

**INDICE**

<b>1.- INTRODUCCIÓN.....</b>	<b>1</b>
<b>2.- DEFINICIONES INFORMALES.....</b>	<b>1</b>
<b>3.- RELACIÓN.....</b>	<b>3</b>
3.1.- INTENCIÓN.....	3
3.2.- EXTENSIÓN.....	4
<b>4.- CLAVES DE LAS RELACIONES.....</b>	<b>4</b>
<b>5.- INTEGRIDAD DE LOS ESQUEMAS RELACIONALES.....</b>	<b>4</b>
5.1.- REGLA DE INTEGRIDAD DE ENTIDADES.....	5
5.2.- REGLA DE INTEGRIDAD REFERENCIAL.....	5
5.3.-OTRAS RESTRICCIONES.....	6
<b>6.- NORMALIZACIÓN.....</b>	<b>7</b>
6.1.- PRIMERA FORMA NORMAL.....	8
6.2.- SEGUNDA FORMA NORMAL.....	8
6.3.- TERCERA FORMA NORMAL.....	9

**1.- INTRODUCCIÓN**

El modelo relacional fue creado por Codd en 1970 y es el modelo lógico en el que se basan la mayoría de los SGBD comerciales en uso hoy en día.

El documento de Codd propone un modelo de datos basado en la teoría de las relaciones, en donde los datos se estructuran lógicamente en forma de relaciones (tablas), siendo el objetivo fundamental del modelo mantener la independencia de esta estructura lógica respecto al modo de almacenamiento y a otras características de tipo físico.

Codd introduce el concepto de relación (tabla) como estructura básica del modelo. Todos los datos de una base de datos se representan en forma de relaciones cuyo contenido varía en el tiempo. Formalmente, una relación es un conjunto de filas en la terminología relacional.

Al encontrarnos en la fase de diseño lógico este modelo debe proporcionar también una parte dinámica que permita crear y manipular las relaciones. Los operadores que se pueden aplicar a las tablas conforman el álgebra relacional y el cálculo relacional.

Por último cabe decir, que el modelo relacional se basa en dos ramas de las matemáticas: la teoría de conjunto y la lógica de predicados de primer orden.

**2.- DEFINICIONES INFORMALES**

Una **relación** es una tabla con columnas y filas

Un **atributo** es el nombre de una columna de la relación.

En el modelo relacional, las relaciones se utilizan para almacenar información sobre los objetos que se representan en la base de datos. Una relación se representa gráficamente como una tabla bidimensional en la que las filas corresponden a registros individuales y las columnas corresponden a los campos o atributos de esos registros. Los

atributos pueden aparecer en la relación en cualquier orden, y están caracterizados por un nombre. Cada columna contiene valores de un solo atributo.

## RELACIÓN

Atributo 1	Atributo 2	.....	Atributo n
xxxxx	xxxxx	xxxxx	xxxxx
xxxxx	xxxxx	xxxxx	xxxxx
.....	.....	.....	.....
xxxxx	xxxxx	xxxxx	xxxxx

El **grado** de una relación es el número de atributos que contiene

Una **tupla** es una fila de la relación. Las tuplas de la relación no siguen ningún orden.

La **cardinalidad** de una relación es el número de tuplas que contiene. Ya que en las relaciones se van insertando y borrando tuplas a menudo, la cardinalidad de las mismas varía constantemente.

Ejemplo: Tabla EMPLEADO

Codigo	Nombre	Apellido	Dirección	Telefono	Fax	Nacionalidad	Edad
11111 A	Pedro	González	Calle 1	620 464 145	95 464 146	Española	29
22222 B	María	Díaz	Calle 2	635 236 258	95 236 259	Francesa	30
33333 C	Juan	Ruiz	Calle 3	620 585 247	95 585 248	Española	31
44444 D	José	Pérez	Calle 4	671 498 125	95 498 126	Española	29
55555 E	Julia	Pons	Calle 5	628 532 165	95 321 655	Italiana	36

Grado de la relación: 8

Cardinalidad: 5

Un **Dominio** es un conjunto finito de valores homogéneos y atómicos de uno o varios atributos.

*Homogéneo: son todos del mismo tipo*

*Atómico: son indivisibles en lo que al modelo se refiere, ya que si se dividen pierden el significado*

Cada atributo de una base de datos relacional se define sobre un dominio, pudiendo haber varios atributos definidos sobre el mismo dominio. En la tabla anterior los atributos Tfo y Fax están definidos sobre el mismo dominio.

Los dominios pueden definirse por intención y por extensión.

- Por intención: El dominio de las edades de los empleados se define como entero de longitud (2) comprendido entre 18 y 65
- Por extensión: El dominio de las nacionalidades serán los nombres de las distintas nacionalidades que admitiésemos en nuestra base de datos

Si definimos el dominio por intención de las nacionalidades como carácter de longitud (10) sería una definición muy pobre porque cualquier cadena de 10 caracteres sería válida.

Ejemplo de dominio

<i>Atributo</i>	<i>Nombre del dominio</i>	<i>Descripción</i>	<i>Definición</i>
Código	NOM_CODIGO	Código de empleado	7 caracteres
Nombre	NOM_NOMBRE_AP	Nombres y apellidos de personas	15 caracteres
Dirección	NOM_DIRECCION	Nombre de direcciones de España	25 caracteres
Tfo	NUM_TEL_FAX	Números de teléfono de España	9 caracteres
Fax	NUM_TEL_FAX	Números de teléfono de España	9 caracteres
Nacionalidad	NOM_NACIONALIDAD	Nombres de las nacionalidades	Francesa, Española, Italiana...
Edad	NUM_EDAD	Edades de Empleados	Entero (2)

Un Atributo A es el papel que tiene un determinado Dominio D en una relación; se dice que D es el dominio de A y se denota como  $\text{dom}(A)$ .

Es muy normal dar el mismo nombre al atributo y al dominio subyacente. En el caso de que sean varios los atributos de una misma tabla definidos sobre el mismo dominio, habrá que darle nombre distinto a los atributos, ya que una tabla no puede tener dos atributos con el mismo nombre.

Las características que ha de tener las relaciones son:

- Cada relación tiene un nombre distinto
- El orden de los atributos no importa: no están ordenados
- Cada tupla es distinta de las demás: no hay tuplas duplicadas
- El orden de las tuplas no importa: no están ordenadas.

### 3.- RELACIÓN

Ejemplo: TIENDA

Tid	Tdir	Tloc
T1	C/ Arroyo, 55	SEVILLA
T2	Carretera Carmona 66	SEVILLA

Existen básicamente dos formas de definir una relación: **la intención** o esquema de la relación y **la extensión** u ocurrencia de la relación.

#### 3.1.- Intención

La intención, denotado como  $R (A_1:D_1, A_2:D_2, \dots, A_n:D_n)$ , es un conjunto de n pares de atributos-dominio subyacente  $\{(A_i:D_i)\}$  donde n es el grado de la relación.

La intención es la parte relativamente estática de la relación, también llamada cabecera de la tabla cuando la relación se percibe como una tabla

Ejemplo:

TIENDAS (Tid:D\_Tid; Tdir:D\_Tdir, Tloc:D\_Tloc)

La intención va a ser siempre invariante con el tiempo y se trata de la definición de los atributos, de los cuales cada uno está definido en su correspondiente dominio.

### 3.2.- Extensión

La extensión u ocurrencia de una relación (llamada simplemente relación) depende del momento específico en el cual la relación es tomada en cuenta, y hace referencia al conjunto de  $m$  tuplas que forman parte de la relación en un instante dado.

Definición matemática:  $r(R)$  es un conjunto de  $m$  tuplas  $\{t_1, t_2, \dots, t_m\}$  donde cada tupla es un conjunto de  $n$  pares de atributos-valor  $\{(A_i: V_{ij})\}$ , donde  $V_{ij}$  es el valor  $j$  del dominio  $D_i$  asociado al atributo  $A_i$ .

La extensión de una relación representa a cada uno de las tuplas pertenecientes a una relación existente en el dominio del problema en un instante dado. La extensión de una relación es dinámica, varía en el transcurso del tiempo.

Ejemplo:

TIENDA

Tid	Tdir	Tloc
T1	C/ Arroyo, 55	SEVILLA
T2	Carretera Carmona 66	SEVILLA

## 4.- CLAVES DE LAS RELACIONES

Ya que en una relación no hay tuplas repetidas estas se pueden distinguir una de otras, es decir, se pueden identificar de forma única, mediante los valores de sus atributos.

Al o los atributos (tal vez compuestos) que satisfacen la propiedad de identificación única de las tuplas de una relación se les denomina **claves candidatas**. Toda relación, por definición, debe tener alguna clave candidata.

De entre todas las claves candidatas de una relación, se deberá especificar cuál de ella se considera como **clave primaria o principal**, denominándose al resto de las claves candidatas como **claves alternas o claves alternativas**.

Una **clave ajena** de una relación es un conjunto no vacío de atributo/s cuyos valores coinciden con los valores de la clave primaria de alguna otra relación (puede ser la misma). **La clave ajena y su correspondiente clave primaria han de estar definida sobre los mismos dominios.**

Ejemplo:

EDITORIAL (Nombre e, Dirección, Ciudad, País,...)

LIBRO (Código, Título, Idioma..., Nombre e)

Se dice que un valor de clave ajena representa “una referencia” a la tupla que contiene el mismo valor en su clave primaria.

## 5.- INTEGRIDAD DE LOS ESQUEMAS RELACIONALES

La intención de un esquema relacional deben satisfacer las siguientes reglas de integridad mediante las cuales se garantiza la consistencia de la información que puede ser manejada sobre la base del esquema:

### 5.1.- Regla de integridad de entidades

Ningún atributo que forme parte de la clave primaria de una relación puede tomar un valor nulo.

Un valor nulo puede significar:

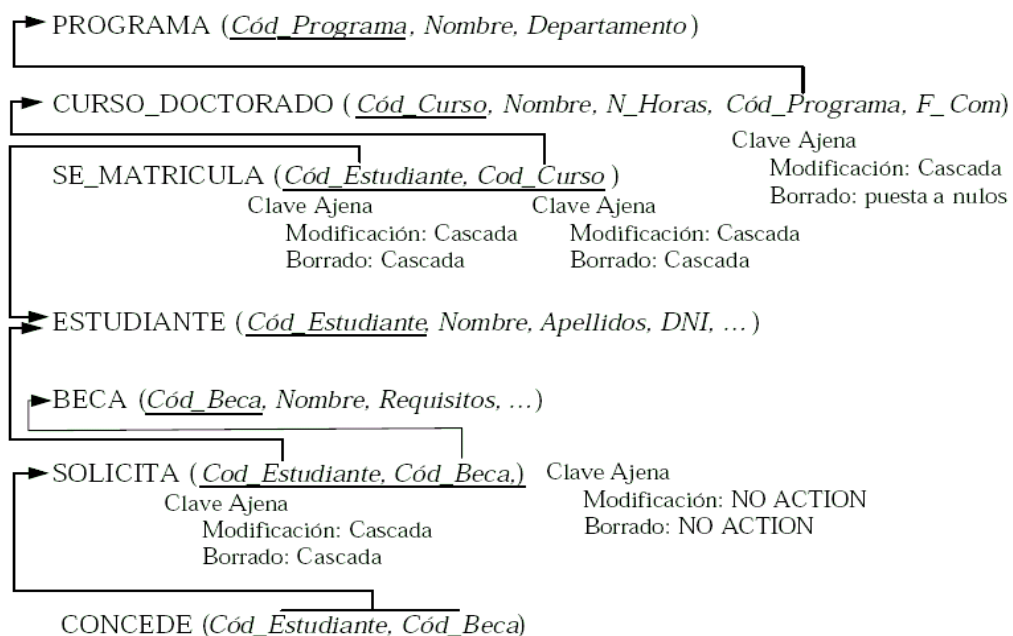
- el atributo no es aplicable a esa fila
- el valor del atributo para esa fila es desconocido
- el valor se conoce pero no se ha registrado todavía.

### 5.2.- Regla de integridad referencial

Si en una relación hay alguna clave ajena, sus valores deben coincidir con valores de la clave primaria a la que hace referencia, o bien, deben ser completamente nulos.

Una vez definidas las claves ajenas, hay que establecer las consecuencias que pueden tener las operaciones de borrado y modificación realizadas sobre las tuplas de la relación referenciada.

Ejemplo:



*Ejemplos de claves ajenas con opciones de borrado y modificación*

Las opciones que se pueden distinguir son:

➤ Operación restringir (RESTRICT o NO ACTION)

El borrado o modificación de tuplas de la relación que contiene la clave primaria sólo se permite si no existen tuplas con dicha clave en la relación que contiene la clave ajena.

*Para poder borrar una beca en nuestra base de datos no tendría que haber nadie que hubiese solicitado esa beca, en caso contrario el sistema impediría el borrado.*

➤ Operación con transmisión en cascada (CASCADE)

El borrado o modificación de tuplas de la relación que contiene la clave primaria referenciada lleva consigo el borrado o modificación en cascada de las tuplas de la relación que contiene la clave ajena.

*Al modificar el código de un estudiante en la relación ESTUDIANTE, se modifica también el código de estudiante en las relaciones SOLICITA y SE MATRICULA (también se cambia en concede)*

➤ Operación con puesta a nulos (SET NULL)

El borrado o la modificación de tuplas de la relación que contiene la clave primaria referenciada lleva consigo poner a nulos los valores de las claves ajenas de la relación que referencia.

*Cuando se borra un programa, a los Cursos de Doctorado que tienen ese programa se les coloca el atributo cod\_programa a nulo. Esta opción sólo es posible cuando el atributo que es clave ajena admite el valor nulo.*

➤ Operación con puesta a valor por defecto (SET DEFAULT)

El borrado o la modificación de tuplas de la relación que contiene la clave primaria referenciada lleva consigo pone el valor por defecto a la clave ajena de la relación que referencia; el valor por defecto habrá sido definido al crear la tabla correspondiente.

➤ Operación que desencadena un procedimiento de usuario

El borrado o la modificación de tuplas de la tabla referenciada pone en marcha un procedimiento definido por el usuario.

La opción seleccionada en caso de borrado es independiente de la de modificación.

### 5.3.-Otras restricciones

En la definición del esquema relacional pueden imponerse, en teoría (dependerá del SGBD) otra serie de restricciones que garanticen la integridad del modelo y, por lo tanto, de la información almacenada en la base de datos.

Estas restricciones pueden ser concernientes a:

- Los valores permitidos por los atributos que forman parte de las relaciones existentes en el esquema. Por ejemplo, valor máximo y mínimo, lista de valores, etc.
- Condiciones que determinan los valores que pueden tomar los atributos. Estas condiciones pueden definirse en base a diferentes predicados: en función del valor de otros atributos de la misma o diferente relación, o al estado de la base de datos, o en función del usuario, etc.

Tipos:**VERIFICACIÓN (CHECK):**

Comprueba, en toda operación de actualización, si el predicado es cierto o falso y, en el segundo caso, rechaza la operación. La restricción de verificación se define sobre un único elemento (dentro de un CREATE TABLE) y puede o no tener nombre.

CHECK N\_HORAS > 30 en CURSO\_DOCTORADO

**ASERCIÓN (ASSERTION):**

Actúa de forma idéntica a la anterior, pero se diferencia de ella en que puede afectar a varios elementos (por ejemplo, a dos tablas distintas). Por tanto, su definición no va unida a la de un determinado elemento del esquema y siempre ha de tener un nombre.

CREATE ASSERTION CONCEDE\_SOLICITA AS  
CHECK (SELECT Cod\_Estudiente, Cod\_Beca FROM CONCEDE) IN  
(SELECT Cod\_Estudiente, Cod\_Beca FROM SOLICITA));

**TRIGGER (DISPARADOR):**

Define una acción procedimental que la base de datos siempre debería realizar cuando se realice alguna operación sobre la base de datos. Esta operación puede ser de inserción, borrado o modificación

## 6.- NORMALIZACIÓN

En el modelo relacional podemos tener diferentes esquemas del mismo problema (algunos serán mejores que otros). El diseño de dichos esquema se puede abordar de dos formas distintas.

- 1º Obteniéndolo directamente a partir de la observación de nuestro Universo del discurso
- 2º Realizando el proceso en dos fases, en la primera se lleva a cabo el diseño conceptual entidad relación extendido y en la segunda se transforma éste en un esquema relacional siguiendo unas determinadas reglas de transformación.

A veces, las relaciones diseñadas pueden presentar algunos problemas como consecuencia de una mala interpretación o un error en el esquema entidad-relación. Los principales problemas que pueden mostrar son:

- Incapacidad de almacenar ciertos datos
- Redundancias o incoherencias
- Pérdidas de información
- Anomalías de inserción, borrado o modificación.

Para evitar estos problemas se utiliza una serie de reglas denominadas “Reglas de normalización de las relaciones”. La teoría en la que se basan dichas reglas utiliza el

método de la descomposición que consiste en dividir una relación en múltiples relaciones.

Para ilustrar las definiciones de las formas normales, consideraremos como ejemplo una tabla en la que se recoge información sobre la organización de las clases presenciales en un determinado centro educativo. En cada curso hay varios grupos de clase, y en cada grupo se imparten varias asignaturas. Cada asignatura es impartida por uno o varios profesores, cada uno de los cuales se encarga de esa asignatura en uno o varios grupos. Cada asignatura tiene, además, un profesor coordinador, que puede ser, o no, uno de los profesores que la imparten. Cada profesor tiene asignado un despacho.

Toda esta información se recoge en la siguiente relación

R (Grupo, Asignatura, Profesor, Despacho, Coordinador)

Los datos que tenemos en el esquema quedan reflejados en la siguiente tabla:

Grupo	Asignatura	Profesor	Despacho	Coordinador
11, 12	Cálculo	F. Díaz	D-23	R. Pérez
13	Cálculo	A. García	D-27	R. Pérez
11	Álgebra	C. Morales	D-34	F. Arranz
12	Álgebra	M. Campos	D-21	F. Arranz
13	Álgebra	F. Arranz	D-32	F. Arranz

### 6.1.- Primera Forma Normal

Se dice que una tabla está en primera forma normal (1FN) si la información asociada a cada una de las columnas es un valor único, y no una colección de valores en número variables.

La relación anterior no está en 1FN porque como vemos en la primera fila hay varios grupos con el mismo profesor.

Podemos forzar que la relación esté en 1FN estableciendo una clave compuesta en la relación R

R (Grupo, Asignatura, Profesor, Despacho, Coordinador)

Los datos en la tabla están en 1FN tal como se indica a continuación:

<u>Grupo</u>	<u>Asignatura</u>	Profesor	Despacho	Coordinador
11	Cálculo	F. Díaz	D-23	R. Pérez
12	Cálculo	F. Díaz	D-23	R. Pérez
13	Cálculo	A. García	D-27	R. Pérez
11	Álgebra	C. Morales	D-34	F. Arranz
12	Álgebra	M. Campos	D-21	F. Arranz
13	Álgebra	F. Arranz	D-32	F. Arranz



## 6.2.- Segunda Forma Normal

Se dice que la tabla está en 2º forma normal (2FN) si está en 1FN y además hay una clave primaria (una columna o combinación de varias) que distingue cada fila, y cada atributo que no forme parte de la clave primaria dependa de toda la clave primaria.

En la tabla anterior la clave primaria es la combinación de las dos primeras columnas.

R (Grupo, Asignatura, Profesor, Despacho, Coordinador)

La tabla R no está en 2FN porque el dato del coordinador de la asignatura no depende de toda la clave primaria, sino sólo de parte de ella (depende de la asignatura, pero no del grupo). Se puede conseguir la 2FN suprimiendo ese dato de la tabla principal, y usando una tabla auxiliar para relacionar la asignatura con el coordinador, tal como se hace en las tablas siguientes, de tal forma que la relación y los datos de las tablas quedaría de la siguiente forma:

R (Asignatura, Coordinador)

R1 (Grupo, Asignatura, Profesor, Despacho) donde R1.Asignatura → R

Datos Tabla R1

<u>Grupo</u>	<u>Asignatura</u>	Profesor	Despacho
11	Cálculo	F. Díaz	D-23
12	Cálculo	F. Díaz	D-23
13	Cálculo	A. García	D-27
11	Álgebra	C. Morales	D-34
12	Álgebra	M. Campos	D-21
13	Álgebra	F. Arranz	D-32

Datos tabla R

<u>Asignatura</u>	Coordinador
Cálculo	R. Pérez
Álgebra	F. Arranz

La 2FN solo se puede violar cuando la clave primaria sea una clave compuesta (conste de más de un atributo).

## 6.3.- Tercera forma normal

Se dice que una tabla está en 3º forma normal (3FN) si satisface el criterio de la 2FN y además el valor de cada columna que no es clave primaria depende directamente de la clave primaria, es decir, no hay dependencias entre columnas que no son clave primaria.

La relación R1 no está en 3FN porque el despacho del profesor depende únicamente del profesor. Puede conseguirse la 3FN eliminando de la tabla cada columna dependiente de otra no clave primaria y usando una relación auxiliar para registrar dicha dependencia.

La relación normalizada en 3FN quedaría por tanto de la siguiente

R (Asignatura, Coordinador)

R1 (Profesor, Despacho)

R2 (Grupo, Asignatura, Profesor)

donde R2.Asignatura → R

R2.Profesor → R1

Y los datos en la tabla quedarán en la siguiente la siguiente manera:

<u>Grupo</u>	<u>Asignatura</u>	Profesor
11	Cálculo	F. Díaz
12	Cálculo	F. Díaz
13	Cálculo	A. García
11	Álgebra	C. Morales
12	Álgebra	M. Campos
13	Álgebra	F. Arranz

<u>Profesor</u>	Despacho
F. Díaz	D-23
A. García	D-27
C. Morales	D-34
M. Campos	D-21
F. Arranz	D-32

<u>Asignatura</u>	Coordinador
Cálculo	R. Pérez
Álgebra	F. Arranz