

# Capítulo 3: Álgebra Relacional

Dr. José Torres Jiménez

Laboratorio de Tecnologías de Información  
Centro de Investigación y de Estudios Avanzados del IPN

Cinvestav-Tamaulipas



# Outline

- 1 El Modelo Relacional
- 2 Grupos de Operadores del Álgebra Relacional
  - Tradicionales de Conjuntos
  - Operadores Especiales de Relaciones
  - Nuevos Operadores Relacionales
- 3 Operadores Relacionales
  - Unión
  - Intersección
  - Diferencia
  - Producto Cartesiano
  - Selección
  - Proyección
  - Join
  - División
  - Operadores básicos
  - Ejercicios de Álgebra relacional
  - Operadores Torres



## El modelo relacional

El modelo relacional consta de tres partes:

- Estructura de datos relacional.
- Reglas de Integridad referencial y de entidad.
- Parte manipulativa que consta de:
  - Álgebra relacional
  - El operador de asignación



S#	SNAME	STATUS	CITY
S1	Smith	20	London
S2	Jones	10	París
S3	Blake	30	París
S4	Clark	20	London
S5	Adams	30	Athens

Tabla 1: Tabla S

P#	PNAME	COLOR	WEIGHT	CITY
P1	Nut	Red	12	London
P2	Bolt	Green	17	París
P3	Screw	Blue	17	Rome
P4	Screw	Red	14	London
P5	Cam	Blue	12	París

Tabla 2: Tabla P

S#	P#	QTY
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P2	200
S4	P4	300
S4	P5	400

Tabla 3: Tabla SP

# Outline

- 1 El Modelo Relacional
- 2 Grupos de Operadores del Álgebra Relacional
  - Tradicionales de Conjuntos
  - Operadores Especiales de Relaciones
  - Nuevos Operadores Relacionales
- 3 Operadores Relacionales
  - Unión
  - Intersección
  - Diferencia
  - Producto Cartesiano
  - Selección
  - Proyección
  - Join
  - División
  - Operadores básicos
  - Ejercicios de Álgebra relacional
  - Operadores Torres



## Tradicionales de subconjuntos

- UNIÓN (*UNION*)
- INTERSECCIÓN (*INTERSECT*)
- DIFERENCIA (*MINUS*)
- PRODUCTO CARTESIANO (*TIMES*)



## Operadores especiales de relaciones

- SELECCIÓN (*WHERE*)
- PROYECCIÓN ([ ])
- JUNTURA (*JOIN*)
- DIVISIÓN (*DIVIDEDBY*)



## Nuevos operadores relacionales

Adicionalmente se han definido algunas extensiones al álgebra relacional entre ellas [TORRES], se contemplan los siguientes operadores:

- DIFERENCIAL
- INTEGRAL
- MÁXIMO
- MÍNIMO
- CUENTA
- SUMA

# Outline

- 1 El Modelo Relacional
- 2 Grupos de Operadores del Álgebra Relacional
  - Tradicionales de Conjuntos
  - Operadores Especiales de Relaciones
  - Nuevos Operadores Relacionales
- 3 Operadores Relacionales
  - Unión
  - Intersección
  - Diferencia
  - Producto Cartesiano
  - Selección
  - Proyección
  - Join
  - División
  - Operadores básicos
  - Ejercicios de Álgebra relacional
  - Operadores Torres



## Unión

La unión de dos relaciones  $A$  y  $B$  que deben ser compatibles a la unión es el conjunto de tuplos que pertenecen a la relación  $A$ , a la relación  $B$  o a ambas relaciones, y se designa por:

$A \textbf{ UNION } B$

## Compatibilidad de la unión

Dos relaciones son compatibles a la unión si tienen el mismo número de atributos(es decir son del mismo grado), y deben existir atributos equivalentes dentro de las dos relaciones, es decir:

El atributo 1 de la relación  $A$  debe estar definido en el mismo dominio del atributo 1 de la relación  $B$ , el atributo 2 de la relación  $A$  debe estar definido en el mismo dominio del atributo 2 de la relación  $B$ , y así sucesivamente.

Gráficamente se verá como se ilustra en la Figura 1.



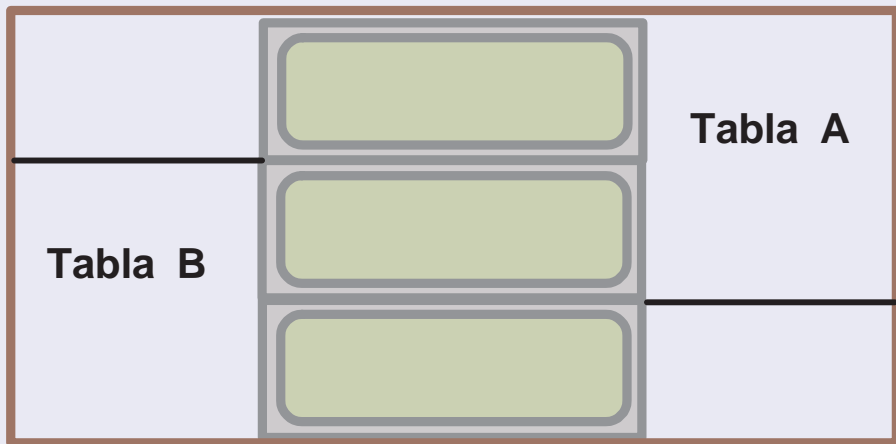


Figura 1: Representación gráfica de la unión de dos tablas

#EMP	SUELDO
E25	10
E30	20
E15	40

Tabla 4: Tabla de Casados

EMP#	SAL
E70	30
E60	40
E85	90

Tabla 5: Tabla de Solteros

EMP#	SAL
E25	10
E30	20
E15	40
E70	30
E60	40
E85	90

Tabla 6: Tabla de Todos



*CASADOS UNION SOLTEROS = TODOS*

*SOLTEROS UNION TODOS = TODOS*

El resultado de la unión conserva los nombres de los atributos de la primer relación.



## Intersección

La intersección de dos relaciones  $A$  y  $B$  que deben ser compatibles a la unión, es el conjunto de tuplos que pertenecen a la relación  $A$  y a la relación  $B$ . Gráficamente esto se verá como se ilustra en la Figura 2.



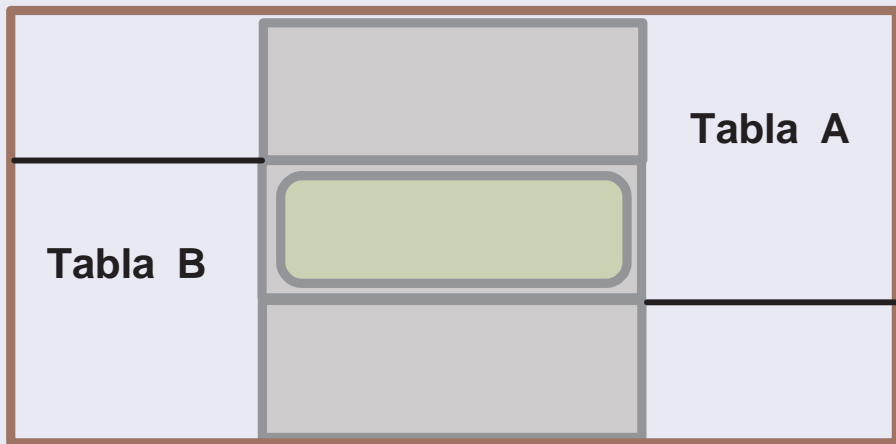


Figura 2: Intersección de dos tablas

Con respecto a la base de datos presentada anteriormente tenemos:

*CASADOS INTERSECT SOLTEROS = VACIO*

*CASADOS INTERSECT TODOS = CASADOS*

## Diferencia

La diferencia de dos relaciones  $A$  y  $B$  que deben ser compatibles a la unión, es el conjunto de tuplos que pertenecen a la relación  $A$  y no a la relación  $B$ . La representación gráfica es ilustrada en la Figura 3.



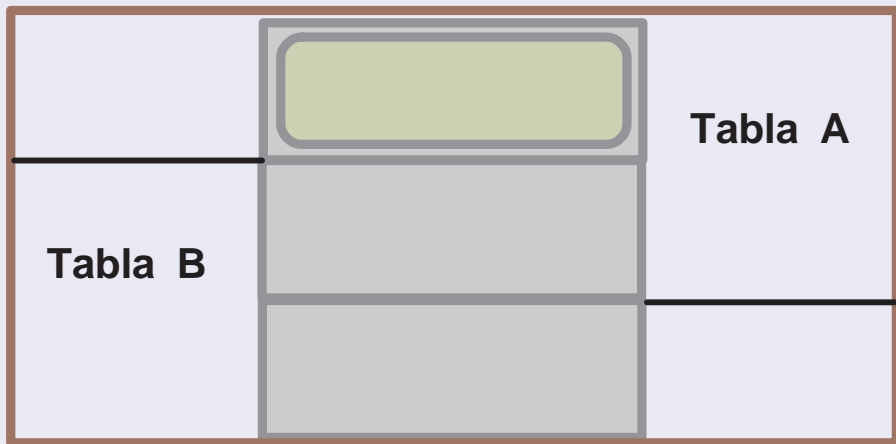


Figura 3: Representación gráfica de la diferencia de tablas

Con respecto a la base de datos presentada anteriormente tenemos:

*CASADOS MINUS TODOS = VACIO*

*TODOS MINUS CASADOS = SOLTEROS*



## Producto cartesiano

El producto cartesiano de dos relaciones  $A$  y  $B$  ( $A$  **TIMES**  $B$ ) es el conjunto de tuplos que resultan de la concatenación de un tuplo de  $A$  con un tuplo de  $B$ .

v.g. *CASADOS* **TIMES** *SOLTEROS* da como resultado la Tabla 4



#EMP	SUELDO	EMP#	SAL
E25	10	E70	30
E25	10	E60	40
E25	10	E85	90
E30	20	E70	30
E30	20	E60	40
E30	20	E85	90
E15	40	E70	30
E15	40	E60	40
E15	40	E85	90

Tabla 7: Tabla Resultante del Producto Cartesiano

E15 40	E15 40 E70 30	E15 40 E60 40	E15 40 E85 90
E30 20	E30 20 E70 30	E30 20 E60 40	E30 20 E85 90
E25 10	E25 10 E70 30	E25 10 E60 40	E25 10 E85 90
	E70 30	E60 40	E85 90

Figura 4: Representación gráfica del producto cartesiano

De este modo el producto cartesiano de una relación de grado  $G_1$  y con cardinalidad  $C_1$  por una relación de grado  $G_2$  y con cardinalidad  $C_2$  produce una relación de grado  $G_1 + G_2$  y con cardinalidad  $C_1 * C_2$ .

NOTA:

Para poder realizar el producto cartesiano de una relación consigo misma es necesario que definamos un *ALIAS* y conservar la UNICIDAD de los nombres de los atributos. v.g.

```
DEFINE ALIAS XS FOR R  
R TIMES XS
```

NOTA:

De los cuatro operadores anteriores solo la diferencia no es conmutativo.

## Selección

La selección de una relación es un subconjunto horizontal de una relación, en este subconjunto aparecen los tuplos que cumplen alguna condición especificada, gráficamente esto se ve en la Figura 5.



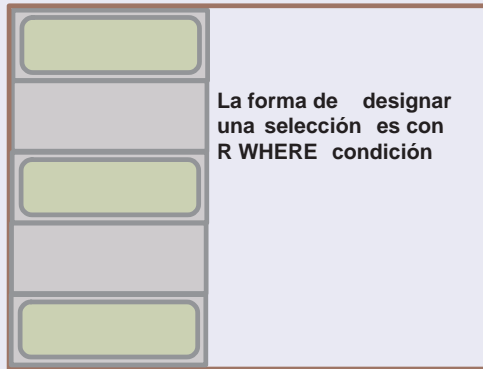


Figura 5: Representación gráfica de la selección de registros de una tabla

**NOTA IMPORTANTE:**

Las siguiente operaciones son equivalentes:

$R \text{ WHERE } C1 \text{ AND } C2 = (R \text{ WHERE } C1) \text{ INTERSECT } (R \text{ WHERE } C2)$

$R \text{ WHERE } C1 \text{ OR } C2 = (R \text{ WHERE } C1) \text{ UNION } (R \text{ WHERE } C2)$

$R \text{ WHERE NOT } C1 = R \text{ MINUS } (R \text{ WHERE } C1)$



## Proyección

La proyección de una relación es un subconjunto vertical con la eliminación de duplicados. Esto se ilustra en la tabla proye.

*		*		
*		*		
*		*		
*		*		
*		*		
*		*		
*		*		

Tabla 8: Tabla Indicando la Proyección

La forma de definir la proyección es encerrando entre paréntesis cuadrados y separados por comas los campos que se desean proyectar:  
(*S* **TIMES** *P*) [*STATUS*, *P.CITY*]



## Join

Es equivalente a un producto cartesiano, seguido de una selección de los tuplos que tengan en los atributos “equivalentes” el mismo valor, y finalmente una proyección para eliminar los atributos duplicados.

Una forma de definirlo será:

$$A \textbf{ JOIN } B = ((A \textbf{ TIMES } B) \textbf{ WHERE } A.C_i = B.C_i \\ \textbf{ AND } \dots A.C_j = B.C_j) \\ [A.A_1, \dots A.A_n, B.B_1, \dots B.B_m]$$


El *JOIN* que se manejará será el conocido como *EQUIJOIN* o *JOIN NATURAL* en el que la condición de la selección es por igualdad, además se han definido otros tipos de *JOIN* cuando la condición involucrada no es por igualdad.

Dadas las tablas 1 y 2 se obtiene lo indicado en la Tabla 9

S#	SNAME	STATUS	CITY	P#	PNAME	COLOR	WEIGHT
S1	Smith	20	London	P1	Nut	Red	12
S1	Smith	20	London	P4	Screw	Red	14
S1	Smith	20	London	P6	Cog	Red	19
S2	Jones	10	París	P2	Bolt	Green	17
S2	Jones	10	París	P5	Cam	Blue	12
S3	Blake	30	París	P2	Bolt	Green	17
S3	Blake	30	París	P5	Cam	Blue	12
S4	Clark	20	London	P1	Nut	Red	12
S4	Clark	20	London	P4	Screw	Red	14
S4	Clark	20	London	P6	Cog	Red	19

Tabla 9: Tabla Resultante  $S \text{ JOIN } P$

## División

La división de una relación  $A$  de grado  $m + n$  entre una relación  $B$  de grado  $n$ , produce una relación  $C$  de grado  $m$ .

Además para poderse realizar la división se debe cumplir que el  $(m + i)$ ésimo atributo de la relación  $A$  este definido en el mismo dominio que el  $i$ ésimo atributo de la relación  $B$ .

Los tuplos resultantes en la relación  $C$  (Cociente) son aquellos atributos  $m$  tales que aparezcan combinados en  $A$  con todos los valores de  $B$ .



v.g. Si tenemos las tablas:

S#	P#
S1	P1
S1	P2
S1	P3
S1	P4
S1	P5
S1	P6
S2	P1
S2	P2
S3	P2
S4	P2
S4	P4
S4	P5

Tabla 10: Tabla X

P#
P1

Tabla 11: Tabla Y

P#
P2
P4

Tabla 12: Tabla Z

P#
P1
P2
P3
P4
P5
P6

Tabla 13: Tabla W



S#
S1
S2

Tabla 14:  $X \text{ DIVIDEDBY } Y$ 

S#
S1
S4

Tabla 15:  $X \text{ DIVIDEDBY } Z$

S#
S1

Tabla 16:  $X \text{ DIVIDEDBY } W$

## NOTAS:

Respecto a la división es importante tener presente que:

$$RESIDUO = DIVIDENDO \textbf{ MINUS } (COCIENTE \textbf{ TIMES } DIVISOR)$$
$$DIVIDENDO = ((COCIENTE \textbf{ TIMES } DIVISOR) \textbf{ UNION } RESIDUO)$$

Si consideramos que el dividendo consta de atributos  $X$ ,  $Y$  y el divisor consta de atributos  $Y$ , una forma de calcular la división será:

$$COCIENTE = DIVIDENDO[X]$$
$$\textbf{ MINUS }$$
$$((DIVIDENDO[X] \textbf{ TIMES } DIVISOR) \textbf{ MINUS } DIVIDENDO)[X]$$


## Operadores básicos

De los 8 operadores vistos solo 5 de ellos son básicos puesto que los otros tres pueden ser definidos en función de los básicos. Los operadores no básicos son:

- **JOIN** Cuya definición ya fue dada.
- **DIVISIÓN** Cuya definición ya fue dada.
- **INTERSECCIÓN** Que equivale a:  $A \text{ MINUS } (A \text{ MINUS } B)$



## Ejercicio 1

Respecto a la base de datos de partes( $P$ ), proveedores( $S$ ) y pedidos( $SP$ ), obtenga las consultas en **ÁLGEBRA RELACIONAL**:

- 1 Obtener los nombres de los proveedores que suministran todas las partes.  
 $((SP[S\#, P\#] \text{ DIVIDEDBY } P[P\#]) \text{ JOIN } S)[SNAME]$
- 2 Obtener los números de proveedor que suministran al menos una parte que sea suministrada por un proveedor que suministra alguna parte de color rojo (RED).  
 $((((P \text{ WHERE } COLOR = 'RED') [P\#] \text{ JOIN } SP)[S\#] \text{ JOIN } SP)[P\#] \text{ JOIN } SP)[S\#]$
- 3 Obtener las parejas de Nombre de Proveedor y Nombre de Parte tales que el proveedor y la parte tengan la misma ciudad.  
 $(S \text{ JOIN } P)[SNAME, PNAME]$

## Continuación

4.- Obtener los números de proveedor que suministran al menos las partes que son suministradas por *S2*.

***SP[S#, P#] DIVIDEDBY (SP WHERE S# = 'S2')[P#]***

5.- Obtener los pedidos en los que el proveedor sea de 'LONDON' o la parte sea de 'PARIS'.

***((S TIMES P TIMES SP) WHERE S.S# = SP.S# AND P.P# = SP.P# AND (P.CITY = 'PARIS' OR S.CITY = 'LONDON'))[SP.S#, SP.P#, QTY]***

## Limitaciones del álgebra relacional

La principal limitación del álgebra relacional reside en que no es posible contestar, por lo menos en forma directa, a preguntas que involucren:

- Obtener la suma de atributos.
- Contar el número de tuplos que cumplan una condición.
- Obtener los tuplos que tengan el valor mínimo respecto a algún(os) atributo(s).
- Obtener los tuplos que tengan el valor máximo respecto a algún(os) atributo(s).
- Obtener algún(os) atributo(s) sí aparecen exactamente  $n$  veces.
- Obtener algún(os) atributo(s) sí aparecen mas de  $n$  veces.
- Obtener algún(os) atributo(s) sí aparecen menos de  $n$  veces.
- Obtener algún(os) atributo(s) sí aparecen por lo menos  $n$  veces.

Se definirán ahora seis operadores nuevos del álgebra relacional que permiten extender su poder de recuperación. Considerando que tenemos una tabla  $A$  que consta de atributos  $X + Y$  donde  $X$  e  $Y$  pueden ser compuestos.



## Mínimo (*MIN*)

Permite obtener un subconjunto horizontal de la tabla *A* de forma que los tuplos seleccionados serán aquellos que tengan el valor mínimo respecto al atributo *X*.

$MIN(A, X) = \text{ALIAS } B \text{ FOR } A$

$A \text{ MINUS } ((A \text{ TIMES } B) \text{ WHERE } A.X > B.X))[A.*]$

Nota:  $A.*$  denota todos los atributos de la tabla *A*.



## Máximo (*MAX*)

Permite obtener un subconjunto horizontal de la tabla *A* de forma que los tuplos seleccionados serán aquellos que tengan el valor máximo respecto al atributo *X*.

$MAX(A, X) = \text{ALIAS } B \text{ FOR } A$   
 $A - ((A \text{ TIMES } B) \text{ WHERE } A.X < B.X))[A. * ]$



## Diferencial (*DIF*)

Permite obtener un subconjunto horizontal de la tabla  $A$ , de forma tal que se elimina de  $A$  un tuplo para cada valor diferente del atributo  $X$ . El tuplo que se elimina es el que tenga el menor valor respecto al atributo  $Y$ .

**DIF**( $A, X$ ) = **ALIAS**  $B$  *FOR*  $A$   
**(( $A$  **TIMES**  $B$ ) **WHERE**  $A.X = B.X$  **AND**  $A.Y > B.Y$ ) $[A.*]$**



## Notas sobre el operador diferencial

Es posible definir diferentes versiones del operador diferencial cambiando la condición que compara los atributos  $Y$  de las tablas  $A$  y  $B$ , de la misma forma que se han definido diferentes versiones del *JOIN*. Además combinando DIFERENCIALES de diferente grado es posible obtener resultados muy interesantes.



## Diferencial enésima

$$\mathbf{DIF}^0(A, X) = A$$

$$\mathbf{DIF}^i(A, X) = \mathbf{DIF}(\mathbf{DIF}^{i-1}(A, X), X)$$

OBTENER LOS TUPLOS QUE APARECEN POR LO MENOS  $n$  VECES

$$\mathbf{DIF}^{n-1}(A, X)$$

OBTENER LOS TUPLOS QUE APARECEN EXACTAMENTE  $n$  VECES

$$\mathbf{DIF}^{n-1}(A, X) \text{ MINUS } \mathbf{DIF}^n(A, X)$$

OBTENER LOS TUPLOS QUE APARECEN MENOS DE  $n$  VECES

$$A \text{ MINUS } \mathbf{DIF}^{n-1}(A, X)$$

## Integral (*INT*)

Es el operador inverso del diferencial, produce una tabla que contiene un tuplo adicional para cada valor diferente de  $X$  el valor del nuevo tuplo respecto a  $Y$  será NULO. Además se tendrá un tuplo completamente NULO (tanto en  $X$  como en  $Y$ ) pero esto violaría la regla de integridad de ENTIDAD.

**$INT(A, X) = A \text{ UNION } (A[X] \text{ TIMES } NULA1) \text{ UNION } NULA2$**

Nota:

La tabla *NULA1* Consta de los  $Y$  atributos todos con valor NULO.

La tabla *NULA2* Consta de un tuplo de  $X + Y$  atributos todos con valor nulo.



## Cuenta

Permite contar el número de tuplos de una tabla.



## Suma (*SUMA*)

Permite obtener la suma de algún campo indicado.

De los seis operadores el operador integral es el que menos se ha utilizado pero se podrá manejar quizás en aplicaciones donde se manejen valores desconocidos o se maneje incertidumbre.





## Aspectos importantes

Un detalle muy importante de los primeros 4 nuevos operadores es que no son básicos es decir que pueden ser definidos en función de los 5 básicos (unión, diferencia, producto cartesiano, selección, proyección). Los últimos dos si son básicos.



## Ejemplos

1.- Obtener los números de parte suministradas por lo menos tres veces.

**DIF<sup>2</sup>**(SP, P#)[P#]

SELECT P#

FROM SP

GROUP BY p#

HAVING COUNT(\*) > 2);



## Ejemplos

2.- Obtener los nombres de partes que sean suministradas por lo menos por dos proveedores que están en 'LONDON'

```
DIF((((S WHERE CITY = 'LONDON') [S#] JOIN SP)[S#, P#], P#)[P#] JOIN P)  
[PNAME]
```

```
SELECT PNAME FROM P WHERE P# IN  
(SELECT A.P#  
FROM SP A, SP B  
WHERE A.P#=B.P#  
AND A.S#<B.S#)  
SELECT PNAME FROM P WHERE P# IN  
(SELECT P#  
FROM SP  
GROUP BY P#  
HAVING COUNT(*) >= 2)
```

## Ejemplos

3.- Obtener los nombres de proveedores que suministran los pedidos con la menor cantidad.

```
((MIN(SP, QTY)[QTY] JOIN SP)[S#] JOIN S)[SNAME]
```

```
SELECT SNAME FROM S WHERE S# IN (  
SELECT S# FROM SP WHERE QTY IN (  
SELECT MIN(QTY) FROM SP))
```

```
SELECT SNAME FROM S WHERE S# IN (  
SELECT S# FROM SP WHERE QTY IN (  
SELECT QTY FROM SP
```

*MINUS*

```
SELECT A.QTY FROM SP A, SP B,  
WHERE A.QTY > B.QTY))
```

## Ejemplos

4.- Obtener los números de proveedores que suministran por lo menos dos partes 'BLUE'.

```
DIF(((P WHERE COLOR = 'BLUE')[P#] JOIN SPJ), S#)[S#]
```



## Ejemplos

5.- Obtener parejas de nombres de proveedor y nombres de partes, tales que el proveedor suministre el pedido más pequeño de cada parte.

```
((SP[P#, QTY] MINUSDIF(SP[P#, QTY], P#)))JOIN SP JOIN S[S#, SNAME]  
JOIN P[P#, PNAME])[SNAME, PNAME]
```



## Ejemplos

6.- Definir el operador de división utilizando el operador diferencial:

Solución:

Si tenemos que la tabla dividendo es *DENDO* y consta de atributos  $X + Y$  y la tabla divisor es *SOR* y consta de atributos  $Y$ , y tanto  $X$  como  $Y$  pueden ser atributos compuestos, tenemos:

$$DENDO \text{ DIVIDEDBY } SOR = \text{DIF}^{CUENTA(SOR)-1}(DENDO \text{ JOIN } SOR, X)$$



## Ejercicio 1

RESPECTO A LA BASE DE DATOS DE EMPLEADOS-CURSOS-INCRITOS OBTENGA LAS CONSULTAS EN ALGEBRA RELACIONAL Y SQL.





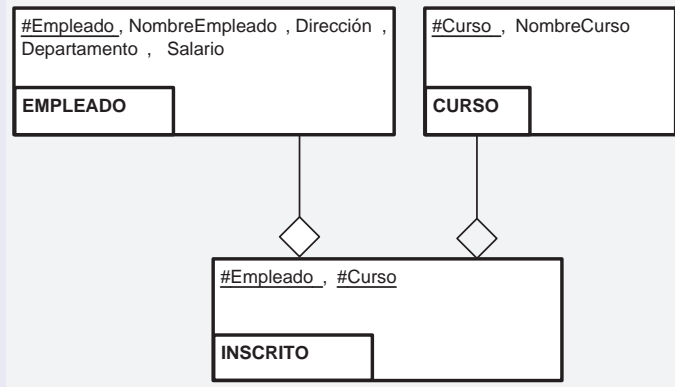


Figura 6: Base de Datos de empleados y cursos

- ❶ Los nombres de empleados que han tomado exactamente tres cursos.
- ❷ Los nombres de cursos que no han sido tomados por ningún empleado.
- ❸ Los números de empleados que no han tomado ningún curso.
- ❹ Los nombres de empleados que han tomado todos los cursos.
- ❺ Los nombres de cursos que han sido tomados por todos los empleados.
- ❻ Los números de empleados que han tomado por lo menos todos los cursos que han sido tomados por el empleado con número E78.
- ❼ Los nombres de cursos que han sido tomados por más de 3 empleados que tengan el mayor salario.
- ❽ Los nombres de cursos que han sido tomados por los empleados con menor salario.

## Ejercicio 2

RESPECTO A LA BASE DE DATOS DE CONTROL ESCOLAR DE LA UNIVERSIDAD OBTENGA LAS CONSULTAS EN ALGEBRA RELACIONAL



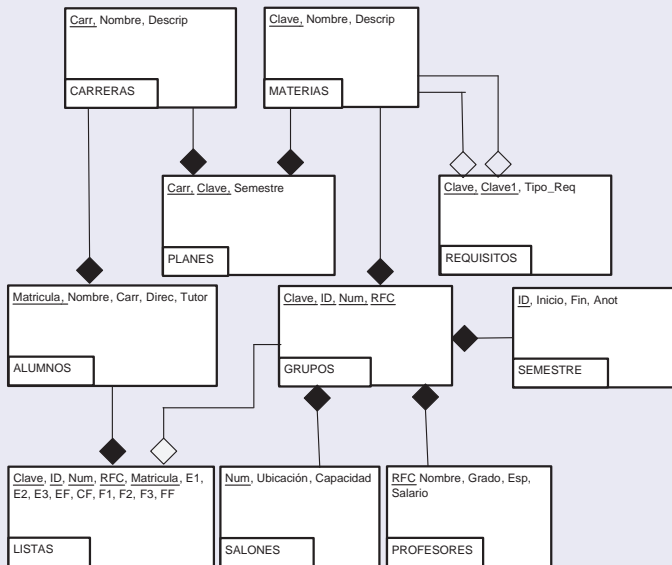


Figura 7: Base de Datos de una universidad

- ❶ Obtener los nombres de los alumnos que han reprobado más de 10 materias.
- ❷ Obtener los nombres de los profesores que imparten por lo menos tres materias.
- ❸ Las materias que sean requisito de dos o más materias.
- ❹ Parejas de carreras tales que tengan en común por lo menos una materia.

