

## EJERCICIO-EJEMPLO DE NORMALIZACIÓN

Dado el siguiente esquema de relación que representa información de una empresa

**EMPRESA** (NSS, nombre-empl, dirección, salario, sexo, fecha-nac, num-depart, nombre-depart, idlugar, nombre-lugar, metros, num-proyecto, nombre-proyecto, presupuesto, duración, numHorasSemanales)

Se considera los siguientes supuestos semánticos:

1. De cada empleado se guarda el nombre completo (pueden coincidir dos empleados llamados iguales), número de seguridad social, dirección, salario, sexo y fecha de nacimiento. Todo empleado está asignado a un departamento y puede trabajar en varios proyectos, guardándose también el número de horas por semana que un empleado trabaja en cada proyecto.
2. Cada departamento se identifica por un número y se guarda el nombre (único) y está distribuido en un lugar.
3. Cada lugar se identifica por un código, tiene un nombre único y se guardan los metros cuadrados.
4. Cada proyecto se identifica por un número y se guarda el nombre (único), presupuesto y duración.

- 1) Comentar los problemas y anomalías de la anterior relación.
- 2) Establecer las dependencias funcionales.
- 3) Establece las claves candidatas (primaria y alternativa)
- 4) Normalizar hasta la FNBC

### 1) COMENTAR LOS PROBLEMAS Y ANOMALÍAS QUE ENCUENTRES EN LA ANTERIOR RELACIÓN.

- **Redundancias:** consiste en la repetición innecesaria de datos.
  - ✓ Para un empleado asignado a un proyecto → nombre-empl, dirección, salario, sexo, fecha-nac, num-depart, nombre-depart, idlugar, nombre-lugar, metros.
  - ✓ Para cada empleado asignado al mismo departamento → nombre-depart, idlugar, nombre-lugar, metros
  - ✓ Para cada proyecto que se le asignen empleados → nombre-proyecto, presupuesto, duración
- **Anomalías de modificación,** debidas a las redundancias: cambiar cualquier dato del empleado, requiere cambiarlo en todos los proyectos asignados y puede ocurrir que se cambien en algunas tuplas y en otras no. Cambiar un dato del proyecto, requiere cambiarlo en todas las filas que ha sido asignado a un empleado y puede ocurrir que se cambien en unas filas y en otras no. Cambiar algún dato del departamento, requiere cambiarlo en todas las filas de los empleados de este departamento en todos los proyectos asignados y puede ocurrir que se modifiquen en unas y en otras no
- **Anomalías de inserción:** no se puede insertar un empleado sin proyectos, ni un departamento nuevo ( la inserción de un nuevo departamento requiere insertar un empleado y asignarle un proyecto). La creación de un nuevo lugar requiere la inserción de un departamento, y para ello habría que insertar un empleado y asignarle un proyecto
- **Anomalías de borrado:** al borrar un empleado se puede borrar de la BD la información del departamento y/o del proyecto (si no está asignado el departamento a otro empleado o el proyecto), y viceversa. Si se borra un departamento se puede perder la información del empleado si este solo está asignado a un proyecto, etc..

## 2) ESTABLECER LAS CLAVES CANDIDATAS.

### DEPENDENCIAS FUNCIONALES

**NSS** -> nombre-empl, dirección, salario, sexo, fecha-nac, num-depart, nombre-depart, idlugar, nombre-lugar, metros

**num-proyecto** -> presupuesto, duración

**num-depart** -> nombre-departa, idlugar, nombre-lugar, metros

**num-depart** <--> **nombre-departa**

**idlugar**-> nombre-lugar, metros

**NSS, num-proyecto** -> **numHorasSemanales**

**Num-proyecto** <--> **nombre-proyecto**

### CLAVES CANDIDATAS:

- ✓ **NSS-empl, num-proyecto**
- ✓ **NSS-empl, nombre-proyecto**

DE LAS CLAVES CANDIDATAS, SE ELIGE UNA COMO CLAVE PRIMARIA Y EL RESTO COMO ALTERNATIVAS.

#### ✓ Clave primaria:

CP-> **NSS-empl, num-proyecto**

#### ✓ Claves alternativas :

CP-> NSS-empl, nombre-proyecto

## 3) PROCESO DE NORMALIZACION.

a) Se señalan en la relación los atributos **primos** (si es miembro de **cualquier clave candidata** de la relación).

**EMPRESA** (**NSS**, nombre-empl, dirección, salario, sexo, fecha-nac, num-depart, nombre-depart, idlugar, nombre-lugar, metros, **num-proyecto**, **nombre-proyecto**, presupuesto, duración, numHorasSemanales)

b) Ver si se cumple la primera forma normal (1FN)

(Una tabla se encuentra en **primera forma normal** si impide que un atributo de una tupla pueda tomar más de un valor (**todos sus atributos son atómicos**). No se aceptan atributos multivaluados ni compuestos)

➤ La relación empresa **si cumple la primera forma normal(1FN)** por que es una relacion y una de las propiedades de las relaciones es que cada atributo solo puede contener un único valor.

c) Ver si se cumple la segunda forma normal (2FN)

(Un esquema de relación R está en **2ª Forma normal** cuando está en **1FN** y además **todo atributo no primo A en R dependen funcionalmente de manera completa** con respecto de cada una de las claves candidatas de R

➤ La relación EMPRESA **NO cumple la segunda forma normal(2FN)**

### Justificación:

Vamos a utilizar la clave primaria para ver las dependencias funcionales (nota: Podíamos escoger cualquier clave candidata)

### Dependencias Funcionales:

**NSS, num-proyecto** -> nombre-empl, dirección, salario, sexo, fecha-nac, num-depart, nombre-depart, idlugar, nombre-lugar, metros, presupuesto, duración, numHorasSemanales

**NSS** -> nombre-empl, dirección, salario, sexo, fecha-nac, num-depart, nombre-depart, idlugar, nombre-lugar, metros

**num-proyecto** -> presupuesto, duración

- ✓ los atributos no primos **nombre-empl, dirección, salario, sexo, fecha-nac, num-depart, nombre-depart, idlugar, nombre-lugar, metros** no tienen una dependencia total con respecto a la clave primaria (NSS, num-proyecto).
- ✓ Los atributos no primos **presupuesto, duración** no tienen una dependencia total con respecto a la clave primaria (NSS, num-proyecto).

Por tanto, **hay que descomponer** la anterior relación **EMPRESA** de la siguiente forma:

**EMPRESA** (NSS, num-proyecto, nombre-proyecto , numHorasSemanales)

**EMPLEADO** (NSS, nombre-empl, dirección, salario, sexo, fecha-nac, num-depart, nombre-depart, idlugar, nombre-lugar, metros)

**PROYECTO** (num-proyecto , presupuesto, duración )

Las tres tablas anteriores ya se encuentra en 2FN.

Le vamos a cambiar el nombre a la relacion EMPRESA por otro que refleje mejor la informacion que almacena. EMPRESA pasa a llamarse EMPLEADO\_PROYECTO

### Tablas resultantes de 2FN:

**EMPLEADO-PROYECTO** (NSS, num-proyecto, nombre-proyecto , numHorasSemanales)

**EMPLEADO** (NSS, nombre-empl, dirección, salario, sexo, fecha-nac, num-depart, nombre-depart, idlugar, nombre-lugar, metros)

**PROYECTO** (num-proyecto , presupuesto, duración )

### **d) Ver si se cumple la tercera forma normal (3FN)**

(Una relación está en **3ª Forma Normal** si está en **2FN** y **ningún atributo no primo de R depende transitivamente de alguna clave**. Es decir, no pueden existir dependencias transitivas entre los atributos que no forman parte de alguna clave de la relación R).

EMPLEADO-PROYECTO -> Si está en 3FN

EMPLEADO -> NO está en 3FN

PROYECTO -> Si está en 3FN

En la relacion empleado existen atributos no primos que dependen transitivamente de alguna clave.

### Justificación:

#### Dependencias Funcionales de EMPLEADO:

**NSS** -> nombre-empl, dirección, salario, sexo, fecha-nac, num-depart, nombre-departa, idlugar, nombre-lugar, metros

**num-depart** -> nombre-departa, idlugar, nombre-lugar, metros

Por tanto, **hay que descomponer** la anterior relación **EMPLEADO** de la siguiente forma:

**EMPLEADO** (**NSS**, nombre-empl, dirección, salario, sexo, fecha-nac, num-depart)

**DEPARTAMENTO** ( **num-depart**, nombre-departa, idlugar, nombre-lugar, metros)

Ahora EMPLEADO si está en 3FN

DEPARTAMENTO NO está en 3FN

### Justificación:

#### Dependencias Funcionales de DEPARTAMENTO:

**num-depart** -> nombre-departa, idlugar, nombre-lugar, metros

**idlugar**-> nombre-lugar, metros

Por tanto, **hay que descomponer** la anterior relación **DEPARTAMENTO** de la siguiente forma:

**DEPARTAMENTO** ( **num-depart**, nombre-departa, idlugar)

**LUGAR** (**idlugar**, nombre-lugar, metros)

Ahora las tablas EMPLEADO, DEPARTAMENTO, LUGAR, PROYECTO, EMPLEADO-PROYECTO si que están en 3FN

### Tablas resultantes de 3FN:

**EMPLEADO-PROYECTO** (**NSS**, **num-proyecto**, nombre-proyecto , numHorasSemanales)

**PROYECTO** (**num-proyecto** , presupuesto, duración )

**EMPLEADO** (**NSS**, nombre-empl, dirección, salario, sexo, fecha-nac, num-depart)

**DEPARTAMENTO** ( **num-depart**, nombre-departa, idlugar)

**LUGAR** (**idlugar**, nombre-lugar, metros)

**e) Ver si se cumple la forma normal de Boyce-Codd (FNBC)**

(Una relación está en FNBC si está en 3FN y si los antecedentes de todas las dependencias funcionales (determinantes) son **claves candidatas** )

**EMPLEADO-PROYECTO -> NO está en FNBC**

PROYECTO -> Si está en FNBC

EMPLEADO -> Si está en FNBC

DEPARTAMENTO -> Si está en FNBC

LUGAR -> Si está en FNBC

**Justificación:**

**Dependencias Funcionales de EMPLEADO-PROYECTO:**

NSS, num-proyecto -> numHorasSemanales

NSS, nombre-proyecto -> numHorasSemanales

Num-proyecto <-> nombre-proyecto

**Antecedentes de las dependencias funcionales**

NSS, num-proyecto **SI ES CLAVE CANDIDATA**

NSS, nombre-proyecto **SI ES CLAVE CANDIDATA**

Num-proyecto **NO ES CLAVE CANDIDATA**

nombre-proyecto **NO ES CLAVE CANDIDATA**

**La descomposición sería:**

**PROYECTO2 (num-proyecto, nombre-proyecto)**

**EMPLEADO-PROYECTO (NSS, num-proyecto, numHorasSemanales)**

Las tablas PROYECTO y PROYECTO2 **se unen** ya que tienen la misma clave.

**Esquema de relacion resultante de la normalización:**

