

Ejercicios - Unidad IV – Álgebra Relacional

Grupo 8

Integrantes:

- ★ María Leonor Albornoz
- ★ Maximiliano Hernán Cocito
- ★ Fabricio Víctor Kinweiler
- ★ Mauricio Nicolás Schefer
- ★ Yamil Apas Moselli
- ★ Juan Ignacio Velazco Gez Schegtel

Ejercicio 1

Explique la afirmación de que los operadores de álgebra relacional pueden "componerse". ¿Por qué es importante la capacidad de componer operadores?

Los operadores se pueden componer para combinar operaciones y condiciones que nos permitan obtener la información que necesitamos.

También podemos decir...

En el álgebra relacional, la afirmación de que los operadores pueden "componerse" se refiere a la capacidad de encadenar o combinar varios operadores para formar una sola expresión o consulta. Esto significa que el resultado de un operador puede ser la entrada de otro.

Por ejemplo, si tienes dos operadores, **selección (σ)** y **proyección (π)**, puedes "componerlos" de la siguiente manera:

$$\Pi_{\text{atributo}}(\sigma_{\text{condición}}(\text{Relación}))$$

En este caso, primero se aplica el operador de selección a una relación para filtrar las tuplas que cumplen con una cierta condición. Luego, el operador de proyección se aplica al resultado para seleccionar solo ciertos atributos.

Ejercicio 2

Dadas dos relaciones R1 y R2, donde R1 contiene N1 tuplas, R2 contiene N2 tuplas, y $N2 > N1 > 0$, encuentre los tamaños mínimo y máximo posible (en tuplas) para la relación resultante al aplicar cada una de las siguientes expresiones de álgebra relacional. En cada caso, establezca los supuestos necesarios sobre los esquemas de R1 y R2 para que la expresión tenga sentido.

a. $R1 \cup R2$

Para poder ser compatible con la unión, las tablas tienen que tener los mismos atributos o campos. También se puede "forzar" esta compatibilidad haciendo una proyección de los campos correspondientes.

El tamaño mínimo será el de R2 (o sea, N2) si R1 no tiene tuplas cargadas. El tamaño máximo, $N1+N2$.

b. $R1 \cap R2$

El tamaño mínimo será 0, una tabla sin tuplas, porque puede pasar que no se encuentre tuplas "en común". El tamaño máximo será N_1 .

La intersección también debe ser compatible con la unión.

c. $R_1 - R_2$

El tamaño mínimo será 0. Lo que significa que se "sacó" todos las tuplas en R_2 que aparecían también en R_1 . El máximo, N_1 , ergo, no se pudo restar nada.

La diferencia de conjuntos o resta debe ser compatible con la unión.

d. $R_1 \times R_2$

En el producto cartesiano hay que tomar la precaución de cambiar el nombre de los campos de ser necesario. De otra manera, se prohibiría la operación al no poder tener una tabla con nombres de campos repetidos, o, en el mejor de los casos, los nombres que sean iguales quedarían indefinidos en la tabla resultante.

El tamaño mínimo será el de R_2 . Este es el caso en que la tabla R_1 , de tamaño N_1 , tenga una cardinalidad de 1. El tamaño máximo está dado por el producto $N_1 \times N_2$.

e. $\sigma_{a=5}(R_1)$

El tamaño mínimo será 0. Esto pasaría si en la tabla R_1 no hay ningún campo "a" con el valor 5. El tamaño máximo será equivalente a la cantidad de veces que aparezca este valor en la tabla o relación.

f. $\pi_a(R_1)$

Para este ejemplo el tamaño máximo es N_1 . El tamaño mínimo podría ser 1 si el valor del campo "a" es el mismo para toda la relación.

g. R_1/R_2

En el álgebra relacional, la operación R_1/R_2 es una operación de división. Esta operación se utiliza para responder a consultas que incluyen la frase "para todos".

"Dividir" R_1/R_2 devuelve una relación con todos los registros en R_1 que están asociados en R_2 con **todos** los registros de R_2 . En cuanto a los campos de la relación resultante, la operación produce una tabla que consta de aquellos campos de R_1 que no están en R_2 . Es decir, si R_1 tiene los campos {A, B, C} y R_2 tiene los campos {B}, entonces la relación resultante tendrá los campos {A, C}.

Antes de analizar el resultado de la operación, analicemos lo que tiene que suceder para que la misma pueda llevarse a cabo: El grado de la relación R_1 tiene que ser mayor que el de la relación R_2 ; además R_2 debe tener al menos un campo que coincida en denominación y dominio con algún campo de R_1 .

Ahora bien, si respetamos la cardinalidad establecida en la consigna donde $N_2 > N_1$, el tamaño será siempre 0, ya que no resultará nada de la operación. Por ello, analizaremos la operación **R_1/R_2** cuando $N_1 > N_2$:

El tamaño máximo de la relación resultante será N_1 . Esto sucederá si cada registro en R_1 está asociado con todos los registros en R_2 .

El tamaño mínimo de la relación resultante será 0. Esto sucederá si no hay ningún registro en R_1 que esté asociado con todos los registros en R_2 .

Ejercicio 3

Considere el siguiente esquema:

- Proveedores(idp: integer, nombrep: string, direccionp: string)

- Partes(idpa:integer, nombrep: string, color: string)
- Catalogo(idp: integer, idpa: integer, precio: real)

Los campos clave están subrayados y el dominio de cada campo se especifica después del nombre. La relación "Catalogo" lista los precios a los que cada proveedor vende cada parte. Escriba las siguientes consultas en álgebra relacional:

1. Encontrar los nombres de los proveedores que proveen alguna parte roja.

$$\pi \text{ nombrep } (\text{proveedores} \bowtie \sigma \text{ color} == \text{'Red'} (\text{partes}) \bowtie \text{catalogo})$$

2. Encontrar los idps de los proveedores que proveen alguna parte roja o verde.

Opción 1:

$$\pi \text{ idp } (\text{proveedores} \bowtie \sigma \text{ color} == \text{'Red'} \text{ or } \text{color} == \text{'Green'} (\text{partes}) \bowtie \text{catalogo})$$

Opción 2:

$$pRojas = \text{proveedores} \bowtie \text{catalogo} \bowtie \sigma \text{ color} == \text{'Red'} (\text{partes})$$

$$pVerdes = \text{proveedores} \bowtie \text{catalogo} \bowtie \sigma \text{ color} == \text{'Green'} (\text{partes})$$

$$\pi \text{ idp } (pRojas) \cup \pi \text{ idp } (pVerdes)$$

3. Encontrar los idps de los proveedores que proveen alguna parte roja o están en Av. 9 de Julio 2222.

$$\pi \text{ idp } (\sigma \text{ direccionp} == \text{'Av. 9 de Julio 2222'} (\text{proveedores})) \cup \pi \text{ idp } (\sigma \text{ color} == \text{'Red'} (\text{partes}) \bowtie \text{catalogo})$$

4. Encontrar los idps de los proveedores que proveen alguna parte roja y alguna parte verde.

$$pRojas = \text{proveedores} \bowtie \text{catalogo} \bowtie \sigma \text{ color} == \text{'Red'} (\text{partes})$$

$$pVerdes = \text{proveedores} \bowtie \text{catalogo} \bowtie \sigma \text{ color} == \text{'Green'} (\text{partes})$$

$$\pi \text{ idp } (pRojas) \cap \pi \text{ idp } (pVerdes)$$

5. Encontrar los idps de los proveedores que proveen todas las partes.

$$\pi \text{ idp, idpa } (\text{catalogo}) \div \pi \text{ idpa } (\text{partes})$$

Recordar que cuando se nos indica que encontremos "todo" de algo, se suele ocupar la división.

6. Encontrar los idps de los proveedores que proveen todas las partes rojas.

$$\pi \text{ idp, idpa } (\text{catalogo}) \div \pi \text{ idpa } (\sigma \text{ color} == \text{'Red'} (\text{partes}))$$

7. Encontrar los idps de los proveedores que proveen todas las partes rojas o verdes.

Opción 1:

$$\pi \text{idp, idpa}(\text{catalogo}) \div \pi \text{idpa}(\sigma \text{ color} == \text{'Red'}(\text{partes})) \cup \pi \text{idp}(\text{catalogo} \bowtie \sigma \text{ color} == \text{'Green'}(\text{partes}))$$

Opción 2:

$$\begin{aligned} \text{todos_rojos} &= \pi \text{idp, idpa}(\text{catalogo}) \div \pi \text{idpa}(\sigma \text{ color} == \text{'Red'}(\text{partes})) \\ \text{alguna_verde} &= (\text{catalogo} \bowtie \sigma \text{ color} == \text{'Green'}(\text{partes})) \\ \text{todos_rojos} &\cup \pi \text{idp}(\text{alguna_verde}) \end{aligned}$$

8. Encontrar los idps de los proveedores que proveen todas las partes rojas o todas las partes verdes.

$$\begin{aligned} \text{todosRojos} &= \pi \text{idp, idpa}(\text{catalogo}) \div \pi \text{idpa}(\sigma \text{ color} == \text{'Red'}(\text{partes})) \\ \text{todosVerdes} &= \pi \text{idp, idpa}(\text{catalogo}) \div \pi \text{idpa}(\sigma \text{ color} == \text{'Green'}(\text{partes})) \\ \text{todosRojos} &\cup \text{todosVerdes} \end{aligned}$$

9. Encontrar pares de idps tales que el proveedor con el primer idp vende más caro alguna parte que el proveedor con el segundo idp.

$$\begin{aligned} \text{aux} &= \rho \text{idp} \rightarrow \text{idp_aux, idpa} \rightarrow \text{idpa_aux, precio} \rightarrow \text{precio_aux}(\text{catalogo}) \\ \pi \text{idp, idp_aux, idpa, idpa_aux, precio, precio_aux} &(\sigma \text{idp} \neq \text{idp_aux and idpa} == \text{idpa_aux and precio} > \text{precio_aux}(\text{catalogo} * \text{aux})) \end{aligned}$$

Proyecté más campos que los solicitados en la consigna para poder comprobar los resultados rápidamente.

10. Encontrar los idpas de las partes que son provistas por al menos dos proveedores diferentes.

$$\begin{aligned} \text{aux} &= \rho \text{idp} \rightarrow \text{idp_aux, idpa} \rightarrow \text{idpa_aux, precio} \rightarrow \text{precio_aux}(\text{catalogo}) \\ \pi \text{idp, idp_aux, idpa, idpa_aux} &(\sigma \text{idp} \neq \text{idp_aux and idpa} == \text{idpa_aux}(\text{catalogo} * \text{aux})) \end{aligned}$$

11. Encontrar los idpas de las partes que son provistas por el proveedor llamado ["Big Red Tool and Die"](#)).

$$\pi \text{nombrep, idpa}(\sigma \text{nombrep} == \text{"Big Red Tool and Die"}(\text{proveedores}) \bowtie \text{catalogo} \bowtie \text{partes})$$

12. Encontrar los idpas de las partes provistas por todos los proveedores a menos de \$200 (si algún proveedor no provee la parte o la cobra más de \$200, ésta no se selecciona).

$$\pi \text{idpa, idp}(\sigma \text{ precio} < 200(\text{catalogo})) \div \pi \text{idp}(\text{proveedores})$$

Recordar que para obtener proveedores que provean todas las partes (ergo, un idp que "tenga" todas las idpa's), la consulta es:

$$\pi \text{idp, idpa}(\text{catalogo}) \div \pi \text{idpa}(\text{partes})$$

En este punto nos piden lo mismo pero al revés: un idpa que "tenga" todos los idp's.

Ejercicio 4

Considere el esquema Proveedores-Partes-Catálogo del punto anterior. Establezca cuál es el resultado de las siguientes consultas.

Las tablas que utilizaremos en este ejemplo son las siguientes:

Proveedores		
idp	nombrep	direccionp
1	Acme Widget Suppliers	1 Grub St., Potemkin Village, IL 61801
2	Big Red Tool and Die	4 My Way, Bermuda Shorts, OR 90305
3	Perfunctory Parts	99999 Short Pier, Terra Del Fuego, TX 41299
4	Alien Aircraft Inc.	Av. 9 de Julio 2222

Partes		
idpa	nombrep	color
1	Left Handed Bacon Stretcher Cover	Red
2	Smoke Shifter End	Black
3	Acme Widget Washer	Red
4	Acme Widget Washer	Silver
5	I Brake for Crop Circles Sticker	Translucent
6	Anti-Gravity Turbine Generator	Cyan
7	Anti-Gravity Turbine Generator	Magenta
8	Fire Hydrant Cap	Red
9	7 Segment Display	Green

Catálogo		
idp	idpa	precio
1	1	1500.00
1	2	0.50
1	3	11.70
1	4	12.00
1	5	1.00
1	6	12.50
1	7	63.45
1	8	15.00
1	9	12.49
2	3	0.55
2	8	7.95
2	1	16.50
3	8	12.50
3	9	1.00
3	1	190.00
4	5	2.20
4	6	1247548.23
4	7	1247548.23
4	1	14.00
4	8	16.35

a. π nombrep (π idpa (σ color == 'Red' (partes)) \bowtie σ precio < 100 (catalogo) \bowtie proveedores)

Arroja el nombre de los proveedores que proveen partes rojas a menos de \$100.

El resultado de la consulta fue:

nombrep
Acme Widget Suppliers
Big Red Tool and Die
Alien Aircraft Inc.
Perfunctory Parts

b. π nombrep (π idpa (σ color == 'Red' (partes) \bowtie σ precio < 100 (catalogo) \bowtie proveedores))

Primero, dentro del paréntesis donde se hacen los _«natural joins»_ toma las partes rojas que son vendidas a menos de \$100. Para dar esta información no haría falta hacer el _«natural join»_ con la relación de proveedores. Este resultado intermedio es el siguiente:

idpa
1
8
3

Luego, trata de hacer la proyección del campo "nombrep" y falla, ya que el mismo no es parte de la relación.

c. π nombrep (σ color == 'Red' (partes) \bowtie σ precio < 100 (catalogo) \bowtie proveedores) \cap π nombrep (σ color == 'Green' (partes) \bowtie σ precio < 100 (catalogo) \bowtie proveedores)

Devuelve los nombres de los proveedores que tienen tanto partes rojas como verdes a menos de \$100.

El resultado de la consulta fue:

nombrep
Acme Widget Suppliers
Perfunctory Parts

d. π idp (σ color == 'Red' (partes) \bowtie σ precio < 100 (catalogo) \bowtie proveedores) \cap π idp (σ color == 'Green' (partes) \bowtie σ precio < 100 (catalogo) \bowtie proveedores)

Ídem al punto anterior. El _«natural join»_ con la relación o tabla de proveedores está de más ya que esta vez se proyectará el idp, dato que se encuentra en la tabla de catálogos.

e. π nombrep (π idp, nombrep (σ color == 'Red' (partes) \bowtie σ precio < 100 (catalogo) \bowtie proveedores) \cap π idp, nombrep (σ color == 'Green' (partes) \bowtie σ precio < 100 (catalogo) \bowtie proveedores))

La consulta devuelve lo mismo que la consulta del punto c. La diferencia es que primero hace una intersección entre una tabla con proveedores que tienen partes rojas a menos de \$100 con otra que tiene lo mismo pero con partes verdes. Este resultado "intermedio" es:

idp	nombrep
1	Acme Widget Suppliers
3	Perfunctory Parts

Finalmente, sobre esta tabla proyecta el campo "idp", obteniendo así el mismo resultado que en el punto c.

Ejercicio 5

Dado el siguiente esquema relacional:

- flights(flno: integer, origin: string, to: string, distance: integer, departs: time, arrives: time, cost: real)
- certified(aid: integer, aname: string, cruisingrange: integer)
- employees(eid: integer, aid: integer)
- flights(eid: integer, ename: string, salary: real)

Tener en cuenta que la relación *employees* describe tanto pilotos como otras clases de empleados; cada piloto está calificado para algún avión (de otra manera no podría calificar como piloto), y sólo los pilotos están calificados para volar.

Escribir las siguientes consultas en álgebra relacional. Tener en cuenta que no todas pueden ser contestadas. En tales casos explicar por qué.

1. Encuentre los eids de pilotos certificados para algún avión Boeing.

Opción 1:

$$\pi \text{ eid } (\sigma \text{ aname} == \text{'Boeing 727'} (\text{aircraft}) \bowtie \text{certified})$$

Sin embargo, esta opción sólo toma los pilotos que están certificados para volar un Boeing 727.

Para hacer más general la consulta, en Relational podemos hacer lo siguiente:

Opción 2:

$$\pi \text{ eid } (\sigma (\text{aname.startswith('Boeing')}) (\text{aircraft}) \bowtie \text{certified})$$

Esta consulta encuentra el eid de todos los pilotos que pueden volar aviones Boeing.

2. Encuentre los nombres de pilotos certificados para algún avión Boeing.

$$\pi \text{ eid, ename } (\sigma (\text{aname.startswith('Boeing')}) (\text{aircraft}) \bowtie \text{certified} \bowtie \text{employees})$$

3. Encuentre los aids de todos los aviones que pueden ser usados para vuelos sin paradas desde Los Ángeles hasta Sydney.

Voy a proyectar también el nombre del avión para que sea más fácil de controlar el ejercicio.

$$\pi \text{ aid, aname } (\sigma \text{ cruisingrange} > \text{distance} (\text{aircraft} * \sigma \text{ origin} == \text{'Los Angeles'} \text{ and } \text{to} == \text{'Sydney'} (\text{flights})))$$

4. Identifique los vuelos que pueden ser piloteados por cada piloto cuyo salario sea mayor a \$260.000.

$\text{aux1} = (\sigma \text{ salary} > 260000 (\text{employees}) \bowtie \text{certified} \bowtie \text{aircraft})$

$\pi \text{ flno, ename } (\sigma \text{ cruisingrange} > \text{distance} (\text{flights} * \text{aux1}))$

5. Encuentre los nombres de pilotos que pueden operar aviones con un rango mayor a 3.000 km pero que NO esten certificados para los aviones Boeing.

$\text{aux1} = \text{employees} \bowtie \text{certified} \bowtie \sigma \text{ cruisingrange} > 3000 (\text{aircraft})$

$\text{aux2} = \text{employees} \bowtie \text{certified} \bowtie \sigma (\text{aname.startswith}(\text{"Boeing"})) (\text{aircraft})$

$\pi \text{ ename } (\text{aux1}) - \pi \text{ ename } (\text{aux2})$

En **aux1** obtenemos una relación con todos los pilotos que pueden volar aviones con una autonomía mayor a los 3000km, incluidos los Boeing; **aux2** es una tabla con pilotos que están certificados para volar aviones Boeing.

Al hacer una diferencia entre estas dos tablas, obtenemos una nueva tabla con pilotos que no están certificados para pilotar aviones Boeing pero sí pueden volar más de 3000km.

6. Encuentre los eids de empleados que ganan el mayor salario.

$\text{aux} = \rho \text{ eid} \rightarrow \text{eid_aux, ename} \rightarrow \text{ename_aux, salary} \rightarrow \text{salary_aux} (\text{employees})$

$\text{aux1} = \pi \text{ eid, ename, salary } (\sigma \text{ eid} \neq \text{eid_aux and salary} < \text{salary_aux} (\text{employees} * \text{aux}))$

$\pi \text{ ename, eid } (\text{employees} - \text{aux1})$

Primero, hacemos la tabla **aux** para poder hacer el producto cartesiano "entre sí" de la tabla **employees**. Luego, en **aux1** obtenemos una tabla donde se excluyó al empleado que gana el mayor salario. De esta manera, al hacer la diferencia con la tabla original, obtenemos el empleado que más gana al estar "eliminando" a todos los demás.

El nombre lo proyectamos para que sea más fácil comprobar el ejercicio.

7. Encuentre los eids de empleados que ganen el segundo mayor salario.

$\text{aux} = \rho \text{ eid} \rightarrow \text{eid_aux, ename} \rightarrow \text{ename_aux, salary} \rightarrow \text{salary_aux} (\text{employees})$

$\text{aux1} = \pi \text{ eid, ename, salary } (\sigma \text{ eid} \neq \text{eid_aux and salary} < \text{salary_aux} (\text{employees} * \text{aux}))$

$\text{aux} = \rho \text{ eid} \rightarrow \text{eid_aux, ename} \rightarrow \text{ename_aux, salary} \rightarrow \text{salary_aux} (\text{aux1})$

$\text{aux2} = \pi \text{ eid, ename, salary } (\sigma \text{ eid} \neq \text{eid_aux and salary} < \text{salary_aux} (\text{aux1} * \text{aux}))$

$\pi \text{ ename, eid } (\text{aux1} - \text{aux2})$

Este ejercicio sigue la misma lógica que el punto anterior, pero repitiendo el proceso una vez más: Digamos que de una tabla "**A**", obtengo una tabla "**B**" que excluye a la persona que gane el mayor salario de la tabla "**A**". Repito el proceso, pero esta vez de la tabla "**B**" obtengo una tabla "**C**" que excluye a la persona que gane el mayor salario de la tabla "**B**".

Finalmente, la diferencia entre la tabla "B" con la "C", nos devuelve la persona con el mayor salario de la tabla "B", que a su vez es la persona con el segundo mejor salario en la tabla "A", que es lo que se nos pide en el enunciado.

8. Encuentre los eids de empleados que están certificados para la mayor cantidad de aviones.

Este ejercicio no se puede resolver con las herramientas que nos brinda el álgebra relacional. Aunque el ejercicio pareciera ser similar al anterior, ahora tendríamos que comparar si el resultado de contar algo (la cantidad de aviones que un piloto puede pilotar) es mayor al resultado de contar otra cosa. **Acumular o contar valores no lo podemos hacer.** Nótese que en el ejercicio anterior podíamos comparar si un salario era mayor a otro porque era simplemente una cuestión de comparar dos valores numéricos "absolutos" almacenados de antemano en un campo de la tabla.

9. Encuentre los eids de los empleados que están certificados para exactamente 3 (tres) aviones.

```
aux1 = ρ eid → eid_aux1, aid → aid_aux1 (certified)
```

```
aux2 = ρ eid → eid_aux2, aid → aid_aux2 (certified)
```

```
aux3 = ρ eid → eid_aux3, aid → aid_aux3 (certified)
```

```
vuelan3_o_mas = σ eid == eid_aux1 == eid_aux2 and aid != aid_aux1 and aid != aid_aux2 and  
aid_aux1 != aid_aux2 (certified * aux1 * aux2)
```

```
vuelan4_o_mas = σ eid == eid_aux1 == eid_aux2 == eid_aux3 and aid != aid_aux1 and aid !=  
aid_aux2 and aid != aid_aux3 and aid_aux1 != aid_aux2 and aid_aux1 != aid_aux3 and aid_aux2 !=  
aid_aux3 (certified * aux1 * aux2 * aux3)
```

```
π eid (vuelan3_o_mas) - π eid (vuelan4_o_mas)
```

10. Encuentre la cantidad total de dinero pagado en concepto de salario.

Para resolver este ejercicio tendríamos que sumar todos los salarios que aparecen en la tabla **employees**. Como sucedía en el ejercicio 8, **al no poder contar o acumular valores** en el álgebra relacional, este ejercicio **no se puede resolver**.

11. ¿Existe alguna secuencia de vuelos de Buenos Aires a Berlín Cada vuelo en la secuencia debe partir de la ciudad que es destino del vuelo anterior; el primer vuelo debe partir de Buenos Aires, el último debe llegar a Berlín y no hay restricción en la cantidad de vuelos intermedios. Su consulta debe determinar cuándo existe alguna secuencia de vuelos desde Buenos Aires a Berlín para alguna instancia de la relación Vuelos.

Al no estar determinada la cantidad de vuelos intermedios o escalas que tiene el vuelo este ejercicio **no se puede resolver**.

El en caso hipotético de que sí estuviese determinado este factor, al ejercicio se lo podría resolver haciendo muchas comparaciones (aunque pueda parecer tedioso). Sin estar determinado el número de paradas, no se puede hacer con las herramientas que tenemos de álgebra relacional.