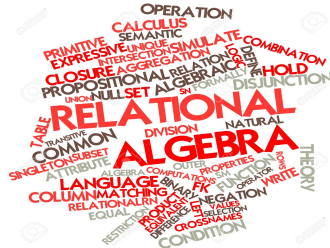


# Álgebra Relacional

1C - 2024



# AR - Marco General

- **Álgebra Relacional.** Lenguaje formal utilizado en el modelo relacional
- Permite a usuarios especificar consultas sobre instancias de relaciones
- El resultado de una consulta es una nueva relación

# AR - Marco General

- **Álgebra Relacional.** Lenguaje formal utilizado en el modelo relacional
- Permite a usuarios especificar consultas sobre instancias de relaciones
- El resultado de una consulta es una nueva relación
- **Importancia.**
  - 1 Provee fundamento formal a las operaciones asociadas al modelo relacional
  - 2 Base para implementar y optimizar queries en RDBMS
  - 3 Principales operaciones y funciones de los módulos internos de la mayoría de los sistemas relacionales están basados en operaciones del AR
- **Técnica.** Procedural (a diferencia del **Cálculo Relacional** que es de tipo declarativo)

# AR - Marco General

- **Álgebra Relacional.** Lenguaje formal utilizado en el modelo relacional
- Permite a usuarios especificar consultas sobre instancias de relaciones
- El resultado de una consulta es una nueva relación
- **Importancia.**
  - 1 Provee fundamento formal a las operaciones asociadas al modelo relacional
  - 2 Base para implementar y optimizar queries en RDBMS
  - 3 Principales operaciones y funciones de los módulos internos de la mayoría de los sistemas relacionales están basados en operaciones del AR
- **Técnica.** Procedural (a diferencia del **Cálculo Relacional** que es de tipo declarativo)
- **Operadores.** Unarios y Binarios

# AR - SELECT

- **Función.** Selecciona un subconjunto de tuplas de una relación que satisface cierta condición
- **Notación.**  $\sigma_{\langle \text{condición de selección} \rangle}(R)$
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

# AR - SELECT

- **Función.** Selecciona un subconjunto de tuplas de una relación que satisface cierta condición
- **Notación.**  $\sigma_{\langle \text{condición de selección} \rangle}(R)$
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\sigma_{\text{Sexo}=\text{F}}(\text{EMPLEADO})$

# AR - SELECT

- **Función.** Selecciona un subconjunto de tuplas de una relación que satisface cierta condición
- **Notación.**  $\sigma_{\langle \text{condición de selección} \rangle}(R)$
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\sigma_{\text{Sexo}=\text{F}}(\text{EMPLEADO})$

DNI	Nombre	Sexo	Salario
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

# AR - SELECT

- **Función.** Selecciona un subconjunto de tuplas de una relación que satisface cierta condición
- **Notación.**  $\sigma_{\langle \text{condición de selección} \rangle}(R)$
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\sigma_{\text{Sexo}=F}(\text{EMPLEADO})$

DNI	Nombre	Sexo	Salario
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\sigma_{\text{Sexo}=F \text{ AND Salario} > \$15.000}(\text{EMPLEADO})$



# AR - SELECT

- **Función.** Selecciona un subconjunto de tuplas de una relación que satisface cierta condición
- **Notación.**  $\sigma_{\langle \text{condición de selección} \rangle}(R)$
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\sigma_{\text{Sexo}=F}(\text{EMPLEADO})$

DNI	Nombre	Sexo	Salario
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\sigma_{\text{Sexo}=F \text{ AND Salario} > \$15.000}(\text{EMPLEADO})$

DNI	Nombre	Sexo	Salario
33456234	Laura	F	\$25.000,00

# AR - SELECT

- **Función.** Selecciona un subconjunto de tuplas de una relación que satisface cierta condición
- **Notación.**  $\sigma_{\langle \text{condición de selección} \rangle}(R)$
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\sigma_{\text{Sexo}=F}(\text{EMPLEADO})$

DNI	Nombre	Sexo	Salario
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\sigma_{\text{Sexo}=F \text{ AND Salario} > \$15.000}(\text{EMPLEADO})$

DNI	Nombre	Sexo	Salario
33456234	Laura	F	\$25.000,00

- Genera una *partición horizontal* de la relación

# AR - SELECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**

# AR - SELECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\sigma_c(R)) = \text{Grado}(R)$

# AR - SELECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\sigma_c(R)) = \text{Grado}(R)$
- **# tuplas.**

# AR - SELECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\sigma_c(R)) = \text{Grado}(R)$
- **# tuplas.**  $|\sigma_c(R)| \leq |R|$

# AR - SELECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\sigma_c(R)) = \text{Grado}(R)$
- **# tuplas.**  $|\sigma_c(R)| \leq |R|$
- La fracción de tuplas seleccionadas se denomina *selectividad* de la condición
- **Conmutatividad.**

# AR - SELECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\sigma_c(R)) = \text{Grado}(R)$
- **# tuplas.**  $|\sigma_c(R)| \leq |R|$
- La fracción de tuplas seleccionadas se denomina *selectividad* de la condición
- **Conmutatividad.**  $\sigma_{c_1}(\sigma_{c_2}(R)) = \sigma_{c_2}(\sigma_{c_1}(R))$



# AR - SELECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\sigma_c(R)) = \text{Grado}(R)$
- **# tuplas.**  $|\sigma_c(R)| \leq |R|$
- La fracción de tuplas seleccionadas se denomina *selectividad* de la condición
- **Conmutatividad.**  $\sigma_{c_1}(\sigma_{c_2}(R)) = \sigma_{c_2}(\sigma_{c_1}(R))$
- **Cascada de SELECTs.**  $\sigma_{c_1}(\sigma_{c_2}(\dots\sigma_{c_n}(R))) = \sigma_{c_1 \text{ AND } c_2 \text{ AND } \dots \text{ AND } c_n}(R)$

# AR - SELECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\sigma_c(R)) = \text{Grado}(R)$
- **# tuplas.**  $|\sigma_c(R)| \leq |R|$
- La fracción de tuplas seleccionadas se denomina *selectividad* de la condición
- **Conmutatividad.**  $\sigma_{c_1}(\sigma_{c_2}(R)) = \sigma_{c_2}(\sigma_{c_1}(R))$
- **Cascada de SELECTs.**  $\sigma_{c_1}(\sigma_{c_2}(\dots\sigma_{c_n}(R))) = \sigma_{c_1 \text{ AND } c_2 \text{ AND } \dots \text{ AND } c_n}(R)$
- **SQL.** Se especifica típicamente en la cláusula **WHERE**
- **Ejemplo.**  $\sigma_{\text{Sexo}=F \text{ AND } \text{Salario}>\$15.000}(\text{EMPLEADO})$  se puede corresponder con:

```
SELECT *  
FROM EMPLEADO  
WHERE Sexo=F AND Salario>$15.000;
```

# AR - PROJECT

- **Función.** Selecciona un subconjunto de columnas de una relación
- **Notación.**  $\pi_{\langle \text{lista de atributos} \rangle}(R)$
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

# AR - PROJECT

- **Función.** Selecciona un subconjunto de columnas de una relación
- **Notación.**  $\pi_{\langle \text{lista de atributos} \rangle}(R)$
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{DNI, Salario}(EMPLEADO)$

# AR - PROJECT

- **Función.** Selecciona un subconjunto de columnas de una relación
- **Notación.**  $\pi_{\langle \text{lista de atributos} \rangle}(R)$
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{\text{DNI}, \text{Salario}}(\text{EMPLEADO})$

DNI	Salario
20222333	\$20.000,00
33456234	\$25.000,00
45432345	\$10.000,00

# AR - PROJECT

- **Función.** Selecciona un subconjunto de columnas de una relación
- **Notación.**  $\pi_{\langle \text{lista de atributos} \rangle}(R)$
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{\text{DNI}, \text{Salario}}(\text{EMPLEADO})$

DNI	Salario
20222333	\$20.000,00
33456234	\$25.000,00
45432345	\$10.000,00

- Genera una *partición vertical* de la relación

# AR - PROJECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**

# AR - PROJECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\pi_{\langle \text{lista de atributos} \rangle}(R)) = | \langle \text{lista de atributos} \rangle |$



# AR - PROJECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\pi_{\langle \text{lista de atributos} \rangle}(R)) = | \langle \text{lista de atributos} \rangle |$
- **# tuplas.**

# AR - PROJECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\pi_{\langle \text{lista de atributos} \rangle}(R)) = | \langle \text{lista de atributos} \rangle |$
- **# tuplas.**  $|\pi_{\langle \text{lista de atributos} \rangle}(R)| \leq |R|$ . Remueve tuplas duplicadas de la relación resultante

# AR - PROJECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\pi_{\langle \text{lista de atributos} \rangle}(R)) = | \langle \text{lista de atributos} \rangle |$
- **# tuplas.**  $|\pi_{\langle \text{lista de atributos} \rangle}(R)| \leq |R|$ . Remueve tuplas duplicadas de la relación resultante
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

# AR - PROJECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\pi_{\langle \text{lista de atributos} \rangle}(R)) = | \langle \text{lista de atributos} \rangle |$
- **# tuplas.**  $|\pi_{\langle \text{lista de atributos} \rangle}(R)| \leq |R|$ . Remueve tuplas duplicadas de la relación resultante
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{\text{Sexo}}(\text{EMPLEADO})$

# AR - PROJECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\pi_{\langle \text{lista de atributos} \rangle}(R)) = | \langle \text{lista de atributos} \rangle |$
- **# tuplas.**  $|\pi_{\langle \text{lista de atributos} \rangle}(R)| \leq |R|$ . Remueve tuplas duplicadas de la relación resultante
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{\text{Sexo}}(\text{EMPLEADO})$

Sexo
M
F

# AR - PROJECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\pi_{\langle \text{lista de atributos} \rangle}(R)) = | \langle \text{lista de atributos} \rangle |$
- **# tuplas.**  $|\pi_{\langle \text{lista de atributos} \rangle}(R)| \leq |R|$ . Remueve tuplas duplicadas de la relación resultante
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{\text{Sexo}}(\text{EMPLEADO})$

Sexo
M
F

- **Conservación # tuplas.**

# AR - PROJECT - Propiedades

- **Operador Unario.** Se aplica a una sola relación
- **Grado.**  $\text{Grado}(\pi_{\langle \text{lista de atributos} \rangle}(R)) = | \langle \text{lista de atributos} \rangle |$
- **# tuplas.**  $|\pi_{\langle \text{lista de atributos} \rangle}(R)| \leq |R|$ . Remueve tuplas duplicadas de la relación resultante
- **Ejemplo.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{\text{Sexo}}(\text{EMPLEADO})$

Sexo
M
F

- **Conservación # tuplas.** En  $\pi_{\langle \text{lista de atributos} \rangle}(R)$ , si  $\langle \text{lista de atributos} \rangle$  es súper clave de  $R$  entonces  $|\pi_{\langle \text{lista de atributos} \rangle}(R)| = |R|$

# AR - PROJECT - Propiedades (Cont.)

- **Conmutatividad.**



## AR - PROJECT - Propiedades (Cont.)

- **Conmutatividad.**  $\pi_{lista_1}(\pi_{lista_2}(R)) = \pi_{lista_1}(R)$ .  $lista_1 \subseteq lista_2$ , de lo contrario lado izq. de la expresión es incorrecto. **Conmutatividad no aplica a PROJECT**

# AR - PROJECT - Propiedades (Cont.)

- **Conmutatividad.**  $\pi_{lista_1}(\pi_{lista_2}(R)) = \pi_{lista_1}(R)$ .  $lista_1 \subseteq lista_2$ , de lo contrario lado izq. de la expresión es incorrecto. **Conmutatividad no aplica a PROJECT**
- **Ejemplo Conmutatividad.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

# AR - PROJECT - Propiedades (Cont.)

- **Conmutatividad.**  $\pi_{lista_1}(\pi_{lista_2}(R)) = \pi_{lista_1}(R)$ .  $lista_1 \subseteq lista_2$ , de lo contrario lado izq. de la expresión es incorrecto. **Conmutatividad no aplica a PROJECT**
- **Ejemplo Conmutatividad.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{Sexo}(\pi_{Nombre, Sexo}(EMPLEADO))$

# AR - PROJECT - Propiedades (Cont.)

- **Conmutatividad.**  $\pi_{lista_1}(\pi_{lista_2}(R)) = \pi_{lista_1}(R)$ .  $lista_1 \subseteq lista_2$ , de lo contrario lado izq. de la expresión es incorrecto. **Conmutatividad no aplica a PROJECT**
- **Ejemplo Conmutatividad.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{Sexo}(\pi_{Nombre, Sexo}(EMPLEADO))$

Nombre	Sexo
Diego	M
Laura	F
Marina	F

# AR - PROJECT - Propiedades (Cont.)

- **Conmutatividad.**  $\pi_{lista_1}(\pi_{lista_2}(R)) = \pi_{lista_1}(R)$ .  $lista_1 \subseteq lista_2$ , de lo contrario lado izq. de la expresión es incorrecto. **Conmutatividad no aplica a PROJECT**
- **Ejemplo Conmutatividad.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{Sexo}(\pi_{Nombre, Sexo}(EMPLEADO))$

Nombre	Sexo
Diego	M
Laura	F
Marina	F

Sexo
M
F

# AR - PROJECT - Propiedades (Cont.)

- **Conmutatividad.**  $\pi_{lista_1}(\pi_{lista_2}(R)) = \pi_{lista_1}(R)$ .  $lista_1 \subseteq lista_2$ , de lo contrario lado izq. de la expresión es incorrecto. **Conmutatividad no aplica a PROJECT**
- **Ejemplo Conmutatividad.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{Sexo}(\pi_{Nombre, Sexo}(EMPLEADO))$

Nombre	Sexo
Diego	M
Laura	F
Marina	F

Sexo
M
F

- $\pi_{Nombre, Sexo}(\pi_{Sexo}(EMPLEADO))$

# AR - PROJECT - Propiedades (Cont.)

- **Conmutatividad.**  $\pi_{lista_1}(\pi_{lista_2}(R)) = \pi_{lista_1}(R)$ .  $lista_1 \subseteq lista_2$ , de lo contrario lado izq. de la expresión es incorrecto. **Conmutatividad no aplica a PROJECT**
- **Ejemplo Conmutatividad.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{Sexo}(\pi_{Nombre, Sexo}(EMPLEADO))$

Nombre	Sexo
Diego	M
Laura	F
Marina	F

Sexo
M
F

- $\pi_{Nombre, Sexo}(\pi_{Sexo}(EMPLEADO))$  ¡NO ES POSIBLE!

# AR - PROJECT - Propiedades (Cont.)

- **Conmutatividad.**  $\pi_{lista_1}(\pi_{lista_2}(R)) = \pi_{lista_1}(R)$ .  $lista_1 \subseteq lista_2$ , de lo contrario lado izq. de la expresión es incorrecto. **Conmutatividad no aplica a PROJECT**
- **Ejemplo Conmutatividad.**

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{Sexo}(\pi_{Nombre, Sexo}(EMPLEADO))$

Nombre	Sexo
Diego	M
Laura	F
Marina	F

Sexo
M
F

- $\pi_{Nombre, Sexo}(\pi_{Sexo}(EMPLEADO))$  ¡NO ES POSIBLE!
- **SQL.** Se especifica típicamente en la cláusula **SELECT DISTINCT**
- **Ejemplo.**  $\pi_{Sexo, Salario}(EMPLEADO)$  se puede corresponder con:

```
SELECT DISTINCT Sexo, Salario  
FROM EMPLEADO
```



# AR - RENAME

- **Función.** Asigna nombre a atributos / relación resultado
- Muy útil para asignar nombre a resultados intermedios
- **Notación.**  $\rho_{S(B_1, B_2, \dots, B_n)}(R)$  ó  $\rho_S(R)$  ó  $\rho_{(B_1, B_2, \dots, B_n)}(R)$
- **Ejemplo 1.** Relaciones

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

# AR - RENAME

- **Función.** Asigna nombre a atributos / relación resultado
- Muy útil para asignar nombre a resultados intermedios
- **Notación.**  $\rho_{S(B_1, B_2, \dots, B_n)}(R)$  ó  $\rho_S(R)$  ó  $\rho_{(B_1, B_2, \dots, B_n)}(R)$
- **Ejemplo 1.** Relaciones

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{Nombre, Sexo}(\sigma_{Salario \geq \$15.000}(EMPLEADO))$

# AR - RENAME

- **Función.** Asigna nombre a atributos / relación resultado
- Muy útil para asignar nombre a resultados intermedios
- **Notación.**  $\rho_{S(B_1, B_2, \dots, B_n)}(R)$  ó  $\rho_S(R)$  ó  $\rho_{(B_1, B_2, \dots, B_n)}(R)$
- **Ejemplo 1.** Relaciones

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $\pi_{Nombre, Sexo}(\sigma_{Salario \geq \$15.000}(EMPLEADO))$ 
  - 1  $SALARIO\_MAYOR \leftarrow \sigma_{Salario \geq \$15.000}(EMPLEADO)$
  - 2  $RESULT \leftarrow \pi_{Nombre, Sexo}(SALARIO\_MAYOR)$

SALARIO\_MAYOR

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00

RESULT

Nombre	Sexo
Diego	M
Laura	F

# AR - RENAME (Cont.)

- Ejemplo 2. Atributos

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

# AR - RENAME (Cont.)

- Ejemplo 2. Atributos

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $EMP(id, Ingreso) \leftarrow \pi_{DNI, Salario}(EMPLEADO)$

# AR - RENAME (Cont.)

- Ejemplo 2. Atributos

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $EMP(id, Ingreso) \leftarrow \pi_{DNI, Salario}(EMPLEADO)$

EMP

id	Ingreso
20222333	\$20.000,00
33456234	\$25.000,00
45432345	\$10.000,00

# AR - RENAME (Cont.)

- Ejemplo 2. Atributos

EMPLEADO

DNI	Nombre	Sexo	Salario
20222333	Diego	M	\$20.000,00
33456234	Laura	F	\$25.000,00
45432345	Marina	F	\$10.000,00

- $EMP(id, Ingreso) \leftarrow \pi_{DNI, Salario}(EMPLEADO)$

id	Ingreso
20222333	\$20.000,00
33456234	\$25.000,00
45432345	\$10.000,00

- SQL. Se especifica típicamente en la cláusula **AS**
- Ejemplo.

```
SELECT EMP.DNI AS id, EMP.Salario AS Ingreso  
FROM EMPLEADO AS EMP
```

## AR - Ejercicio 1

VUELO

Número	Origen	Destino	Salida
345	MAD	CDG	12:30
321	MAD	ORY	19:05
165	LHR	CDG	09:55
903	CDG	LHR	14:40
447	CDG	LHR	17:00

AEROPUERTO

Código	Nombre	Ciudad
MAD	Barajas	Madrid
LGW	Gatwick	Londres
LHR	Heathrow	Londres
ORY	Orly	París
CDG	Charles de Gaulle	París

PASAJERO

DNI	Nombre
123	María
456	Pedro
789	Isabel

RESERVA

DNI	Nro.Vuelo	Fecha	Precio
789	165	07-01-11	210
123	345	20-12-10	170
789	321	15-12-10	250
456	345	03-11-10	190

- 1 Retornar Código y Nombre de los aeropuertos de Londres
- 2 ¿Qué retorna  $Cities(City) \leftarrow \pi_{Ciudad}(\sigma_{Código=ORY \text{ OR } Código=CDG}(AEROPUERTO))$
- 3 Obtener los números de vuelo que van desde CDG hacia LHR
- 4 Obtener los números de vuelo que van desde CDG hacia LHR o viceversa
- 5 Devolver las fechas de reservas cuyos precios son mayores a \$200



# AR - UNION, INTERSECTION, MINUS

- **Función.** Equivalente a operaciones matemáticas sobre conjuntos
- **Notación.**  $R \cup S$ ,  $R \cap S$ ,  $R - S$
- **Duplicados.** La relación resultante no contiene duplicados

# AR - UNION, INTERSECTION, MINUS

- **Función.** Equivalente a operaciones matemáticas sobre conjuntos
- **Notación.**  $R \cup S$ ,  $R \cap S$ ,  $R - S$
- **Duplicados.** La relación resultante no contiene duplicados
- **Unión Compatible.** Se dice que dos relaciones  $R(A_1, A_2, \dots, A_n)$  y  $S(B_1, B_2, \dots, B_n)$  son **unión compatibles** (o compatibles por tipos) si:
  - Ambas tienen grado  $n$
  - $(\forall i, 1 \leq i \leq n) \text{ tipo}(A_i) = \text{tipo}(B_i)$

# AR - UNION, INTERSECTION, MINUS

- **Función.** Equivalente a operaciones matemáticas sobre conjuntos
- **Notación.**  $R \cup S$ ,  $R \cap S$ ,  $R - S$
- **Duplicados.** La relación resultante no contiene duplicados
- **Unión Compatible.** Se dice que dos relaciones  $R(A_1, A_2, \dots, A_n)$  y  $S(B_1, B_2, \dots, B_n)$  son **unión compatibles** (o compatibles por tipos) si:
  - Ambas tienen grado  $n$
  - $(\forall i, 1 \leq i \leq n) \text{ tipo}(A_i) = \text{tipo}(B_i)$
- **UNION.**  $R \cup S$ . Relación que incluye todas las tuplas que están en R, S o en ambas relaciones a la vez. Duplicados son eliminados
- **INTERSECTION.**  $R \cap S$ . Relación que incluye todas las tuplas que están a la vez en R y S
- **SET DIFFERENCE (o MINUS).**  $R - S$ . Relación que incluye todas las tuplas que están en R, pero no incluye a aquellas que aparecen en S

# AR - UNION, INTERSECTION, MINUS

- **Función.** Equivalente a operaciones matemáticas sobre conjuntos
- **Notación.**  $R \cup S$ ,  $R \cap S$ ,  $R - S$
- **Duplicados.** La relación resultante no contiene duplicados
- **Unión Compatible.** Se dice que dos relaciones  $R(A_1, A_2, \dots, A_n)$  y  $S(B_1, B_2, \dots, B_n)$  son **unión compatibles** (o compatibles por tipos) si:
  - Ambas tienen grado  $n$
  - $(\forall i, 1 \leq i \leq n) \text{ tipo}(A_i) = \text{tipo}(B_i)$
- **UNION.**  $R \cup S$ . Relación que incluye todas las tuplas que están en R, S o en ambas relaciones a la vez. Duplicados son eliminados
- **INTERSECTION.**  $R \cap S$ . Relación que incluye todas las tuplas que están a la vez en R y S
- **SET DIFFERENCE (o MINUS).**  $R - S$ . Relación que incluye todas las tuplas que están en R, pero no incluye a aquellas que aparecen en S
- **Convención.** Relación resultante conserva los nombres de atributo de la primer relación.

# AR - UNION, INTERSECTION, MINUS (Cont.)

- Ejemplo 1. UNION

ALUMNOS\_BDs

id	Nombre
1	Diego
2	Laura
3	Marina

ALUMNOS\_TLENG

id	Nombre
2	Laura
4	Alejandro

# AR - UNION, INTERSECTION, MINUS (Cont.)

- Ejemplo 1. UNION

ALUMNOS\_BDs

id	Nombre
1	Diego
2	Laura
3	Marina

ALUMNOS\_TLENG

id	Nombre
2	Laura
4	Alejandro

- $\text{RESULT\_1} \leftarrow \text{ALUMNOS\_BD} \cup \text{ALUMNOS\_TLENG}$

# AR - UNION, INTERSECTION, MINUS (Cont.)

- Ejemplo 1. UNION

ALUMNOS\_BDs

id	Nombre
1	Diego
2	Laura
3	Marina

ALUMNOS\_TLENG

id	Nombre
2	Laura
4	Alejandro

- $RESULT\_1 \leftarrow ALUMNOS\_BD \cup ALUMNOS\_TLENG$

RESULT\_1

id	Nombre
1	Diego
2	Laura
3	Marina
4	Alejandro

# AR - UNION, INTERSECTION, MINUS (Cont.)

## ● Ejemplo 2. INTERSECTION

ALUMNOS\_BDs

id	Nombre
1	Diego
2	Laura
3	Marina

ALUMNOS\_TLENG

id	Nombre
2	Laura
4	Alejandro



# AR - UNION, INTERSECTION, MINUS (Cont.)

- Ejemplo 2. INTERSECTION

ALUMNOS\_BDs

id	Nombre
1	Diego
2	Laura
3	Marina

ALUMNOS\_TLENG

id	Nombre
2	Laura
4	Alejandro

- $\text{RESULT\_2} \leftarrow \text{ALUMNOS\_BD} \cap \text{ALUMNOS\_TLENG}$

# AR - UNION, INTERSECTION, MINUS (Cont.)

## • Ejemplo 2. INTERSECTION

ALUMNOS\_BDs

id	Nombre
1	Diego
2	Laura
3	Marina

ALUMNOS\_TLENG

id	Nombre
2	Laura
4	Alejandro

•  $RESULT\_2 \leftarrow ALUMNOS\_BD \cap ALUMNOS\_TLENG$

RESULT\_2

id	Nombre
2	Laura

# AR - UNION, INTERSECTION, MINUS (Cont.)

## ● Ejemplo 3. SET DIFFERENCE

ALUMNOS\_BDs

id	Nombre
1	Diego
2	Laura
3	Marina

ALUMNOS\_TLENG

id	Nombre
2	Laura
4	Alejandro

# AR - UNION, INTERSECTION, MINUS (Cont.)

- Ejemplo 3. SET DIFFERENCE

ALUMNOS\_BDs

id	Nombre
1	Diego
2	Laura
3	Marina

ALUMNOS\_TLENG

id	Nombre
2	Laura
4	Alejandro

- $RESULT\_3 \leftarrow ALUMNOS\_BD - ALUMNOS\_TLENG$

# AR - UNION, INTERSECTION, MINUS (Cont.)

- Ejemplo 3. SET DIFFERENCE

ALUMNOS\_BDs

id	Nombre
1	Diego
2	Laura
3	Marina

ALUMNOS\_TLENG

id	Nombre
2	Laura
4	Alejandro

- $\text{RESULT\_3} \leftarrow \text{ALUMNOS\_BD} - \text{ALUMNOS\_TLENG}$

RESULT\_3

id	Nombre
1	Diego
3	Marina

# AR - UNION, INTERSECTION, MINUS (Cont.)

- **Conmutatividad.**

# AR - UNION, INTERSECTION, MINUS (Cont.)

- **Conmutatividad.**

- $R \cup S = S \cup R$
- $R \cap S = S \cap R$
- En general,  $R - S \neq S - R$

# AR - UNION, INTERSECTION, MINUS (Cont.)

- **Conmutatividad.**

- $R \cup S = S \cup R$
- $R \cap S = S \cap R$
- En general,  $R - S \neq S - R$

- **Asociatividad.**



# AR - UNION, INTERSECTION, MINUS (Cont.)

- **Conmutatividad.**

- $R \cup S = S \cup R$
- $R \cap S = S \cap R$
- En general,  $R - S \neq S - R$

- **Asociatividad.**

- $R \cup (S \cup T) = (R \cup S) \cup T$
- $R \cap (S \cap T) = (R \cap S) \cap T$

# AR - UNION, INTERSECTION, MINUS (Cont.)

- **Conmutatividad.**

- $R \cup S = S \cup R$
- $R \cap S = S \cap R$
- En general,  $R - S \neq S - R$

- **Asociatividad.**

- $R \cup (S \cup T) = (R \cup S) \cup T$
- $R \cap (S \cap T) = (R \cap S) \cap T$

- **Equivalencia.**  $R \cap S =$

# AR - UNION, INTERSECTION, MINUS (Cont.)

- **Conmutatividad.**

- $R \cup S = S \cup R$
- $R \cap S = S \cap R$
- En general,  $R - S \neq S - R$

- **Asociatividad.**

- $R \cup (S \cup T) = (R \cup S) \cup T$
- $R \cap (S \cap T) = (R \cap S) \cap T$

- **Equivalencia.**  $R \cap S = ((R \cup S) - (R - S)) - (S - R)$

# AR - UNION, INTERSECTION, MINUS (Cont.)

- **Conmutatividad.**

- $R \cup S = S \cup R$
- $R \cap S = S \cap R$
- En general,  $R - S \neq S - R$

- **Asociatividad.**

- $R \cup (S \cup T) = (R \cup S) \cup T$
- $R \cap (S \cap T) = (R \cap S) \cap T$

- **Equivalencia.**  $R \cap S = ((R \cup S) - (R - S)) - (S - R)$

- **SQL 1.** Operaciones en SQL **UNION**, **INTERSECT**, **EXCEPT** funcionan como en AR
- **SQL 2.** Operaciones en SQL **UNION ALL**, **INTERSECT ALL**, **EXCEPT ALL** no eliminan duplicados

## AR - Ejercicio 2

*VUELO*

Número	Origen	Destino	Salida
345	MAD	CDG	12:30
321	MAD	ORY	19:05
165	LHR	CDG	09:55
903	CDG	LHR	14:40
447	CDG	LHR	17:00

*AEROPUERTO*

Código	Nombre	Ciudad
MAD	Barajas	Madrid
LGW	Gatwick	Londres
LHR	Heathrow	Londres
ORY	Orly	París
CDG	Charles de Gaulle	París

*PASAJERO*

DNI	Nombre
123	María
456	Pedro
789	Isabel

*RESERVA*

DNI	Nro.Vuelo	Fecha	Precio
789	165	07-01-11	210
123	345	20-12-10	170
789	321	15-12-10	250
456	345	03-11-10	190

- 1 Devolver los número de vuelo que tienen reservas generadas (utilizar  $\cap$ )
- 2 Devolver los número de vuelo que aún no tienen reservas
- 3 Retornar los códigos de aeropuerto de los que parten o arriban los vuelos

# AR - CARTESIAN PRODUCT

- **Función.** Produce una nueva relación que combina cada tupla de una relación con cada una de las tuplas de la otra relación
- **Notación.**  $R \times S$
- **Ejemplo.** PERSONA

Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

# AR - CARTESIAN PRODUCT

- **Función.** Produce una nueva relación que combina cada tupla de una relación con cada una de las tuplas de la otra relación

- **Notación.**  $R \times S$

- **Ejemplo.** PERSONA

Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

## NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

- $RESULT \leftarrow PERSONA \times NACIONALIDADES$

# AR - CARTESIAN PRODUCT

- **Función.** Produce una nueva relación que combina cada tupla de una relación con cada una de las tuplas de la otra relación

- **Notación.**  $R \times S$

- **Ejemplo.** PERSONA

Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

## NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

- $RESULT \leftarrow PERSONA \times NACIONALIDADES$

## RESULT

Nombre	Nacionalidad	IDN	Detalle
Diego	AR	AR	Argentina
Diego	AR	BR	Brasileira
Diego	AR	CH	Chilena
Laura	BR	AR	Argentina
Laura	BR	BR	Brasileira
Laura	BR	CH	Chilena
Marina	AR	AR	Argentina
Marina	AR	BR	Brasileira
Marina	AR	CH	Chilena



## AR - CARTESIAN PRODUCT (Cont.)

- **Unión compatible.** Las relaciones no tienen que ser unión compatibles

## AR - CARTESIAN PRODUCT (Cont.)

- **Unión compatible.** Las relaciones no tienen que ser unión compatibles
- **Grado.**

## AR - CARTESIAN PRODUCT (Cont.)

- **Unión compatible.** Las relaciones no tienen que ser unión compatibles
- **Grado.** Si  $T = R \times S$  entonces  $\text{grado}(T) = \text{grado}(R) + \text{grado}(S)$

## AR - CARTESIAN PRODUCT (Cont.)

- **Unión compatible.** Las relaciones no tienen que ser unión compatibles
- **Grado.** Si  $T = R \times S$  entonces  $\text{grado}(T) = \text{grado}(R) + \text{grado}(S)$
- **SQL. CROSS JOIN**

# AR - JOIN

- **Función.** Permite combinar pares de tuplas relacionadas entre dos relaciones
- **Notación.**  $R \bowtie_{\langle \text{condición} \rangle} S$
- **Ejemplo.**

PERSONA

Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasilera
CH	Chilena

# AR - JOIN

- **Función.** Permite combinar pares de tuplas relacionadas entre dos relaciones
- **Notación.**  $R \bowtie_{\langle \text{condición} \rangle} S$
- **Ejemplo.**

PERSONA

Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

- $RESULT \leftarrow PERSONA \bowtie_{Nacionalidad=IDN} NACIONALIDADES$

# AR - JOIN

- **Función.** Permite combinar pares de tuplas relacionadas entre dos relaciones
- **Notación.**  $R \bowtie_{\langle \text{condición} \rangle} S$
- **Ejemplo.**

PERSONA

Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

- $RESULT \leftarrow PERSONA \bowtie_{Nacionalidad=IDN} NACIONALIDADES$

RESULT

Nombre	Nacionalidad	IDN	Detalle
Diego	AR	AR	Argentina
Laura	BR	BR	Brasileira
Marina	AR	AR	Argentina

## AR - JOIN (Cont.)

- **CARTESIAN PRODUCT vs JOIN.**
  - **CARTESIAN PRODUCT** aparecen todas las combinaciones de tuplas
  - **JOIN** aparecen sólo combinaciones de tuplas que satisfacen condición



## AR - JOIN (Cont.)

- **CARTESIAN PRODUCT vs JOIN.**
  - **CARTESIAN PRODUCT** aparecen todas las combinaciones de tuplas
  - **JOIN** aparecen sólo combinaciones de tuplas que satisfacen condición
- **Condición 1.** En general, formato de condición de **JOIN** entre  $R$  y  $S$ :  
 $\langle \text{condición} \rangle \text{ AND } \langle \text{condición} \rangle \text{ AND...AND } \langle \text{condición} \rangle$
- **Condición 2.** Forma de  $\langle \text{condición} \rangle$  es  $A_i \theta B_j$ , siendo  $A_i$  atributo de  $R$  y  $B_j$  atributo de  $S$
- **Condición 3.**  $\text{dom}(A_i) = \text{dom}(B_j)$
- **Condición 4.**  $\theta \in \{=, <, \leq, >, \geq, \neq\}$

## AR - JOIN (Cont.)

- **CARTESIAN PRODUCT vs JOIN.**
  - **CARTESIAN PRODUCT** aparecen todas las combinaciones de tuplas
  - **JOIN** aparecen sólo combinaciones de tuplas que satisfacen condición
- **Condición 1.** En general, formato de condición de **JOIN** entre  $R$  y  $S$ :  
 $\langle \text{condición} \rangle \text{ AND } \langle \text{condición} \rangle \text{ AND...AND } \langle \text{condición} \rangle$
- **Condición 2.** Forma de  $\langle \text{condición} \rangle$  es  $A_i \theta B_j$ , siendo  $A_i$  atributo de  $R$  y  $B_j$  atributo de  $S$
- **Condición 3.**  $\text{dom}(A_i) = \text{dom}(B_j)$
- **Condición 4.**  $\theta \in \{=, <, \leq, >, \geq, \neq\}$
- **NULL.** Tuplas cuyos atributos de **JOIN** son NULL o cuya condición es *falsa* no aparecen en el resultado

## AR - JOIN (Cont.)

- **EQUIJOIN.** JOIN donde sólo se utiliza la operación  $=$  en la  $\langle \textit{condición} \rangle$ .

## AR - JOIN (Cont.)

- **EQUIJOIN.** JOIN donde sólo se utiliza la operación  $=$  en la  $\langle \text{condición} \rangle$ .
- **Duplicación de campos.** Al utilizar la igualdad, se generan campos duplicados.
- **Ejemplo.**

PERSONA

Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

# AR - JOIN (Cont.)

- **EQUIJOIN.** JOIN donde sólo se utiliza la operación = en la *< condición >*.
- **Duplicación de campos.** Al utilizar la igualdad, se generan campos duplicados.
- **Ejemplo.**

PERSONA	
Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

NACIONALIDADES	
IDN	Detalle
AR	Argentina
BR	Brasilera
CH	Chilena

- $RESULT \leftarrow PERSONA \bowtie_{Nacionalidad=IDN} NACIONALIDADES$

# AR - JOIN (Cont.)

- **EQUIJOIN.** JOIN donde sólo se utiliza la operación = en la *< condición >*.
- **Duplicación de campos.** Al utilizar la igualdad, se generan campos duplicados.
- **Ejemplo.**

PERSONA

Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

- $RESULT \leftarrow PERSONA \bowtie_{Nacionalidad=IDN} NACIONALIDADES$

RESULT

Nombre	Nacionalidad	IDN	Detalle
Diego	AR	AR	Argentina
Laura	BR	BR	Brasileira
Marina	AR	AR	Argentina

## AR - JOIN (Cont.)

- **EQUIJOIN.** JOIN donde sólo se utiliza la operación = en la *< condición >*.
- **Duplicación de campos.** Al utilizar la igualdad, se generan campos duplicados.
- **Ejemplo.**

PERSONA	
Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

NACIONALIDADES	
IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

- $RESULT \leftarrow PERSONA \bowtie_{Nacionalidad=IDN} NACIONALIDADES$

RESULT			
Nombre	Nacionalidad	IDN	Detalle
Diego	AR	AR	Argentina
Laura	BR	BR	Brasileira
Marina	AR	AR	Argentina

- **NATURAL JOIN.** Realiza el JOIN entre campos de mismo nombre y deja sólo uno de los campos duplicados
- **Notación.**  $R \bowtie S$  (también en la bibliografía  $R * S$ )
- **Requerimiento.** Requiere que atributos de JOIN tengan el mismo nombre. De no ser el caso, se debe hacer un RENAME previo

## AR - JOIN (Cont.)

- Ejemplo NATURAL JOIN.

PERSONA

Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasilera
CH	Chilena



# AR - JOIN (Cont.)

- Ejemplo NATURAL JOIN.

PERSONA

Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasilera
CH	Chilena

- $NACIONALIDADES\_TEMP(Nacionalidad, Detalle) \leftarrow \pi_{IDN, Detalle}(NACIONALIDADES)$
- $RESULT \leftarrow PERSONA \bowtie NATACIONALIDADES\_TEMP$

## AR - JOIN (Cont.)

- Ejemplo NATURAL JOIN.

PERSONA

Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

- $NACIONALIDADES\_TEMP(Nacionalidad, Detalle) \leftarrow \pi_{IDN, Detalle}(NACIONALIDADES)$
- $RESULT \leftarrow PERSONA \bowtie NACIONALIDADES\_TEMP$

RESULT

Nombre	Nacionalidad	Detalle
Diego	AR	Argentina
Laura	BR	Brasileira
Marina	AR	Argentina

# AR - JOIN (Cont.)

- Ejemplo NATURAL JOIN.

PERSONA	
Nombre	Nacionalidad
Diego	AR
Laura	BR
Marina	AR

NACIONALIDADES	
IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

- $NACIONALIDADES\_TEMP(Nacionalidad, Detalle) \leftarrow \pi_{IDN, Detalle}(NACIONALIDADES)$
- $RESULT \leftarrow PERSONA \bowtie NACIONALIDADES\_TEMP$

RESULT		
Nombre	Nacionalidad	Detalle
Diego	AR	Argentina
Laura	BR	Brasileira
Marina	AR	Argentina

- Tamaño resultado JOIN(S,R). Puede ir de 0 a  $S * R$  registros
- Selectividad de JOIN. Es una tasa y corresponde a:  $\frac{|resultado JOIN(S,R)|}{|S| * |R|}$

## AR - JOIN (Cont.)

- **SQL.** Se Puede realizar de múltiples maneras.
- **Ejemplo.**

$NACIONALIDADES\_TEMP(Nacionalidad, Detalle) \leftarrow \pi_{IDN, Detalle}(NACIONALIDADES)$

$RESULT \leftarrow PERSONA \bowtie NACIONALIDADES\_TEMP$

se puede corresponder con:

**SELECT** Persona.Nombre, Persona.Nacionalidad, NACIONALIDADES.Detalle  
**FROM** PERSONA, NACIONALIDADES  
**WHERE** PERSONA.Nacionalidad=NACIONALIDADES.IDN;

## AR - OUTER JOIN

- **INNER JOIN.** JOIN donde tuplas que no cumplen condición de JOIN son eliminadas del resultado (Ej. NULL en atributo de JOIN)
- **OUTER JOIN.** JOIN en el cual se incorpora adicionalmente al resultado las tuplas de  $R$ ,  $S$ , o ambas relaciones, que no cumplen la condición de JOIN

## AR - OUTER JOIN

- **INNER JOIN.** JOIN donde tuplas que no cumplen condición de JOIN son eliminadas del resultado (Ej. NULL en atributo de JOIN)
- **OUTER JOIN.** JOIN en el cual se incorpora adicionalmente al resultado las tuplas de  $R$ ,  $S$ , o ambas relaciones, que no cumplen la condición de JOIN
- **LEFT OUTER JOIN.**  $R \bowtie S$ . Conserva todas las tuplas de  $R$ . Si no se encuentra ninguna tupla de  $S$  que cumpla con condición de JOIN, entonces los atributos de  $S$  en el resultado se completan en NULL

# AR - OUTER JOIN

- Ejemplo.

PERSONA

Nombre	Nacionalidad
Diego	BR
Laura	NULL
Marina	AR
Santiago	UY

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasilera
CH	Chilena

# AR - OUTER JOIN

- Ejemplo.

PERSONA

Nombre	Nacionalidad
Diego	BR
Laura	NULL
Marina	AR
Santiago	UY

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasilera
CH	Chilena

- $RESULT \leftarrow PERSONA \bowtie_{Nacionalidad=IDN} NACIONALIDADES$



# AR - OUTER JOIN

- Ejemplo.

PERSONA

Nombre	Nacionalidad
Diego	BR
Laura	NULL
Marina	AR
Santiago	UY

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

- $RESULT \leftarrow PERSONA \bowtie_{Nacionalidad=IDN} NACIONALIDADES$

RESULT

Nombre	Nacionalidad	IDN	Detalle
Diego	BR	BR	Brasileira
Marina	AR	AR	Argentina

# AR - OUTER JOIN

- Ejemplo.

PERSONA

Nombre	Nacionalidad
Diego	BR
Laura	NULL
Marina	AR
Santiago	UY

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

- $RESULT \leftarrow PERSONA \bowtie_{Nacionalidad=IDN} NACIONALIDADES$

RESULT

Nombre	Nacionalidad	IDN	Detalle
Diego	BR	BR	Brasileira
Marina	AR	AR	Argentina

- $RESULT \leftarrow PERSONA \Join_{Nacionalidad=IDN} NACIONALIDADES$

# AR - OUTER JOIN

- Ejemplo.

PERSONA

Nombre	Nacionalidad
Diego	BR
Laura	NULL
Marina	AR
Santiago	UY

NACIONALIDADES

IDN	Detalle
AR	Argentina
BR	Brasileira
CH	Chilena

- $RESULT \leftarrow PERSONA \bowtie_{Nacionalidad=IDN} NACIONALIDADES$

RESULT

Nombre	Nacionalidad	IDN	Detalle
Diego	BR	BR	Brasileira
Marina	AR	AR	Argentina

- $RESULT \leftarrow PERSONA \Join_{Nacionalidad=IDN} NACIONALIDADES$

RESULT

Nombre	Nacionalidad	IDN	Detalle
Diego	BR	BR	Brasileira
Laura	NULL	NULL	NULL
Marina	AR	AR	Argentina
Santiago	UY	NULL	NULL

## AR - OUTER JOIN (Cont.)

- **RIGHT OUTER JOIN.**  $R \bowtie_r S$ . Conserva todas las tuplas de  $S$ . Si no se encuentra ninguna tupla de  $R$  que cumpla con condición de JOIN, entonces los atributos de  $R$  en el resultado se completan en NULL
- **FULL OUTER JOIN.**  $R \bowtie_{fs} S$ . Conserva todas las tuplas de  $R$  y  $S$ . Si no se encuentra ninguna tupla de  $R$  o  $S$  que cumpla con condición de JOIN, entonces sus atributos en el resultado se completan en NULL
- **SQL.** Las tres operaciones de OUTER JOIN son parte del estándar SQL2

## AR - Ejercicio 3

*VUELO*

Número	Origen	Destino	Salida
345	MAD	CDG	12:30
321	MAD	ORY	19:05
165	LHR	CDG	09:55
903	CDG	LHR	14:40
447	CDG	LHR	17:00

*AEROPUERTO*

Código	Nombre	Ciudad
MAD	Barajas	Madrid
LGW	Gatwick	Londres
LHR	Heathrow	Londres
ORY	Orly	París
CDG	Charles de Gaulle	París

*PASAJERO*

DNI	Nombre
123	María
456	Pedro
789	Isabel

*RESERVA*

DNI	Nro.Vuelo	Fecha	Precio
789	165	07-01-11	210
123	345	20-12-10	170
789	321	15-12-10	250
456	345	03-11-10	190

- 1 Devolver el nombre de la ciudad de partida del vuelo número 165
- 2 Retornar el nombre de las personas que realizaron reservas a un valor menor a \$200
- 3 Obtener Nombre, Fecha y Destino del Viaje de todos los pasajeros que vuelan desde Madrid

# AR - DIVISION

- **Función.** Retorna los valores de  $R$  que se encuentran emparejados con **TODOS** los valores de  $S$
- **Notación.**  $R \div S$ . Requiere que atributos de  $S \subseteq$  atributos de  $R$ . Resultado contiene atributos de  $R$  menos atributos de  $S$

- **Ejemplo.** ALUMNOS

MATERIAS\_1MATERIAS\_2MATERIAS\_3

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↺ 🔍 ↻

# AR - DIVISION (Cont.)

● Ejemplo. ALUMNOS

Nombre	Materia
Diego	BD
Diego	PLP
Laura	BD
Laura	PLP
Laura	TLENG
Marina	BD
Marina	TLENG
Santiago	BD
Santiago	PLP
Santiago	TLENG

MATERIAS\_1

Materia
BD

MATERIAS\_2

Materia
BD
TLENG

MATERIAS\_3

Materia
BD
PLP
TLENG

$ALUMNOS \div MATERIAS_1$



# AR - DIVISION (Cont.)

● Ejemplo. ALUMNOS

Nombre	Materia
Diego	BD
Diego	PLP
Laura	BD
Laura	PLP
Laura	TLENG
Marina	BD
Marina	TLENG
Santiago	BD
Santiago	PLP
Santiago	TLENG

MATERIAS\_1

Materia
BD

MATERIAS\_2

Materia
BD
TLENG

MATERIAS\_3

Materia
BD
PLP
TLENG

$ALUMNOS \div MATERIAS_1$

Nombre
Diego
Laura
Marina
Santiago

# AR - DIVISION (Cont.)

● Ejemplo. ALUMNOS

Nombre	Materia
Diego	BD
Diego	PLP
Laura	BD
Laura	PLP
Laura	TLENG
Marina	BD
Marina	TLENG
Santiago	BD
Santiago	PLP
Santiago	TLENG

MATERIAS\_1

Materia
BD

MATERIAS\_2

Materia
BD
TLENG

MATERIAS\_3

Materia
BD
PLP
TLENG

$ALUMNOS \div MATERIAS_1$

Nombre
Diego
Laura
Marina
Santiago

$ALUMNOS \div MATERIAS_2$

# AR - DIVISION (Cont.)

● Ejemplo. ALUMNOS

Nombre	Materia
Diego	BD
Diego	PLP
Laura	BD
Laura	PLP
Laura	TLENG
Marina	BD
Marina	TLENG
Santiago	BD
Santiago	PLP
Santiago	TLENG

MATERIAS\_1

Materia
BD

MATERIAS\_2

Materia
BD
TLENG

MATERIAS\_3

Materia
BD
PLP
TLENG

$ALUMNOS \div MATERIAS\_1$

Nombre
Diego
Laura
Marina
Santiago

$ALUMNOS \div MATERIAS\_2$

Nombre
Laura
Marina
Santiago

# AR - DIVISION (Cont.)

● Ejemplo. ALUMNOS

Nombre	Materia
Diego	BD
Diego	PLP
Laura	BD
Laura	PLP
Laura	TLENG
Marina	BD
Marina	TLENG
Santiago	BD
Santiago	PLP
Santiago	TLENG

MATERIAS\_1

Materia
BD

MATERIAS\_2

Materia
BD
TLENG

MATERIAS\_3

Materia
BD
PLP
TLENG

$ALUMNOS \div MATERIAS_1$

Nombre
Diego
Laura
Marina
Santiago

$ALUMNOS \div MATERIAS_2$

Nombre
Laura
Marina
Santiago

$ALUMNOS \div MATERIAS_3$

# AR - DIVISION (Cont.)

● Ejemplo. ALUMNOS

Nombre	Materia
Diego	BD
Diego	PLP
Laura	BD
Laura	PLP
Laura	TLENG
Marina	BD
Marina	TLENG
Santiago	BD
Santiago	PLP
Santiago	TLENG

MATERIAS\_1

Materia
BD

MATERIAS\_2

Materia
BD
TLENG

MATERIAS\_3

Materia
BD
PLP
TLENG

$ALUMNOS \div MATERIAS_1$

Nombre
Diego
Laura
Marina
Santiago

$ALUMNOS \div MATERIAS_2$

Nombre
Laura
Marina
Santiago

$ALUMNOS \div MATERIAS_3$

Nombre
Laura
Santiago

# AR - DIVISION

- **Operación compuesta.** Se puede expresar como secuencia de otras operaciones ( $\pi$ ,  $X$ ,  $-$ )

# AR - DIVISION

- **Operación compuesta.** Se puede expresar como secuencia de otras operaciones ( $\pi$ ,  $\times$ ,  $-$ )

En ejemplo anterior ( $ALUMNOS \div MATERIAS_3$ ):

- $TEMP_1 \leftarrow \pi_{Nombre}(ALUMNOS)$
- $TEMP_2 \leftarrow \pi_{Nombre}((TEMP_1 \times MATERIAS_3) - ALUMNOS)$
- $RESULT \leftarrow TEMP_1 - TEMP_2$

*TEMP\_1*

Nombre
Diego
Laura
Marina
Santiago

*TEMP\_2*

Nombre
Diego
Marina

*RESULT*

Nombre
Laura
Santiago

# AR - DIVISION

- **Operación compuesta.** Se puede expresar como secuencia de otras operaciones ( $\pi$ ,  $\times$ ,  $-$ )

En ejemplo anterior ( $ALUMNOS \div MATERIAS_3$ ):

- $TEMP_1 \leftarrow \pi_{Nombre}(ALUMNOS)$
- $TEMP_2 \leftarrow \pi_{Nombre}((TEMP_1 \times MATERIAS_3) - ALUMNOS)$
- $RESULT \leftarrow TEMP_1 - TEMP_2$

*TEMP\_1*

Nombre
Diego
Laura
Marina
Santiago

*TEMP\_2*

Nombre
Diego
Marina

*RESULT*

Nombre
Laura
Santiago

- **SQL.** No suele implementar DIVISION



# AR - Bibliografía

- Capítulo 6 (hasta 6.5 inclusive) Elmasri/Navathe - Fundamentos de Sistemas de Bases de Datos, 5th Ed., Pearson, 2007.

