

# Práctica 10

## Normalización

### 1. Objetivo General

Conocer y aplicar formas normales al diseño de una base de datos.

### 2. Objetivos Secundarios

- Conocer y practicar con dependencias funcionales.
- Reducir inconsistencias y redundancias de las bases de datos.

### 3. Introducción

Las bases de datos pueden modelarse a través de diversos diseños debido a la complejidad del contexto o a la experiencia del diseñador. Esto nos llevará a encontrar maneras distintas de ordenar los datos y las tablas que formarán parte de una base de datos.

El creador del modelo relacional, Edgar F. Codd, estableció una serie de reglas para las bases de datos con el propósito de que los modelos relacionales resultantes fueran eficientes. Estas reglas se ocupan de mantener la calidad organizacional de los datos y las tablas, mediante una estructura abstracta que se puede aplicar de manera universal a cualquier modelo relacional. A éstas reglas se les llama "Reglas de Codd"

Por otra parte, existen criterios sobre los modelos relacionales que permiten determinar que tan vulnerable es una tabla a volverse inconsistente. A estos criterios se les conoce como Formas Normales, mientras que al proceso de aplicarlas a una base de datos se le llama Normalización.

Para ejemplificar la utilidad de la normalización, transformaremos una tabla con el objetivo de alcanzar una o más formas normales. Como ejemplo tomaremos una pizzería con entrega a domicilio. Supongamos que se ha tomado la decisión de controlar los datos de los clientes y los detalles de las órdenes que éstos solicitan.

La Tabla 10.1 muestra los datos tal como se manejan hasta el día de hoy en dicha pizzería.

Tabla 10.1 - Datos sin normalizar de la Pizzería.

No. Orden	Cliente	Teléfono	Dirección	Orden		
1	Pedro	55894320	Calle 1 No. 20 Col. Condesa	1 Pizza hawaiana	1 Pizza Carnes Frías	2 Refrescos Limon Grande
2	Edna	59131677	Venustiano Juárez 83 Col. Polanco	2 Pizzas Jamón	1 Jugo Mango Mediano	NULL
3	Carlos	22267457	Av. Baja California 545 Col. Roma	1 Pizza 4 quesos chica	1 orden papas	1 refresco cola grande
4	Daniel	56983640	Benito Carranza 3 Col. Del Valle	1 Pizza Carnes frías grande	NULL	NULL
5	Genaro	34298561	Av. Universidad 1080 Col. Del valle	4 pizzas jamón grandes	4 pizzas carnes frías grandes	5 refrescos cola grande
6	Sara	90277506	Tuxpan 67 Col. Roma	3 pizzas 4 quesos	NULL	NULL

Se puede apreciar que los datos que se almacenan en esta tabla son: el número de orden, los datos del cliente y el detalle de la orden. Un breve análisis muestra que aunque los datos se

encuentren organizados en columnas, no supone una organización que permita realizar consultas precisas.

Por ejemplo, si se requiere saber cuántas pizzas de carnes frías compró Genaro, se tendría que realizar un análisis de la cadena de caracteres almacenada en *Orden* para averiguarlo, lo cual complica la interpretación de los datos para su posterior transformación en información. Para evitar éste tipo de situaciones se deberá de aplicar el proceso de normalización, es decir, transformar las tablas originales para alcanzar la forma normal deseada.

En el proceso de transformación, entre otros procedimientos, se utiliza principalmente la descomposición de una tabla en dos o más. Esta división será tal que no se pierdan datos ni la información que de ellos se derive, a éste paso se le denomina descomposición sin pérdida o *Lossless-Join*.

A continuación se presentan las principales formas normales:

1. Primera Forma Normal:

***Todos los atributos deben ser atómicos.*** En otras palabras, la primera forma normal asegura que las columnas de una tabla no sean multivaluadas. Las columnas multivaluadas son aquellas que contienen más de un valor del dominio para un solo renglón, por ejemplo, la columna *Orden* de la Tabla 10.1.

Esta forma normal incluye también los siguientes hechos:

- No hay orden de arriba-abajo en los registros, es decir, el orden de éstas no aporta mayor información.
- No hay orden de izquierda a derecha en atributos, análogamente al punto anterior, el orden de los atributos tampoco aporta mayor información.
- No hay registros duplicados.
- Cada intersección fila – columna contiene exactamente un valor del dominio.

Para hacer que la tabla cumpla con la primera forma normal se deben reorganizar los datos contenidos en esa columna de tal manera que sean únicos por renglón, es decir, que sólo exista un artículo por cada renglón. La Tabla 10.2 muestra un arreglo de la Tabla 10.1 que intenta llegar a la primera forma normal.

Tabla 10.2 - Primera arreglo de los datos de la tabla órdenes de la Pizzería.

<b><i>No. Orden</i></b>	<b><i>Cliente</i></b>	<b><i>Teléfono</i></b>	<b><i>Dirección</i></b>	<b><i>Orden</i></b>
1	Pedro	55894320	Calle 1 No. 20 Col. Condesa	1 Pizza hawaiana mediana
1	Pedro	55894320	Calle 1 No. 20 Col. Condesa	1 Pizza Carnes Frías grande
1	Pedro	55894320	Calle 1 No. 20 Col. Condesa	2 Refrescos Limón Grande

2	Edna	59131677	Venustiano Juárez 83 Col. Polanco	2 Pizzas Jamón Mediana
2	Edna	59131677	Venustiano Juárez 83 Col. Polanco	1 Jugo Mango Mediano
3	Carlos	22267457	Av. Baja California 545 Col. Roma	1 Pizza 4 quesos chica
3	Carlos	22267457	Av. Baja California 545 Col. Roma	1 orden papas grande
3	Carlos	22267457	Av. Baja California 545 Col. Roma	1 refresco cola mediano
4	Daniel	56983640	Benito Carranza 3 Col. Del Valle	1 Pizza Carnes frías grande
5	Genaro	34298561	Av. Universidad 1080 Col. Del valle	4 pizzas jamón grandes
5	Genaro	34298561	Av. Universidad 1080 Col. Del valle	4 pizzas carnes frías grandes
5	Genaro	34298561	Av. Universidad 1080 Col. Del valle	5 refrescos cola grande
6	Sara	90277506	Tuxpan 67 Col. Roma	3 pizzas 4 quesos medianas

Nótese que no se ha perdido información entre la existente en la Tabla 10.1 y la almacenada en el nuevo esquema de la Tabla 10.2.

En la Tabla 10.2 aunque cada renglón representa un único artículo en la columna Orden, se puede apreciar que cada registro almacena información multivaluada de diverso dominio, por ejemplo, en la misma celda se almacenan datos como el producto, la cantidad y el tamaño. Por lo tanto no se encuentra en primera forma normal.

En la Tabla 10.3 se propone una nueva descomposición del atributo *Orden* que lleve a la tabla a primera forma normal.

Tabla 10.3 - Primer arreglo para lograr la primera Forma Normal de la tabla de órdenes de la Pizzería.

No. Orden	Cliente	Teléfono	Dirección	Orden		
				Artículo	Cantidad	Tamaño
1	Pedro	55894320	Calle 1 No. 20 Col. Condesa	Pizza hawaiana	1	Mediana
1	Pedro	55894320	Calle 1 No. 20 Col. Condesa	Pizza Carnes Frías	1	Grande
1	Pedro	55894320	Calle 1 No. 20 Col. Condesa	Refresco Limón	2	Grande
2	Edna	59131677	Venustiano Juárez 83 Col. Polanco	Pizza Jamón	2	Mediana
2	Edna	59131677	Venustiano Juárez 83 Col. Polanco	Jugo Mango	1	Mediano
3	Carlos	22267457	Av. Baja California 545 Col. Roma	Pizza 4 quesos	1	Chica
3	Carlos	22267457	Av. Baja California 545 Col. Roma	orden papas	1	Grande
3	Carlos	22267457	Av. Baja California 545 Col. Roma	refresco cola	1	Mediano
4	Daniel	56983640	Benito Carranza 3 Col. Del Valle	Pizza Carnes frías	1	Grande
5	Genaro	34298561	Av. Universidad 1080 Col. Del valle	pizza jamón	4	Grande
5	Genaro	34298561	Av. Universidad 1080 Col. Del valle	pizza carnes frías	4	Grande
5	Genaro	34298561	Av. Universidad 1080 Col. Del valle	refresco cola	5	Grande
6	Sara	90277506	Tuxpan 67 Col. Roma	pizza 4 quesos	3	Mediana

Análogamente podemos descomponer el atributo Dirección en los atributos Calle, Número y Colonia, con lo que obtendríamos un nuevo esquema presentado en la Tabla 10.4.

Tabla 10.4 - Segundo arreglo para lograr la primera Forma Normal de la tabla de órdenes de la Pizzería.

No. Orden	Cliente	Teléfono	Dirección			Orden		
			Calle	Número	Colonia	Artículo	Cantidad	Tamaño
1	Pedro	55894320	Calle 1	20	Condesa	Pizza hawaiana	1	Mediana
1	Pedro	55894320	Calle 1	20	Condesa	Pizza Carnes Frías	1	Grande
1	Pedro	55894320	Calle 1	20	Condesa	Refresco Limón	2	Grande
2	Edna	59131677	Venustiano Juárez	83	Polanco	Pizza Jamón	2	Mediana
2	Edna	59131677	Venustiano Juárez	83	Polanco	Jugo Mango	1	Mediano
3	Carlos	22267457	Av. Baja California	545	Roma	Pizza 4 quesos	1	Chica
3	Carlos	22267457	Av. Baja California	545	Roma	orden papas	1	Grande
3	Carlos	22267457	Av. Baja California	545	Roma	refresco cola	1	Mediano
4	Daniel	56983640	Benito Carranza	3	Del Valle	Pizza Carnes frías	1	Grande
5	Genaro	34298561	Av. Universidad	1080	Del Valle	pizza jamón	4	Grande
5	Genaro	34298561	Av. Universidad	1080	Del Valle	pizza carnes frías	4	Grande
5	Genaro	34298561	Av. Universidad	1080	Del Valle	refresco cola	5	Grande
6	Sara	90277506	Tuxpan	67	Roma	pizza 4 quesos	3	Mediana

Podemos apreciar que la Tabla 10.4 se encuentra en la primera forma normal ya que no existe ninguna columna que contenga más de un valor por registro. La primera forma normal incrementa la probabilidad de explotación de los datos de una base, pero también, incrementa la redundancia de los mismos, es decir, encontramos múltiples registros que almacenan los mismos datos más de una vez, ocupando grandes cantidades de espacio.

Para solucionar esta nueva problemática se debe normalizar la tabla a la segunda forma normal.

## 2. Segunda Forma Normal:

**Todo atributo no llave depende del atributo llave.** Las dependencias funcionales como se puede intuir por su nombre, son una especie de función que recibe un valor de una columna y le asigna un solo valor de otra columna. La notación  $X \rightarrow Y$  indica que el valor de X determina funcionalmente al valor de Y. Es por esto que la segunda forma normal quiere decir, que todo valor en un registro que no sea llave primaria deberá estar determinado por el valor de la llave primaria.

Con ésta información se pueden apreciar las dependencias funcionales existentes. Ver Tabla 10.5.

Tabla 10.5. Dependencias Funcionales de la Tabla 10.4.

<i>No. Orden</i>	→	<i>Cliente</i>
<i>Cliente</i>	→	<i>Teléfono</i>
<i>Cliente</i>	→	<i>Dirección</i>

La Columna *No. De Orden* que es la llave primaria, determina el valor de la columna *Cliente*, pero el valor de la columna *Cliente* determina tanto al valor de la columna *Teléfono* y la columna *Dirección*. Por lo tanto no se encuentra en segunda forma normal ya que las columnas *Teléfono* y *Cliente* no dependen del valor de la llave primaria (*No. De Orden*), sino del valor de la columna *Cliente*.

Una alternativa para corregir éste conflicto es construir otra tabla para separar la información que se encuentra determinada funcionalmente por un atributo que no es llave primaria. En éste caso conviene separar la información de la *Orden* de la información del *Cliente*, las Tablas 10.6 y 10.7 muestran ambas tablas con la información ya dividida.

Tabla 10.6 Tabla R1 obtenida después de la descomposición.

<b><i>No. Orden</i></b>	<b><i>Cliente</i></b>	<b><i>Artículo</i></b>	<b><i>Cantidad</i></b>	<b><i>Tamaño</i></b>
1	Pedro	Pizza hawaiana	1	Mediana
1	Pedro	Pizza Carnes Frías	1	Grande
1	Pedro	Refresco Limón	2	Grande
2	Edna	Pizza Jamón	2	Mediana
2	Edna	Jugo Mango	1	Mediano
3	Carlos	Pizza 4 quesos	1	Chica
3	Carlos	orden papas	1	Grande
3	Carlos	refresco cola	1	Mediano
4	Daniel	Pizza Carnes frías	1	Grande
5	Genaro	pizza jamón	4	Grande
5	Genaro	pizza carnes frías	4	Grande
5	Genaro	refresco cola	5	Grande
6	Sara	pizza 4 quesos	3	Mediana

Tabla 10.7 - Tabla R2 obtenida después de la descomposición.

<i>Cliente</i>	<i>Teléfono</i>	<i>Calle</i>	<i>Número</i>	<i>Colonia</i>
Pedro	55894320	Calle 1	20	Condesa
Edna	59131677	Venustiano Juárez	83	Polanco
Carlos	22267457	Av. Baja California	545	Roma
Daniel	56983640	Benito Carranza	3	Del Valle
Genaro	34298561	Av. Universidad	1080	Del Valle
Sara	90277506	Tuxpan	67	Roma

A pesar de que ambas tablas han sido divididas, se aprecia que la Tabla 10.6 que contiene los detalles de la orden continúa almacenando registros redundantes y posiblemente inconsistentes ocupando espacio de almacenamiento, por ejemplo el valor de la columna *Cliente*. Además, la Tabla 10.7 se encuentra violando la definición de llave primaria, ya que la columna *No. Orden* no contiene valores únicos.

En este caso se deberá dividir nuevamente ésta tabla para lograr que todos los atributos no llave dependan de la llave de cada tabla en particular. Las Tablas 10.8, 10.9 y 10.10 muestran un arreglo para solucionar éste problema.

Tabla 10.8 - Tabla R1-A obtenida después de la descomposición de R1.

<u><i>No. Orden</i></u>	<u><i>ID Artículo</i></u>	<i>Cantidad</i>
1	1	1
1	2	1
1	7	2
2	5	2
2	8	1
3	3	1
3	11	1
3	9	1
4	2	1
5	6	4
5	2	4
5	10	5
6	4	3

Tabla 10.9 - Tabla R1-B obtenida después de la descomposición de R1.

<u>ID Artículo</u>	Artículo	Tamaño
1	Pizza Hawaiana	Mediana
2	Pizza Carnes Frías	Grande
3	Pizza 4 quesos	Chica
4	Pizza 4 quesos	Mediana
5	Pizza Jamón	Mediana
6	Pizza Jamón	Grande
7	Refresco Limón	Grande
8	Jugo Mango	Mediano
9	Refresco Cola	Mediano
10	Refresco Cola	Grande
11	Orden Papas	Grande

Tabla 10.10 - Tabla R1-C obtenida después de la descomposición de R1.

<u>No. Orden</u>	Cliente
1	Pedro
2	Edna
3	Carlos
4	Daniel
5	Genaro
6	Sara

En las tablas anteriores se realizaron dos divisiones, la primera división fue separar la Orden, Tabla 10.8, y el Cliente, Tabla 10.10, al que pertenece, tomando como llave primaria la columna *No. Orden*. La segunda división fue separar los datos de los Artículos, Tabla 10.9, es decir, la descripción del producto y los posibles tamaños de cada uno, identificándolos de manera única por la columna *ID\_Artículo*, que se seleccionó como llave primaria de ésta tabla.

De manera que la tabla del detalle de la orden quedó con las columnas *No. Orden*, *ID\_Artículo* y *Cantidad*, seleccionando como llave primaria las columnas *No. Orden* y *ID\_Artículo*, ya que de ésta manera se puede identificar de manera única un registro de la tabla.

### 3. Tercera Forma Normal:

**No existen dependencias transitivas entre atributos no llave.** Esta forma normal se asegura de eliminar las redundancias mediante el correcto manejo de las llaves primarias. Se puede apreciar que la tabla que contiene los datos de los clientes,

Tabla 10.7, presenta las siguientes dependencias funcionales que se muestran en la a continuación. Ver Tabla 10.11.

Tabla 10.11 - Dependencias Funcionales de la tabla *Cientes* en segunda forma normal.

<i>Cliente</i>	→	<i>Teléfono</i>
<i>Cliente</i>	→	<i>Dirección (Calle, Número y Colonia)</i>
<i>Teléfono</i>	→	<i>Dirección (Calle, Número y Colonia)</i>

Si suponemos que la tabla tiene por llave primaria a la columna *Cliente*, entonces *Cliente* determina a la columna *Teléfono*, pero también *Teléfono* determina a *Dirección*.

La existencia de la transitividad entre las dependencias funcionales que se muestran en la Tabla 10.11 limita a la base en general de alcanzar una mayor eficiencia. Para resolver esto se debe continuar descomponiendo la tabla en diferentes tablas que sigan cumpliendo las formas normales. Por esta razón se descompondrá la tabla con los datos de los clientes. Las Tablas 10.12 y 10.13 muestran un arreglo para ello.

Tabla 10.12 - Tabla R2-A obtenida después de la descomposición de R2.

<i>Cliente</i>	<i>Teléfono</i>
Pedro	55894320
Edna	59131677
Carlos	22267457
Daniel	56983640
Genaro	34298561
Sara	90277506

Tabla 10.13 - Tabla R2-B obtenida después de la descomposición de R2.

<i>Cliente</i>	<i>Calle</i>	<i>Número</i>	<i>Colonia</i>
Pedro	Calle 1	20	Condesa
Edna	Venustiano Juárez	83	Polanco
Carlos	Av. Baja California	545	Roma
Daniel	Benito Carranza	3	Del Valle
Genaro	Av. Universidad	1080	Del Valle
Sara	Tuxpan	67	Roma

La organización que se ha alcanzado con la tercera forma normal es mucho más eficiente para su explotación y almacenamiento que la original. Aun así, se debe de alcanzar la forma normal más alta, dependiendo del contexto del problema, para obtener un mejor desempeño de la base de datos que se pretende normalizar.



4. Forma Normal Boyce-Codd:

**Todo determinante debe ser llave.** Las dependencias funcionales se componen de dos partes, el determinante y el determinado, siendo el determinante el valor del registro que puede identificar de manera única a otro valor del mismo registro (determinado). Por lo tanto en la dependencia funcional  $X \rightarrow Y$ , podemos decir que el determinante es X mientras que el determinado es Y.

A continuación se presentan las tablas finales con las llaves que se seleccionaron. Ver Tablas 10.14, 10.15, 10.16, 10.17 y 10.18.

Tabla 10.14 - Tabla Cliente\_Teléfono normalizada hasta la Forma Normal de Boyce-Codd.

<u>Cliente</u>	<u>Teléfono</u>
Pedro	55894320
Edna	59131677
Carlos	22267457
Daniel	56983640
Genaro	34298561
Sara	90277506

Tabla 10.15 - Tabla Cliente normalizada hasta la Forma Normal de Boyce-Codd.

<u>Cliente</u>	<u>Calle</u>	<u>Número</u>	<u>Colonia</u>
Pedro	Calle 1	20	Condesa
Edna	Venustiano Juárez	83	Polanco
Carlos	Av. Baja California	545	Roma
Daniel	Benito Carranza	3	Del Valle
Genaro	Av. Universidad	1080	Del Valle
Sara	Tuxpan	67	Roma

Tabla 10.16 - Tabla Orden\_Cliente normalizada hasta la Forma Normal de Boyce-Codd.

<u>No. Orden</u>	<u>Cliente</u>
1	Pedro
2	Edna
3	Carlos
4	Daniel
5	Genaro
6	Sara

Tabla 10.17 - Tabla Detalle\_Orden normalizada hasta la Forma Normal de Boyce-Codd.

<u>No. Orden</u>	<u>ID Artículo</u>	<u>Cantidad</u>
1	1	1
1	2	1
1	7	2
2	5	2
2	8	1
3	3	1
3	11	1
3	9	1
4	2	1
5	6	4
5	2	4
5	10	5
6	4	3

Tabla 10.17 - Tabla Artículo normalizada hasta la Forma Normal de Boyce-Codd.

<u>ID Artículo</u>	<u>Artículo</u>	<u>Tamaño</u>
1	Pizza Hawaiana	Mediana
2	Pizza Carnes Frías	Grande
3	Pizza 4 quesos	Chica
4	Pizza 4 quesos	Mediana
5	Pizza Jamón	Mediana
6	Pizza Jamón	Grande
7	Refresco Limón	Grande
8	Jugo Mango	Mediano
9	Refresco Cola	Mediano
10	Refresco Cola	Grande
11	Orden Papas	Grande

Para fines de este manual de prácticas, se considerará que la base de datos tiene un buen diseño si cada una de las tablas que la conforman se encuentran en la Forma Normal Boyce – Codd.

#### 4. Ejercicios

*(NOTA: Resuelve los siguientes ejercicios en relación al proyecto que realizarás durante el curso, en dado caso que no tengas un proyecto, utiliza la información en el apéndice NFL-ONEFA parte 10 al final de esta práctica para realizarlos)*

1. Realiza un análisis a tus tablas y determina la **Forma Normal** en la que se encuentra cada una. Justifica tu respuesta mediante las dependencias funcionales de cada una.
2. Normaliza cada tabla hasta llevarla a la forma normal de Boyce-Codd y mediante el uso de las **Dependencias Funcionales identificadas** justifica la forma normal de cada una de las tablas de la base de datos.

#### Entregables requeridos para prácticas subsecuentes:

- Forma Normal de cada una de las tablas de la base de datos
- Dependencias Funcionales identificadas

## 5. Apéndice NFL-ONEFA parte 10

Siguiendo los supuestos que se encuentran en el Apéndice NFL-ONEFA parte 01, encuentra todas las dependencias funcionales. Utiliza éste resultado para justificar en que Forma Normal se encuentra cada tabla y por tanto cuál es la forma normal máxima que alcanza la base de datos en total.

A continuación se presentan las dependencias funcionales identificadas en NFL-ONEFA:

### Jugador

$id\_jugador \rightarrow Jugador$   
 $fecha\_nacimiento \rightarrow categoría$

### Receiver

$\{rec, yds\} \rightarrow avg$

### Runner

$\{carries, yds\} \rightarrow avg$

### Quarterback

$\{tds, ints, yds, att, comp\} \rightarrow rating$

### Punter

$\{net, punts\} \rightarrow avg$

### Kicker

$\{fga, fgm\} \rightarrow pctg$

### Runner

$\{carries, yds\} \rightarrow avg$

### Runner

$\{carries, yds\} \rightarrow avg$

### Equipo

$id\_equipo \rightarrow Equipo$

### Entrenador

$id\_entrenador \rightarrow Entrenador$

### Estadio

$id\_estadio \rightarrow Estadio$

### Entrenador

$id\_entrenador \rightarrow Entrenador$

### Partido

$id\_partido \rightarrow Partido$

### jugador\_equipo

$\{id\_equipo, id\_jugador, anyo\_adquirido\} \rightarrow jugador\_equipo$

### equipo\_ciudad

$\{id\_equipo, id\_ciudad\} \rightarrow equipo\_ciudad$

### equipo\_estadio

$\{id\_equipo, id\_estadio\} \rightarrow equipo\_estadio$