

## TP3 – Normalización

Materia: Base de Datos I

Profesor: Gustavo Ramoscelli

Ayudantes: María Victoria Ruiz – Fernando Damián Ene

Grupo: 15

## Ejercicio 2 – Estadías

### Paso 1: Dependencias funcionales

$\text{cod\_hotel} \rightarrow \text{cantidad\_habitaciones}, \text{direccion\_hotel}, \text{ciudad\_hotel}, \text{dni\_gerente}$

$\text{dni\_gerente} \rightarrow \text{nombre\_gerente}$

$\text{dni\_cliente} \rightarrow \text{nombre\_cliente}, \text{ciudad\_cliente}$

$(\text{dni\_cliente}, \text{cod\_hotel}, \text{habitacion}, \text{fecha\_inicio\_hospedaje}) \rightarrow \text{cant\_dias\_hospedaje}$

$\text{cod\_hotel}, \text{habitacion} \rightarrow \text{identifica una habitación dentro de un hotel.}$

### Paso 2: Atributos primos y no primos

Atributos primos:  $\text{dni\_cliente}, \text{cod\_hotel}, \text{habitacion}, \text{fecha\_inicio\_hospedaje}$

Atributos no primos:  $\text{cantidad\_habitaciones}, \text{direccion\_hotel}, \text{ciudad\_hotel}, \text{dni\_gerente}, \text{nombre\_gerente}, \text{nombre\_cliente}, \text{ciudad\_cliente}, \text{cant\_dias\_hospedaje}$ .

### Paso 3: Claves candidatas

Clave candidata:  $(\text{dni\_cliente}, \text{cod\_hotel}, \text{habitacion}, \text{fecha\_inicio\_hospedaje})$ .

Cada estadía se identifica de forma única por cliente, hotel, habitación y fecha de inicio.

### Paso 4: Normalización (1FN → 2FN → 3FN)

#### 1FN:

La relación original ya está en Primera Forma Normal, ya que todos los atributos son atómicos (no existen multivaluados ni compuestos).

## 2FN:

Se eliminan las dependencias parciales, es decir, aquellas en las que algunos atributos dependen solo de parte de la clave primaria compuesta.

Dependencias parciales detectadas:

- cod\_hotel → cantidad\_habitaciones, direccion\_hotel, ciudad\_hotel, dni\_gerente
- dni\_gerente → nombre\_gerente
- dni\_cliente → nombre\_cliente, ciudad\_cliente

## Descomposición:

- R1 (**HOTEL**) = (cod\_hotel, cantidad\_habitaciones, direccion\_hotel, ciudad\_hotel, dni\_gerente)
- R2 (**GERENTE**) = (dni\_gerente, nombre\_gerente)
- R3 (**CLIENTE**) = (dni\_cliente, nombre\_cliente, ciudad\_cliente)
- R4 (**ESTADIA**) = (dni\_cliente, cod\_hotel, habitacion, fecha\_inicio\_hospedaje, cant\_dias\_hospedaje)

## 3FN:

Se eliminan dependencias transitivas, es decir, cuando un atributo depende de otro atributo no clave.

- cod\_hotel → dni\_gerente y dni\_gerente → nombre\_gerente (ya separadas en HOTEL y GERENTE).
- Además, se crea la tabla **HABITACION** para controlar el par (cod\_hotel, habitacion), ya que el número de habitación puede repetirse entre hoteles, pero no dentro del mismo hotel.

R5 (**HABITACION**) = (cod\_hotel, habitacion)

## Paso 5: Tablas finales en 3FN

### **CLIENTE**

(dni\_cliente PK,  
nombre\_cliente,  
ciudad\_cliente)

### **GERENTE**

(dni\_gerente PK,  
nombre\_gerente)

### **HOTEL**

(cod\_hotel PK,  
Direccion\_hotel,  
Ciudad\_hotel,  
cantidad\_habitaciones,  
Dni\_gerente FK→GERENTE)

### **HABITACION**

(cod\_hotel PK/FK→HOTEL,  
habitacion PK)

### **ESTADIA**

(dni\_cliente PK/FK→CLIENTE,  
cod\_hotel PK/FK→HOTEL,  
habitacion PK/FK→HABITACION,  
fecha\_inicio\_hospedaje PK,

cant\_dias\_hospedaje)

## Ejercicio 3 – Programas de radio

### Paso 1: Dependencias funcionales

radio, anio → frecuencia\_radio, gerente

radio, anio, programa → conductor

### Paso 2: Atributos primos y no primos

Atributos primos: radio, anio, programa

Atributos no primos: frecuencia\_radio, gerente, conductor

### Paso 3: Claves candidatas

Clave candidata: (radio, anio, programa).

Cada programa queda identificado por la radio, el año y su nombre.

### Paso 4: Normalización (1FN → 2FN → 3FN)

#### 1FN:

Todos los atributos son atómicos.

#### 2FN:

Eliminamos dependencias parciales, ya que algunos atributos dependen de solo una parte de la clave compuesta (radio, anio, programa).

Dependencias parciales detectadas:

- radio, anio → frecuencia\_radio, gerente

#### Descomposición:

- R1 (**RADIO\_ANIO**) = (radio, anio, frecuencia\_radio, gerente)

- R2 (**PROGRAMA**) = (radio, anio, programa, conductor)

### **3FN:**

No existen dependencias transitivas, ya que todos los atributos no clave dependen directamente de la clave primaria completa de su tabla.

## **Paso 5: Tablas finales en 3FN**

### **RADIO\_ANIO**

(radio PK,  
anio PK,  
Frecuencia\_radio,  
gerente)

### **PROGRAMA**

(radio PK/FK→RADIO\_ANIO,  
anio PK/FK→RADIO\_ANIO,  
programa PK,  
conductor)

## **Ejercicio 4 – Talleres de autos**

### **Paso 1: Dependencias funcionales**

codigo\_sucursal → domicilio\_sucursal, telefono\_sucursal  
 codigo\_sucursal, codigo\_fosa → largo\_fosa, ancho\_fosa  
 patente\_auto → marca\_auto, modelo\_auto, dni\_cliente  
 dni\_cliente → nombre\_cliente, celular\_cliente  
 dni\_mecanico → nombre\_mecanico, email\_mecanico  
 (codigo\_sucursal, codigo\_fosa, patente\_auto, dni\_mecanico) → define una reparación.

### **Paso 2: Atributos primos y no primos**

Atributos primos: codigo\_sucursal, codigo\_fosa, patente\_auto, dni\_mecanico

Atributos no primos: domicilio\_sucursal, telefono\_sucursal, largo\_fosa, ancho\_fosa, marca\_auto, modelo\_auto, dni\_cliente, nombre\_cliente, celular\_cliente, nombre\_mecanico, email\_mecanico

### Paso 3: Claves candidatas

Clave candidata: (codigo\_sucursal, codigo\_fosa, patente\_auto, dni\_mecanico).

### Paso 4: Normalización (1FN → 2FN → 3FN)

#### 1FN:

La tabla original ya está en 1FN porque todos los atributos son atómicos.

#### 2FN:

Se eliminan las dependencias parciales.

Dependencias parciales detectadas:

- codigo\_sucursal → domicilio\_sucursal, telefono\_sucursal
- codigo\_sucursal, codigo\_fosa → largo\_fosa, ancho\_fosa
- patente\_auto → marca\_auto, modelo\_auto, dni\_cliente
- dni\_cliente → nombre\_cliente, celular\_cliente
- dni\_mecanico → nombre\_mecanico, email\_mecanico

#### Descomposición:

- R1 (**SUCURSAL**) = (codigo\_sucursal, domicilio\_sucursal, telefono\_sucursal)
- R2 (**FOSA**) = (codigo\_sucursal, codigo\_fosa, largo\_fosa, ancho\_fosa)
- R3 (**CLIENTE**) = (dni\_cliente, nombre\_cliente, celular\_cliente)
- R4 (**AUTO**) = (patente\_auto, marca\_auto, modelo\_auto, dni\_cliente FK→CLIENTE)
- R5 (**MECANICO**) = (dni\_mecanico, nombre\_mecanico, email\_mecanico)

- R6 (**REPARACION**) = (codigo\_sucursal FK→SUCURSAL, codigo\_fosa FK→FOSA, patente\_auto FK→AUTO, dni\_mecanico FK→MECANICO)  
 $(PK \text{ compuesta: } (codigo\_sucursal, codigo\_fosa, patente\_auto, dni\_mecanico))$

### **3FN:**

Eliminamos dependencias transitivas:

- patente\_auto → dni\_cliente y dni\_cliente → nombre\_cliente (ya separadas en AUTO y CLIENTE).
- codigo\_sucursal → domicilio\_sucursal y codigo\_sucursal, codigo\_fosa → largo\_fosa (ya separadas).

También se aclara que (codigo\_sucursal, codigo\_fosa) forma una **clave compuesta** para identificar de forma única cada fosa dentro de una sucursal.

## Paso 5: Tablas finales en 3FN

### **SUCURSAL**

(codigo\_sucursal PK,

Domicilio\_sucursal,

telefono\_sucursal)

### **FOSA**(codigo\_sucursal PK/FK→SUCURSAL,

codigo\_fosa PK,

Largo\_fosa,

ancho\_fosa)

### **CLIENTE**

(dni\_cliente PK,

nombre\_cliente,  
celular\_cliente)

**AUTO**

(patente\_auto PK,  
Marca\_auto,  
Modelo\_auto,  
dni\_cliente FK→CLIENTE)

**MECANICO**

(dni\_mecanico PK,  
Nombre\_mecanico,  
email\_mecanico)

**REPARACION**

(codigo\_sucursal PK/FK→SUCURSAL,  
codigo\_fosa PK/FK→FOSA,  
patente\_auto PK/FK→AUTO,  
dni\_mecanico PK/FK→MECANICO)

## Ejercicio 5 – Torneos de ciclismo

### Paso 1: Dependencias funcionales

cod\_torneo → nombre\_torneo  
cod\_torneo, cod\_corredor → nyap\_corredor  
cod\_torneo, cod\_bicicleta → marca\_bicicleta  
sponsor → dni\_presidente\_sponsor, dni\_medico  
cod\_torneo, cod\_corredor, cod\_bicicleta, sponsor → define\_participación.

## Paso 2: Atributos primos y no primos

Atributos primos: cod\_torneo, cod\_corredor, cod\_bicicleta, sponsor

Atributos no primos: nombre\_torneo, marca\_bicicleta, nyap\_corredor, dni\_presidente\_sponsor, dni\_medico

## Paso 3: Claves candidatas

Clave candidata: (cod\_torneo, cod\_corredor, cod\_bicicleta, sponsor)

## Paso 4: Normalización (1FN → 2FN → 3FN)

### 1FN:

La relación inicial ya está en 1FN, pues todos los atributos son atómicos.

### 2FN:

Se eliminan dependencias parciales, separando atributos que dependen solo de parte de la clave compuesta (cod\_torneo, cod\_corredor, cod\_bicicleta, sponsor).

Dependencias parciales detectadas:

- cod\_torneo → nombre\_torneo
- cod\_torneo, cod\_corredor → nyap\_corredor
- cod\_torneo, cod\_bicicleta → marca\_bicicleta
- sponsor → dni\_presidente\_sponsor, dni\_medico

### Descomposición:

- R1 (**TORNEO**) = (cod\_torneo, nombre\_torneo)
- R2 (**CORREDOR**) = (cod\_torneo, cod\_corredor, nyap\_corredor)
- R3 (**BICICLETA**) = (cod\_torneo, cod\_bicicleta, marca\_bicicleta)
- R4 (**SPONSOR**) = (sponsor, dni\_presidente\_sponsor, dni\_medico)

- R5 (**PARTICIPACION**) = (cod\_torneo FK→TORNEO, cod\_corredor FK→CORREDOR, cod\_bicicleta FK→BICICLETA, sponsor FK→SPONSOR)

### **3FN:**

Eliminamos dependencias transitivas:

- sponsor → dni\_presidente\_sponsor, dni\_medico (ya separadas en SPONSOR).  
En las demás relaciones, los atributos no clave dependen únicamente de sus respectivas claves primarias.

## **Paso 5: Tablas finales en 3FN**

### **TORNEO**

(cod\_torneo PK,  
nombre\_torneo)

### **CORREDOR**

(cod\_torneo PK/FK→TORNEO,  
cod\_corredor PK,  
nyap\_corredor)

### **BICICLETA**

(cod\_torneo PK/FK→TORNEO,  
cod\_bicicleta PK,  
marca\_bicicleta)

### **SPONSOR**

(sponsor PK,  
Dni\_presidente\_sponsor,

dni\_medico)

#### PARTICIPACION

(cod\_torneo PK/FK→TORNEO,  
cod\_corredor PK/FK→CORREDOR,  
cod\_bicicleta PK/FK→BICICLETA,  
sponsor PK/FK→SPONSOR)

## Ejercicio 6 – Juegos Olímpicos

### Paso 1: Dependencias funcionales

anio.olimpiada → pais.olimpiada  
nombre.deportista → pais.deportista  
(anio.olimpiada, nombre.deportista) → nombre.disciplina, asistente

### Paso 2: Atributos primos y no primos

Atributos primos: anio.olimpiada, nombre.deportista

Atributos no primos: pais.olimpiada, pais.deportista, nombre.disciplina, asistente

### Paso 3: Claves candidatas

Clave candidata: (anio.olimpiada, nombre.deportista)

### Paso 4: Normalización (1FN → 2FN → 3FN)

#### 1FN:

La relación está en 1FN porque todos los atributos son atómicos.

#### 2FN:

Se eliminan dependencias parciales, separando atributos que dependen solo de parte de la clave (anio.olimpiada, nombre.deportista).

Dependencias parciales detectadas:

- $\text{anio\_olimpiada} \rightarrow \text{pais\_olimpiada}$
- $\text{nombre\_deportista} \rightarrow \text{pais\_deportista}$

**Descomposición:**

- R1 (**OLIMPIADA**) = (anio.olimpiada, pais.olimpiada)
- R2 (**DEPORTISTA**) = (nombre.deportista, pais.deportista)
- R3 (**PARTICIPACION**) = (anio.olimpiada FK→OLIMPIADA, nombre.deportista FK→DEPORTISTA, nombre.disciplina, asistente)

**3FN:**

No existen dependencias transitivas, ya que en todas las relaciones los atributos no clave dependen directamente de sus respectivas claves primarias.

La relación PARTICIPACION representa correctamente la asociación entre deportista y olimpiada.

## Paso 5: Tablas finales en 3FN

### **OLIMPIADA**

(anio.olimpiada PK,  
pais.olimpiada)

### **DEPORTISTA**

(nombre.deportista PK,  
pais.deportista)

### **PARTICIPACION**

(anio.olimpiada PK/FK→OLIMPIADA,  
nombre.deportista PK/FK→DEPORTISTA,

Nombre\_disciplina,

asistente)