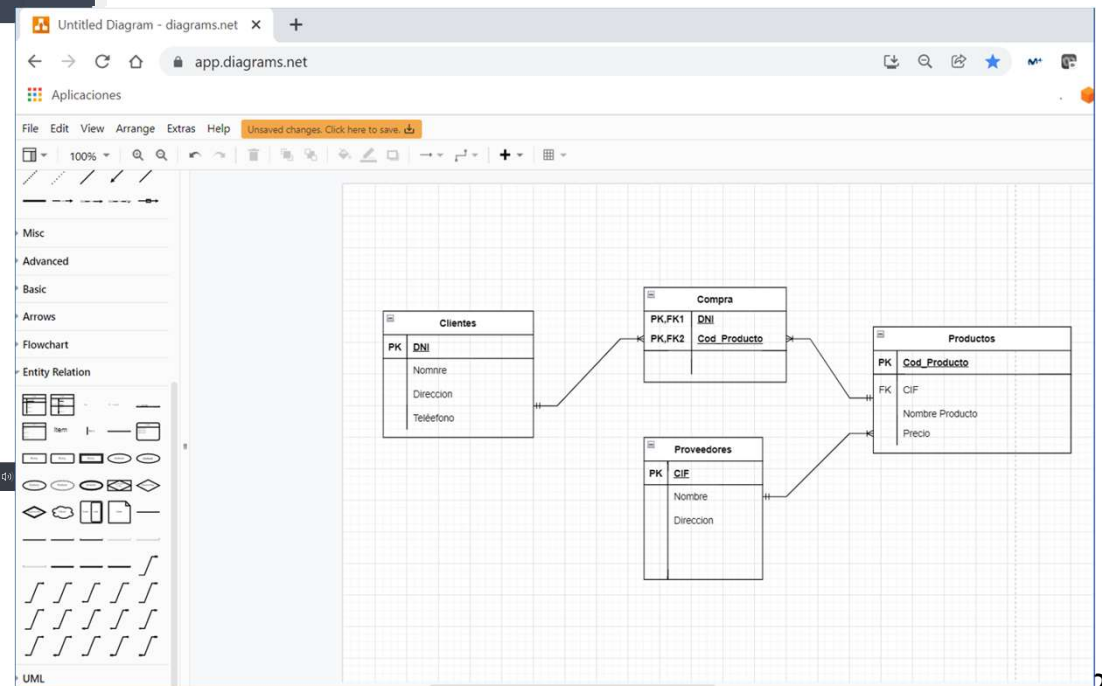
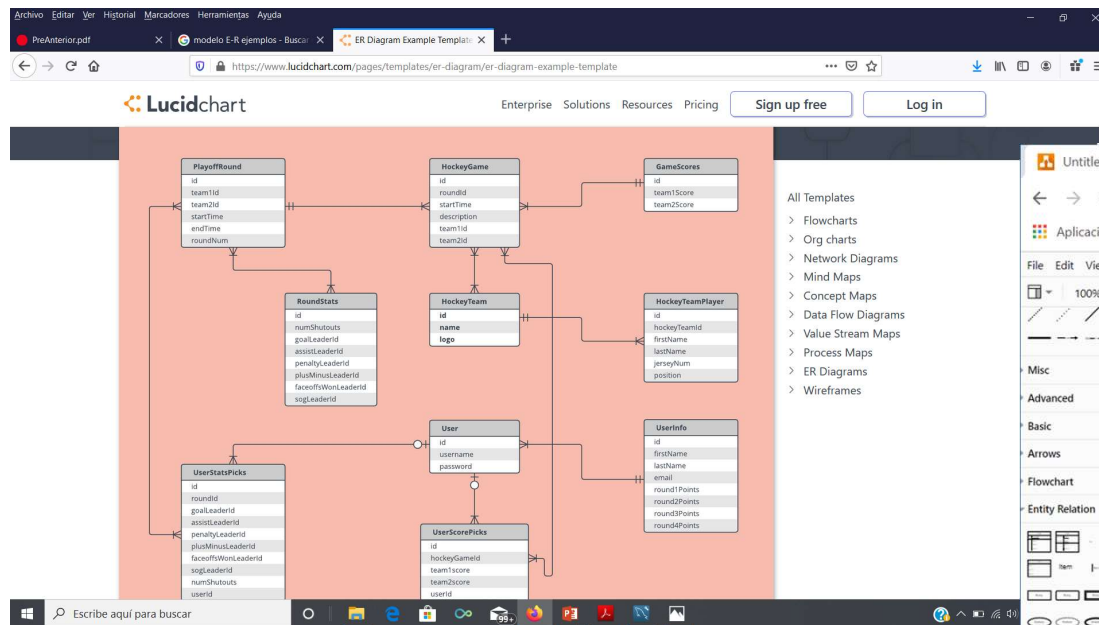
Se encargan de analizar los datos que se van a almacenar y de elegir las estructuras adecuadas para representar y almacenar dichos datos. Para ello, habrán tenido en cuenta todos los requerimientos de uso de la BD por parte de aplicaciones y usuarios.

### ➤ USUARIOS FINALES

Los usuarios finales son las personas/aplicaciones cuyos trabajos requieren acceso a la base de datos para consultar, actualizar y generar informes

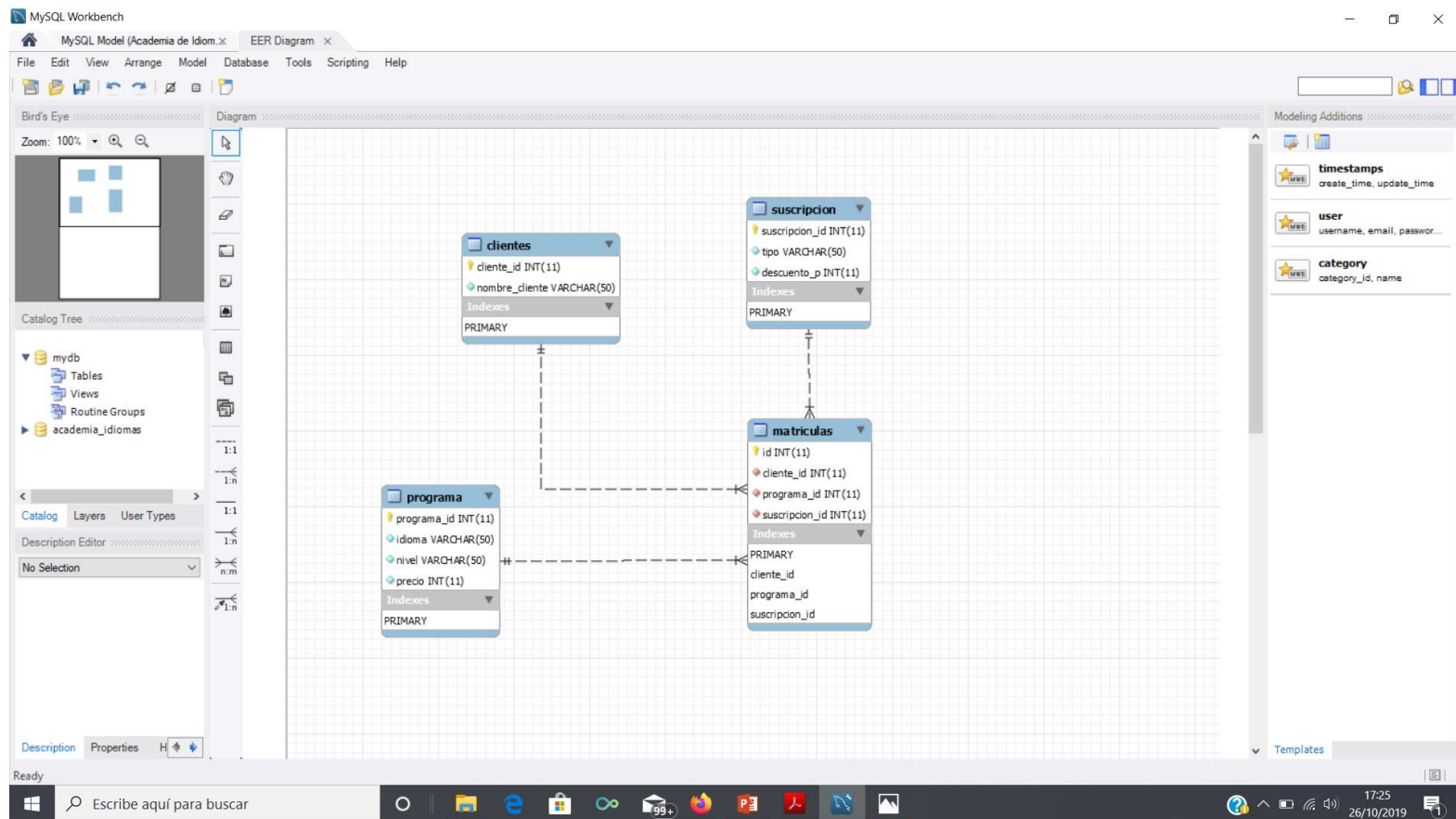
# Herramientas para diseñar modelos E/R y y diagramas

## Lucidchart/Drawio



# Herramientas gráficas para crear esquemas de BBDD

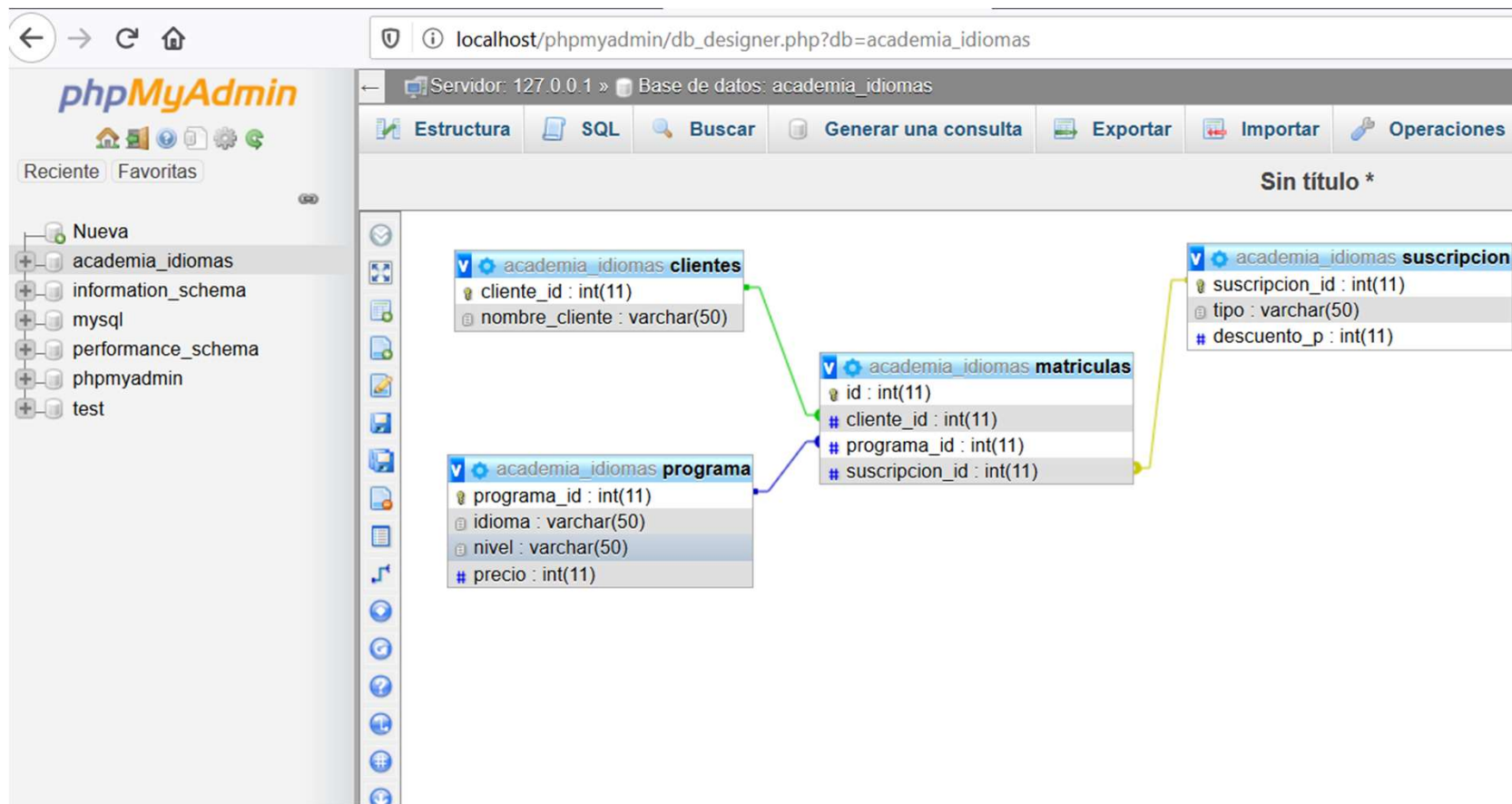
## Workbench de MySQL





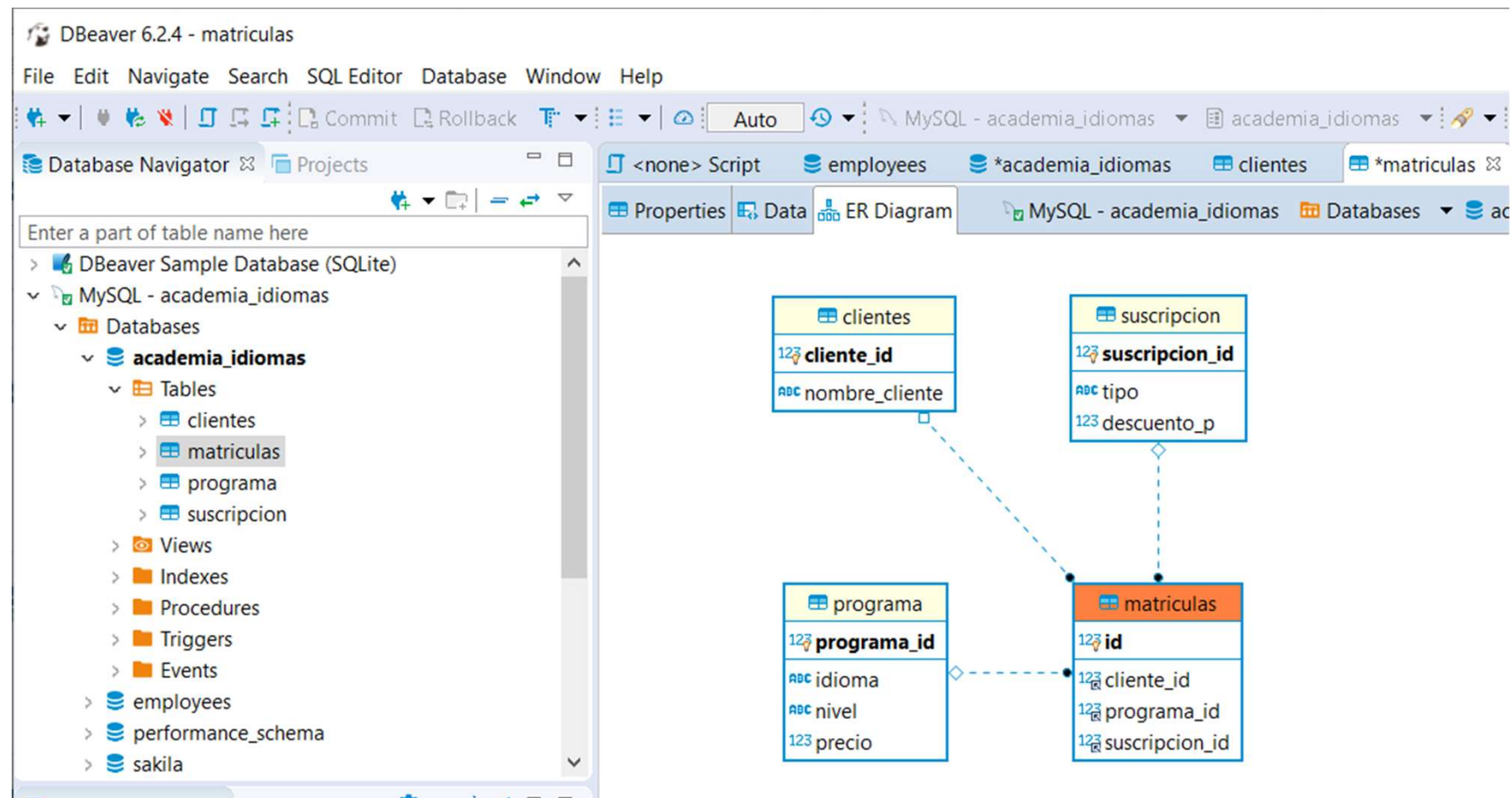
# Herramientas gráficas para crear esquemas de BBDD

## phpMyAdmin



# Herramientas gráficas para crear esquemas de BBDD

## DBeaver





 Calle Playa de Liencres, 2 bis  
(entrada por calle Rozabella)  
Parque Europa Empresarial  
Edificio Madrid  
28290 Las Rozas, Madrid

 900 373 379  [info@u-tad.com](mailto:info@u-tad.com)

 [SOLICITA MÁS INFORMACIÓN](#)



CENTRO ADSCRITO A:

 **Universidad  
Camilo José Cela**

PROYECTO COFINANCIADO POR:

