

Formas Normales

- Normalización - Introducción
- Primera Forma Normal
- Segunda Forma Normal
- Tercera Forma Normal
- Forma Normal de Boyce-Codd
- Dependencias Multivaluadas
- Cuarta Forma Normal

Normalización

◆ Introducción

- En el proceso de normalización se somete un esquema relación (er) a una serie de pruebas para “certificar” si pertenece o no a una cierta **forma normal**.
- Puede considerarse como un proceso durante el cual los er insatisfactorios se descomponen repartiendo sus atributos entre ers más pequeños que poseen propiedades deseables.
- Las formas normales, sin considerar otros factores, no garantizan un buen diseño de BD. Props adicionales:
 - » **Join sin pérdida**
 - » **Preservación de dependencias**

Superclave

◆ Superclave

- Una superclave de $R = \{A_1, \dots, A_n\}$ es un conjunto de atributos $S \subseteq R$ tal que no existen 2 tuplas distintas t_1 y t_2 en ningún r tal que $t_1[S] = t_2[S]$.

◆ Clave

- Una clave K es una superclave que cumple que si se le quita alguno de sus atributos, deja de ser superclave.

Clave Candidata

◆ Clave candidata, clave primaria

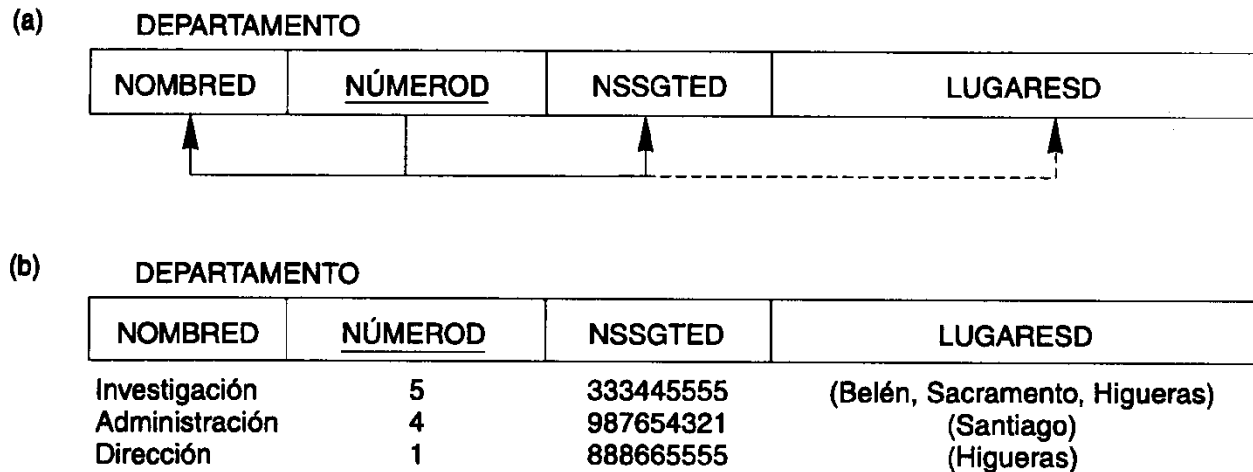
- Si una relacion tiene mas de una clave, cada una es una clave candidata. Una de ellas es arbitrariamente designada como clave primaria. El resto son secundarias.

Primera Forma Normal (1NF)

◆ Definición

- Los dominios de los atributos deben incluir solo valores atómicos (los atributos no pueden ser multivaluados ni compuestos)

◆ Ejemplo



2NF: Definiciones Previas

◆ **Atributo Primo**

- Un atributo del esquema relación R es **primo** si es miembro de alguna clave de R.

2NF: Definiciones Previas

◆ Dependencia Total

- $X \rightarrow Y$ es una df **total** si la eliminación de cualquier atributo A de X hace que la df deje de ser válida. (no tiene atr. redundantes a la izq.)

2NF: Definiciones Previas

◆ Dependencia Parcial

- $X \rightarrow Y$ es una df **parcial** si es posible eliminar un atributo A de X, y la df sigue siendo válida.

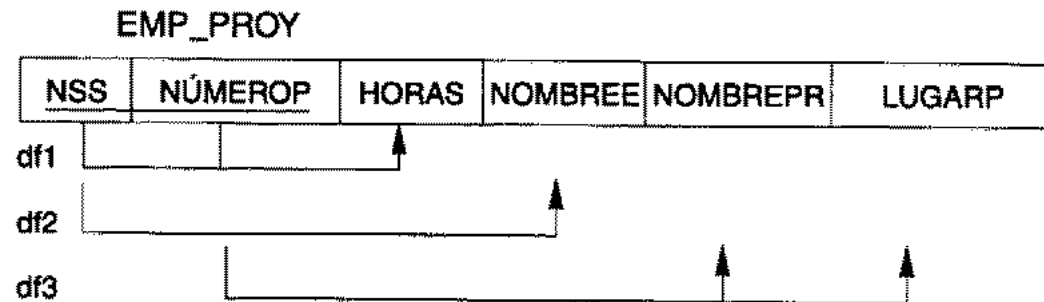
Segunda Forma Normal (2NF)

◆ Definición

- Un er R está en 2NF si ningún atributo no primo A de R depende parcialmente de cualquier clave de R.

Segunda Forma Normal (2NF)

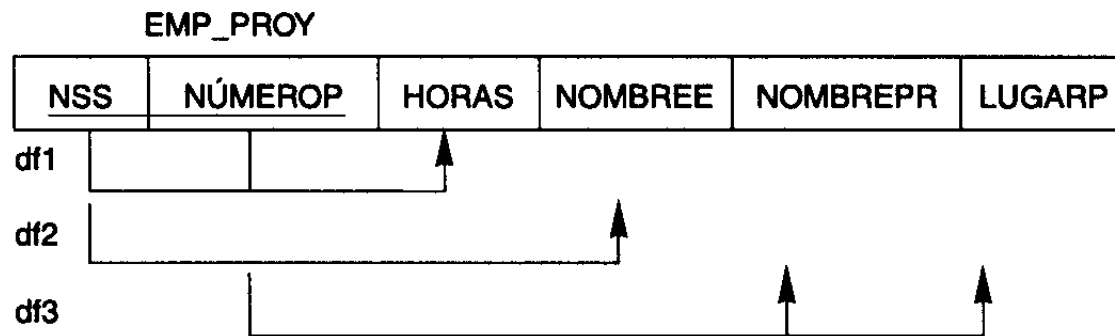
◆ Ejemplo



- El atributo no primo NOMBREE viola 2NF, por la df2
- Los atr no primos NOMBREPR y LUGARP violan 2NF, por la df3

Segunda Forma Normal (2NF)

(a)



NORMALIZACIÓN 2FN



3NF: Definiciones Previas

◆ Dependencia Transitiva

- Una df $X \rightarrow Y$ en un er R es una df **transitiva** si existe un conjunto de atributos Z que no sea un subconjunto de una clave de R, y se cumplen tanto $X \rightarrow Z$ como $Z \rightarrow Y$.

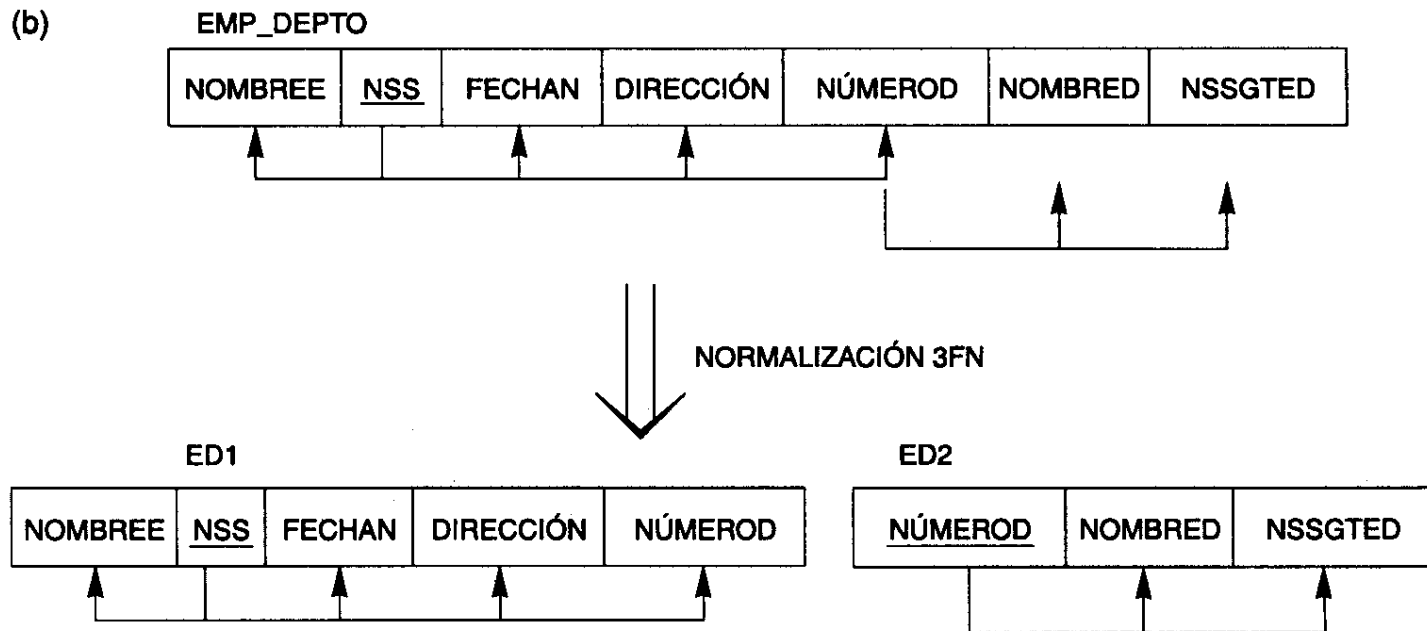
Tercera Forma Normal (3NF)

◆ Definición

- Un er R está en 3NF si está en 2NF y ningún atributo no primo de R depende transitivamente de una clave de R.
- Un er R está en 3NF si, siempre que una df $X \rightarrow A$ se cumple en R, o bien (a) X es una superclave de R, o (b) A es un atributo primo de R.

Tercera Forma Normal (3NF)

◆ Ejemplo

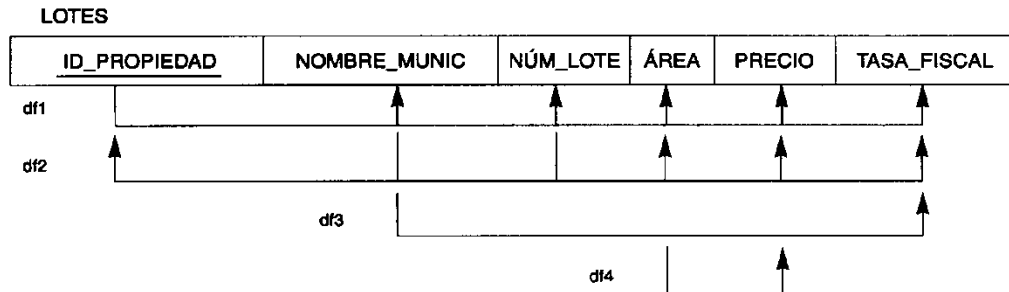


–Las dfs $\text{NUMEROD} \rightarrow \text{NOMBRED}$ y $\text{NUMEROD} \rightarrow \text{NSSGTED}$ violan 3NF

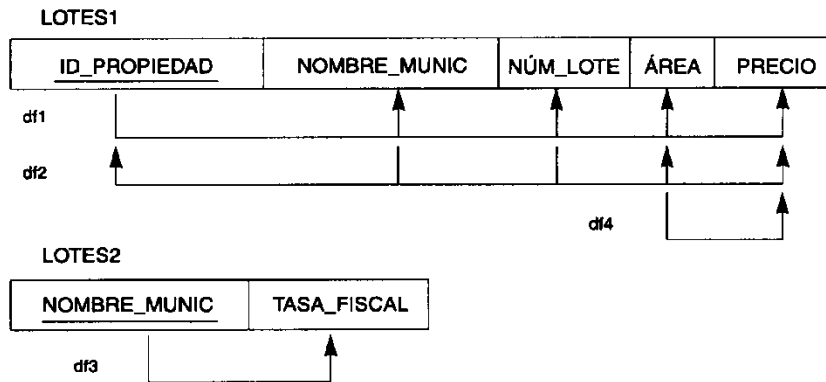
Tercera Forma Normal (3NF)

◆ Ejemplo

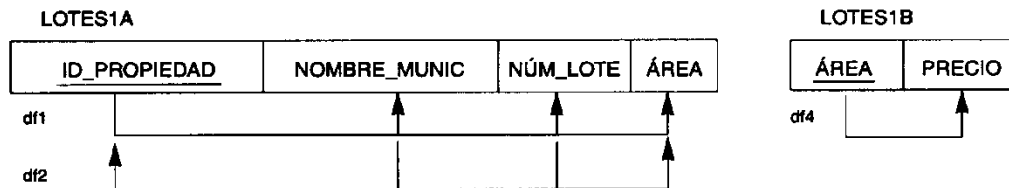
(a)



(b)



(c)

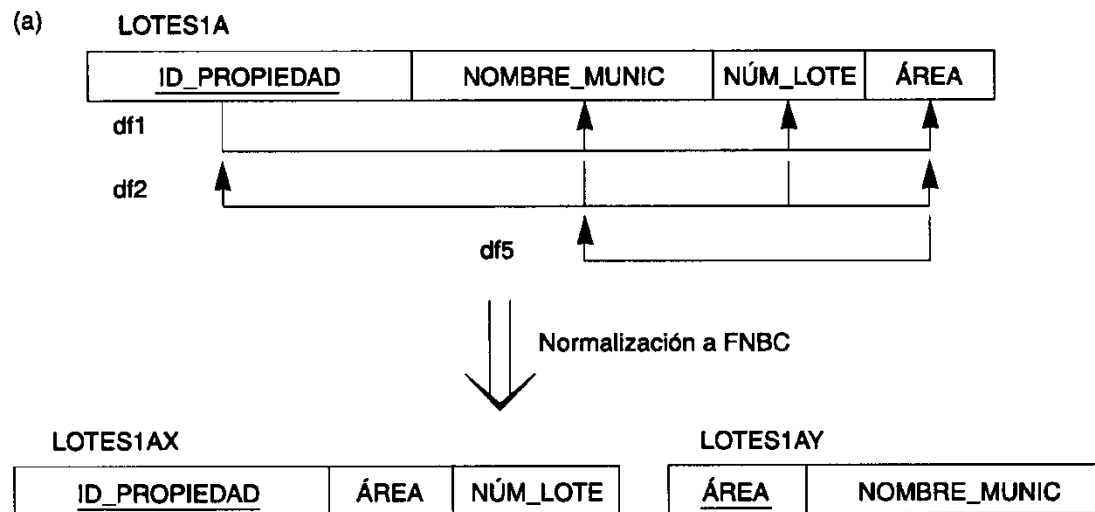


Forma Normal de Boyce-Codd (BCNF)

◆ Definición

- Un er R está en BCNF si, siempre que una df $X \rightarrow A$ se cumple en R, entonces X es una superclave de R.

◆ Ejemplo



Forma Normal de Boyce-Codd (BCNF)

◆ Ejemplo

Imparte (estudiante, curso, profesor)

$F = \{ \text{estudiante, curso} \rightarrow \text{profesor}$
 $\text{profesor} \rightarrow \text{curso} \}$

3NF

Claves: {estudiante, curso} y {estudiante, profesor}

Descomposiciones posibles:

{estudiante, profesor} y {estudiante, curso}

{curso, profesor} y {curso, estudiante}

{profesor, curso} y {profesor, estudiante}

Algoritmos de diseño

- Descomposición de relaciones
- Preservación de dependencias
- Descomposición en 3NF preservando las dfs
- Join sin pérdida. Propiedad
- Test de join sin pérdida
- Descomposición en BCNF con JSP
- Descomposición en 3NF con JSP y pres de dfs
- Problemas con valores nulos y tuplas colgantes

Descomposición de relaciones

◆ Esquema relación universal R

- $R = (A_1, A_2, \dots, A_n)$, que contiene todos los atributos de la BD

Descomposición de relaciones

◆ Descomposición de R, D

- $D = (R_1, R_2, \dots, R_m)$, que se obtiene mediante los algoritmos que realizan la descomposición utilizando las dependencias funcionales
- Se debe verificar: $\cup_{i=1}^m R_i = R$

Preservación de dependencias

◆ Proyección de un conjunto de dependencias sobre un Esquema de Relación

- Dado un conjunto de dfs F sobre R , la **proyección** de F sobre R_i , $\Pi_{R_i}(F)$, donde R_i es un subconj de R , es el conj de dfs $X \rightarrow Y$ en F^+ tal que los atributos en $X \cup Y$ estén todos contenidos en R_i .

Preservación de dependencias

◆ Preservación de dependencias

- Una descomposición $D = (R_1, R_2, \dots, R_m)$ de R preserva las dependencias respecto a F si se cumple: $((\Pi_{R_1}(F)) \cup \dots \cup (\Pi_{R_m}(F)))^+ = F^+$

Descomposición en 3NF con pres de dfs

♦ Algoritmo

1. Encontrar un cubrimiento minimal G para F ;
2. Para cada miembro izq X de una df que aparezca en G
crear un er $\{X \cup A_1 \cup A_2 \dots \cup A_m\}$ en D , donde
 $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_m$ sean las únicas dfs en G
con X como miembro izq;
3. Colocar todos los atributos restantes (que no fueron colocados en ningún er) en un solo er para asegurar la prop. de preservación de dependencias;

Join sin Pérdida

◆ Definición

- Una descomposición $D = (R_1, R_2, \dots, R_m)$ de R tiene la propiedad de **JSP** respecto al conjunto de dfs F sobre R , si por cada instancia de relación r de R que satisfaga F , se cumple lo siguiente:

$$* (\Pi_{R_1}(r), \dots, \Pi_{R_m}(r)) = r$$

Join sin Pérdida

◆ Propiedad

- $D = (R_1, R_2)$ de R tiene JSP respecto a F sobre R sii
 - la df $(R_1 \cap R_2) \rightarrow (R_1 - R_2)$ está en F^+
ó - la df $(R_1 \cap R_2) \rightarrow (R_2 - R_1)$ está en F^+

Test de join sin pérdida

♦ Algoritmo

1. crear una matriz S con una fila i por cada relación R_i en la desc D , y una columna j por cada atributo A_j en R ;
2. hacer $S(i,j) := b_{ij}$ para todas las entradas de la matriz;
3. para cada fila i que represente el r_i en R_i
 para cada columna j que represente el atributo A_j
 si R_i incluye a A_j entonces hacer $S(i,j) := a_j$;
4. repetir hasta que una ejecución no modifique S
 para cada df $X \rightarrow Y$ en F
 igualar los símbolos en los atributos de Y para
 aquellas filas que coinciden en los atributos de X ;
5. si una fila tiene todos símbolos “a”, la desc es con JSP, en caso contrario, no lo es;

Test de join sin pérdida

◆ Ejemplo

- (c) $R = \{NSS, NOMBREE, NÚMEROP, NOMBREPR, LUGARP, HORAS\}$ $D = \{R1, R2, R3\}$
 $R1 = EMP = \{NSS, NOMBREE\}$
 $R2 = PROYECTO = \{NÚMEROP, NOMBREPR, LUGARP\}$
 $R3 = TRABAJA_EN = \{NSS, NÚMEROP, HORAS\}$
 $F = \{NSS \rightarrow NOMBREE; NÚMEROP \rightarrow \{NOMBREPR, LUGARP\}; \{NSS, NÚMEROP\} \rightarrow HORAS\}$

	NSS	NOMBREE	NÚMEROP	NOMBREPR	LUGARP	HORAS
R1	a_1	a_2	b_{13}	b_{14}	b_{15}	b_{16}
R2	b_{21}	b_{22}	a_3	a_4	a_5	b_{26}
R3	a_1	b_{32}	a_3	b_{34}	b_{35}	a_6

(matriz original S al principio del algoritmo)

	NSS	NOMBREE	NÚMEROP	NOMBREPR	LUGARP	HORAS
R1	a_1	a_2	b_{13}	b_{14}	b_{15}	b_{16}
R2	b_{21}	b_{22}	a_3	a_4	a_5	b_{26}
R3	a_1	b_{32} a_2	a_3	b_{34} a_4	b_{35} a_5	a_6

(la matriz S después de aplicar las dos primeras dependencias funcionales - la última fila sólo tiene símbolos "a", así que nos detenemos)

Descomposición en BCNF con JSP

♦ Algoritmo

```
1. hacer  $D := \{ R \}$ ;  
2. mientras haya un  $Q$  en  $D$  que no esté en BCNF hacer  
    comenzar  
        escoger un  $Q$  en  $D$  que no esté en BCNF;  
        encontrar una  $df\ X \rightarrow Y$  en  $Q$  que viole BCNF;  
        reemplazar  $Q$  en  $D$  por dos esquemas  $(Q - Y)$  y  $(X \cup Y)$   
    fin;
```

Desc en 3NF con JSP y pres de dfs

♦ Algoritmo

1. encontrar un cubrimiento minimal G para F;
2. para cada miembro izq X de una df que aparezca en G
crear un er $\{X \cup A_1 \cup A_2 \dots \cup A_{1m}\}$ en D, donde
 $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_m$ sean todas las dfs en G con X
como miembro izq;
3. si ninguno de los er contiene una clave de R, crear un er
adicional que contenga atributos que formen una clave de R;
4. eliminar er redundantes, es decir que estén contenidos en otro
ya generado

Problemas con nulos

◆ Ejemplo

(a)

EMPLEADO				
NOMBREE	<u>NSS</u>	FECHAN	DIRECCIÓN	NÚMD
Silva, José B.	123456789	09-ENE-55	Fresnos 731, Higuera, MX	5
Vizcarra, Federico T.	333445555	08-DIC-45	Valle 638, Higuera, MX	5
Zapata, Alicia J.	999887777	19-JUL-58	Castillo 3321, Sucre, MX	4
Valdés, Jazmín S.	987654321	20-JUN-31	Bravo 291, Belén, MX	4
Nieto, Ramón K.	666884444	15-SEP-52	Espiga 975, Heras, MX	5
Esparza, Josefa A.	453453453	31-JUL-62	Rosas 5631, Higuera, MX	5
Jabbar, Ahmed V.	987987987	29-MAR-54	Dalias 980, Higuera, MX	4
Botello, Jaime E.	888665555	10-NOV-27	Sorgo 450, Higuera, MX	1
Bernal, Andrés C.	999775555	26-ABR-55	Becerra 6530, Belén, MX	nulo
Benítez, Carlos M.	888664444	09-ENE-53	Bejuco 7654, Higuera, MX	nulo

DEPARTAMENTO

NOMBRED	<u>NÚMD</u>	NSSGTED
Investigación	5	333445555
Administración	4	987654321
Dirección	1	888665555

(b)

Problemas con nulos

NOMBREE	<u>NSS</u>	FECHAN	DIRECCIÓN	NÚMD	NOMBRED	NSSGTED
Silva, José B.	123456789	09-ENE-55	Fresnos 731, Higueras, MX	5	Investigación	333445555
Vizcarra, Federico T.	333445555	08-DIC-45	Valle 638, Higueras, MX	5	Investigación	333445555
Zapata, Alicia J.	999887777	19-JUL-58	Castillo 3321, Sucre, MX	4	Administración	987654321
Valdés, Jazmín S.	987654321	20-JUN-31	Bravo 291, Belén, MX	4	Administración	987654321
Nieto, Ramón K.	666884444	15-SEP-52	Espiga 975, Heras, MX	5	Investigación	333445555
Esparza, Josefa A.	453453453	31-JUL-62	Rosas 5631, Higueras, MX	5	Investigación	333445555
Jabbar, Ahmed V.	987987987	29-MAR-59	Dalias 980, Higuera, MX	4	Administración	987654321
Botello, Jaime E.	888665555	10-NOV-27	Sorgo 450, Higueras, MX	1	Dirección	888665555

(c)

NOMBREE	<u>NSS</u>	FECHAN	DIRECCIÓN	NÚMD	NOMBRED	NSSGTED
Silva, José B.	123456789	09-ENE-55	Fresnos 731, Higueras, MX	5	Investigación	333445555
Vizcarra, Federico T.	333445555	08-DIC-45	Valle 638, Higueras, MX	5	Investigación	333445555
Zapata, Alicia J.	999887777	19-JUL-58	Castillo 3321, Sucre, MX	4	Administración	987654321
Valdés, Jazmín S.	987654321	20-JUN-31	Bravo 291, Belén, MX	4	Administración	987654321
Nieto, Ramón K.	666884444	15-SEP-52	Espiga 975, Heras, MX	5	Investigación	333445555
Esparza, Josefa A.	453453453	31-JUL-62	Rosas 5631, Higueras, MX	5	Investigación	333445555
Jabbar, Ahmed V.	987987987	29-MAR-59	Dalias 980, Higuera, MX	4	Administración	987654321
Botello, Jaime E.	888665555	10-NOV-27	Sorgo 450, Higueras, MX	1	Dirección	888665555
Bernal, Andrés C.	999775555	26-ABR-55	Becerra 6530, Belén, MX	nulo	nulo	nulo
Benitez, Carlos M.	888664444	09-ENE-53	Bejuco 7654, Higueras, MX	nulo	nulo	nulo

Problemas con tuplas colgantes

(a) EMPLEADO_1

NOMBREE	<u>NSS</u>	FECHAN	DIRECCIÓN
Silva, José B.	123456789	09-ENE-55	Fresnos 731, Higuerras, MX
Vizcarra, Federico T.	333445555	08-DIC-45	Valle 638, Higuerras, MX
Zapata, Alicia J.	999887777	19-JUL-58	Castillo 3321, Sucre, MX
Valdés, Jazmín S.	987654321	20-JUN-31	Bravo 291, Belén, MX
Nieto, Ramón K.	666884444	15-SEP-52	Espiga 975, Heras, MX
Esparza, Josefa A.	453453453	31-JUL-62	Rosas 5631, Higuerras, MX
Jabbar, Ahmed V.	987987987	29-MAR-54	Dalias 980, Higuerras, MX
Botello, Jaime E.	888665555	10-NOV-27	Sorgo 450, Higuerras, MX
Bernal, Andrés C.	999775555	26-ABR-55	Becerra 6530, Belén, MX
Benítez, Carlos M.	888664444	09-ENE-53	Bejuco 7654, Higuerras, MX

(c) EMPLEADO_3

<u>NSS</u>	NÚMD
123456789	5
333445555	5
999887777	4
987654321	4
666884444	5
453453453	5
987987987	4
888665555	1

Resumen

- ◆ **Se presentaron las ideas básicas de la teoría de diseño relacional.**
 - Dependencia Funcional
 - Forma Normal
 - Descomposición
 - Preservación de Dependencias
 - Join Sin Pérdida
 - Algoritmos de Normalización.

Resumen

