

# **Lenguajes Formales para la manipulación de datos:**

## **Álgebra Relacional**

## **Cálculo Relacional**

**Los lenguajes de manipulación de datos – también llamados lenguajes de consulta - se clasifican en:**

**LENGUAJES PROCEDIMENTALES:** se refiere al tipo de lenguajes donde se debe indicar cómo el SGBD debe ubicar los datos solicitados en una consulta. Se deben especificar las operaciones o procedimientos de acceso a los datos.

**LENGUAJES NO PROCEDIMENTALES O DECLARATIVOS:** se refiere al tipo de lenguajes donde se describen los datos deseados sin especificar las operaciones o procedimientos requeridos para obtenerlos.

Los lenguajes usados para consultar las relaciones en el modelo lógico relacional, son, a su vez, de dos tipos: los lenguajes formales y los lenguajes de consultas - derivados de los lenguajes formales - que están implementados en los SGBD.

## ÁLGEBRA RELACIONAL

Es un lenguaje procedimental que consta de una serie de operadores que consultan una o varias relaciones para obtener otra relación resultado, sin que cambien las relaciones originales.

Tanto los operandos como los resultados son **RELACIONES**, por lo que la salida de una operación puede ser la entrada de otra operación. Esto permite anidar expresiones del álgebra, del mismo modo que se pueden anidar las expresiones aritméticas.

$$R = \text{operador}_1((\text{operador}_2(S)))$$

A esta propiedad se le denomina **clausura**: las relaciones son cerradas bajo el álgebra, del mismo modo que los números son cerrados bajo las operaciones aritméticas.

## ÁLGEBRA RELACIONAL

**Consta de operaciones primitivas y operaciones complejas.**

**Las operaciones primitivas son:**

- 1. Selección o Elegir.**
- 2. Proyección.**
- 3. Producto Cartesiano**
- 4. Unión.**
- 5. Diferencia.**

## ÁLGEBRA RELACIONAL

**Las operaciones complejas son:**

- 1. Intersección.**
- 2. Join Natural.**
- 3. División.**

Si el número de relaciones que aparecen como argumento del operador es igual a **uno (1)** se dice que el operador es **unario** y si es **dos (2)** se dice que el operador es **binario**.

## ÁLGEBRA RELACIONAL: Operadores

### Operador de Selección o Elegir:

**Es un operador unario que permite escoger un conjunto de tuplas que cumplen con un predicado.**

**Sintaxis:**  $\sigma_{\langle \text{predicado} \rangle}(R)$

Pieza(p#, nombre-p, color-p, peso-p, ciudad-alm)

$\sigma_{\text{color-p}=\text{"negro"}}(\text{Pieza})$

Pieza	<u>p#</u>	nombre-p	color-p	peso-p	ciudad-alm
	p1	tornillo	negro	1	Caracas
	p2	tuerca	negro	2	Valencia
	p3	cable	gris	5	Porlamar
	p4	arandela	negro	1	Caracas
	p5	monitor	plateado	15	Valencia

$R \leftarrow \sigma_{\text{color-p}=\text{"negro"}}(\text{Pieza})$

R	<u>p#</u>	nombre-p	color-p	peso-p	ciudad-alm
	p1	tornillo	negro	1	Caracas
	p2	tuerca	negro	2	Valencia
	p4	arandela	negro	1	Caracas

Pieza	<u>p#</u>	nombre-p	color-p	peso-p	ciudad-alm
	p1	tornillo	negro	1	Caracas
	p2	tuerca	negro	2	Valencia
	p3	cable	gris	5	Porlamar
	p4	arandela	negro	1	Caracas
	p5	monitor	plateado	15	Valencia

$R \leftarrow \sigma_{\text{color-p}=\text{"negro"} \wedge \text{peso-p} < 2}(\text{Pieza})$

R	<u>p#</u>	nombre-p	color-p	peso-p	ciudad-alm
	p1	tornillo	negro	1	Caracas
	p4	arandela	negro	1	Caracas



Pieza	<u>p#</u>	nombre-p	color-p	peso-p	ciudad-alm
	p1	tornillo	negro	1	Caracas
	p2	tuerca	negro	2	Valencia
	p3	cable	gris	5	Porlamar
	p4	arandela	negro	1	Caracas
	p5	monitor	plateado	15	Valencia

$R \leftarrow \sigma_{\text{peso-p} < 2} (\sigma_{\text{color-p} = \text{"negro"}}(\text{Pieza}))$

R	<u>p#</u>	nombre-p	color-p	peso-p	ciudad-alm
	p1	tornillo	negro	1	Caracas
	p4	arandela	negro	1	Caracas

Precio	<u>p#</u>	precio-p	precio-p-sug
	p1	20	19
	p2	15	17
	p3	80	78
	p4	12	12
	p5	100	100

$R \leftarrow \sigma_{\text{precio-p} = \text{precio-p-sug}} (\text{Precio})$

R	<u>p#</u>	precio-p	precio-p-sug
	p4	12	12
	p5	100	100

## ÁLGEBRA RELACIONAL: Operadores

### Operador de Proyección:

**Es un operador unario que permite escoger un conjunto de columnas de una relación.**

**Sintaxis:**  $\pi_{\langle \text{lista-atributos} \rangle}(R)$

Pieza(p#, nombre-p, color-p, peso-p, ciudad-alm)

$\pi_{p\#, ciudad-alm}(Pieza)$

Pieza	<u>p#</u>	nombre-p	color-p	peso-p	ciudad-alm
	p1	tornillo	negro	1	Caracas
	p2	tuerca	negro	2	Valencia
	p3	cable	gris	5	Porlamar
	p4	arandela	negro	1	Caracas
	p5	monitor	plateado	15	Valencia

$R \leftarrow \pi_{p\#, \text{ciudad-alm}} (\text{Pieza})$

R	<u>p#</u>	ciudad-alm
	p1	Caracas
	p2	Valencia
	p3	Porlamar
	p4	Caracas
	p5	Valencia

Precio	<u>p#</u>	precio-p	precio-p-sug
	p1	20	19
	p2	15	17
	p3	80	78
	p4	12	12
	p5	100	100

$R \leftarrow \pi_{p\#, \text{precio-p}} (\sigma_{\text{precio-p} = \text{precio-p-sug}} (\text{Precio}))$

R	<u>p#</u>	precio-p
	p4	12
	p5	100

Pieza

<u>p#</u>	nombre-p	color-p	peso-p	ciudad-alm
p1	tornillo	negro	1	Caracas
p2	tuerca	negro	2	Valencia
p3	cable	gris	5	Porlamar
p4	arandela	negro	1	Caracas
p5	monitor	plateado	15	Valencia

Precio

<u>p#</u>	precio-p	precio-p-sug
p1	20	19
p2	15	17
p3	80	78
p4	12	12
p5	100	100

*¿Cómo se pueden obtener las ciudades donde se almacenan las piezas cuyos precio-p y precio-p-sug son iguales?*

### ÁLGEBRA RELACIONAL: Operadores

#### Operador de Producto Cartesiano:

**Es un operador binario que permite relacionar dos relaciones. La primera relación se llama externa y la segunda se llama interna. En la relación resultante cada tupla de la relación externa se relaciona con todas las tuplas de la relación interna.**

**Sean  $R(a_1, a_2, \dots, a_n)$  y  $S(b_1, b_2, \dots, b_m)$  dos relaciones, el producto cartesiano de  $R \times S$  se define como:**

$$R \times S = \{(r, s) / r \in R \wedge s \in S\}$$

La cardinalidad de  $R \times S$  es  $|R| \times |S|$  y su grado es  $n+m$

### Pieza X Precio

Pieza.p#	nombre-p	color-p	peso-p	ciudad-alm	Precio.p#	precio-p	precio-p-sug
p1	tornillo	negro	1	Caracas	p1	20	19
p1	tornillo	negro	1	Caracas	p2	15	17
p1	tornillo	negro	1	Caracas	p3	80	78
p1	tornillo	negro	1	Caracas	p4	12	12
p1	tornillo	negro	1	Caracas	p5	100	100
p2	tuerca	negro	2	Valencia	p1	20	19
p2	tuerca	negro	2	Valencia	p2	15	17
p2	tuerca	negro	2	Valencia	p3	80	78
p2	tuerca	negro	2	Valencia	p4	12	12
p2	tuerca	negro	2	Valencia	p5	100	100
.							
.							
.							
p5	monitor	plateado	15	Valencia	p1	20	19
p5	monitor	plateado	15	Valencia	p2	15	17
p5	monitor	plateado	15	Valencia	p3	80	78
p5	monitor	plateado	15	Valencia	p4	12	12
p5	monitor	plateado	15	Valencia	p5	100	100

**25 tuplas y 8 columnas: Cardinalidad 25 y Grado 8**



### Pieza X Precio

Pieza.p#	nombre-p	color-p	peso-p	ciudad-alm	Precio.p#	precio-p	precio-p-sug
p1	tornillo	negro	1	Caracas	p1	20	19
p1	tornillo	negro	1	Caracas	p2	15	17
p1	tornillo	negro	1	Caracas	p3	80	78
p1	tornillo	negro	1	Caracas	p4	12	12
p1	tornillo	negro	1	Caracas	p5	100	100
p2	tuerca	negro	2	Valencia	p1	20	19
p2	tuerca	negro	2	Valencia	p2	15	17
p2	tuerca	negro	2	Valencia	p3	80	78
p2	tuerca	negro	2	Valencia	p4	12	12
p2	tuerca	negro	2	Valencia	p5	100	100
.							
.							
.							
p5	monitor	plateado	15	Valencia	p1	20	19
p5	monitor	plateado	15	Valencia	p2	15	17
p5	monitor	plateado	15	Valencia	p3	80	78
p5	monitor	plateado	15	Valencia	p4	12	12
p5	monitor	plateado	15	Valencia	p5	100	100

¿Cómo se pueden obtener las ciudades donde se almacenan las piezas cuyos precio-p y precio-p-sug son iguales?

## Pieza X Precio

Pieza.p#	nombre-p	color-p	peso-p	ciudad-alm	Precio.p#	precio-p	precio-p-sug
p1	tornillo	negro	1	Caracas	p1	20	19
p1	tornillo	negro	1	Caracas	p2	15	17
p1	tornillo	negro	1	Caracas	p3	80	78
p1	tornillo	negro	1	Caracas	p4	12	12
p1	tornillo	negro	1	Caracas	p5	100	100
p2	tuerca	negro	2	Valencia	p1	20	19
p2	tuerca	negro	2	Valencia	p2	15	17
p2	tuerca	negro	2	Valencia	p3	80	78
p2	tuerca	negro	2	Valencia	p4	12	12
p2	tuerca	negro	2	Valencia	p5	100	100
.							
.							
.							
p5	monitor	plateado	15	Valencia	p1	20	19
p5	monitor	plateado	15	Valencia	p2	15	17
p5	monitor	plateado	15	Valencia	p3	80	78
p5	monitor	plateado	15	Valencia	p4	12	12
p5	monitor	plateado	15	Valencia	p5	100	100

$\pi_{\text{Pieza.p\#, ciudad-alm}}(\sigma_{\text{precio-p=precio-p-sug} \wedge \text{Pieza.p\#}=\text{Precio.p\#}}(\text{Pieza X Precio}))$

## ÁLGEBRA RELACIONAL: Operadores

### Operador de Unión y Diferencia:

**Ambos son operadores binarios y conjuntistas.**

**La unión une las tuplas de una relación con las de otra eliminando las repetidas.**

**La diferencia permite encontrar las tuplas que están en una relación y no en otra.**

**Sean  $R(a_1, a_2, \dots, a_n)$  y  $S(b_1, b_2, \dots, b_n)$  dos relaciones donde los atributos  $a_i$  y  $b_i$  con  $i = 1, \dots, n$  se definen sobre los mismos dominios uno a uno, entonces se definen:**

$$R \cup S = \{t / t \in R \vee t \in S\}$$

$$R - S = \{t / t \in R \wedge t \notin S\}$$

## ÁLGEBRA RELACIONAL: Operadores

### Operador de Unión y Diferencia:

**EstudianteComputacion(ci-ec, nombre-ec, direccion-ec)**

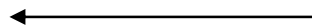
**EstudianteQuimica(ci-eq, nombre-eq, direccion-eq)**

**$R \leftarrow \text{EstudianteComputacion} \cup \text{EstudianteQuimica}$**

**Curso(ci-c, codigo-a)**

**Aprobo(ci-a, codigo-asig)**

**$S \leftarrow \text{Curso} - \text{Aprobo}$**



S contiene los  
estudiantes que reprobaron  
la asignatura relacionada.

**EstudianteComputacion(ci-ec, nombre-ec, direccion-ec)**

**EstudianteQuimica(ci-eq, nombre-eq, direccion-eq)**

**Curso(ci-c, codigo-a)**

**Aprobo(ci-a, codigo-asig)**

**R**  $\leftarrow \pi$  nombre-ec( $\sigma$  ci-c=ci-ec(  
     $\pi$  ci-c (Curso - Aprobo) X  
     $\pi$  ci-ec, nombre-ec( $\sigma$  direccion-ec='Naguanagua'(EstudianteComputacion))))

**R1**  $\leftarrow \sigma$  direccion-ec='Naguanagua'(EstudianteComputacion)

**R2**  $\leftarrow \pi$  ci-ec, nombre-ec(R1)

**R3**  $\leftarrow \pi$  ci-c (Curso - Aprobo)

**R4**  $\leftarrow$  R3 X R2

**R5**  $\leftarrow \sigma$  ci-c=ci-ec(R4)

**R6**  $\leftarrow \pi$  nombre-ec(R5)

## Fuentes consultadas:

[1] Silberchatz, Korth. ,  
"Fundamentos de Bases de Datos".

[2] Prof. Elsa Liliana Tovar.  
Notas de clase compiladas entre 1997-2016.

[3] [http://www3.uji.es/~mmarques/f47/apun/  
node58.html](http://www3.uji.es/~mmarques/f47/apun/node58.html)