

# T2 - Normalización

La normalización es uno de los principales **principios** en el diseño de una base de datos.

- Organiza los datos siguiendo reglas
- Minimiza la redundancia
- Reduce anomalías
- Puede mejorar la mantenibilidad y según el caso, la performance

¿Cómo funciona?

- Toma una relación grande como **entrada** y la descompone en relaciones más **pequeñas** las cuales están libre de redundancia de datos y otras anomalías como la de inserción/eliminación.
- La normalización se aplica a base de datos relacionales.

## Conceptos generales

### Anomalía

Una **anomalía** se refiere a un problema o comportamiento no deseado que ocurre cuando los datos no están bien organizados, normalmente porque la base de datos no esta normalizada.

### Dependencia funcional

Es una restricción entre subconjuntos de atributos de una relación.

Una dependencia funcional es una relación entre atributos (columnas) de una tabla que indica que un atributo depende de otro.

✦ Un atributo **Y** depende funcionalmente de un atributo **X** (se escribe  $X \rightarrow Y$ ) si **para cada valor de X hay un único valor asociado de Y**.

Ejemplos :

## Dependencia Funcional (df)

### Ejemplo 1:

—Dada la relación

PERSONA(dni, nombre, edad, fechaNacimiento)

—Y valga en PERSONA la

df: dni->nombre,edad,fechaNac



La df enunciada, indica que, si dos tuplas t1 y t2 de la relación PERSONA tienen el mismo valor en el atributo dni, deben necesariamente tener los mismos valores en los atributos nombre, edad y fechaNac

## Dependencia Funcional (df)

### Ejemplo 2:

—Dada la relación: PERSONA(dni, nombre, edad, fechaNac, nroLegajo)

*Donde*

- *Una persona posee un único número de legajo asignado*
- *Un número de legajo pertenece a una sola persona*

—Se pueden enunciar las siguientes dfs

df1) dni -> nombre, edad, fechaNac, nroLegajo

df2) nroLegajo -> nombre, edad, fechaNac, dni

### • Dependencia Funcional (df)

#### Ejemplo 3:

—Dada la relación: PERSONA(dni, nombre, edad, fechaNac, nroLegajo, carrera)

»Donde

- *Una persona puede cursar diversas carreras*
- *Nombre indica como se llama la persona*
- *Una persona posee un único número de legajo asignado para cada carrera que cursa*
- *Un número de legajo pertenece a una sola persona de una carrera*

—Se pueden enunciar las siguientes dfs

df1) dni -> nombre, edad, fechaNac

df2) nroLegajo, carrera -> dni

df3) dni, carrera -> nroLegajo

- No se puede identificar a una persona con solo el legajo ya que puede repetirse el legajo en otra carrera e identificar a otra persona.
- Lo mismo para dni y legajo, no podemos saber la carrera. Ya que se puede repetir para otra carrera.

## Dependencia funcional trivial

Una dependencia funcional trivial es un tipo de dependencia que siempre se cumple de manera obvia, porque el lado derecho (el dependiente) está contenido en el lado izquierdo ( el determinante ).

- ✦ Se dice trivial porque no aporta información nueva.
- ✦ Una dependencia funcional  $X \rightarrow Y$  es **trivial** si  $Y \subseteq X$ .

### Ejemplos :

Tenemos un conjunto de atributos:

- $X = \{\text{DNI, Nombre}\}$
- $Y = \{\text{Nombre}\}$

👉 Entonces:  $\{\text{DNI, Nombre}\} \rightarrow \{\text{Nombre}\}$  es una **dependencia funcional trivial**, porque **Nombre** ya está incluido en el lado izquierdo.

- $\{\text{DNI, Nombre}\} \rightarrow \{\text{DNI}\}$  ✓ trivial
- $\{\text{DNI, Nombre}\} \rightarrow \{\text{DNI, Nombre}\}$  ✓ trivial (el conjunto completo)
- $\{\text{DNI}\} \rightarrow \{\text{Nombre}\}$  ✗ no trivial (no está incluido en X)

## Dependencia Funcional trivial

⚙ Ejemplo:

—Dada la relación:

CONTRATADO(nroContratado, dni, nombrePersona, inicioActividad)

Donde valen las siguientes dependencias funcionales:

- df1) dni -> nombrePersona
- df2) nroContratado, dni -> inicioActividad



Algunas dependencias funcionales triviales válidas en CONTRATADO son:

- dft1) dni-> dni
- dft2) nroContratado, dni -> nroContratado

## Forma normal

## Primera forma normal

- La tabla no puede tener datos repetidos ni atributos multivaluados (Listas, conjuntos, etc).
- Cada celda debe contener un solo valor atómico.
- No puede haber grupos repetidos.

## Segunda forma normal

- Debe cumplir 1FN + los atributos no clave deben depender de toda la clave primaria, no solo de una parte (solo se aplica si la clave primaria es compuesta).

Ejemplo :

```
Venta(idProducto, idFactura, nombreProducto, cantidad)
```

👉 `nombreProducto` depende solo de `idProducto`, no de la combinación completa.

```
Producto(idProducto, nombreProducto)
```

```
Factura(idFactura)
```

```
Venta(idProducto, idFactura, cantidad)
```

## Tercera forma normal

- Debe cumplir 2FN + eliminar dependencias transitivas
- Un atributo no clave no debe depender de otro atributo que no es clave

Ejemplo :

```
Empleado(dni, nombre, idDepto, nombreDepto)
```

Clave primaria: dni

👉 `nombreDepto` depende de `idDepto`, y `idDepto` depende de `dni`. Esto es una dependencia transitiva.

```
Empleado(dni, nombre, idDepto)
```

```
Departamento(idDepto, nombreDepto)
```