

Introducción a las Bases de Datos

**Contenido**

Introducción ...................................................................................................................... 2

Bases de Datos Relacionales ........................................................................................... 2

¿Qué es una tabla? ....................................................................................................... 2

Tipos de Datos .............................................................................................................. 3

Etapas básicas del diseño. ............................................................................................... 5

Introducción al modelo Entidad-Relación. ..................................................................... 6

Diagrama ER de un ejemplo. ........................................................................................ 7

# INTRODUCCIÓN

Lo primero y fundamental para ingresar a la programación con Bases de Datos es lo siguiente:



*Etapas de un proyecto.*

En la figura vemos bien definidas las tres etapas de todo proyecto. La primera es **DATOS**, que son los elementos básicos o fundamentales con los que cuento en el momento de inicio de la tarea a realizar, por si solos, ellos no poseen ningún valor agregado. La segunda es **PROCESO**, que es un conjunto de acciones o tareas que se realizan en un orden específico con un principio y un fin bien definidos. ***Los procesos pueden ser: identificar donde está ubicado el dato, combinar el dato o realizar una determinada operación.*** Por último, la tercera es **INFORMACIÓN** y es el resultado que se obtiene como consecuencia de procesar los datos, y el fin primordial que se busca en todo sistema o programa informático es la obtención de información certera y precisa en el menor tiempo posible. ***El dato por si solo no tiene ningún valor, salvo que se le aplique un proceso y se obtenga información.***

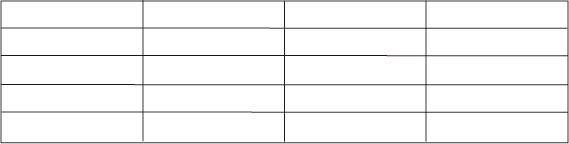
# Bases de Datos Relacionales

Una Base de Datos relacional consiste en un conjunto de tablas o relaciones, donde cada una de ellas varía o puede variar con el transcurso del tiempo y se identifica de manera única por medio de un nombre.

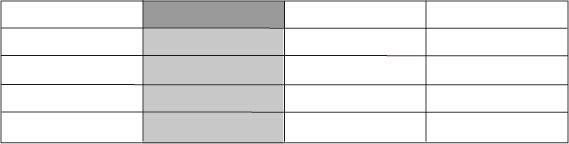
**¿Qué es una tabla?**

Una tabla es el único objeto de la base de datos en la que se guardan los datos. Está conformada por campos y registros formando una celda en la intersección de estos.

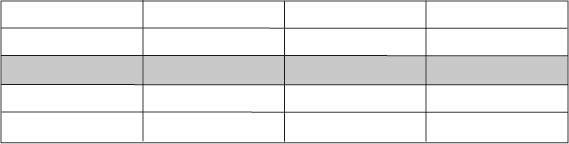
***Las tablas con objetos únicos con nombres únicos, por lo que ningún otro elemento de nuestra base de datos puede tener el mismo nombre que nuestra tabla.***



**Campo o Atributo**: También llamado Propiedad son los elementos que componen las entidades. Representa, por ejemplo, los apellidos de todos nuestros clientes, el teléfono, etc. Se representa por columna.



**Registro**: Equivale a una fila de atributos en el cual consta todas las propiedades de un mismo miembro de datos. Por ejemplo, los datos personales de un Alumno (Nombre, Dirección, Teléfono, CP, email, etc.).



**Tipos de Datos**

Dentro de la definición de las tablas nos encontraremos con distintos tipos de datos que podemos asignarle a cada campo, estos tipos de datos difieren según la base de datos con la cual nos encontremos trabajando, datos numéricos (enteros chicos, grandes, decimales, etc.), fecha y hora, cadenas de caracteres, booleanos, etc.

En el sentido más amplio, se podría considerar que una base de datos es simplemente un conjunto de información (básicamente una serie de “hojas con una determinada estructura”).

Por ejemplo, una base de datos muy sencilla podría ser una agenda de direcciones en la que anotemos datos de nuestros amigos. Tendríamos todas las letras con espacios para ingresar a cada uno de estos amigos. En cada espacio, a su vez, existirá una serie de apartados, como el nombre, la dirección, el teléfono, etc.

|  |  |  |  |
| --- | --- | --- | --- |
| ***Nombre*** | ***Dirección*** | ***Ciudad*** | ***Teléfono*** |
| José Luis | Ruta 197 2312 | Pacheco | 1111-1111 |
| Mariana | Av. Cazón 1239 | Tigre | 2222-2222 |
| Juan | Estrada 231 | Pacheco | 3333-3333 |

Pero en la práctica, una “**base de datos**” real suele estar formada por más de una tabla. Por ejemplo, la base de datos que utiliza una empresa “normal” para su gestión deberá almacenar datos sobre clientes, proveedores, artículos, facturas, etc.

Cada uno de estos “bloques” de datos será una tabla, y estas tablas estarán relacionadas entre sí (por ejemplo: un artículo será suministrado por un cierto proveedor, y ese artículo aparecerá en ciertas facturas, cada una de las cuales corresponderá a un cierto cliente).

Todo este conjunto de información que forman las tablas, las relaciones entre ellas y algunas cosas más será nuestra “base de datos”.

• *Base de Datos es un* ***conjunto exhaustivo no redundante(datos no repetidos)*** *de datos* ***estructurados organizados independientemente de su utilización*** *y su implementación en máquina* ***accesibles en tiempo real*** *y compatibles con* ***usuarios concurrentes con necesidad de información diferente(varios usuarios pueden acceder o hacer consultas al mismo tiempo a la bbdd)****.*

Nosotros trabajaremos en concreto, sobre lo que se conoce como una “**base de datos relacional**”.

***Bases de datos Estática: la información que tiene la bbdd nunca se va a modificar. Ejemplo: un censo, ya que la información que se obtuvo con ese censo en particular no cambia.***

***Bases de datos Dinámica: la información se va modificando a lo largo del tiempo. Ejemplo: bbdd de stock de productos.***

***El concepto de estático y dinámico también se pueden aplicar a los campod:***

***Tablas Maestras: con tablas que alimentan de determinada información a otras tablas. Son estáticas ya que sus campos no cambian. Un ejemplo son las tablas de provincias***

***Tablas dinámicas: se modifican los registros constantemente. Ejemplo tabla venta, cada vez que hago una venta se carga una registro. Si bien el registro es estático, porque la información de esa venta particular no cambia. La tabla es dinámica, porque le estoy agregando registros estáticos con cada venta.***

**Etapas básicas del diseño (son 3).**

Primero y principal, no te apresures a escribir código nunca. Si no diseñas primero la solución, te vas a encontrar después con miles de problemas en el camino en los que vas a gastar más tiempo del que hubieras gastado realizando un buen diseño.

***Antes de diseñar las bases, lo primero que hay que hacer es el relevamiento de la info necesaria (cuales son los proveedores, artículos, clientes o lo que se necesita) y de se relevamiento salen los requerimientos.*** Lo primero será documentar bien los requerimientos, la necesidad que vas a resolver/solucionar, los datos, como agruparlos y sus restricciones. Con los requerimientos bien documentados podrás ir diseñando cada uno de los objetos de tu base de datos. En nuestra cursada utilizaremos a forma de relevamiento los enunciados de los ejercicios.

## Diseño conceptual - D.E.R. (Diagrama Entidad Relación)

Se trata de un modelo para representar nuestras tablas (entidades) y las relaciones que existirán entre ellas. En esta etapa se recomienda realizarlo a mano, en una hoja donde podremos tachar, borronear y volver a escribir, o en una aplicación para representar este tipo de diagramas, teniendo en cuenta que se trata de un diagrama previo a la creación de la base de datos.

***Nunca confundir un DER con un Diagrama de Base de Datos. El DER se genera como una instancia previa antes de crear la base de datos y el Diagrama de Base de Datos es una foto de como quedó creada esa base de datos.***

* **Diseño lógico**

En esta fase, debemos **pensar en los campos que conformarán cada una de nuestras tablas** y en cómo normalizar nuestras tablas para evitar duplicidad de

información y para ahorrar espacio de almacenamiento. Esto último (ahorrar

espacio) ya no es tan importante como hace algunos años, pero es necesario para

optimizar el funcionamiento de una base de datos relacional ***Acá también de definen las claves primarias de cada tabla***

* **Diseño físico**

En esta última fase ya debemos revisar con detalle los tipos de datos que utilizaremos, sus dominios (qué valores va a permitir), cuales índices debemos crear para optimizar las consultas, entre otros. Aquí ya transformaremos nuestro diseño en una base de datos física, ya sea mediante la interfaz gráfica o escribiendo nuestro SQL en el motor de bases de datos elegido.

***Las Bases de Datos Relaciones se utilizan para bases de datos que perduren en el tiempo (ejemplo banco, universidad, etc).***

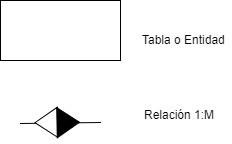
***Las Bases de datos No Sql tiene un propósito y se utilizan para análisis, estadísticas, relevamientos de datos, etc.***

**Introducción al modelo Entidad-Relación.**

Este es un modelo que nos permitirá “dibujar” las entidades y las relaciones que existen entre ellas. Nosotros usaremos un “Diagrama Entidad-Relación” (**DER**, de aquí en adelante). Existen varias notaciones ligeramente distintas. Vamos a utilizar la que consideramos más sencilla.

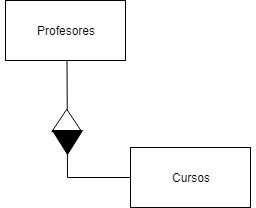
En esta notación se representan las entidades como un rectángulo y las relaciones binarias como un rombo partido por la mitad. **Las relaciones siempre serán para nosotros de uno a muchos (1:M),** una de las mitades (la que corresponde al

“muchos”) deberá estar sombreada.



*(Nota: hemos visto las relaciones expresadas también de otras formas, por otros autores; de momento, emplearemos esta notación y más adelante comentaré otras notaciones que es posible utilizar o encontrar en otros textos).*

**Diagrama ER de un ejemplo.**



Así de sencillo: tenemos 2 entidades (Profesores y Cursos) y una relación (impartir, entre profesores y cursos, 1:M).

Pensemos primero qué campos o atributos nos podrían interesar para integrar nuestras tablas o entidades, agregando también una tabla de “alumnos”.

Profesores:

DNI.



Nombre

Dirección

Ciudad

Teléfono

Título

Sueldo

Cuenta bancaria

Cursos:

Nombre del curso



Fecha de comienzo

Duración (horas)

Importe (pesos)

Número máximo de alumnos

Alumnos: Legajo



DNI (Documento Nacional de Identidad)

Nombre

Dirección

Ciudad

Teléfono

Fecha de nacimiento

Fecha de alta en el centro

Fotografía

***Tipos de datos:***

***Char: cadena de caracteres de longitud fija (máximo 20)***

***Varchar: cadena de caracteres de longitud variable***

***Numérico: únicamente para campos calculables.***

***Todos aquellos números que NO vayan a ser calculables, van a ser cadenas de caracteres (dni, legajo, teléfono, cuit, cbu)***

***En base a esto, las Primary Keys siempre van a ser cadenas de caracteres de longitud fija,***

***Con respecto a las imágenes, en una bbdd no se almacenan nunca imágenes. En su lugar se almacena la ruta al repositorio que tiene alojada la imagen.***

***Tipos de Claves:***

1. ***Primaria***
2. ***Foránea***
3. ***Unique: asume la propiedad de no poder repetirse en una tabla, evita la redundancia.(ejemplo si a un campo dni le declaro al lado unique, significa que en esa tabla no puedo cargar dos veces el mismo dni)***

***Las claves primarias concatenadas se permiten únicamente en las tablas intermedias, cuando las relaciones son de muchos a muchos.***

***Integridad referencial: las tablas intermedias mantienen una integridad referencial con las tabas con las que se relacionan, es decir hacen referencia a las tablas que la componen. Solo le puedo cargar datos que existan en las tablas a las que se hacen referencia (Ej tengo la tabla intermedia cusosxalumnos, y solo puede tener datos que ya estén cargados en la tabla cursos y en la tabla alumnos). A su vez los campos que relacionan o referencian a las tablas tiene que ser del mismo tipo exacto de dato y la misma longitud.***

Es sólo un ejemplo. En este momento estamos pensando en qué datos queremos almacenar, lo que debería ser el resultado de los requerimientos de nuestro relevamiento.

Deberíamos pensar en qué tipo de dato le asignaremos a cada campo de cada tabla, teniendo en cuenta que los únicos campos a los que les asignaremos tipo de dato numérico son aquellos que serán calculables, y que según el sistema de bases de datos que empleemos realmente, puede ocurrir que sea incómodo (o incluso imposible) trabajar con algunos de estos datos que hemos previsto (por ejemplo, la

“fotografía” del alumno).

Luego definiremos cual de esos datos nos permitirá **distinguir un registro de otro**. Esto se hace porque podemos tener dos alumnos con el mismo nombre, pero claramente son personas distintas, y debemos saber qué cursos ha realizado cada uno de ellos sin posibilidad de confusión, para no dar a uno el diploma que corresponda a otro, ni cobrarle un dinero de otro.

En el caso de los alumnos, no son datos únicos los siguientes: el nombre (puede repetirse, incluso con apellidos), la dirección (dos hermanos o dos amigos pueden vivir en la misma casa), el teléfono (ocurre lo mismo), la fecha de nacimiento (también podemos encontrar dos alumnos que hayan nacido el mismo día), etc. Lo que realmente distinguirá a un alumno de otro es su número de DNI (Documento Nacional de Identidad), o pasaporte, o CUIT/CUIL, que sí es único.

Pues bien, este dato que puede distinguir un alumno de otro (o en general un registro de otro) es lo que llamaremos la “**clave”**.

Puede ocurrir que no exista nada que nos sirva claramente como clave, como es el caso de los cursos: no es único el nombre (podemos impartir más de un curso con el mismo contenido), ni la fecha de comienzo (varios cursos pueden comenzar el mismo día), ni la duración, ni el importe, ni el número máximo de alumnos. En estos casos se suele añadir algo arbitrario, un **código**, que nos permita distinguir un curso de otro (en general un registro de otro). En nuestro caso, incluiríamos un nuevo atributo, llamado “Código de curso”.

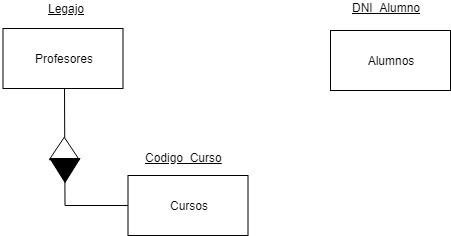
**Debemos tener en cuenta que para relacionar dos tablas debe existir entre ellas al menos un campo coincidente, con**

**exactamente el mismo tipo de dato relacionando registros en base a una igualdad en sus campos coincidentes.**

Un último comentario antes de ver cómo quedaría nuestro DER. Puede ocurrir que nuestra tabla tenga varios atributos únicos, todos los cuales puedan servir como clave. Entonces escogemos una de ellas (o varias) como “**clave principal**”, (luego veremos más tipos de claves) y el resto serán “**datos**” que no llegaremos a usar como claves. En el diagrama, el atributo que vaya a utilizarse como clave principal aparecerá subrayado.

En este caso se muestra la representación de un DER (diagrama de entidad relación).

Ahora ya sí, nuestro diagrama quedaría así (no incluimos todos los atributos que habíamos pensado, sólo las claves, que es con los que trabajaremos a partir de ahora):



Teniendo en cuenta que nosotros dijimos que nuestras relaciones **siempre** van a ser de **uno a muchos** y que un alumno va a asistir a muchos cursos, pero un curso va a tener a muchos alumnos, necesitamos formar de alguna manera una relación de **muchos a muchos** para lo cual agregaremos una tabla auxiliar que va a relacionar a las dos que queremos relacionar de esta forma.

