

## Ejercicios 1

Para explicar con un ejemplo en qué consiste el proceso de Normalización y cada una de las reglas, vamos a considerar el siguiente esquema de datos (orden).

**orden** (id\_orden, fecha, id\_cliente, nom\_cliente, ciudad, num\_art, nom\_art, cant, precio)

orden

id\_orden fecha id\_cliente nom\_cliente ciudad num\_art nom\_art cant, precio

### FORMAS NORMALES

#### Orden

Id_orden	Fecha	Id_cliente	Nom_cliente	Ciudad	Num_art	nom_art	cant	Precio
2301	23/02/11	101	Martin	Montevideo	3786	Red	3	35,00
2301	23/02/11	101	Martin	Montevideo	4011	Raqueta	6	65,00
2301	23/02/11	101	Martin	Montevideo	9132	Paq-3	8	4,75
2302	25/02/11	107	Herman	Maldonado	5794	Paq-6	4	5,00
2303	27/02/11	110	Pedro	Rocha	4011	Raqueta	2	65,00
2303	27/02/11	110	Pedro	Rocha	3141	Funda	2	10,00

#### PRIMERA FORMA NORMAL (1FN)

Si observamos los registros de la tabla Orden, podemos darnos cuenta, por ejemplo, que contienen un grupo de datos que se repite. Si observas los 3 renglones correspondientes a la orden 2301 para cada NUM\_ART, NOM\_ART, CANT y PRECIO se repite la misma información en ID\_ORDEN, FECHA, ID\_CLIENTE, NOM\_CLIENTE, CIUDAD.

Podríamos decir que esto es lo que ocurre cuando estamos en primera forma normal, existe redundancia de información.

Dicho de otra forma, analizando la relación existente entre los datos, (dependencias funcionales) observamos lo siguiente:

Para un id\_orden existe un único correspondiente en fecha, id\_cliente, nom\_cliente y ciudad.

Para un num\_art existe un único correspondiente en nom\_art y en el precio del artículo.

Para una pareja de datos (id\_orden, num\_art) existe un único correspondiente en cant.

A este tipo de relación que existe entre los datos (atributos) le llamamos dependencia funcional.

En este sentido en el esquema que estamos analizando podríamos decir que se cumplen las siguientes dependencias funcionales

**id\_orden** → fecha, id\_cliente, nom\_cliente, ciudad

**id\_cliente** → nom\_cliente, ciudad

**num\_art** → nom\_art, precio

**id\_orden, num\_art** → cant,

si recordamos la definición de **clave**, como conjunto mínimo de atributos que nos permiten identificar funcionalmente a todos los atributos del esquema (ver reglas de Armstrong) la clave en este esquema seria la siguiente:

**id\_orden, id\_cliente, num\_art** → id\_orden, fecha, id\_cliente, nom\_cliente, ciudad, num\_art, nom\_art, cant, precio

Para eliminar los grupos repetidos vamos a quitar del esquema las dependencias funcionales que cumplan que su parte izquierda sea una parte o este contenida en la parte izquierda de la clave transformándolas en una tabla.

En este sentido tomaremos la dependencia (**id\_orden** → fecha, id\_cliente, nom\_cliente, ciudad).

Observemos que su parte izquierda, id\_orden, es parte o esta contenida en la parte izquierda de la clave.

Por lo tanto, debemos sacar esa dependencia transformándola en una tabla.

El vinculo que mantendremos entre la tabla original y la nueva tabla es la parte izquierda de la dependencia que estamos quitando. Este proceso lo debemos hacer con cada dependencia que cumpla con estas condiciones. A este proceso lo llamaremos pasaje o transformación de primera forma normal a segunda forma normal

Los pasos a seguir son:

- Tenemos que eliminar los grupos repetidos.
- Tenemos que crear una nueva tabla con la PK de la tabla base y el grupo repetido.

Aplicando lo antes dicho, los registros de la tabla original quedan ahora en dos tablas que llamaremos: ORDEN y ARTICULO\_ORDEN

*orden* (id\_orden, fecha, id\_cliente, nom\_cliente, Ciudad)

*Articulo\_orden* (id\_orden, num\_art, nom\_art, cant, precio)

#### Orden

<b>Id_orden</b>	<b>Fecha</b>	<b>Id_cliente</b>	<b>Nom_cliente</b>	<b>Ciudad</b>
2301	23/02/11	101	Martin	Montevideo
2302	25/02/11	107	Herman	Maldonado
2303	27/02/11	110	Pedro	Rocha

#### Articulo\_orden

<b>Id_orden</b>	<b>Num_art</b>	<b>nom_art</b>	<b>cant</b>	<b>Precio</b>
2301	3786	Red	3	35,00
2301	4011	Raqueta	6	65,00
2301	9132	Paq-3	8	4,75
2302	5794	Paq-6	4	5,00
2303	4011	Raqueta	2	65,00
2303	3141	Funda	2	10,00

Si observamos ahora, veremos que la tabla orden tiene como clave a:

id\_orden →, fecha, id\_cliente, nom\_cliente, Ciudad

y el resto de las dependencias de esta tabla no cumplen que su parte izquierda este contenida (no sea una parte de) en la parte izquierda de la clave. Esto hace que podamos decir que la tabla Orden estaría en segunda forma normal. Lo mismo podemos observar en la tabla Articulo\_Orden con la clave

Id\_orden, Num\_art → Id\_Orden, Num\_art, nom\_art, cant, precio.

Aquí tampoco existen dependencias que cumplan esas condiciones por lo que también podemos decir que se encuentra en segunda forma normal.

## SEGUNDA FORMAL NORMAL (2FN)

Ahora procederemos a analizar estando en la segunda formal normal.

En este caso podríamos observar que en la tabla **Articulo\_orden** que tiene como clave Id\_orden, Num\_art  $\rightarrow$  Id\_Orden, Num\_art, nom\_art, cant, precio y una dependencia funcional como la siguiente Num\_art  $\rightarrow$  Nom\_art.

Cuando tenemos esta situación, en la que la parte izquierda de la dependencia no esta contenida en la parte izquierda de la clave en este caso particular decimos que se cumple una transitividad ya que con el id\_orden puedo obtener el Num\_art y con el Num\_art puedo obtener el nom\_art.

Para seguir depurando nuestro esquema ahora tenemos que eliminar cualquier columna no clave que no dependa de la clave primaria de la tabla.

Para el ejemplo nom\_art no depende de la clave primaria de la tabla, depende de Num\_art y al igual que el precio.

Los pasos a seguir son:

- Determinar cuáles columnas que no son llave no dependen de la llave primaria de la tabla.
- Eliminar esas columnas de la tabla base.
- Crear una segunda tabla con esas columnas y la(s) columna(s) de la PK de la cual dependen.

La tabla ORDEN está en 2FN. Cualquier valor único de ID\_ORDEN determina un sólo valor para cada columna. Por lo tanto, todas las columnas son dependientes de la llave primaria ID\_ORDEN.

Por su parte, la tabla ARTICULOS\_ORDENES tiene las columnas PRECIO y NOM\_ART que son dependientes de NUM\_ART, pero no son dependientes de ID\_ORDEN. Lo que haremos a continuación es eliminar estas columnas de la tabla ARTICULOS\_ORDENES y crear una tabla ARTICULOS con dichas columnas y la llave primaria de la que dependen.

Para ello aplicamos el mismo procedimiento que aplicamos en el primer caso cuando particionamos la tabla Orden. Eliminamos la dependencia que no cumple transformando esta en una tabla y manteniendo la parte izquierda como vinculo en la nueva partición.

Aplicando lo antes dicho entonces, las tablas quedan ahora de la siguiente manera:

*Articulos\_ordenes* (id\_orden, num\_art, cant) donde id\_orden, num\_art  $\rightarrow$  cant

*Articulos* ( num\_art, nom\_art, precio) donde num\_art  $\rightarrow$  nom\_art, precio

En este sentido entonces la table original que estamos particionano (**Articulos\_ordenes**) se descompone en 2 tablas manteniendo la parte izquierda de la dependencia que estamos quitando transformandola en table como vinculo entre ambas tablas

Las tablas entonces quedarian de la siguiente forma\_

*Articulos\_ordenes* (id\_orden, num\_art, cant)

**Articulos\_ordenes**

<b>Id_orden</b>	<b>Num_art</b>	<b>cant</b>
2301	3786	3
2301	4011	6
2301	9132	8
2302	5794	4
2303	4011	2
2303	3141	2

*Articulos* ( num\_art, nom\_art, precio)

**Articulos**

<b>Num_art</b>	<b>nom_art</b>	<b>Precio</b>
3786	Red	35,00
4011	Raqueta	65,00
9132	Paq-3	4,75
5794	Paq-6	5,00
3141	Funda	10,00

### **TERCERA FORMAL NORMAL (3FN)**

La tercera forma normal nos dice que tenemos que eliminar cualquier columna no llave que sea dependiente de otra columna no llave. Los pasos a seguir son:

- Determinar las columnas que son dependientes de otra columna no llave.
- Eliminar esas columnas de la tabla base.
- Crear una segunda tabla con esas columnas y con la columna no llave de la cual son dependientes.

Al observar las tablas que hemos creado, nos damos cuenta que tanto la tabla ARTICULOS, como la tabla ARTICULOS\_ORDENES se encuentran en 3FN. Sin embargo, la tabla ORDENES no lo está, ya que NOM\_CLIENTE y CIUDAD son dependientes de ID\_CLIENTE, y esta columna no es la llave primaria.

Para normalizar esta tabla, moveremos las columnas no llave y la columna llave de la cual dependen dentro de una nueva tabla CLIENTES. Las nuevas tablas CLIENTES y ORDENES se muestran a continuación.

*ordenes* (id\_orden, fecha, id\_cliente)

#### **Ordenes**

<b>Id_orden</b>	<b>Fecha</b>	<b>Id_cliente</b>
2301	23/02/11	101
2302	25/02/11	107
2303	27/02/11	110

*Clientes* (id\_cliente, nom\_cliente, Ciudad)

#### **Ordenes**

<b>Id_cliente</b>	<b>Nom_cliente</b>	<b>Ciudad</b>
101	Martin	Montevideo
107	Herman	Maldonado
110	Pedro	Rocha

**Por lo tanto la base de datos queda de la siguiente manera:**

*ordenes* (id\_orden, fecha, id\_cliente)

*Clientes* (id\_cliente, nom\_cliente, Ciudad)

*Articulos* ( num\_art, nom\_art, precio)

*Articulos\_ordenes* (id\_orden, num\_art, cant)

**En resumen**

**La tabla original orden** (id\_orden, fecha, id\_cliente, nom\_cliente, ciudad, num\_art, nom\_art, cant, precio) se descompone en una primera instancia en:

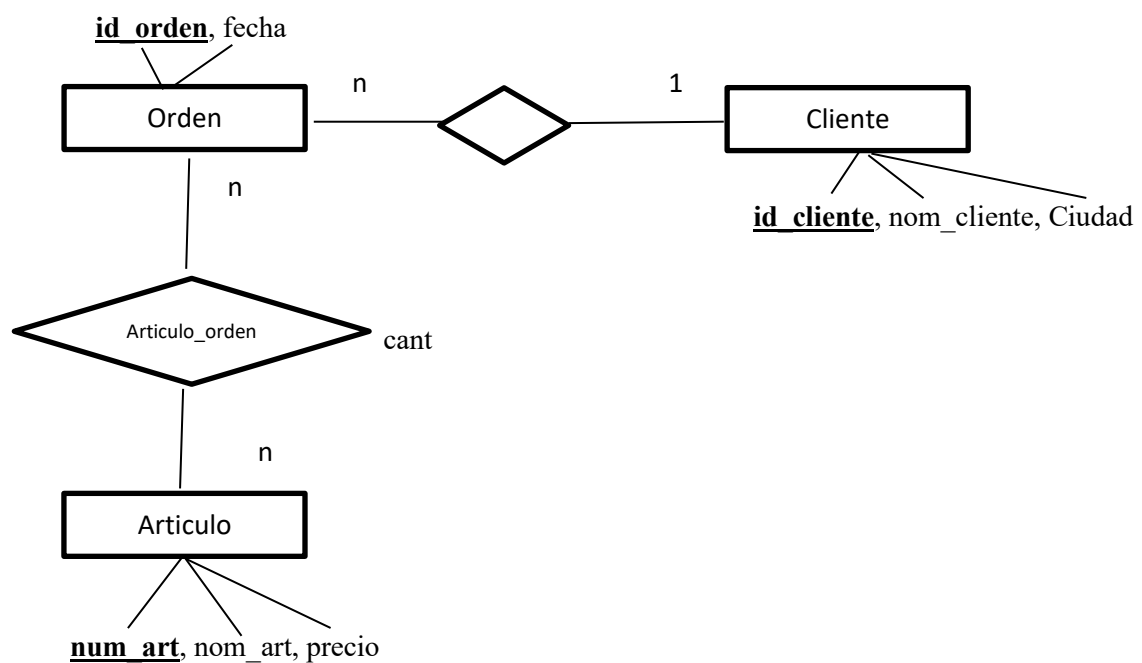
- **orden** (id\_orden, fecha, id\_cliente, nom\_cliente, Ciudad) se descompone en
- **Articulo\_orden** (id\_orden, num\_art, nom\_art, cant, precio)

Luego cada una de esas tablas resultantes se descomponen como se ve a continuación:

1. **orden** (id\_orden, fecha, id\_cliente, nom\_cliente, Ciudad) se descompne en
  - a. **orden** (id\_orden, fecha, id\_cliente)
  - b. **Clientes** (id\_cliente, nom\_cliente, Ciudad)
2. **Articulo\_orden** (id\_orden, num\_art, nom\_art, cant, precio)
  - a. **Articulos\_ordenes** (id\_orden, num\_art, cant)
  - b. **Articulos** (num\_art, nom\_art, precio)

**Quedando como resultado final una descomposición en 3 forma normal**

**orden** (id\_orden, fecha, id\_cliente)  
**Cliente** (id\_cliente, nom\_cliente, Ciudad)  
**Articulo** (num\_art, nom\_art, precio)  
**Articulo\_orden** (id\_orden, num\_art, cant)



## Ejercicios 2

**FACTURA DE COMPRA VENTA:** La empresa COLOMBIAN SYSTEMS lo ha contratado como el “Ingeniero Encargado” para sistematizar la facturación. En la siguiente FACTURA DE COMPRA VENTA, usted debe analizar toda la información disponible y aplique el proceso de normalización, hasta llegar a la Tercera Forma Normal.

Se pide realizar la respectiva justificación detallada de cada uno de los pasos que conduzcan al resultado final.

**Factura**(NUM\_FAC, FECHA\_FAC, NOM\_CLIENTE, DIR\_CLIENTE, RIF\_CLIENTE, CIUDAD\_CLIENTE, TELEF\_CLIENTE, CATEGORIA, COD\_PROD, DESP\_PROD, VAL\_UNIT, CANT\_PROD)

**Donde:**

**NUM\_FAC:** Número de la factura de compra venta

**FECHA\_FAC:** Fecha de la factura de compra venta

**NOM\_CLIENTE:** Nombre del cliente

**DIR\_CLIENTE:** Dirección del cliente

**RIF\_CLIENTE:** Rif del cliente

**CIUDAD\_CLIENTE:** Ciudad del cliente

**TELEF\_CLIENTE:** Teléfono del cliente

**CATEGORIA:** Categoría del producto

**COD\_PROD:** Código del producto

**DESCRIPCION:** Descripción del producto

**VAL\_UNIT:** Valor unitario del producto

**CANT\_PROD:** Cantidad de productos q compra el cliente

La llave primaria es Número de Factura de venta: NUM\_FAC

## Ejercicios 3

**EMPRESA DE ENVIO DE MERCANCIA:** a continuación se agrupan todos los atributos que hacen parte de la base de datos para aplicarle las reglas de normalización. Donde se incluyen los nombres de los atributos con su significado

- \* GUIA\_NO = Numero de Guia
- \* GUIA\_FECHA= Fecha de la Guia
- \* GUIA\_HORA= Hora de la Guia
- \* ORGN\_RIF = Identificacion de Empresa Origen
- \* ORGN\_NOM = Nombre de Empresa Origen
- \* ORGN\_ACT = Actividad Comercial de Empresa Origen
- \* ORGN\_CIUADAD= Ciudad de Empresa Origen
- \* ORGN\_DIR = Direccion de Empresa Origen
- \* ORGN\_TEL = Telefono de Empresa Origen
- \* ORGN\_CEL = Celular de Empresa Origen
- \* DEST\_ID = Identificacion del destinatario
- \* DEST\_NOM = Nombre del destinatario
- \* DEST\_COD\_CIUADAD =Codigo de la ciudad del destinatario
- \* DEST\_CIUADAD= Ciudad del destinatario
- \* DEST\_DIR = Direccion del destinatario
- \* DEST\_TEL = Telefono del destinatario
- \* DEST\_KM = Distancia kilometraje de Ciudad origen a ciudad del destinatario
- \* CODIGO = Codigo del paquete
- \* TIPO = Tipo de paquete
- \* NOMBRE = Nombre del paquete
- \* DESCRIPCION = Descripción del paquete
- \* VALR\_FLETE = Valor del flete



## Ejercicios 4

**Video club:** En una tienda de video se necesita mantener información de alrededor de 3000 casetas cada uno de los casetes tiene asignado un número por cada película se necesita conocer un título y categoría por ejemplo: comedia, suspenso, drama, acción, ciencia ficción, etc. Se mantienen algunas copias de muchas películas. Se le da a cada película una identificación y se mantiene seguimiento de lo que contiene cada casete. Un casete puede venir en varios formatos y una película es grabada en un solo casete; frecuentemente las películas son pedidas de acuerdo a un actor específico Tom Cruise y Demi More son los más populares es por esto que se debe mantener información de los actores que pertenecen a cada película.

No en todas las películas actúan artistas famosos, a los clientes de la tienda le gusta conocer datos como el nombre real del actor, y su fecha de nacimiento.

En la tienda se mantienen información solo de los actores que aparecen en las películas y que se tiene a disposición. Solo se alquila videos a aquellos que pertenecen al club de videos. Para pertenecer al club se debe tener un buen crédito. Por cada miembro del club se mantiene una ficha con su nombre, teléfono y dirección, cada miembro del club tiene asignado un número de membresía. Se desea mantener información de todos los casetes que un cliente alquila, cuando un cliente alquila un casete se debería conocer el nombre de la película, la fecha en la que se alquila y la fecha de devolución.

Se pide aplicar las reglas de normalización hasta la tercera forma normal, teniendo las siguientes entidades con sus respectivos atributos:

**Alquiler** (cod\_alquiler, num\_membresia, cod\_cliente, nom\_cliente, dir\_cliente, telef\_cliente, cod\_cassette, fecha\_alquiler, fecha\_dev, valor\_alquiler, cantidad)

**Cassette** (cod\_cassette, num\_copias, formato, cod\_pelicula, titulo, categoría, cod\_actor, nom\_actor, fechanac\_actor, cod\_tipo)

### Donde:

cod\_alquiler = Código del alquiler  
 num\_membresia = Numero de membresia  
 cod\_cliente = código del cliente  
 nom\_cliente = nombre del cliente  
 dir\_cliente = dirección del cliente  
 telef\_cliente = teléfono del cliente  
 cod\_cassette = código del cassette  
 fecha\_alquiler = fecha del alquiler de la película  
 fecha\_dev = fecha de devolución de la película  
 valor\_alquiler = valor del alquiler de la película  
 cantidad = cantidad de película alquilada  
 num\_copias = números de copias de cassette  
 formato = formato del cassette  
 titulo = nombre de la película  
 categoría = categoría de la película  
 cod\_actor = código del actor  
 nom\_actor = nombre del actor  
 fechanac\_actor = fecha de nacimiento del actor  
 cod\_tipo = código del tipo de película.

## Ejercicios 5

Dada la siguiente relación PRESTAMO\_LIBROS (Colegio, profesor, asignatura/ habilidad, aula, curso, libro, editorial, fecha\_prestamo) que contiene información relativa a los préstamos que realizan las editoriales a los profesores de primaria de los colegios para su evaluación en alguna de las asignaturas/habilidades que imparten. Se pide aplicar las reglas de normalización y obtener su modelo relacional, indicar sus claves, atributos principales.

Colegio	Profesor	Asignatura/ habilidad	Aula	Curso	Libro	Editorial	Fecha_prestamo
C.P Cervantes	Juan Pérez	Pensamiento Lógico	1.A01	1er Grado	Aprender y enseñar en educación infantil	Graó	09/09/2010
C.P Cervantes	Juan Pérez	Escritura	1.A01	1er Grado	Preescolar Rubio,N56	Técnicas Rubio	05/05/2010
C.P Cervantes	Juan Pérez	Pensamiento Numérico	1.A01	1er Grado	Aprender y Enseñar en educación infantil	Graó	05/05/2010
C.P Cervantes	Alicia García	Pensamiento Espacial, Temporal y causal	1.B01	1er Grado	Educación Infantil N9	Prentice Hall	06/05/2010
C.P Cervantes	Alicia García	Pensamiento Numérico	1.B01	1er Grado	Aprender y enseñar en educación infantil	Graó	06/05/2010
C.P Cervantes	Andrés Fernández	Escritura	1.A01	2do Grado	Aprender y enseñar en educación infantil	Graó	09/09/2010
C.P Cervantes	Andrés Fernández	Ingles	1.A01	2do Grado	Saber educar: guía para Padres y Profesores	Temas de Hoy	05/05/2010
C.P Quevedo	Juan Méndez	Pensamiento Lógico	2.B01	1er Grado	Saber educar: guía para Padres y Profesores	Temas de Hoy	18/12/2010
C.P Quevedo	Juan Méndez	Pensamiento Numérico	2.B01	1er Grado	Aprender y enseñar en educación infantil	Graó	06/05/2010

## Ejercicios 6

Se tiene una relación del REPORTE\_MATRICULA (código\_alumno, nombre\_alumno, especialidad, código\_curso, nombre\_curso, nombre\_docente, oficina, sección) se pide aplicar las reglas de normalización llegando hasta las 3FN.

Código/ alumno	Nombre/ alumno	Especialidad	Código/ curso	Nombre_curso	Nombre/ docente	Oficina	curso
382145A	Luis Zuloaga	Industrial	MA123	Matemática 2	Carlos Arambulo	CB-214	U
382145A	Luis Zuloaga	Industrial	QU514	Física Química	Petra Rondinel	CB-110	U
382145A	Luis Zuloaga	Industrial	AU521	Descriptiva	Víctor Moncada	CB-120	W
360247k	Raúl Rojas	Sistemas	PA714	Investigación 1	Cesar Fernandez	SC-220	V
360247k	Raúl Rojas	Sistemas	MA123	Matemática 2	Carlos Arambulo	CB-214	V
360247k	Raúl Rojas	Sistemas	AU511	Dibujo	Víctor Moncada	CB-120	U

## Ejercicios 7

Se presenta una base de datos de una biblioteca, aplicar las reglas de normalización simplificando hasta la tercera forma normal.

**Prestamos\_libro** (codLibro, Titulo, Autor, Editorial, nombreLector, Fechadev)

codLibro	Titulo	Autor	Editorial	nombreLector	Fechadev
1001	Variable compleja	Murray Spiegel	McGraw Hill	Pérez Gómez, Juan	15/04/2005
1004	Visual Basic 5	E. Petroustos	Anaya	Ríos Terán, Ana	17/04/2005
1005	Estadística	Murray Spiegel	McGraw Hill	Roca, René	16/04/2005
1006	Oracle University	Nancy Greenberg y Priya Nathan	Oracle Corp.	García Roque, Luis	20/04/2005
1007	Clipper 5.01	Ramalho	McGraw Hill	Pérez Gómez, Juan	18/04/2005