

Algebra relacional

Con un presente dominado por los avances de la IA generativa y el análisis de inmensos volúmenes de datos para crear conocimiento y decisión, en los más variados ámbitos...



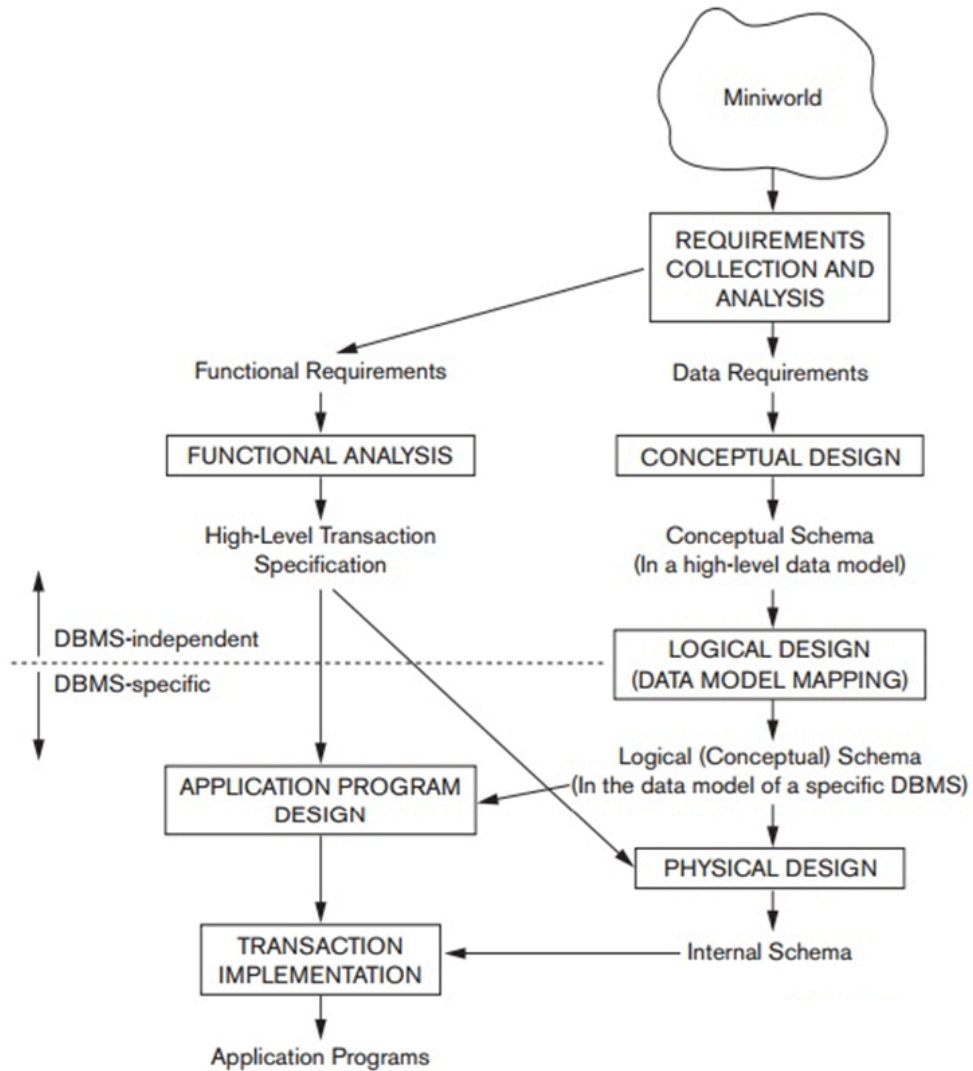
¿Cómo un modelo creado en los años 70 ha sentado las bases para el impacto más significativo en el mundo de la informática de la gestión de datos?

Objetivo: Revisión del modelo relacional. Diagrama ER. Profundizar en operadores más complejos como unión natural, theta join, equi join y en operadores de totales como COUNT, SUM, AVG, MIN, MAX, aplicándolos en ejercicios prácticos de álgebra relacional.

...descubriendo el poder de los datos



Modelo Relacional



FASE 1: Recopilación y análisis de requisitos

FASE 2: Diseño conceptual de la BD

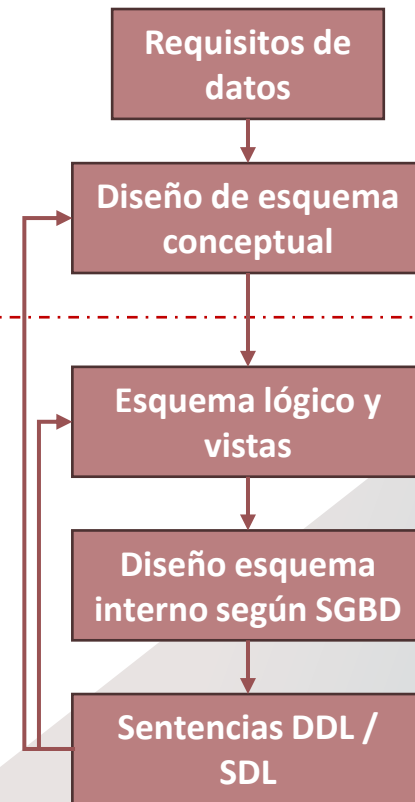
FASE 3: Elección de un SGBD

FASE 4: Mapeo del modelo de datos (diseño lógico)

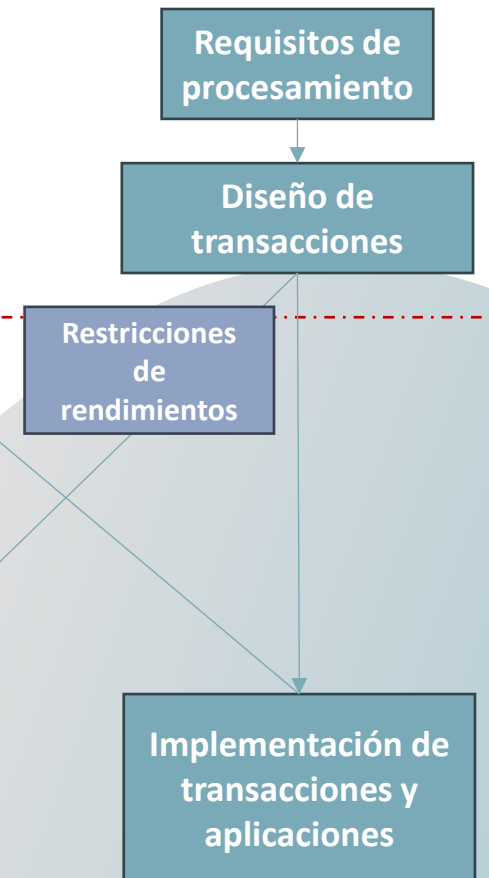
FASE 5: Diseño Físico

FASE 6: Implementación y puesta a punto del sistema

Contenido de datos y estructura



Aplicaciones de bases de datos





Restricciones modelo relacional



Resumen de las Principales Restricciones:

1. **Dominio:** Controla los tipos de datos permitidos.
2. **Clave Primaria:** Asegura que cada fila sea identificada de manera única.
3. **Clave Foránea:** Mantiene la integridad referencial entre tablas.
4. **Integridad de Entidad:** La clave primaria no puede ser `NULL`.
5. **Integridad Referencial:** Mantiene la consistencia en las relaciones entre tablas.
6. **Unicidad:** Asegura valores únicos en un conjunto de atributos.
7. **CHECK:** Impone condiciones sobre los valores permitidos.
8. **NOT NULL:** Evita que una columna tenga valores `NULL`.
9. **DEFAULT:** Define valores por defecto para columnas.

Modelo Relacional



Ejercicio: Sistema de Gestión Hospitalaria

Proceso Analizado: Un hospital cuenta con un sistema de gestión para el manejo de pacientes, doctores, consultas y tratamientos. Los pacientes pueden ser atendidos por doctores especializados en diferentes áreas, y los doctores prescriben tratamientos a los pacientes en función de sus diagnósticos.

Consignas:

1. Identificación de Entidades: Define las entidades relevantes, sus atributos, y las claves primarias y foráneas.
2. Relaciones entre Entidades: Define las relaciones entre las entidades, asignando las cardinalidades y restricciones adecuadas.
3. Diagrama ER
4. Diagrama BD



- Independencia física
- Independencia lógica
- Flexibilidad
- Uniformidad
- Sencillez

Algebra relacional



RELACION (TABLA) CLIENTES

ATRIBUTOS (CABECERA)

ID	CUIT	RAZON SOCIAL	DIRECCION	EMAIL
2152	30225308322	Martinez Herreria	San Martin 445	Martinezh@mh.com
1352	30051255823	Gentiles Ferrum	Montañeses 5001	Adm@gferrum.com
6523	23210004453	STI Soluciones	Congreso 1599	supply@sti.com

TUPLAS (FILAS)

CARDINALIDAD

- Relación 1-1
- Relación 1-N o N-1
- Relación N-M

DOMINIOS

CLAVES CANDIDATAS

CLAVES PRIMARIAS

CLAVES EXTERNAS

SUPERCLAVE

MANIPULACION DE DATOS

La manipulación en el modelo relacional se puede realizar mediante:

- **Algebra Relacional:** especifica consultas y manipula datos mediante operaciones como selección, proyección, unión y diferencia, sobre relaciones (tablas). Se centra en la aplicación secuencial de operaciones para obtener resultados deseados, ofreciendo una forma concisa y directa de expresar las instrucciones.
- **Cálculo Relacional:** expresa las consultas en términos de predicados lógicos o fórmulas, especificando las condiciones que deben cumplir las tuplas seleccionadas y centrando la lógica para cumplir con las tuplas de la relación resultante.

MODELO DE DATOS:

Conjunto de herramientas conceptuales que permiten describir los datos, sus relaciones, límites de integridad que les afectan y terminología a emplear



SELECCION

Algebra relacional



Sea R una relación y θ una condición $\sigma_{\theta}(R)$

da como resultado otra relación con esquema igual que el de R y con instancia igual el conjunto de tuplas de las instancias de R que cumplen con la condición θ .

- **Seleccionar empleados mayores de 30 años**

Consulta $\sigma_{\text{edad} > 30}(\text{EMPLEADOS})$

emp_ID	nombre	edad	dpt_ID
3	Pedro	40	101

- **Seleccionar empleados del departamento 101 menor a 35 años**

Consulta $\sigma_{\text{edad} < 35 \text{ AND } \text{dpt_ID} = 101}(\text{EMPLEADOS})$

emp_ID	nombre	edad	dpt_ID
1	Juan	30	101

El operador Selección devuelve una nueva relación que contiene solo las tuplas de la relación original que **satisfacen una condición** dada.

E M P L E A D O S	emp_ID	nombre	edad	dpt_ID
	1	Juan	30	101
	2	Mariela	25	102
	3	Pedro	40	101
	4	Martin	28	103

Dentro de la condición de selección podemos incorporar distintos tipos de operadores:

- Operadores relacionales ($>$, $<$, \leq , \geq , $=$)
- Operadores lógicos (AND, OR, NOT):
- Paréntesis:
- $(A1 > 3 \text{ AND } A2 = 'b') \text{ OR } (A1 = 3 \text{ AND } A2 = 'a')$

La selección no modifica la estructura de la relación (los atributos), solo se filtran las tuplas.

\cup	UNION
$-$	DIFERENCIA
\times	PRODUCTO CARTESIANO
σ	SELECCIÓN / RESTRICCIÓN
π	PROYECCIÓN
\cap	INTERSECCIÓN
\bowtie	COMBINACIÓN / JOIN
$/$	DIVISION
ρ	RENOMBRAR



SELECCION

Algebra relacional



Selecciona todos los clientes de Madrid

σ Ciudad = 'Madrid' (CLIENTES)

Selecciona todos los clientes de Madrid y mayores a 30

σ Ciudad = 'Madrid' AND Edad > 30 (CLIENTES)

Selecciona todos los clientes de Madrid y mayores a 30 o menores de 25

σ Ciudad = 'Madrid' AND (Edad > 30 OR Edad < 25) (CLIENTES)

C
L
I
E
N
T
E
S

Id_Cliente	nombre	Ciudad	Edad
1	Juan Pérez	Madrid	34
2	Ana García	Barcelona	28
3	Luis Martínez	Madrid	42
4	Marta López	Valencia	23
5	José Díaz	Sevilla	30



PROYECCION \pi

Algebra relacional



El operador Proyección devuelve una nueva relación que contiene solo los atributos (columnas) especificados de la relación original.

Permite obtener tuplas con un cierto conjunto de atributos

$$\pi_{\langle \text{ATRIBUTOS} \rangle}(\mathbf{R})$$

donde $\langle \text{ATRIBUTOS} \rangle$ (A1,A2,A3..An) es una lista de atributos a aparecer en la relación resultado.

- **Proyectar los nombres y edades de los empleados**

Consulta $\pi_{\text{nombre, edad}}(\text{EMPLEADOS})$

nombre	edad
Juan	30
Mariela	25
Pedro	40
Martin	28

- **Uso combinado: Seleccionar empleados mayores de 30 años y proyectar solo nombres y edades**

Consulta $\pi_{\text{nombre, edad}}(\sigma_{\text{edad} > 30}(\text{EMPLEADOS}))$

nombre	edad
Pedro	40

EMPLEADOS

emp_ID	nombre	edad	dpt_ID
1	Juan	30	101
2	Mariela	25	102
3	Pedro	40	101
4	Martin	28	103

Eliminación de Duplicados:

La proyección elimina las tuplas duplicadas en el resultado, asegurando que cada combinación de valores de los atributos proyectados sea única.

Reducción de Atributos:

Solo se incluyen en el resultado los atributos especificados, todas las demás columnas de la relación original se omiten.

∪	UNION
−	DIFERENCIA
×	PRODUCTO CARTESIANO
σ	SELECCIÓN / RESTRICCIÓN
π	PROYECCIÓN
∩	INTERSECCIÓN
⋈	COMBINACIÓN / JOIN
/	DIVISION
ρ	RENOMBRAR



PROYECCION π

Algebra relacional



Proyecta las columnas Nombre, Ciudad y Edad para obtener una lista de todas los nombres, ciudades y edades de los clientes.

π nombre, ciudad, edad (CLIENTES)

nombre	Ciudad	Edad
Juan Pérez	Madrid	34
Ana García	Barcelona	28
Luis Martínez	Madrid	42
Marta López	Valencia	23
José Díaz	Sevilla	30

C
L
I
E
N
T
E
S

Id_Cliente	nombre	Ciudad	Edad
1	Juan Pérez	Madrid	34
2	Ana García	Barcelona	28
3	Luis Martínez	Madrid	42
4	Marta López	Valencia	23
5	José Díaz	Sevilla	30



UNION $R \cup S$ (o R UNION S): incluye todas las tuplas que **están en R, en S o en ambas**, eliminando duplicados.

Algebra relacional



Las operaciones de unión, intersección y diferencia del álgebra relacional, las relaciones deben tener la misma estructura. Esto significa que ambas relaciones deben tener:

- **El mismo número de atributos (columnas).**
- **Tipos de datos compatibles para los atributos correspondientes – mismos Dominios**

R UNION S	
A1	A2
a	1
a	2
b	3
b	4



DIFERENCIA RESTA $R - S$ (o R MINUS S): incluye todas las tuplas que **están en R pero no en S**.

R MINUS S	
A1	A2
a	1
b	3

S MINUS R	
A1	A2
b	4



INTERSECCION

$S \cap R$	
A1	A2
a	2

$R \cap S$ (o R INTERSECTION S): incluye **aquellas tuplas que están a la vez en ambas R y en S**.

P
R
O
P
I
E
D
A
D
E
S

Unión (U):

- **Asociativa:** $R \cup (S \cup T) = (R \cup S) \cup T$
- **Conmutativa:** $R \cup S = S \cup R$
- **Distributiva sobre Intersección:** $R \cup (S \cap T) = (R \cup S) \cap (R \cup T)$

Intersección (∩):

- **Asociativa:** $R \cap (S \cap T) = (R \cap S) \cap T$
- **Conmutativa:** $R \cap S = S \cap R$
- **Distributiva sobre Unión:** $R \cap (S \cup T) = (R \cap S) \cup (R \cap T)$

Diferencia (-):

- **No Asociativa:** $(R - S) - T \neq R - (S - T)$
- **No Conmutativa:** $R - S \neq S - R$
- **Distributiva sobre Unión:** $R - (S \cup T) = (R - S) \cap (R - T)$
- **No Distributiva sobre Intersección:** $R - (S \cap T) \neq (R - S) \cup (R - T)$

R	
A1	A2
a	1
a	2
b	3

S	
A1	A2
a	2
b	4

∪	UNION
-	DIFERENCIA
×	PRODUCTO CARTESIANO
σ	SELECCIÓN / RESTRICCIÓN
π	PROYECCIÓN
∩	INTERSECCIÓN
⋈	COMBINACIÓN / JOIN
/	DIVISION
ρ	RENOMBRAR



Cross Join

Algebra relacional



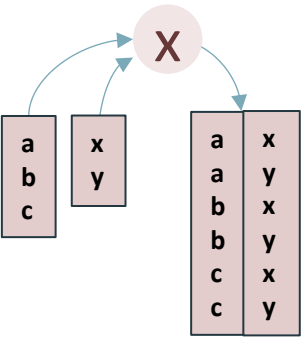
Es una operación binaria que combina todas las tuplas de una relación con todas las tuplas de otra relación, generando una nueva relación que contiene todas las combinaciones posibles de tuplas de las dos relaciones originales.

PRODUCTO CARTESIANO \times

El producto cartesiano de dos relaciones es una nueva relación que contiene todas las combinaciones posibles de tuplas de ambas relaciones.

$(R_1) \times (R_2)$

- El producto cartesiano de Empleados x Departamentos



EMPLEADOS

emp_ID	nombre	dpt_ID
1	Juan	101
2	Mariela	102
3	Pedro	101
4	Martin	103

DPTOS

dpt_ID	desc
101	HR
102	IT
103	Sales

emp_ID	nombre	dpt_ID	dpt_ID	desc
1	Juan	101	101	HR
1	Juan	101	102	IT
1	Juan	101	103	Sales
2	Mariela	102	101	HR
2	Mariela	102	102	IT
2	Mariela	102	103	Sales
3	Pedro	101	101	HR
3	Pedro	101	102	IT
3	Pedro	101	103	Sales
4	Martin	103	101	HR
4	Martin	103	102	IT
4	Martin	103	103	Sales

Cada tupla de la relación R1 se combina con cada tupla de la relación R2.

Si R1 tiene n tuplas y R2 tiene m tuplas, el resultado del producto cartesiano tendrá $n \times m$ tuplas.

Los atributos de las tuplas resultantes son la concatenación de los atributos de R1 y R2.

- \cup UNION
- $-$ DIFERENCIA
- \times PRODUCTO CARTESIANO
- σ SELECCIÓN / RESTRICCIÓN
- π PROYECCIÓN
- \cap INTERSECCIÓN
- \bowtie COMBINACIÓN / JOIN
- $/$ DIVISION
- ρ RENOMBRAR



Combinación
Concatenación
Reunión

Algebra relacional



Theta Join (θ -Join)

Un Theta Join combina las tuplas de dos relaciones donde las tuplas cumplen una condición específica (θ), que puede ser cualquier expresión booleana.

Esta operación es equivalente a:

$$R_1 \bowtie_{\theta} R_2 \quad \sigma_{\theta} ((R_1) \times (R_2))$$

- Combinar las tuplas de Empleados y de Departamentos donde nombre sea igual a Pedro

Consulta $\sigma_{\text{nombre} = \text{'Pedro'}} ((\text{EMPLEADOS}) \times (\text{DPTOS}))$

emp_ID	nombre	dpt_ID	dpt_ID	desc
3	Pedro	101	101	HR
3	Pedro	101	102	IT
3	Pedro	101	103	Sales

El operador Join (\bowtie) en álgebra relacional es una operación fundamental que permite combinar tuplas de dos relaciones basándose en una condición específica. Hay varios tipos de joins y todos ellos relacionan filas de una tabla con filas de otra tabla.

EMPLEADOS

emp_ID	nombre	dpt_ID
1	Juan	101
2	Mariela	102
3	Pedro	101
4	Martin	103

DPTOS

dpt_ID	desc
101	HR
102	IT
103	Sales

Es una relación resultante cuya cantidad de atributos es igual a la suma de los atributos de las relaciones (tablas).

La cantidad de tuplas resultantes contiene una tupla por cada combinación de tuplas de las relaciones que satisfaga una condición de conexión.

La principal diferencia entre el PRODUCTO CARTESIANO y la Theta Join es que en esta última sólo aparecen en el resultado las combinaciones de tuplas que satisfacen la condición de conexión, mientras que en el PRODUCTO CARTESIANO se incluyen todas

\cup	UNION
$-$	DIFERENCIA
\times	PRODUCTO CARTESIANO
σ	SELECCIÓN / RESTRICCIÓN
π	PROYECCIÓN
\cap	INTERSECCIÓN
\bowtie	COMBINACIÓN / JOIN
$/$	DIVISION
ρ	RENOMBRAR



Combinación
Concatenación
Reunión

Equi Join

El Equi Join es un caso especial de Theta Join donde la condición es una igualdad entre columnas de ambas tablas.

Algebra relacional



El operador Join (\bowtie) en álgebra relacional es una operación fundamental que permite combinar tuplas de dos relaciones basándose en una condición específica. Hay varios tipos de joins y todos ellos relacionan filas de una tabla con filas de otra tabla.

$$R_1 \bowtie_{R_1.At = R_2.At} R_2$$

- Combinar las tuplas de Empleados y de Departamentos el ID de departamento sean iguales

Consulta $\sigma_{EMPLEADOS.dpt_ID = DPTOS.dpt_ID} ((EMPLEADOS) \times (DPTOS))$

emp_ID	nombre	dpt_ID	dpt_ID	desc
1	Juan	101	101	HR
2	Mariela	102	102	IT
3	Pedro	101	101	HR
4	Martin	103	103	Sales

E M P L E A D O S	emp_ID	nombre	dpt_ID	D P T O S	dpt_ID	desc
	1	Juan	101		101	HR
	2	Mariela	102		102	IT
	3	Pedro	101		103	Sales
	4	Martin	103			

Es una relación resultante cuya cantidad de atributos es igual a la suma de los atributos de las relaciones (tablas). Tienen atributos (columnas) duplicadas.

La cantidad de tuplas resultantes contiene una tupla por cada combinación de tuplas de las relaciones que satisfaga la igualdad de los mismos atributos conexiónados de ambas relaciones.

Equi Join siempre es un tipo de **Inner Join** porque usa una igualdad para combinar filas, pero este último puede agregar otros tipos de condiciones booleanas para combinar filas y proyección para seleccionar atributos resultantes

$$\pi_{\text{atributos}} \sigma_{\theta} ((R_1) \times (R_2))$$

U	UNION
-	DIFERENCIA
x	PRODUCTO CARTESIANO
σ	SELECCIÓN / RESTRICCIÓN
π	PROYECCIÓN
\cap	INTERSECCIÓN
\bowtie	COMBINACIÓN / JOIN
/	DIVISION
ρ	RENOMBRAR



Combinación
Concatenación
Reunión

Natural Join

El Natural Join es un tipo de Equi Join que elimina las columnas duplicadas en las relaciones combinadas. Combina las relaciones automáticamente basándose en todos los atributos con el mismo nombre.

$$R_1 \bowtie R_2$$

- Combinar las tuplas de Empleados y de Departamentos el ID de departamento sean iguales

Consulta (EMPLEADOS) \bowtie (DPTOS)

emp_ID	nombre	dpt_ID	desc
1	Juan	101	HR
2	Mariela	102	IT
3	Pedro	101	HR
4	Martin	103	Sales

Algebra relacional



El operador Join (\bowtie) en álgebra relacional es una operación fundamental que permite combinar tuplas de dos relaciones basándose en una condición específica. Hay varios tipos de joins y todos ellos relacionan filas de una tabla con filas de otra tabla.

E M P L E A D O S	emp_ID	nombre	dpt_ID	D P T O S	dpt_ID	desc
	1	Juan	101		101	HR
	2	Mariela	102		102	IT
	3	Pedro	101		103	Sales
	4	Martin	103			

Es una relación resultante cuya cantidad de atributos es igual a la suma de los atributos de las relaciones (tablas) sin duplicados.

La cantidad de tuplas resultantes contiene una tupla por cada combinación de tuplas de las relaciones que satisfaga la igualdad de los mismos atributos conexonados de ambas relaciones.

Auto Join o **Self Join** es cuando una tabla se une consigo misma para comparar filas de la misma tabla

$$\sigma_{\theta}((R_1) \times (R_1))$$

∪	UNION
−	DIFERENCIA
×	PRODUCTO CARTESIANO
σ	SELECCIÓN / RESTRICCIÓN
π	PROYECCIÓN
∩	INTERSECCIÓN
⋈	COMBINACIÓN / JOIN
/	DIVISION
ρ	RENOMBRAR



Combinación
Concatenación
Reunión

Outer Join

El Outer Join extiende el resultado del join para incluir todas las tuplas de una relación, y tuplas de la otra relación que no tienen coincidencias se completan con valores nulos.

Left Outer Join: devuelve todas las filas de la R1 izquierda, y las filas coincidentes de la R2 derecha. Si no hay coincidencia, las filas de la R2 derecha contendrán NULL.

$$R_1 \bowtie R_2$$

Consulta (EMPLEADOS) \bowtie (DPTOS)

emp_ID	nombre	dpt_ID	desc
1	Juan	101	HR
2	Mariela	102	IT
3	Pedro	101	HR
4	Martin	NULL	NULL

Right Outer Join: devuelve todas las filas de la R2 derecha, y las filas coincidentes de la R1 izquierda. Si no hay coincidencia, las filas de la R1 izquierda contendrán NULL.

$$R_1 \bowtie R_2$$

Consulta (EMPLEADOS) \bowtie (DPTOS)

emp_ID	nombre	dpt_ID	desc
1	Juan	101	HR
2	Mariela	102	IT
3	Pedro	101	HR
NULL	NULL	103	Sales

Algebra relacional



El operador Join (\bowtie) en álgebra relacional es una operación fundamental que permite combinar tuplas de dos relaciones basándose en una condición específica. Hay varios tipos de joins y todos ellos relacionan filas de una tabla con filas de otra tabla.

EMPLEADOS

emp_ID	nombre	dpt_ID
1	Juan	101
2	Mariela	102
3	Pedro	101
4	Martin	NULL

DPTOS

dpt_ID	desc
101	HR
102	IT
103	Sales

Full Outer Join: devuelve todas las filas cuando hay una coincidencia en una de las tablas y si no hay coincidencia, las partes que no coinciden se completan con NULL.

$$R_1 \bowtie R_2$$

Consulta (EMPLEADOS) \bowtie (DPTOS)

emp_ID	nombre	dpt_ID	desc
1	Juan	101	HR
2	Mariela	102	IT
3	Pedro	101	HR
4	Martin	NULL	NULL
NULL	NULL	103	Sales



Combinación
Concatenación
Reunión

Algebra relacional



El operador Join (\bowtie) en álgebra relacional es una operación fundamental que permite combinar tuplas de dos relaciones basándose en una condición específica. Hay varios tipos de joins y todos ellos relacionan filas de una tabla con filas de otra tabla.

Semi Join

Un Semi Join devuelve las tuplas de la primera relación (tabla) que tienen al menos una coincidencia en la segunda relación. No devuelve ningún atributo (columna) de la segunda relación, solo de la primera.

$$R_1 \bowtie R_2$$

Consulta (EMPLEADOS) \bowtie (DPTOS)

emp_ID	nombre	dpt_ID
1	Juan	101
2	Mariela	102
3	Pedro	101

EMP
LEA
DOS

emp_ID	nombre	dpt_ID
1	Juan	101
2	Mariela	102
3	Pedro	101
4	Martin	NULL

D
PT
OS

dpt_ID	desc
101	HR
102	IT
103	Sales

Anti Join

Un Anti Join devuelve las tuplas de la primera relación que no tienen ninguna coincidencia en la segunda relación. No devuelve ningún atributo (columna) de la segunda relación, solo de la primera.

$$R_1 \nabla R_2$$

Consulta (EMPLEADOS) ∇ (DPTOS)

emp_ID	nombre	dpt_ID
4	Martin	NULL

- \cup UNION
- $-$ DIFERENCIA
- \times PRODUCTO CARTESIANO
- σ SELECCIÓN / RESTRICCIÓN
- π PROYECCIÓN
- \cap INTERSECCIÓN
- \bowtie COMBINACIÓN / JOIN
- $/$ DIVISION
- ρ RENOMBRAR



DIVISION

Algebra relacional



Es una operación binaria que devuelve las tuplas de una relación que están asociadas con todas las tuplas de otra relación. La división se utiliza cuando queremos encontrar todos los elementos de un conjunto que están relacionados con todos los elementos de otro conjunto.

La división se usa para responder consultas del tipo "encuentra todos los R1 que están relacionados con todos los R2".

$$R_1 / R_2$$

emp_ID	nombre
1	Juan
3	Pedro

Paso 1: proyección de los atributos relevantes en Empleados

$\pi_{\text{emp_ID, nombre}}(\text{EMPLEADOS}) \setminus \{1, 2, 3\}$

Paso 2: producto cartesiano de la proyección con Proyectos

$\pi_{\text{emp_ID, nombre}}(\text{EMPLEADOS}) \times (\text{PROYECTOS}) \setminus \{1,101\}, \{1,102\}, \{2,101\} \dots \{3,102\}\}$

E M P L E A D O S	emp_ID	nombre	proj_ID	P R O Y E C T O S	proj_ID
	1	Juan	101		101
	1	Juan	102		102
	2	Mariela	102		
	2	Mariela	103		
	3	Pedro	101		
	3	Pedro	102		
	3	Pedro	103		

Cada tupla de la relación R1 tiene $a_1..a_n; b_1..b_m$ atributos y se relaciona con las todas las tuplas de la relación R2 que tiene $b_1..b_m$ atributos.

El resultado es una relación con $a_1..a_n$ atributos y todas las tuplas de la R1 que estén relacionadas a todas las tuplas de R2.

Paso 3: restamos Empleados del Paso 2

$(\pi_{\text{emp_ID, nombre}}(\text{EMPLEADOS}) \times (\text{PROYECTOS})) - (\text{EMPLEADOS}) \setminus \{2,101\}\}$

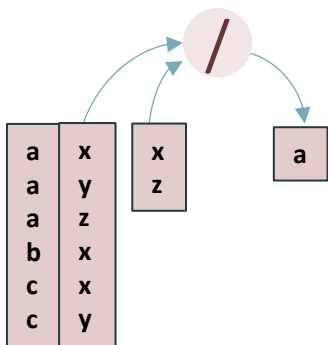
Paso 4: proyectamos los atributos relevantes del Paso 3

$\pi_{\text{emp_ID, nombre}}((\pi_{\text{emp_ID, nombre}}(\text{EMPLEADOS}) \times (\text{PROYECTOS})) - (\text{EMPLEADOS})) \setminus \{2\}\}$

Paso 5: restamos el resultado del Paso 4 del Paso 1

$\pi_{\text{emp_ID, nombre}}(\text{EMPLEADOS}) \sim$

$-\pi_{\text{emp_ID, nombre}}((\pi_{\text{emp_ID, nombre}}(\text{EMPLEADOS}) \times (\text{PROYECTOS})) - (\text{EMPLEADOS})) \setminus \{1,3\}\}$





RENOMBRAR

Algebra relacional



La operación de renombrar puede cambiar el nombre de una relación, los nombres de sus atributos, o ambos. Esto es especialmente importante para evitar conflictos cuando se realizan operaciones con varias relaciones que tienen atributos con nombres similares.

Renombra R1 como R2 siendo los B1,B2,..Bn los nuevos nombres de los atributos. Se puede renombrar solo una relación, solo los atributos o la relación y los atributos.

$$\rho_{R_2(B1,B2,...Bn)}(R_1)$$

- Renombrar la relación EMPLEADOS a STAFF
 $\rho_{STAFF}(EMPLEADOS)$
- Renombrar la relación EMPLEADOS a STAFF y los atributos a (id, empleado, area)
 $\rho_{STAFF(id, empleado, \text{área})}(EMPLEADOS)$

E M P L E A D O S	emp_ID	nombre	dpt_ID
	1	Juan	101
	2	Mariela	102
	3	Pedro	101
	4	Martin	103

D P T O S	dpt_ID	desc
	101	HR
	102	IT
	103	Sales

S T A F F	id	empleado	area
	1	Juan	101
	2	Mariela	102
	3	Pedro	101
	4	Martin	103

∪	UNION
−	DIFERENCIA
×	PRODUCTO CARTESIANO
σ	SELECCIÓN / RESTRICCIÓN
π	PROYECCIÓN
∩	INTERSECCIÓN
⋈	COMBINACIÓN / JOIN
/	DIVISION
ρ	RENOMBRAR



Algebra relacional



OPERADORES DE AGREGACION (TOTALES)

$\gamma_{A,CALC(B)}(R)$

- γ (gamma) representa el operador de agregación.
- **A** es un atributo o un conjunto de atributos por los cuales se agrupa.
- **CALC(B)** es el cálculo del atributo B.
- **R** es la relación sobre la cual se aplica la agregación.

SUM: Suma de las cantidades vendidas por item

$\gamma_{ITEM,SUM(CANTIDAD)}(VENTAS)$

item	Sum(cantidad)
101	5
102	1
103	5

COUNT: Contar el número de ventas por item

$\gamma_{ITEM,COUNT(SALES)}(VENTAS)$

item	Sum(cantidad)
101	2
102	1
103	1

AVG: Promedio del precio de venta por item

$\gamma_{ITEM,COUNT(SALES)}(VENTAS)$

item	Sum(cantidad)
101	10
102	20
103	15

Los operadores de agregación se utilizan para realizar cálculos sobre conjuntos de tuplas y producir resultados resumidos. Estos operadores son fundamentales para realizar tareas como contar tuplas, sumar valores, calcular promedios, encontrar valores máximos y mínimos

VENTAS	sale_ID	item	cantidad	precio
	1	101	2	10
	2	102	1	20
	3	101	3	10
	4	103	5	15

Los operadores de agregación en álgebra relacional (**SUM, COUNT, AVG, MAX, MIN**) permiten realizar cálculos agregados sobre las tuplas de una relación, proporcionando resúmenes valiosos de los datos.

Estos operadores se utilizan junto con la notación γ (gamma) para definir grupos y aplicar funciones de agregación sobre ellos

Otra nomenclatura:

$\gamma_{\text{item}, \text{SUM(cantidad)}}(\text{VENTAS})$

Operadores de Totales en SELECT

Principales operadores de totales:

- **COUNT()**: Cuenta el número de filas que cumplen con una condición.
- **SUM()**: Suma los valores de una columna numérica.
- **AVG()**: Calcula el promedio de una columna numérica.
- **MIN()**: Devuelve el valor mínimo de una columna.
- **MAX()**: Devuelve el valor máximo de una columna.

```
SELECT COUNT(*) AS TotalEmpleados FROM Empleados;
```

```
SELECT SUM(Salario) AS TotalSalarios FROM Empleados;
```

```
SELECT AVG(Salario) AS SalarioPromedio FROM Empleados;
```

```
SELECT MIN(Salario) AS SalarioMinimo, FROM Empleados;
```

```
SELECT MAX(Salario) AS SalarioMaximo FROM Empleados;
```

Uso de operadores de totales con GROUP BY

El uso del operador GROUP BY es fundamental para aplicar agregaciones por categorías.

```
SELECT DeptoID, COUNT(*) AS TotalEmpleados, AVG(Salario) AS SalarioPromedio  
FROM Empleados  
GROUP BY DeptoID;
```



Conceptos clave sobre varrel:

- **Varrel** (variable de relación): Es una tabla cuyos datos (tuplas o filas) pueden ser modificados mediante operaciones como inserciones, eliminaciones o actualizaciones. (J.DALE) – Relación modificable con INSERT, DELETE y UPDATE → SQL
- En teoría relacional, una **relación** es un conjunto inmutable de tuplas, pero en un sistema de base de datos real, las tablas no son inmutables; pueden cambiar, y por eso se les llama varrel, ya que son variables que pueden contener relaciones diferentes en distintos momentos.

Algebra relacional



Ejercicio: Gestión de Hospitales y Pacientes

Tablas:

- Pacientes (id_paciente, nombre_paciente, edad, sexo)
- Doctores (id_doctor, nombre_doctor, especialidad)
- Consultas (id_consulta, id_paciente, id_doctor, fecha_consulta, diagnostico)
- Tratamientos (id_tratamiento, nombre_tratamiento, tipo_tratamiento)
- Prescripciones (id_consulta, id_tratamiento, dosis, frecuencia, duracion_dias)

Consignas:

1. Encuentra los nombres de los pacientes que han sido atendidos por doctores especializados en "Cardiología" y han recibido tratamientos con el tipo "Medicamento".
2. Selecciona los doctores que han recetado un tratamiento con frecuencia mayor a 3 veces al día para pacientes mayores de 60 años.
3. Muestra los nombres de los pacientes que han recibido más de un tratamiento durante una consulta.

Ejercicio de Integración



Ejercicio: Gestión de Ventas en una Empresa Online

Proceso Analizado: Una empresa de ventas online necesita gestionar información sobre clientes, productos, y las órdenes de compra realizadas. Queremos diseñar un sistema para realizar consultas que ayuden a la toma de decisiones, como encontrar clientes que han gastado más de una cantidad específica, analizar ventas por productos, y descubrir productos que no han sido comprados.

Consulta 1: Encuentra los nombres de los clientes que han gastado más de 1000 en total.

Consulta 2: Encuentra los productos más vendidos, aquellos que han sido comprados más de 50 veces.

Consulta 3: Encuentra los productos que no han sido comprados.

Práctica conceptual



Ejercicio de tablas: Gestión de Ventas

CLIENTES

ClienteID	Nombre	Ciudad
1	Juan	Madrid
2	María	Barcelona
3	Pedro	Valencia

PRODUCTOS

ProductoID	NombreProducto	Precio
101	Televisor	500
102	Portátil	1000
103	Tablet	300

VENTAS

VentaID	ClienteID	ProductoID	Cantidad	Fecha
1	1	101	2	15/1/2024
2	2	102	1	16/1/2024
3	3	103	3	17/1/2024
4	1	102	1	18/1/2024
5	1	101	1	1/2/2024
6	2	103	2	3/2/2024
7	3	102	1	5/2/2024
8	1	103	1	10/3/2024
9	3	101	2	15/3/2024
10	2	102	2	20/3/2024

¿Cuál fue el cliente que compró más productos en el mes de enero de 2024?

Debes obtener:

- la expresión en álgebra relacional,
- el resultado final y
- el orden de ejecución.

Modelo relacional



Bibliografía:

Introducción a los Sistemas de Bases de Datos - C. J. Date – 7º Edición

- **CAPÍTULO 3: Una introducción a las bases de datos relacionales.**
- **CAPÍTULO 5: Dominios, relaciones y varrels base.**

Sistemas de Bases de Datos – Conceptos Fundamentales - Elmasri / Navathe – 5º Edición.

- **Capítulo 5: El modelo de datos relacional y las restricciones de una base de datos relacional.**
- **Capítulo 6: El álgebra relacional y los cálculos relacionales.**

Próximo encuentro



Primer Parcial → Módulo 1 y 2

Fundamentos y Conceptos Básicos

Creación de estructuras, Modelo relacional y Algebra Relacional

Módulo 3: SQL y Seguridad

Vistas: creación y usos. Funciones del sistema.

Reutilización de consultas y la creación de abstracciones de datos
que mejoran la seguridad y la eficiencia.

INGENIERÍA DE DATOS I – Prof. Alfonso Fernández Buttera