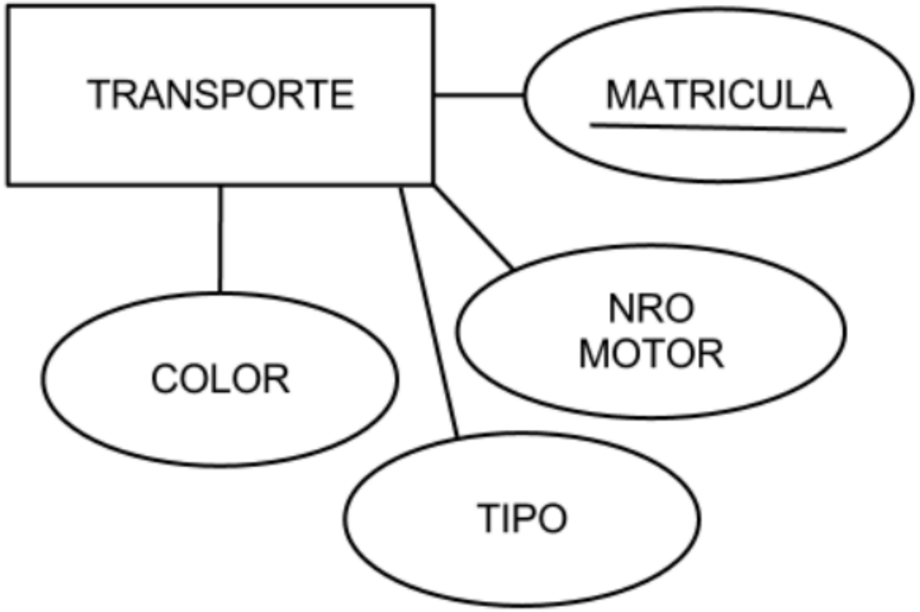
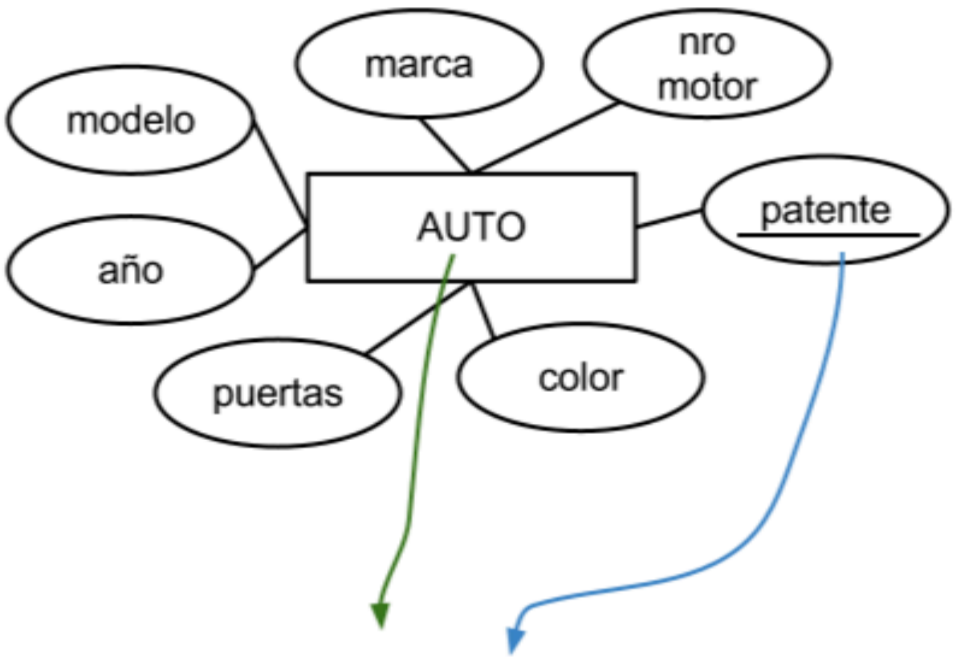
De una Entidad a una Relación



- **Cada atributo de la entidad, se transforma en un atributo de una relación**

    - Transporte(<u>matricula</u>, nroMotor, color, Tipo)

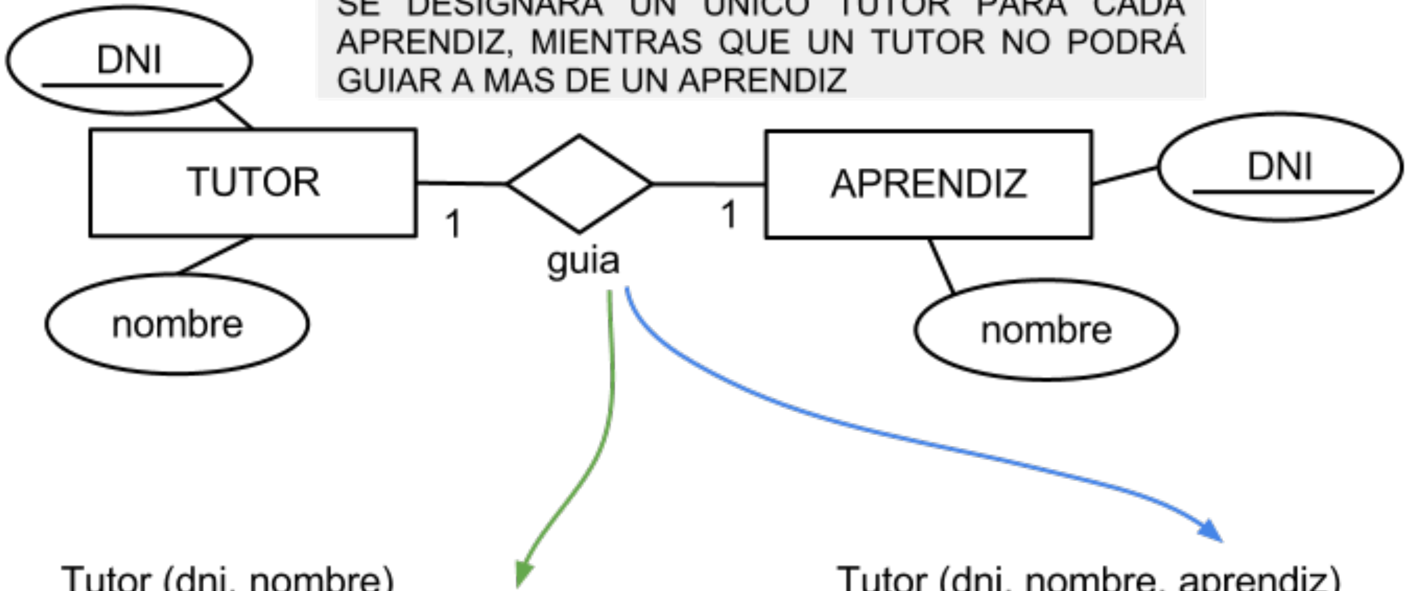
DEL DER AL Modelo Relacional (MR)



Auto (patente, color, nro_motor, marca, puertas, año, modelo)

Transformación de Relaciones 1.1

SE DESIGNARÁ UN ÚNICO TUTOR PARA CADA APRENDIZ, MIENTRAS QUE UN TUTOR NO PODRÁ GUIAR A MAS DE UN APRENDIZ



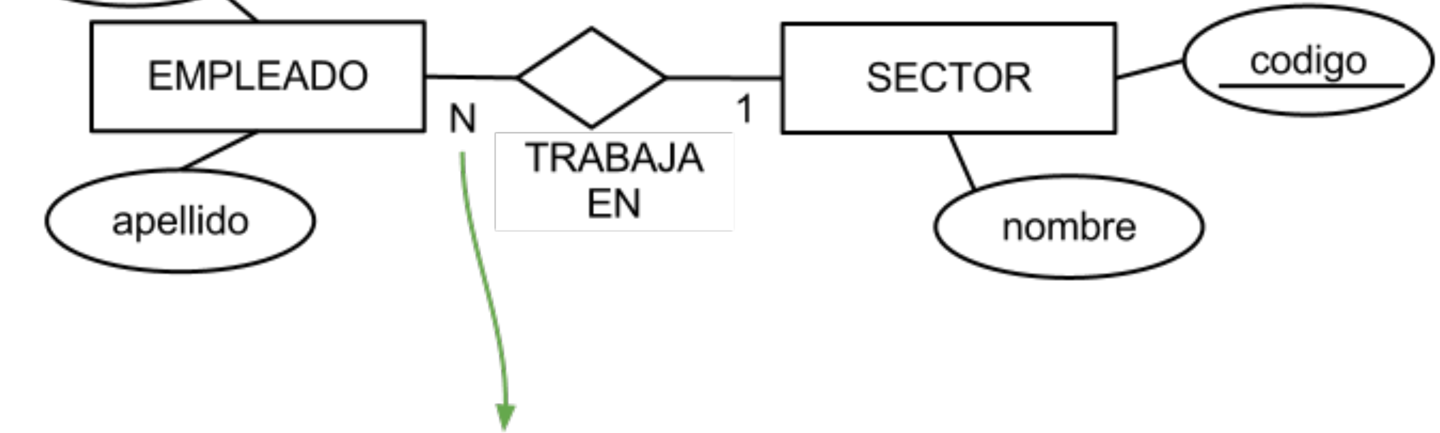
Tutor (dni, nombre)
Aprendiz (dni, nombre, tutor)

Tutor (dni, nombre, aprendiz)
Aprendiz (dni, nombre)

COMO AMBAS ENTIDADES SON MANDATORIAS, PODEMOS ELEGIR DONDE COLOCAR EL ATRIBUTO FORANEO

Relaciones 1.N

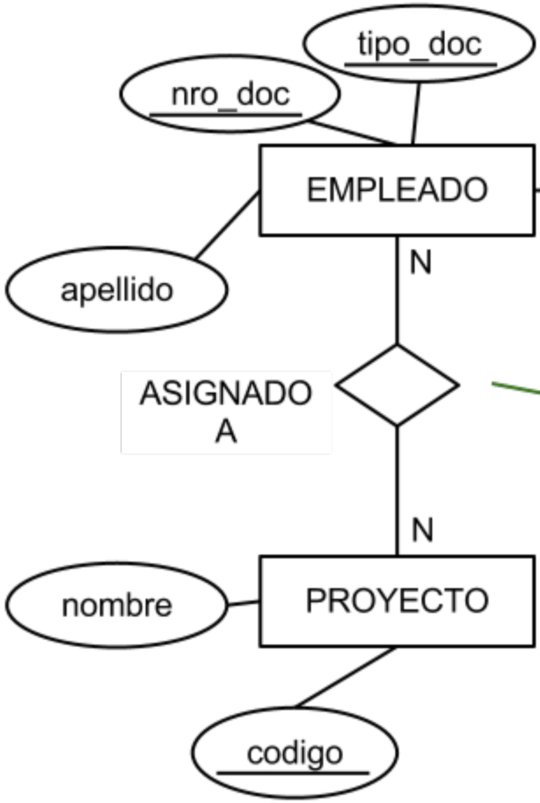
UN SECTOR TIENE MUCHOS EMPLEADOS ASIGNADOS. CADA EMPLEADO PUEDE ESTAR ASIGNADO A UN SOLO SECTOR



Empleado (Legajo, apellido, sector_codigo)
Sector (codigo, nombre)

AQUI, LA ENTIDAD MANDATORIA ES SECTOR. ENVIAMOS LA CLAVE CODIGO DE SECTOR A EMPLEADO

Relaciones N.N



UN EMPLEADO ESTA ASIGNADO A VARIOS PROYECTOS Y UN PROYECTO ES REALIZADO POR VARIOS EMPLEADOS

Empleado (<u>nro_doc, tipo_doc</u>, nombre, apellido)

Asignado_a (<u>empleado_tipo, empleado_nro_doc, proyecto_codigo</u>)

Proyecto (<u>codigo</u>, nombre)

1. CREAMOS LAS RELACIONES EMPLEADO Y PROYECTO
2. CREAMOS LA RELACION ASIGNADO_A
3. TOMAMOS LOS ATRIBUTOS CLAVES DE AMBAS ENTIDADES Y LOS COLOCAMOS EN ASIGNADO_A
4. LOS DEFINIMOS COMO FORANEOS, POR SER DE OTRA ENTIDAD Y COMO CLAVE

Ejemplo de N.N con datos

EMPLEADO		
TIPO_DOC	NRO_DOC	NOMBRE
DNI	21.343.121	JUANA
PASAPORTE	84.618.121	ROBERTO

TOMO LOS CAMPOS CLAVES Y LOS TRANSPORTO

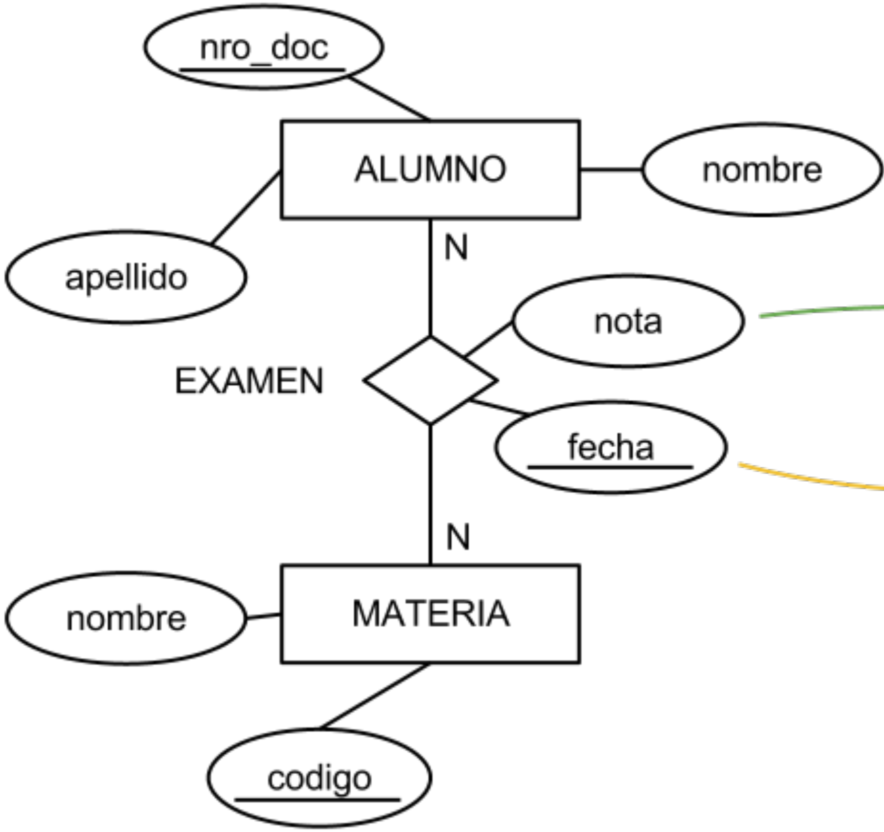
ASIGNADO A		
TIPO_DOC	NRO_DOC	P_CODIGO
DNI	21.343.121	P1
DNI	21.343.121	P2
PASAPORTE	84.618.121	P1

VEMOS QUE UN EMPLEADO PUEDE ESTAR ASIGNADO A MÁS DE UN PROYECTO Y UN PROYECTO ESTAR ASIGNADO A VARIOS EMPLEADOS

PROYECTO	
CODIGO	NOMBRE
P1	INSTALACION
P2	REFORMA

TOMO EL CAMPO CLAVE Y LO TRANSPORTO

Ejemplo 2 N.N con atributos



UN ALUMNO PUEDE REALIZAR UN EXAMEN EN MATERIAS Y EN UNA MATERIA SE PUEDEN EXAMINAR A VARIOS ALUMNOS

Alumno (nro_doc, nombre, apellido)

Examen (nro_doc, codigo, fecha, nota)

Materia (codigo, nombre)

1. HACEMOS LO MISMO QUE EN EL CASO ANTERIOR,
2. AGREGAMOS LOS ATRIBUTO DE LA RELACIÓN FECHA Y NOTA.
3. LA CLAVE DE EXAMEN ESTA DADA POR LAS CLAVES DE LAS ENTIDADES MÁS LA CLAVE MARCADA EN LA RELACIÓN.
4. MARCAMOS LOS ATRIBUTOS QUE SON FORANEOS

Dependencia Funcional

X → Y

X: Conjunto Determinante
Y: Conjunto Determinado

"al tener siempre el mismo valor en X, obtendremos el mismo valor en Y"
Ya que Y depende funcionalmente de X.

Ejemplos de Dependencias funcionales (DF)

Patente → Marca	tipo doc, nro doc → Nombre	HistClinica → Paciente
usuario → Apellido	nro Cuenta → saldo	Pasaporte → País, Nombre
isbn → nombre libro	nro tel → propietario	Código HTML → Color

No es Bidireccional!

Pasaporte	País
123213123	México
423423423	México
A12312312	Argentina
A43543534	Brasil

No es Bidireccional!

Fecha de Nacimiento	Edad
18/9/2000	21
19/9/2000	20
18/9/2001	20
14/10/2000	20
31/08/2000	21

Axiomas de Armstrong

- Aumento
  - X → Y => X Z → Y Z
- Transitividad
  - X → Y e Y → Z => X → Z
- Reflexividad
  - X → X

$DNI \longrightarrow nombre$

$DNI, dirección \longrightarrow nombre, dirección$

$FechaDeNacimiento \longrightarrow Edad$

$Edad \longrightarrow Conducir$

$FechaDeNacimiento \longrightarrow Edad \longrightarrow Conducir$

Reglas derivadas

- Unión
  - $X \rightarrow Y$ e $X \rightarrow Z$ *entonces* $X \rightarrow YZ$

- Descomposición
  - $X \rightarrow YZ \Rightarrow X \rightarrow Y$ *entonces* $X \rightarrow Z$

- Pseudo Transitividad
  - $X \rightarrow Y$ y $WY \rightarrow Z$ *entonces* $WX \rightarrow Z$

Unión => $X \rightarrow Y$ e $X \rightarrow Z$ *entonces* $X \rightarrow YZ$

1. $X \rightarrow Y$ (dada)
2. $X \rightarrow YX$ (Aumento x)
3. $X \rightarrow Z$ (dado)
4. $XY \rightarrow ZY$ (Aumento Y el punto 3)
5. $X \rightarrow ZY$ (transitividad 2 y 4 $X \rightarrow YX$ e $XY \rightarrow ZY$)

Descomposición:

$X \rightarrow YZ \Rightarrow X \rightarrow Y$ *entonces* $X \rightarrow Z$

1. $X \rightarrow YZ$ (dada)
2. $YZ \rightarrow Z$ (Reflexividad)
3. $YZ \rightarrow Y$ (Reflexividad)
4. $X \rightarrow Z$ (Transitividad 1 y 2 => $X \rightarrow YZ$ e $YZ \rightarrow Z$)
5. $X \rightarrow Y$ (transitividad 2 y 3 $X \rightarrow YZ$ e $YZ \rightarrow Y$)

Pseudo Transitividad:

$X \rightarrow Y$ y $WY \rightarrow Z$ *entonces* $WX \rightarrow Z$

1. $X \rightarrow Y$ (dada)
2. $XW \rightarrow YW$ (Aumento 1)
3. $WY \rightarrow Z$ (Dado)
4. $XW \rightarrow Z$ (Transitividad 1 y 3 => $XW \rightarrow YW$ e $WX \rightarrow Z$)

Pseudotransitividad

- Pseudo Transitividad
  - X → Y y WY → Z *entonces* WX → Z

R(ABCDEFG)
Fmin = {B → C, C → A, C → F, F → B, C → D, B → E}

{GB} = { GBCEAFD }

GB → **ABCDEFG**
Y ADEMAS: F → B ENTONCES, POR PSEUDOTRANSITIVIDAD,
FG → **ABCDEFG es decir, DETERMINA A TODO EL CONJUNTO DE R**
LO MISMO, SIRVE PARA EL EJEMPLO DE
C → B → CG → ABCDEFG

CONJUNTO DE DEPENDENCIAS FUNCIONALES

- Dentro de una Relacion R, tendremos asociado un conjunto “F” de dependencias funcionales

R (<u>dni</u>, nombre, apellido, fechaNac, edad)

F = { dni $\rightarrow$ nombre, apellido, fechaNac ,
fechaNac $\rightarrow$ edad }

Formas Normales

- Reglas aplicadas a las relaciones de una base de datos para determinar su estado de normalización

- Es decir, se validará si todas sus relaciones tienen claves, como se forman esas claves, etc

Clausura de Un elemento

- La clausura de un elemento refiere a que atributos alcanza directa o indirectamente un atributo (indirectamente es a traves de las reglas de Armstrong).

R(X, Y, Z, W, Q)

- Si tenemos a $X \rightarrow Y$, $Y \rightarrow Z$ y $W \rightarrow Q$, Podemos decir que:

$\{ X \}^{+} = \{ X, Y, Z \}$

$\{ Y \}^{+} = \{ Y, Z \}$

$\{Z\}^{+} = \{ Z \}$

$\{W\}^{+} = \{ W, Q \}$

$\{ Q \}^{+} = \{ Q \}$

CLAUSURA: Ejemplo con columnas reales

R(TIPO_DOC, FECHA_NAC,EDAD, NRO_DOC, APELLIDO, NOMBRE, PREFIJO, LOCALIDAD, NRO_TELEFONO)

F { TIPO_DOC, NRO_DOC → FECHA_NAC

TIPO_DOC, NRO_DOC → APELLIDO, NOMBRE, LOCALIDAD

FECHA_NAC → EDAD

LOCALIDAD → PREFIJO

TIPO_DOC, NRO_DOC → NRO_TELEFONO

}

{TIPO_DOC} = {TIPO_DOC} → 1ra A. TODO ELEMENTO, DETERMINA A SI MISMO.

{FECHA_NAC} = {FECHA_NAC, EDAD}

{TIPO_DOC, NRO_DOC} = {TIPO_DOC, NRO_DOC, FECHA_NAC, APELLIDO, NOMBRE, EDAD , NRO_TELEFONO, LOCALIDA) PREFIJO}

{PREFIJO} = {PREFIJO}

{LOCALIDAD} = {LOCALIDAD, PREFIJO}

Super Clave (SC)

- Un conjunto de atributos es super clave si determina funcionalmente al resto de atributos de la Relación

R (<u>dni</u>, nombre, apellido, fechaNac, edad)

F = { dni → nombre, apellido, fechaNac ,

fechaNac → edad }

LA CLAUSURA DE <u>DNI</u>, NOS DARÁ TODOS LOS ATRIBUTOS DEL CONJUNTO R, POR LO QUE SERÍA UNA CLAVE

POR OTRO LADO, LA CLAUSURA DE { DNI, NOMBRE } TAMBIEN DETERMINARÁ TODO EL CONJUNTO R, POR LO QUE TAMBIEN ES UNA SUPER CLAVE.

R (<u>dni</u>, nombre, apellido, fechaNac, edad)

F = { dni $\rightarrow$ nombre, apellido, fechaNac ,
fechaNac $\rightarrow$ edad }

Clausuras:

{DNI,NOMBRE} = {DNI,NOMBRE, APELLIDO, FECHAHAC,EDAD}

ENTONCES, EL CONJUNTO DNI+NOMBRE, ¿ES UNA CLAVE CANDIDATA?

¡NO!

¿COMO LO SABEMOS? * DEBEMOS HACER LAS CLAUSURAS DE LOS ELEMENTOS POR SEPARADO, Y VER SI SOLOS, ALCANZAN O NO, EL RESTO DE ATRIBUTOS DE LA RELACIÓN

$\{NOMBRE\}^{+}$ = {NOMBRE}

$\{DNI\}^{+}$ = {DNI, NOMBRE, APELLIDO, FECHANAC, EDAD}

ENTONCES, COMO DNI CUBRE TODA LA RELACIÓN, LA CC ES DNI, NO {DNI+NOMBRE}, SIENDO ESTO, UNA SUPER CLAVE

Clave Candidata (CC)

- Una clave candidata es una super clave, que tiene además la propiedad de ser MINIMA.
- Ser minima significa que, si le quito alguno de sus elementos, deja de ser una clave.

- R(XYZWQGT)
- F = { X → Y, Y → WZQ, Q → TG }
- CLAUSURA DE X e Y => $\{ XY \}^{+}$ = { XYWZQTG }
- XY es una superclave o una clave candidata? => para saberlo, tenemos uqe hacer la clausura de X e Y y ver si cada una determina al conjunto R. Si ninguna por su cuenta determina a R, XY es una CC. si alguna por su cuenta determina a R, XY es solo una SC.

Clausura de X => {X} = {XY……….}

CLAVES CANDIDATAS CON DISTINTA CANTIDAD DE ATRIBUTOS

ALUMNO(LEGAJO, TIPO_DOC, NRO_DOC, NOMBRE, APELLIDO, TELEFONO, DIRECCIÓN}

F = {

LEGAJO → TIPO_DOC, NRO_DOC

TIPO_DOC, NRO_DOC → LEGAJO, NOMBRE, APELLIDO, TELEFONO, DIRECCION

}

{ TIPO_DOC, NRO_DOC}+ = {LEGAJO, NOMBRE, APELLIDO, TELEFONO, DIRECCION}

{LEGAJO}+ = { LEGAJO, TIPO_DOC, NRO_DOC,NOMBRE, APELLIDO, TELEFONO, DIRECCION}

CONJUNTO DE CLAVES CANDIDAS → CC = { LEGAJO, {TIPO_DOC, NRO_DOC} }

Conjunto Minimo de DF

- Existe un algoritmo para minimizar el conjunto de DF y lograr que este sea más fácil de trabajar y procesar luego.

- Reglas del algoritmo:
- 1. disminuir lados derechos: Es decir, aplicar la propiedad de division
  - Si X → YZ entonces dividir en X → Y , X → Z

- 2. Intentar disminuir los lados Izquierdos o determinates.
  - XW → Z tenemos que hacer las clausuras de X y W y ver si por si mismas determinan a Z. Si es así, Podemos reemplazar la DF por X → Z o W → Z

- 3. Intentar eliminar DF que puedan obtenerse a través de otas. Es decir que si tenemos las DF X → Z, X → Y e Y → Z, podremos eliminar X → Z ya que es un dato que puede igualmente obtenerse en la clausura de X

Ejemplo Fmin

R(ABCDEFG) con

F = { B → CD, C → AF, F → B, FC → D, ACB → ED, BD → A }

PRIMER PASO. DIVIDIR LADOS DERECHOS.

EJEMPLO. SI TENEMOS B → CD, LO REEMPLAZAMOS POR B → C Y B → D.

F’={B → C, B → D, C → A, C → F, F → B, FC → D, ACB → E,
ACB → D, BD → A }

Ejemplo Fmin

R(ABCDEFG) con

- **F = { B → CD, C → AF, F → B, FC → D, ACB → ED, BD → A }**

F' = {B → C, B → D, C → A, C → F, F → B, FC → D,
$A\cancel{C}B$ → E, $A\cancel{C}B$ → D, BD → A }

SEGUNDO PASO. INTENTAR ELIMINAR ATRIBUTOS DEL LADO IZQUIERDO. EJEMPLO.

Tomamos ACB → E y queremos reducir el lado IZQUIERDO. Por lo que intentaremos quedarnos con alguna de las siguientes opciones

A → E | B → E | C → E | AB → E | AC → E | BC → E

$\{A\}^+ = \{A\}$ $\{B\}^+ = \{BCDAFE\}$ $\{C\}^+ = \{CAFBDE\}$

F'' = {B → C, B → D, C → A, C → F, F → B, C → D,
B → E, B → A }

F = fB

Ejemplo Fmin

R(ABCDEFG) con

F = { B → CD, C → AF, F → B, FC → D, ACB → ED, BD → A }

F''={B → C, ~~B → D~~, C → A, C → F, F → B, C → D, B → E, B → A }

TERCER PASO. Intentar Reducir la cantidad de DF del conjunto F'
$\{B\}^+$ (sin contar B → C) = {BDAE} como el determinado en la DF seleccionada no aparece en la clausura, no puedo eliminarla por no ser redundante.

$\{B\}^+$ (sin contar B → D) = {BCAFD} por transitividad, B alcanza la D, utilizando las siguientes DF, B → C y C → D. Entonces la DF B → D puede ser suprimida por ser redundante

Siguiendo este razonamiento para el resto de DF, nos queda:

F'''= {B → C, ~~B → D~~, C → A, C → F, F → B, C → D, B → E, ~~B → A~~ }

Ejemplo Fmin

R(ABCDEFG) con

F = { B → CD, C → AF, F → B, FC → D, ACB → ED, BD → A }

Fmin= {B → C, C → A, C → F, F → B, C → D, B → E}

{B}+ = { BCAFDE } ES TODO EL CONJUNTO? NO. FALTA

{G}+= {G}

{BG}+ = { BCAFDEG } ES TODO EL CONJUNTO? SI, ENTONCES ES CC

{CG}+ = {GCAFBDE} —

{FG}+ = {F**GC**…. . }

{AFG}+ = {ABCDEFG}

CC = {{ BG}, {CG}, {FG} }

Atributos Primos

- Atributos que forman parte de una clave **compuesta**

- Ejemplo:

  - PERSONA (tipoDoc, NroDoc, nombre, apellido, fechaNac)
  - F = {tipoDoc, NroDoc → nombre, apellido, fechaNac}

  - Clave candidata: {tipoDoc, NroDoc}

  - Atributos Primos: tipoDoc, NroDoc

Atributos Primos

R(ABCDEFG) con

F = { B → CD, C → AF, F → B, FC → D, ACB → ED, BD → A

CC = { {BG} , {CG} , {FG} }

PRIMOS SON: B, G, C, F

ALUMNO(LEGAJO, TIPO_DOC, NRO_DOC, NOMBRE, APELLIDO, TELEFONO, DIRECCIÓN}

F = { LEGAJO → TIPO_DOC, NRO_DOC

TIPO_DOC, NRO_DOC → LEGAJO, NOMBRE, APELLIDO, TELEFONO, DIRECCION }

CC = { LEGAJO, {TIPO_DOC, NRO_DOC} }

PRIMOS: TIPO_DOC, NRO_DOC

NOTA: LEGAJO NO ES PRIMO, PORQUE NO ES PARTE DE UNA CLAVE CANDIDATA **COMPUESTA**

Primera Forma Normal (1FN)

- Toda Relación debe tener Clave.
  - La clave debe tener valores únicos y No Nulos!

Segunda Forma Normal (2FN)

- No debe existir la dependencia parcial de una clave:

- Ejemplo. Si tenemos un R(x,y,z,w) y su clave es {X,Y}
  Para cumplir con 2FN, no debemos tener una DF que sea del tipo

$$Y \rightarrow Z$$

Ya que de esta forma, Z dependería parcialmente de la clave XY y no del total de la clave

Ejemplo que no cumple con 2FN

R (<u>tipoDocumento</u>, <u>numeroDocumento</u>,
nombre, apellido, f_Nac, edad, tieneVencimiento)

F = { tipoDocumento, numeroDocumento $\boxed{?}$ nombre, apellido,
f_Nac ,
f_Nac → edad ,
tipoDocumento → tieneVencimiento }

CC = {tipoDocumento, numeroDocumento}

Tercera Forma Normal (3FN)

- Para que una DF este en 3FN, el determinante debe ser Super Clave (o clave candidata) o el determinado un atributo Primo.

- R(XYZWQ)
- F = { XY → ZWQ, WQ → Y }

- {XY} ES UNA CC

Forma normal de Boyce-Codd (FNBC)

- Igual que 3FN, pero solo cumple con la parte de que el determinante sea Super clave O clave candidata.

- R(XYZWQ)
- F = { XY → ZWQ, WQ → Y }

- {XY} ES UNA CC

WQ → Y, POR LO TANTO, NO CUMPLE CON FNBC

Algoritmo para alcanzar 3FN (Sin Pérdida de Info y sin Perdida de DF)

- **Paso 1. Armar el Fmin.**
- **Paso 2. Tomamos los distintos determinantes y armamos nuevas relaciones con los determinantes**
- **Paso 3. Sumamos los determinados a la nueva relación de cada determinante**
- **Paso 4 (condicional) ¿Esta alguna de las claves originales en alguno de los R nuevos?**
    - **Si no se cuentra, armar una nueva relación con una de las claves**

Pérdida de Información

1. Al dividir una relación, no poder construir la misma tupla por tener información faltante
2. No pueda reconstruir, por tener información de más.
3. Al juntar relaciones tengo información que antes no tenía.

Paso 1. Armar el Fmin

R(ABCDEFG) con

F = { B → CD, C → AF, F → B, FC → D, ACB → ED, BD → A }

F′={B → C, B → D, C → A, C → F, F → B, FC → D,
ACB → E, ACB → D, BD → A }

F′′ = {B → C, B → D, C → A, C → F, F → B,
C → D, B → E, B → A }

F′′′= {B → C, C → F, F → B, C → D, B → E, B → A}
= **Fmin**

CC: {BG, FG, CG}

Paso 2. Tomamos los distintos determinantes.

{ B → C, C → F, F → B, C → D, B → E, B → A}

Paso 3. Armamos los nuevos esquemas R con los determinantes y los determinados.

R1 (BCEA) **R2 (CFD)** **R3 (FB)**

Paso 4. Armamos los conjuntos de DF correspondientes.

R1 (BCEA)	**F1= {B → C, B → E, B → A}**	**Cc: {B}**
R2 (CFD)	**F2= {C → F, C → D}**	**cc: {C}**
R3 (FB)	**F3= {F → B}**	**cc: {F}**

Paso 4. ¿Esta alguna de las claves originales en alguno de los R nuevos?

R4 (BG)	**F4= { }**	**cc: {BG}**

Algoritmo FNBC (Sin Pérdida de Info y puede perder DF)

- Paso 1. (no obligatorio) Armar el Fmin
- Paso 2. Dividimos el esquema en R1 y R2.
  - Poniendo en R1 los atributos de la 1er DF que no cumpla con BC, por definición, tomaremos siempre de Izquierda a Derecha
  - Poniendo en R2 el resto de atributos que esten en la relación, menos el determinado por la DF tomada
- Paso 3. Si R2 no se encuentra en FNBC, realizar el paso 2 nuevamente, pero con R2. Repetir esto hasta que el Rx2 este en FNBC.

Paso 1. Armar Fmin

R (A, B, C, D) con **F= { AB → C, C → D, D → A }**

F = Fmin **CC = { AB, DB, CB }**

Resul=
R1, R21 R22

Paso 2. Dividir R en R1 y R2.

~~R~~ **(A, B, C, D)** con **F= { AB → C, C → D, D → A }**

R1 (CD) F1 = {C → D} CC = { c }
~~R2~~ (ABC) F2 = {AB → C, C → A } (por transitividad, C → D y D → A => C → A
CC = { AB} + CB

R21 (CA) F21 = { C → A }
R22 (BC) F22 = { } bc → c es trivial, por lo que el conjunto queda vacio

Ejemplo armado de un conjunto F

Dada la siguiente relación **Registro (NombreCurso, Profesor, Hora, Aula, Estudiante, Nota)**

Y las siguientes restricciones (dadas por las reglas del negocio) identificar las Dependencias Funcionales:

- Cada curso es impartido por varios profesores.

~~CURSO → PROFESOR~~

PROFESOR → NOMBRECURSO

- A una hora y en un aula se imparte un solo curso.

Hora, Aula → NombreCurso

- A una hora determinada, un profesor está es una única aula.

Hora, Profesor → Aula | Hora, Aula ? Profesor

- Cada estudiante obtiene una nota en cada curso tomado (solo puede tomar una vez cada curso).

Estudiante, NombreCurso → Nota

- A una hora determinada, un estudiante puede estar en una sola aula.

Hora, Estudiante → Aula | ~~Hora, Aula → Estudiante?~~

Ejercicios Clase Proxima

(Practica de DF)

Ejercicios:

- 1
- 3
- 6
- 10
- 16