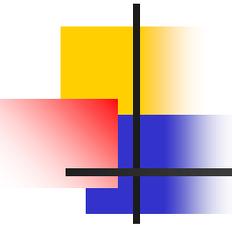


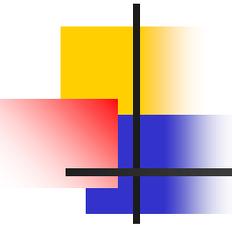
Bases de Datos

Álgebra Relacional



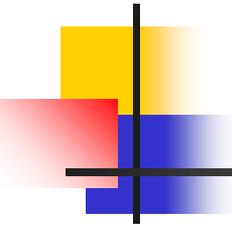
Álgebra Relacional

- Lenguajes de acceso a BD
 - Álgebra Relacional
 - Lenguaje procedimental (se indica qué obtener y cómo obtenerlo)
 - Lenguajes de Usuario
 - SQL (Structured Query Language), basado en álgebra relacional
 - Se usan para construir consultas sobre una Base de Datos



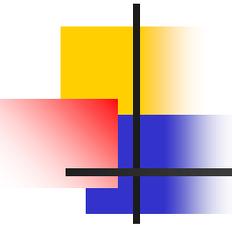
Álgebra Relacional

- Definición: Conjunto de operaciones sobre Relaciones
 - Actúan sobre Relaciones
 - Producen Relaciones como resultados
 - Pueden combinarse para construir expresiones más complejas
- Operadores
 - Básicos y Derivados
 - Propios de BD y de Conjuntos



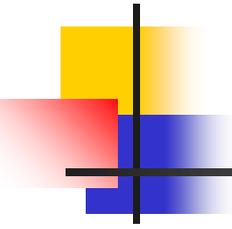
Operadores Básicos

- Propios de BD
 - Selección (σ) y Proyección (π)
- De Conjuntos
 - Unión (\cup), Diferencia ($-$), Producto Cartesiano (\times)
- Son operacionalmente completos
- Permiten expresar cualquier consulta sobre una BD



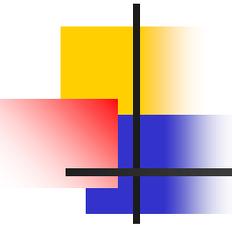
Operadores Derivados

- Propios de BD
 - Join condicional ($|X|_{condicion}$)
 - Join natural ($|X|$)
- De Conjuntos
 - Intersección (\cap), División ($/$)
- Se derivan combinando los operadores básicos
- No añaden nada nuevo. Fueron definidos para simplificar la construcción de consultas
- Se usan con frecuencia



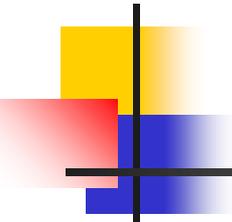
Operadores Tradicionales sobre Conjuntos

- Para todos, con excepción del producto cartesiano, join natural, join condicional y división:
 - Las dos relaciones operandos deben tener el mismo grado
 - Los j -ésimos atributos de las dos relaciones (j en el rango de 1 a n) deben tener el mismo dominio subyacente



Selección: σ

- Operador unario: $\sigma_c: R(N) \rightarrow R(N)$
- $S = \sigma_c (R)$: el resultado una relación S cuyas tuplas son el subconjunto de tuplas de R que cumplen con una determinada condición c
 - c es una expresión booleana sobre los atributos de la relación
- R puede ser una expresión algebraica relacional (cuyo resultado es una relación)
- Es una partición horizontal de R en 2 conjuntos de tuplas –las que cumplen y son seleccionadas y las que no cumplen y son descartadas –



Selección: σ (continuación)

- Es una operación conmutativa
 - $\sigma_{c_1} (\sigma_{c_2} (R)) = \sigma_{c_2} (\sigma_{c_1} (R))$
 - Por lo tanto, una secuencia de SELECTs anidados puede ser evaluada en cualquier orden
- Adicionalmente, podemos remplazar una secuencia de SELECTs anidados por un único SELECT cuya condición es la conjunción de las condiciones de cada SELECT individual
 - $\sigma_{c_1} (\sigma_{c_2} (\dots (\sigma_{c_n} (R)) \dots)) = \sigma_{c_1 \text{ AND } c_2 \text{ AND } \dots \text{ AND } c_n} (R)$

Selección: Ejemplo

Ingenieros

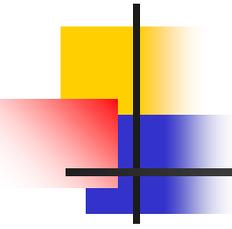
E#	Nombre	Edad
320	José	34
322	Rosa	37
323	María	25

$\sigma_{\text{edad}>35}(\text{Ingenieros})$

E#	Nombre	Edad
322	Rosa	37

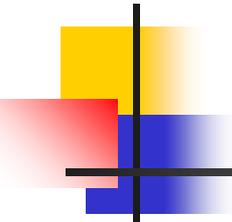
$\sigma_{\text{edad}>45}(\text{Ingenieros})$

E#	Nombre	Edad



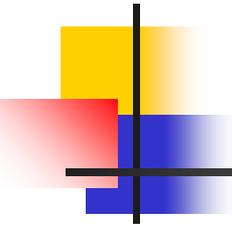
Selección: Ejercicios

- Dado el esquema de la figura 3.7 (página 75) y el estado de la figura 3.6 (página 72) de Elmasri-Navate (6ta. edición), resolver:
 - Seleccionar los empleados que trabajan en el departamento número 4
 - Seleccionar los empleados cuyo salario es mayor que 30 mil pesos
 - Seleccionar los empleados que, o bien trabajan en el departamento 4 y ganan más que 25 mil pesos, o trabajan en el departamento 5 y ganan más de 30 mil pesos



Proyección: π

- Operador unario: $\pi_{\langle a_1 \dots a_m \rangle}: R(N) \rightarrow R(M), M \leq N$
- $S = \pi_{\langle a_1 \dots a_m \rangle} (R)$: el resultado una relación S cuyos atributos, $\langle a_1 \dots a_m \rangle$, son un subconjunto de los de R y cuyas tuplas son todas las de R , salvo que haya duplicadas (se eliminan cuando $\langle a_1 \dots a_m \rangle$ no es una superclave de R)
- El orden de los atributos de S es el de $\langle a_1 \dots a_m \rangle$
- R puede ser una expresión algebraica relacional (cuyo resultado es una relación)
- Es una partición vertical de R en 2 conjuntos de tuplas –uno con los atributos seleccionados y otro con los descartados –



Proyección: π (continuación)

- Adicionalmente, la expresión $\pi_{\langle \text{list}_1 \rangle} (\pi_{\langle \text{list}_2 \rangle} (R)) = \pi_{\langle \text{list}_1 \rangle} (R)$ es válida siempre que $\langle \text{list}_1 \rangle \subseteq \langle \text{list}_2 \rangle$

Proyección: Ejemplo

Ingenieros

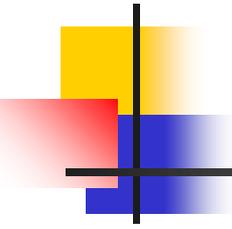
E#	Nombre	Edad
320	José	34
322	Rosa	37
323	María	25
326	José	29

$\pi_{\text{Nombre, Edad}}$ (Ingenieros)

Nombre	Edad
José	34
Rosa	37
María	25
José	29

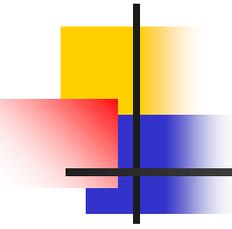
π_{Nombre} (Ingenieros)

Nombre
José
Rosa
María



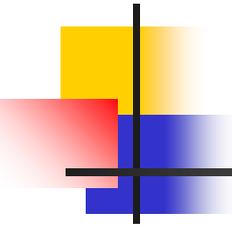
Proyección: Ejercicios

- Dado el esquema de la figura 3.7 (página 75) y el estado de la figura 3.6 (página 72) de Elmasri-Navate (6ta. edición), resolver:
 - Obtener una relación que contenga nombre y apellido de todos los empleados
 - Obtener una relación que contenga número y nombre de todos los departamentos
 - Obtener una relación que contenga número, nombre y ubicación de todos los proyectos



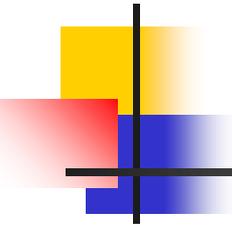
Rename: ρ

- Operador unario: $\rho_{S(a_1 \dots a_n)}: R(N) \rightarrow R(N)$
- $\rho_{S(B_1, B_2, \dots, B_n)}(R)$
 - Renombra la relación y sus atributos
- $\rho_S(R)$
 - Renombra sólo la relación
- $\rho_{(B_1, B_2, \dots, B_n)}(R)$
 - Renombra sólo los atributos



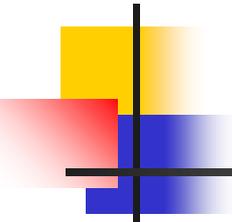
Unión: U

- Operador binario: $U: R(N) \times R(N) \rightarrow R(N)$
- $T = R \cup S$: el resultado es una relación T que incluye todas las tuplas de R o todas las de S o las de ambas, excluyendo repeticiones
- R y S deben ser *type compatible*
 - R y S deben tener el mismo grado n
 - (i.e.: misma cantidad de atributos)
 - $\text{dom}(R_i) = \text{dom}(S_i)$, para $1 \leq i \leq n$
 - (i.e.: cada par de atributos correspondientes deben pertenecer al mismo dominio)



Unión: \cup (continuación)

- Es una operación conmutativa
 - $R \cup S = S \cup R$
- Es una operación asociativa
 - $R \cup (S \cup T) = (R \cup S) \cup T$



Unión: Ejemplo

Ingenieros

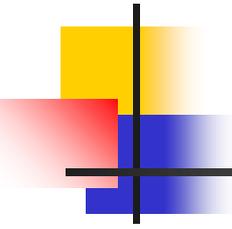
E#	Nombre	Edad
320	José	34
322	Rosa	37
323	María	25

Jefes

E#	Nombre	Edad
320	José	34
421	Jorge	48

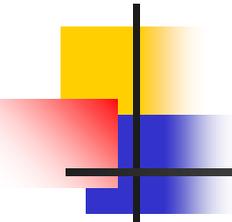
Ingenieros U Jefes

E#	Nombre	Edad
320	José	34
322	Rosa	37
323	María	25
421	Jorge	48



Unión: Ejercicios

- Dado el esquema de la figura 3.7 (página 75) y el estado de la figura 3.6 (página 72) de Elmasri-Navate (6ta. edición), resolver:
 - Obtener los *Social Security numbers* de los empleados que, o bien trabajan en el departamento 5, o bien los que supervisan a empleados que trabajan en el departamento 5



Diferencia: $-$

- Operador binario: $-: R(N) \times R(N) \rightarrow R(N)$
- $T = R - S$: el resultado es una relación T que incluye todas las tuplas que pertenecen a R y NO a S
- R y S deben ser *type compatible*
 - R y S deben tener el mismo grado n
 - (i.e.: misma cantidad de atributos)
 - $\text{dom}(R_i) = \text{dom}(S_i)$, para $1 \leq i \leq n$
 - (i.e.: cada par de atributos correspondientes deben pertenecer al mismo dominio)
- NO es una operación conmutativa
 - $R - S \neq S - R$

Diferencia: Ejemplo

Ingenieros

E#	Nombre	Edad
320	José	34
322	Rosa	37
323	María	25

Jefes

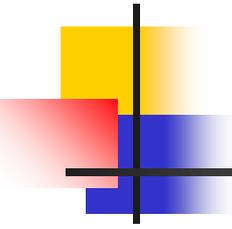
E#	Nombre	Edad
320	José	34
421	Jorge	48

Ingenieros - Jefes

E#	Nombre	Edad
322	Rosa	37
323	María	25

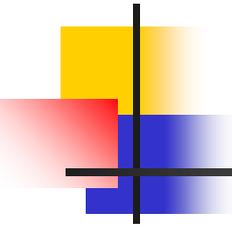
Jefes - Ingenieros

E#	Nombre	Edad
421	Jorge	48



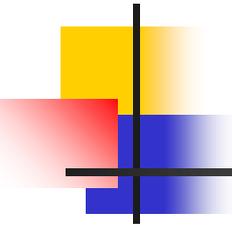
Diferencia: Ejercicios

- Dado el esquema de la figura 3.7 (página 75) y el estado de la figura 3.6 (página 72) de Elmasri-Navate (6ta. edición), resolver:
 - Obtener los *Social Security numbers* de todos los empleados que no son supervisores



Intersección: \cap

- Operador binario: $\cap: R(N) \times R(N) \rightarrow R(N)$
- $T = R \cap S$: el resultado es una relación T que incluye todas las tuplas que pertenecen a la vez a R y a S
- R y S deben ser *type compatible*
 - R y S deben tener el mismo grado n
 - (i.e.: misma cantidad de atributos)
 - $\text{dom}(R_i) = \text{dom}(S_i)$, para $1 \leq i \leq n$
 - (i.e.: cada par de atributos correspondientes deben pertenecer al mismo dominio)



Intersección: Continuación

- Equivalencia con operadores básicos
 - $A \cap B \equiv A - (A - B)$
- Es una operación conmutativa
 - $R \cap S = S \cap R$
- Es una operación asociativa
 - $R \cap (S \cap T) = (R \cap S) \cap T$

Intersección: Ejemplo

Ingenieros

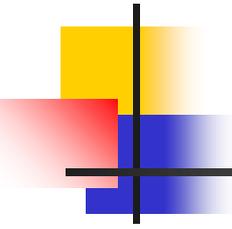
E#	Nombre	Edad
320	José	34
322	Rosa	37
323	María	25

Jefes

E#	Nombre	Edad
320	José	34
421	Jorge	48

Ingenieros \cap Jefes

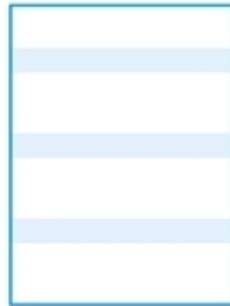
E#	Nombre	Edad
320	José	34



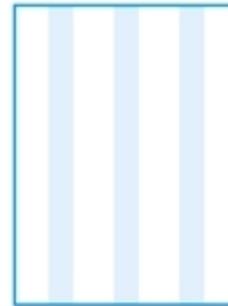
Intersección: Ejercicios

- Dado el esquema de la figura 3.7 (página 75) y el estado de la figura 3.6 (página 72) de Elmasri-Navate (6ta. edición), resolver:
 - Obtener los *Social Security numbers* de los empleados que trabajan en el departamento 5 y que supervisan a empleados que trabajan en el departamento 5

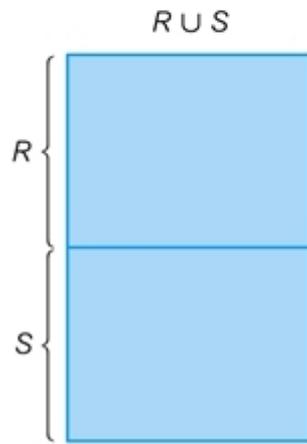
Operadores: Vista esquemática



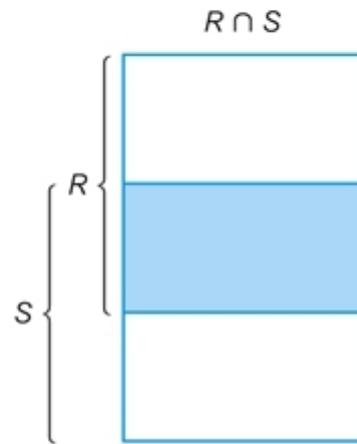
(a) Selection



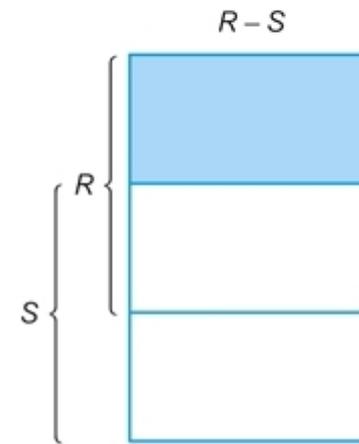
(b) Projection



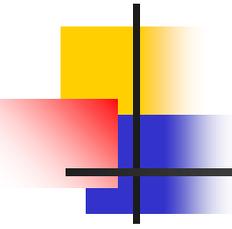
(c) Union



(d) Intersection



(e) Set difference



Producto Cartesiano: X

- Operador binario: $X: R(N) \times R(M) \rightarrow R(N + M)$
- $T = R X S$: el resultado es una relación T que incluye todas las combinaciones posibles de cada tupla de R con cada tupla de S , y sus atributos corresponden a los de R seguidos por los de S
- NO es una operación conmutativa
 - $R X S \neq S X R$
- Es una operación asociativa
 - $R X (S X T) = (S X R) X T$

Producto Cartesiano: Ejemplo

Ingenieros

E#	Nombre	D#
320	José	D1
322	Rosa	D3

Departamentos

D#	Descripcion
D1	Central
D3	I + D

Proyectos

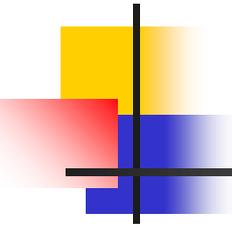
Proyecto	Tiempo
RX338A	21
PY254Z	32

Ingenieros X Proyectos

E#	Nombre	D#	Proyecto	Tiempo
320	José	D1	RX338A	21
320	José	D1	PY254Z	32
322	Rosa	D3	RX338A	21
322	Rosa	D3	PY254Z	32

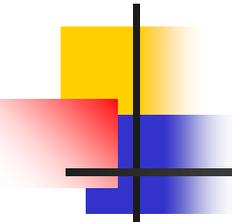
Ingenieros X Departamentos

E#	Nombre	D#	DD	Descripcion
320	José	D1	D1	Central
320	José	D1	D3	I + D
322	Rosa	D3	D1	Central
322	Rosa	D3	D3	I + D



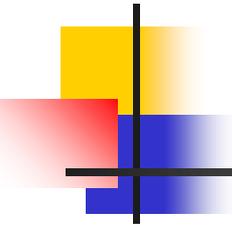
Producto Cartesiano: Ejercicios

- Dado el esquema de la figura 3.7 (página 75) y el estado de la figura 3.6 (página 72) de Elmasri-Navate (6ta. edición), resolver:
 - Obtener los nombres y la relación de los familiares de empleadas mujeres



Join natural: \bowtie

- Operador binario: $\bowtie R(N) \times R(M) \rightarrow R(N + M - 1)$
- $T = R \bowtie S$: el resultado es una relación T que incluye todas las combinaciones posibles de cada tupla de R con cada tupla de S , siempre que la combinación satisfaga que el atributo de R con el mismo nombre que el atributo de S tengan el mismo valor.
 - Debe haber al menos un atributo con el mismo nombre entre ambas relaciones
- Equivalencia
 - $R \bowtie_{\langle \text{condition} \rangle} S \equiv \sigma_{\langle \text{condition} \rangle} (R \times S)$



Join natural: $|X|$ (continuación)

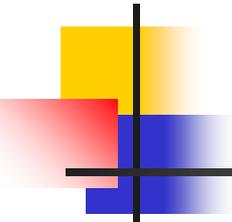
- NO es una operación conmutativa
 - $R|X|S \neq S|X|R$
- Es una operación asociativa
 - $R|X|(S|X|T) = (S|X|R)|X|T$

Join natural: Ejemplos

Departamento			
Nombre_Dept	Num_Dept	Id_Jefe	Fecha_Inicio_Jefe
Dirección	1	1122	12-8-95
Desarrollo	2	5594	25-5-97
Ventas	3	2234	12-9-2001
Compras	4	3355	5-6-2003

Localización_Departamento	
Num_Dept	Localización
1	Quilmes
2	Buenos Aires
2	Córdoba
2	Rosario
0	Catamarca
0	Mendoza

Departamento X Localización_Departamento				
Nombre_Dept	Num_Dept	Id_Jefe	Fecha_Inicio_Jefe	Localización
Dirección	1	1122	12-8-95	Quilmes
Desarrollo	2	5594	25-5-97	Buenos Aires
Desarrollo	2	5594	25-5-97	Córdoba
Desarrollo	2	5594	25-5-97	Rosario

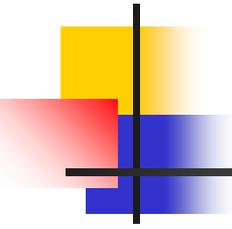


Join natural: Ejercicios

- Dado el esquema de la figura 3.7 (página 75) y el estado de la figura 3.6 (página 72) de Elmasri-Navate (6ta. edición), resolver:
 - Obtener el nombre de los departamentos ubicados en Houston.
 - Obtener los nombres de los familiares de los empleados que trabajan en el proyecto número 1.

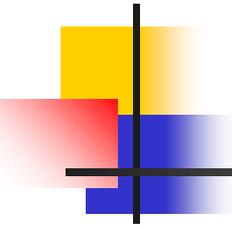
Join condicional: \bowtie

- Operador binario: $\bowtie R(N) \times R(M) \rightarrow R(N \cup M)$
- $T = R \bowtie_{\langle \text{condition} \rangle} S$: el resultado es una relación T que incluye todas las combinaciones posibles de cada tupla de R con cada tupla de S , siempre que la combinación satisfaga la condición del join; sus atributos corresponden a los de R seguidos por los de S
 - La condición del JOIN es de la forma $\langle \text{condition}_1 \rangle \text{ AND } \dots \text{ AND } \langle \text{condition}_n \rangle$, donde cada $\langle \text{condition} \rangle$ es de la forma $\langle A_i \beta B_j \rangle$; A_i es un atributo de R , B_j es un atributo de S , A_i y B_j tienen el mismo dominio y β (beta) es uno de los operadores de comparación $\{=, <, \leq, >, \geq, \neq\}$
 - NO debe haber atributos con el mismo nombre
- Equivalencia
 - $R \bowtie_{\langle \text{condition} \rangle} S \equiv \sigma_{\langle \text{condition} \rangle} (R \times S)$



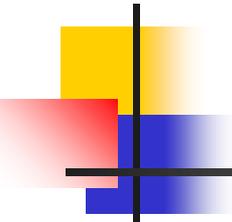
Join condicional: $|X|$ (continuación)

- NO es una operación conmutativa
 - $R|X| S \neq S|X| R$
- Es una operación asociativa
 - $R|X|_{\langle \text{condition} \rangle} (S|X|_{\langle \text{condition} \rangle} T) = (S|X|_{\langle \text{condition} \rangle} R)|X|_{\langle \text{condition} \rangle} T$



Join condicional: Ejercicios

- Dado el esquema de la figura 3.7 (página 75) y el estado de la figura 3.6 (página 72) de Elmasri-Navate (6ta. edición), resolver:
 - Obtener los nombres de los familiares de empleadas mujeres
 - Obtener los nombres de los gerentes de cada departamento



Outer-Join

- Un Outer-Join devuelve todas las tuplas que devolvería un Join y agrega las tuplas que no cumplen con la condición del mismo. Asigna NULL a los valores que no tienen correspondencia en el Join
 - **Left-Outer-Join:** $R \text{ /* } \langle \text{condition} \rangle S$, sólo agrega las tuplas de la relación de la izquierda (**R**) que no cumplen con $\langle \text{condition} \rangle$
 - **Right-Outer-Join:** $R \text{ */ } \langle \text{condition} \rangle S$, sólo agrega las tuplas de la relación de la derecha (**S**) que no cumplen con $\langle \text{condition} \rangle$
 - **Full-Outer-Join:** $R \text{ /*/ } \langle \text{condition} \rangle S$, agrega las tuplas de ambas relaciones que no cumplen con $\langle \text{condition} \rangle$

Left-Outer-Join: Ejemplo

Departamento			
Nombre_Dept	Num_Dept	Id_Jefe	Fecha_Inicio_Jefe
Dirección	1	1122	12-8-95
Desarrollo	2	5594	25-5-97
Ventas	3	2234	12-9-2001
Compras	4	3355	5-6-2003

Localización_Departamento	
Número_Dept	Localización
1	Quilmes
2	Buenos Aires
2	Córdoba
2	Rosario
0	Catamarca
0	Mendoza

Departamento /* Departamento.Num_Dept = Localización_Departamento.Número_Dept Localización_Departamento					
Nombre_Dept	Num_Dept	Id_Jefe	Fecha_Inicio_Jefe	Número_Dept	Localización
Dirección	1	1122	12-8-95	1	Quilmes
Desarrollo	2	5594	25-5-97	2	Buenos Aires
Desarrollo	2	5594	25-5-97	2	Córdoba
Desarrollo	2	5594	25-5-97	2	Rosario
Ventas	3	2234	12-9-2001	null	null
Compras	4	3355	5-6-2003	null	null

Right-Outer-Join: Ejemplo

Departamento			
Nombre_Dept	Num_Dept	Id_Jefe	Fecha_Inicio_Jefe
Dirección	1	1122	12-8-95
Desarrollo	2	5594	25-5-97
Ventas	3	2234	12-9-2001
Compras	4	3355	5-6-2003

Localización_Departamento	
Número_Dept	Localización
1	Quilmes
2	Buenos Aires
2	Córdoba
2	Rosario
0	Catamarca
0	Mendoza

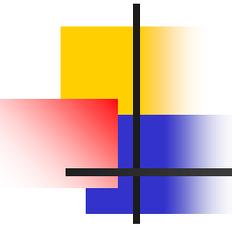
Departamento */ Departamento.Num_Dept = Localización_Departamento.Número_Dept Localización_Departamento					
Nombre_Dept	Num_Dept	Id_Jefe	Fecha_Inicio_Jefe	Número_Dept	Localización
Dirección	1	1122	12-8-95	1	Quilmes
Desarrollo	2	5594	25-5-97	2	Buenos Aires
Desarrollo	2	5594	25-5-97	2	Córdoba
Desarrollo	2	5594	25-5-97	2	Rosario
null	null	null	null	0	Catamarca
null	null	null	null	0	Mendoza

Full-Outer-Join: Ejemplo

Departamento			
Nombre_Dept	Num_Dept	Id_Jefe	Fecha_Inicio_Jefe
Dirección	1	1122	12-8-95
Desarrollo	2	5594	25-5-97
Ventas	3	2234	12-9-2001
Compras	4	3355	5-6-2003

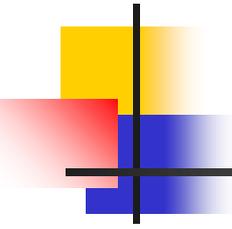
Localización_Departamento	
Número_Dept	Localización
1	Quilmes
2	Buenos Aires
2	Córdoba
2	Rosario
0	Catamarca
0	Mendoza

Departamento /*/ <small>Departamento.Num_Dept = Localización_Departamento.Número_Dept</small> Localización_Departamento					
Nombre_Dept	Num_Dept	Id_Jefe	Fecha_Inicio_Jefe	Número_Dept	Localización
Dirección	1	1122	12-8-95	1	Quilmes
Desarrollo	2	5594	25-5-97	2	Buenos Aires
Desarrollo	2	5594	25-5-97	2	Córdoba
Desarrollo	2	5594	25-5-97	2	Rosario
Ventas	3	2234	12-9-2001	null	null
Compras	4	3355	5-6-2003	null	null
null	null	null	null	0	Catamarca
null	null	null	null	0	Mendoza



División

- El operador de división (\div) define una relación sobre el conjunto de atributos C, incluido en la relación A, y que contiene el conjunto de valores de C, que en las tuplas de A están combinadas con cada una de las tuplas de B



División

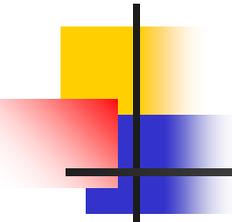
- Condiciones

- $\text{grado}(A) > \text{grado}(B)$
- conjunto atributos de B \subset conjunto de atributos de A

- Equivalencia

- $X_1 = \pi_c(A) \quad X_2 = \pi_c((B \times X_1) - A)$

$$X = X_1 - X_2$$



División: Ejemplo

R1

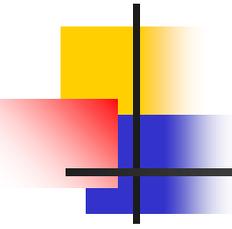
E#	Proyecto
320	RX338A
320	PY254Z
323	RX338A
323	PY254Z
323	NC168T
324	NC168T
324	KT556B

R2

Proyecto
RX338A
PY254Z

R1 / R2

E#
320
323



AR: Ejercicios combinados

- Dado el esquema de la figura 3.7 (página 75) y el estado de la figura 3.6 (página 72) de Elmasri-Navate (6ta. edición), resolver:
 - Obtener nombre y domicilio de todos los empleados que trabajan en el departamento de *Research*
 - Obtener nombres de todos los empleados que no tienen dependientes
 - Obtener nombres de todos los gerentes que tienen por lo menos un dependiente