

Algebra relacional

BASES DE DATOS

Profesor: Héctor Gómez Gauchía

Materiales: Héctor Gómez Gauchía, Mercedes García Merayo

Lenguajes de consulta

- ▶ El M.R. me indica **cómo** almacenar los datos en forma de tablas.
- ▶ Me falta un medio para poder **obtener datos combinados de las tablas**:
 - ▶ Cómo hago una consulta que me devuelva los datos que quiero?
→ necesito un Lenguaje de Consulta
 - ▶ Ej: Compras hechas con la tarjeta 123 en la BD ejemplo
- ▶ Con un lenguaje de consulta puedo solicitar información de la BD:
 - ▶ L. Procedimentales: cómo obtener la información
 - ▶ Indicas una lista ordenada de operaciones para obtener el resultado
 - ▶ **Algebra relacional**: cuya implementación es el **SQL**
 - ▶ L. no procedimentales: qué información
 - ▶ Describes la información deseada de una forma declarativa
 - ▶ Basados en el Cálculo de Predicados de la lógica.
 - ▶ Son el Cálculo Relacional de Tuplas (CRT) y el C.R. de Dominios (CRD)
 - Cuyas implementaciones son Quel y QBE (en Access de MS)

Algebra Relacional

- ▶ Basado en la operaciones del *álgebra de conjuntos*, éstos conjuntos pueden ser:
 - ▶ Una relación o tabla
 - ▶ O bien, el resultado de otra operación: así se encadenan operaciones
- ▶ El resultado de cualquier operación
 - ▶ Siempre es una nueva relación
- ▶ **Conjunto de operaciones**
 - ▶ Aplicadas a dos relaciones compatibles (ambas con los mismos atributos)
 - ▶ unión, resta o diferencia, división, [intersección]
 - ▶ Aplicadas a dos relaciones cualesquiera
 - ▶ producto cartesiano, Reunión Condicional, Reunión Natural
 - ▶ Aplicadas a una relación
 - ▶ selección, proyección

Operaciones entre Conjuntos

- ▶ Dadas las relaciones/tablas:

Cliente: $CL(DNI, NombreC, Dirección)$

Moroso: $MO(DNI, NombreC, Dirección)$

- ▶ Consulta:

- ▶ Obtener los datos de todos los clientes: morosos y normales

- ▶ Necesito: operación unión de dos tablas CL y MO con los mismos atributos

$(CL \cup MO)(DNI, NombreC, Dirección)$

- ▶ Uso: Tener datos juntos de varias tablas con los mismos atributos (compatibles)

- ▶ Consulta:

- ▶ Obtener los datos de los clientes que son a la vez morosos y normales

- ▶ Necesito: operación intersección de dos tablas CL y MO con los mismos atributos

$(CL \cap MO)(DNI, NombreC, Dirección)$

- ▶ Uso: Tener datos que están a la vez en dos tablas compatibles

Operaciones entre Conjuntos: definición Formal

- ▶ Son operaciones binarias que requieren:
 - ▶ Dos esquemas de relación que deben tener el mismo numero de atributos y con igual dominio (en la misma posición):

$$R(A_1, \dots, A_n) \text{ y } S(B_1, \dots, B_n), \forall i \ 1 \leq i \leq n: \text{Dom}(A_i) = \text{Dom}(B_i)$$

- ▶ Dados $R(A_1, \dots, A_n)$, $S(B_1, \dots, B_n)$
 - ▶ La **operación unión** de R y S, que se denota $(R \cup S)(A_1, \dots, A_n)$, produce un
 - ▶ esquema de relación cuyas instancias validas
 - son las instancias validas de R o de S.
 - ▶ La **operación intersección**, que se denota $(R \cap S)(A_1, \dots, A_n)$ produce un
 - ▶ esquema de relación cuyas instancias validas
 - ▶ son las instancias validas que están en ambas, R y en S.

Operaciones entre Conjuntos

- ▶ Consulta:

- ▶ Obtener los datos de los clientes normales que **NO son** morosos

- ▶ Necesito: operación diferencia de dos tablas CL y MO con los mismos atributos

$(CL \setminus MO)(\text{DNI}, \text{NombreC}, \text{Dirección})$

→ Mejor, usaremos la notación $(CL - MO)(\text{DNI}, \text{NombreC}, \text{Dirección})$

- ▶ Uso:

- ▶ quitar de una tabla algunas tuplas / filas

- ▶ que por alguna situación especial están también en otra.

- ▶ En la descripción de Requisitos aparecen

- ▶ negaciones, ej.: clientes que no compran

- ▶ el mejor, el máximo, el mayor. Ej.: compra máximo importe

Operaciones entre Conjuntos: definición Formal

Dados dos esquemas de relación $R(A_1, \dots, A_n), S(B_1, \dots, B_n)$

- ▶ La **operación diferencia** de R y S, que se denota $(R \setminus S)(A_1, \dots, A_n)$ produce
 - ▶ un esquema de relación cuyas instancias validas son
 - las instancias validas de R pero no de S

- ▶ Resumen y Combinación de operaciones:

Queremos todas las instancias del esquema de relación A y B .

Todos $\longleftarrow A \cup B$

Queremos conocer las instancias que están tanto en A como en B.

Comunes $\longleftarrow A \cap B$

Instancias de A que no están en B.

SoloA $\longleftarrow A - B$

Instancias que solo están en A o en B.

SoloAóB $\longleftarrow \text{Todos} - \text{Comunes}$

Renombramiento / Asignación de nombres

- ▶ Consulta:

- ▶ Llamar “BuenosClientes” a los clientes que no son morosos

- ▶ Necesito: operación renombramiento del resultado de la operación

$$\rho_{BuenosClientes} (CL - MO)$$

- ▶ Uso:

- ▶ dar nombre al resultado de una operación o a una relación o atributos
 - ▶ para usarlo en otras operaciones después
 - ▶ y manejarlos mejor

- ▶ Letra ro minúscula: ρ

Renombramiento / Asignación de nombres: Formal

- ▶ La *operación renombramiento* permite renombrar relaciones.
- ▶ Dado un esquema de relación $R(A_1, \dots, A_n)$
 - ▶ la operación de renombramiento denotada por $\rho_{S(B_1, \dots, B_n)}(R)$
 - ▶ produce un esquema de relación que tiene:
 - Nombre S
 - Atributos: (B_1, \dots, B_n) , donde cada B_i es
 - un nuevo nombre de atributo con el mismo dominio que A_i
 - Instancias validas las instancias validas de R
- ▶ Si solo se quieren renombrar los atributos se puede omitir S .
- ▶ Si solo se quiere cambiar el nombre de la relación se puede omitir (B_1, \dots, B_n)

Renombramiento

Esquema Relacional EMP (DNI, Nombre, Apellido 1, Apellido 2, Sueldo)

1. Renombramiento de DNI

$$\rho_{EMP(D, Nombre, Apellido 1, Apellido 2, SUELDO)} (EMP)$$

2. Renombramiento de todos los atributos

$$\rho_{EMP(D, N, A1, A2, S)} (EMP)$$

3. Copia de EMP a EMPLEADO sin renombrar atributos

$$\rho_{EMPLEADO} (EMP)$$

Operaciones de Recortar una Relación: Selección

▶ Consulta:

▶ Quiero los clientes de Madrid (teléfono = '91*')

▶ Que no son morosos

▶ Necesito: operación selección sobre el resultado de la operación diferencia

$$\sigma_{telefono = '91*' CL - MO}$$

▶ Uso:

▶ Solo necesito unas pocas filas de una relación

▶ que cumplan una condición

□ que, a su vez, puede ser compuesta por varias condiciones simples

▶ Recortar la relación en horizontal: quito filas que no quiero

▶ Letra sigma minúscula: σ

Op. de Recortar una Relación: Selección (Formal)

- ▶ La *operación selección* permite seleccionar tuplas de las relaciones.
- ▶ Dado un esquema de relación $R(A_1, \dots, A_n)$,
 - ▶ la operación de selección de R respecto a una condición C
 - ▶ se denota $\sigma_C(R)$
 - ▶ y produce un esquema de relación que tiene:
 - como atributos A_1, \dots, A_n ,
 - como instancias validas las instancias validas de R que cumplen C .

Selección : la Condición Formal

- ▶ C es la **condición de selección**. Es una formula construida de la siguiente forma:

- $C := \langle \text{nombreatributo} \rangle \langle \text{opcomp} \rangle \langle \text{nombreatributo} \rangle$
- $C := \langle \text{nombreatributo} \rangle \langle \text{opcomp} \rangle \langle \text{constante} \rangle$
- $C := C \wedge C$
- $C := C \vee C$
- $C := \neg C$

donde $\langle \text{opcomp} \rangle$ representa uno de los operadores booleanos:

$< ; > ; <= ; >= ; =$

Selección

a) Empleado con DNI 27347234T

$$\sigma_{(DNI=27347234T)}(EMP)$$

b) Empleados con salario entre 1200 y 1500 euros (ambas cantidades inclusive)

$$\sigma_{(salario \geq 1200) \wedge (salario \leq 1500)}(EMP)$$

c) Empleados que se apellidan García o ganan menos de 1000 euros

$$\sigma_{(apellido1="Garcia") \vee (salario \leq 1000)}(EMP)$$

Operaciones de Recortar una Relación: Proyección

- ▶ Consulta:

- ▶ Quiero solo las direcciones y nombres de los clientes
 - ▶ Que no son morosos (los que hemos llamado “BuenosClientes”)

- ▶ Necesito: operación proyección sobre el resultado anterior

$$\Pi_{(Nombres, Direccion)} (BuenosClientes)$$

→ recordamos cómo obtuvimos BuenosClientes

$$\rho_{BuenosClientes} (CL - MO)$$

- ▶ Uso:

- ▶ Solo necesito unos pocos atributos / columnas de una relación
- ▶ Recortar la relación en vertical: quito atributos que no quiero

- ▶ Letra pi mayúscula: Π

Proyección : formal

- ▶ La *operación proyección* permite extraer columnas de las relaciones.
- ▶ Dado un esquema de relación $R(A_1, \dots, A_n)$,
 - ▶ la operación de proyección de R respecto a
 - ▶ un conjunto de atributos $\{B_1, \dots, B_k\} \subseteq \{A_1, \dots, A_n\}$
 - ▶ se denota $\Pi_{(B_1, \dots, B_k)}(R)$
 - ▶ y produce un esquema de relación que tiene:
 - los atributos son B_1, \dots, B_k
 - y las instancias validas son
 - los valores en B_1, \dots, B_k de las instancias validas de R .

Proyección

- ▶ **Una proyección no repite tuplas**
 - ▶ porque su resultado es una relación
 - ▶ que a su vez es un conjunto de tuplas **distintas**

a) DNI y Sueldo de los empleados

$$\Pi_{(DNI, sueldo)} (EMP)$$

b) DNI de los empleados con salario superior a 1200 euros.

$$\Pi_{(DNI)} (\sigma_{(salario \geq 1200)} (EMP))$$

O bien

$$\text{Emp1} \longleftarrow \sigma_{(salario \geq 1200)} (EMP)$$

$$\text{Emp2} \longleftarrow \Pi_{(DNI)} (\text{Emp1})$$

Restricciones de integridad referencial con Proyección

- ▶ Las claves ajenas o FK se pueden representar así
- ▶ Ejemplo de Restaurantes

$$\Pi_{\text{restaurante}}(\text{Áreas Cobertura}) \subseteq \Pi_{\text{código}}(\text{Restaurantes})$$

$$\Pi_{\text{restaurante}}(\text{Horarios}) \subseteq \Pi_{\text{código}}(\text{Restaurantes})$$

$$\Pi_{\text{restaurante}}(\text{Platos}) \subseteq \Pi_{\text{código}}(\text{Restaurantes})$$

$$\Pi_{\text{cliente}}(\text{Pedidos}) \subseteq \Pi_{\text{DNI}}(\text{Clientes})$$

$$\Pi_{\text{restaurante, plato}}(\text{Contiene}) \subseteq \Pi_{\text{restaurante, nombre}}(\text{Platos})$$

$$\Pi_{\text{restaurante}}(\text{Contiene}) \subseteq \Pi_{\text{código}}(\text{Restaurantes})$$

$$\Pi_{\text{pedido}}(\text{AplicadoA}) \subseteq \Pi_{\text{código}}(\text{Pedidos})$$

$$\Pi_{\text{descuento}}(\text{AplicadoA}) \subseteq \Pi_{\text{código}}(\text{Descuentos})$$

Operaciones de Combinación de Relaciones

- ▶ Combinan dos relaciones no compatibles
 - ▶ Con algunos o todos los atributos diferentes entre ellas
- ▶ Tipos de operaciones de Combinación
 - ▶ Producto cartesiano $(R \times S)$
 - ▶ Reunión condicional $R \bowtie_C S = \sigma_C (R \times S)$
 - ▶ Reunión natural / Join natural $R \bowtie S$
- ▶ Extensiones del Algebra Relacional:
 - ▶ Combinación Externa de R y S
 - ▶ Combinación exterior izquierda $R \bowtie_{\leftarrow} S$
 - ▶ Combinación exterior derecha $R \bowtie_{\rightarrow} S$
 - ▶ Combinación exterior completa $R \bowtie_{\leftarrow \rightarrow} S$
- ▶ Operación División $R \div S$

Producto Cartesiano

- ▶ Dadas las relaciones/tablas:

Cliente: CL(DNI, NombreC, Dirección)

Ofertas: OF(Producto, descuento, caducidad)

- ▶ Consulta:

- ▶ Quiero datos para enviar a todos los clientes todas las ofertas

- ▶ Necesito combinar cada cliente con todas las ofertas

- ▶ Necesito: operación producto cartesiano sobre el resultado anterior

$(CL \times OF)$

no es $(CL \cup OF)$

- ▶ Uso:

- ▶ Combinar todas las filas de una relación con todas de otra relación

- ▶ La relación resultante tiene todos los atributos de ambas

- ▶ Suele ser la 1ª op. para aplicar otras después al resultado y

- ▶ conseguir unos datos más sofisticados

- ▶ Letra X mayúscula

Producto Cartesiano : Formal

- ▶ Dados dos esquemas de relación A y B ,
 - ▶ la *operación producto cartesiano* de A y B
 - ▶ se denota por $A \times B$
 - ▶ y se define como:
 - Atributos: (atributos de A) \cup (atributos de B)
- ▶ Si tienen algún nombre de atributo 'AC' común,
 - ▶ Se crean dos: $A.AC$, $B.AC$.
 - ▶ Para evitarlo utilizar renombramiento
- ▶ Instancias o filas: Son de la forma $a \times b$,
 - ▶ con a instancia de A , b instancia de B .

Producto Cartesiano: ejemplos

- Combinación de varias operaciones después de aplicar Producto Cartesiano

EMP (DNI, Nombre, Apellido1, Apellido2, Sueldo)

PROYECTOS (Codigo, DNIDir, Descripcion)

Queremos conocer los nombres y apellidos de los empleados que dirigen algún proyecto:

1. Hacemos el producto cartesiano de los empleados y los proyectos y nos quedamos con las tuplas en las que el DNI del empleado coincida con el DNI del director del proyecto

$$\text{DirProyecto} \longleftarrow \sigma_{(DNI=DNIDir)}(EMP \times PROYECTOS)$$

2. Nos quedamos con los datos de los directores de proyecto

$$\text{DatosDirProyecto} \longleftarrow \Pi_{(Nombre, Apellido1, Apellido2)}(\text{DirProyecto})$$

Producto Cartesiano: ejemplos

EMP (DNI, Nombre, Apellido1, Apellido2, Sueldo)

PROYECTOS (Codigo, DNIDir, Descripcion)

DISTRIBUCION (CodigoPr, DNIEmp, Horas)

Queremos conocer los nombres y apellidos de los empleados que trabajan en algún proyecto mas de 10 horas

1. Proyecto10 $\longleftarrow \sigma_{(Horas > 10)}(DISTRIBUCION)$
2. Emp10 $\longleftarrow \Pi_{(DNIEmp)}(Proyecto10)$
3. DNIEmp10 $\longleftarrow \sigma_{(DNI = DNIEmp)}(EMP \times Emp10)$
4. DatosEmp10 $\longleftarrow \Pi_{(Nombre, Apellido1, Apellido2)}(DNIEmp10)$

Producto Cartesiano: ejemplos

EMP (DNI, Nombre, Apellido1, Apellido2, Sueldo)

FAMILIA(DNI, Nombre)

Queremos conocer los DNI de los empleados que tienen al menos dos familiares registrados.

1. FamR $\longleftarrow \rho_{(DNI_F, Nombre_F)}(FAMILIA)$
2. Fam2 $\longleftarrow FAMILIA \times FamR$
3. ResA $\longleftarrow \sigma_{(DNI=DNIF) \wedge (Nombre \neq Nombre_F)}(Fam2)$
4. Resultado $\longleftarrow \pi_{(DNI)}(ResA)$

Producto Cartesiano: ejemplos

EMP (DNI, Nombre, Apellido1, Apellido2, Sueldo, Jefe)

Queremos conocer los DNI de los empleados hasta dos niveles por debajo del empleado con DNI='5278693L'

1. JefeNivel1 $\longleftarrow \rho_{(DNISub1)}(\Pi_{DNI}(\sigma_{Jefe="5278693L"}(EMP)))$
2. JefeNivel2 $\longleftarrow \Pi_{DNI}(\sigma_{(Jefe=DNISub1)}(JefeNivel1 \times EMP))$
3. Resultado $\longleftarrow JefeNivel1 \cup JefeNivel2$

Reunión condicional / Reunión/ theta-union/Join

► Consulta:

- Quiero datos para enviar a todos los clientes todas las ofertas
 - Pero solo los clientes con Telefono = '91*' (de Madrid)

► Necesito: operación Reunión Condicional

$$CL \bowtie_{\text{telefono} = '91*'} OF \Rightarrow \text{equivale a } \sigma_{\text{telefono} = '91*'} (CL \times OF)$$

► Uso:

- Combinar todas las filas de una relación con todas de otra relación
 - Aplicando condiciones para seleccionar solo algunas filas
- Es más eficiente que las dos operaciones por separado

► Signo: $\bowtie_{\text{condición}}$

- la condición puede ser entre atributos de las dos tablas

Reunión condicional / Reunión/ theta-union/Join

- ▶ Dados los esquemas de relación $R(A_1, \dots, A_n)$ y $S(B_1, \dots, B_m)$
 - ▶ se define la **reunión condicional** de R y S como:

$$R \bowtie_C S = \sigma_C (R \times S)$$

donde la condición C es idéntica en su forma

a las condiciones de selección de la operación **selección**

- ▶ Se llama **equijoin /equi-uni3n** a toda reuni3n cuya condici3n es
 - ▶ una conjunci3n de igualdades.
- ▶ El resultado puede ser el conjunto vac3o si
 - ▶ ninguna de las tuplas del producto cartesiano cumple la condici3n
- ▶ Las tuplas cuyos atributos de reuni3n sean nulos
 - ▶ no aparecen en el resultado.

Reunión condicional / Reunión/Join: Ejemplos

EMP (DNI, Nombre, Apellido1, Apellido2, Sueldo)
DISTRIBUCION (CodigoPr, DNIEmp, Horas)

Queremos conocer los nombres y apellidos de los empleados que trabajan en algún proyecto mas de 10 horas

1. Proyecto10 $\longleftarrow \sigma_{(Horas > 10)}(DISTRIBUCION)$
2. DNIEmp10 $\longleftarrow EMP \bowtie_{(DNI = DNIEmp)} Proyecto10$
4. DatosEmp10 $\longleftarrow \Pi_{(Nombre, Apellido1, Apellido2)}(DNIEmp10)$

Reunión Natural / Join Natural

- ▶ Dadas las relaciones/tablas:

Cliente: CL(DNI, NombreC, Dirección)

Invierte: IN (DNI, NombreE, Cantidad, Tipo)

- ▶ Consulta:

- ▶ Quiero los clientes y sus inversiones

- ▶ Solo aquellos que invierten en tipo 'BonoBueno'

- ▶ Necesito: operación Reunión Natural

- clientes que invierten $CL \bowtie IN$

- en tipo 'BonoBueno' $\sigma_{Tipo='BonoBueno'} (CL \bowtie IN)$

- ▶ Uso:

- ▶ Combinar todas las filas de una relación con todas de otra relación

- ▶ Y aplicar automáticamente la condición de

- que coincidan los valores de sus atributos comunes

- ▶ Es como una unión condicional con las condiciones implícitas

Reunión Natural /Join Natural: Formal

- ▶ La **combinación o reunión natural** permite combinar
 - ▶ ciertas selecciones y
 - ▶ un producto cartesiano en una sola operación
- ▶ Sean $R(A_1, \dots, A_n)$ y $S(B_1, \dots, B_m)$ dos esquemas de relación
- ▶ y (C_1, \dots, C_j) la lista de los atributos comunes a ambas relaciones.
- ▶ La operación de **reunión natural** se denota por $R \bowtie S$ y se define como:
 - ▶ Atributos: $\{A_1, \dots, A_n\} \cup \{B_1, \dots, B_m\}$
 - ▶ Instancias : Combinación de instancias validas de A y B
 - ▶ Donde coinciden los valores de $\{C_1, \dots, C_j\}$
- ▶ Si las relaciones no tuvieran atributos comunes,
 - ▶ la operación $R \bowtie S$ sería un producto cartesiano

Reunión Natural /Join Natural : Formal

- ▶ Está compuesta por las siguientes operaciones implícitas:

Sean dos esquemas de relación $R(A_1, \dots, A_n)$ y $S(B_1, \dots, B_m)$.

Entonces, $R \bowtie S =$

$$\rho_{(A_1, \dots, A_{n-j}, B_1, \dots, B_{m-j}, C_1, \dots, C_j)} (\Pi_{(A_1, \dots, A_{n-j}, B_1, \dots, B_{m-j}, R.C_1, \dots, R.C_j)} (\sigma_C (R \times S)))$$

$$\text{con } C = (R.C_1 = S.C_1 \wedge \dots \wedge R.C_j = S.C_j)$$

Reunión Natural /Join Natural : Ejemplos

EMP (DNI, Nombre, Apellido1, Apellido2, Sueldo)

PROYECTOS (Codigo, DNIDir Descripcion, CodigoDept)

DEP (CodigoDept, Nombre)

La combinación PROYECTOS \bowtie DEPT crea una relación con los atributos

(Codigo, DNIDir Descripcion, CodigoDept, Nombre)

que contiene los proyectos junto al nombre del departamento al que están asociados

La combinación EMP \bowtie DEPT devolvería una relación vacía (nombre de empleado y de departamento difícilmente coincidirán)

Extensión del algebra relacional

- ▶ Nuevos operadores necesarios para
 - ▶ consultas comunes en bases de datos relacionales.
- ▶ Sean $R(A_1, \dots, A_n)$ y $S(B_1, \dots, B_m)$ dos esquemas de relación.
- ▶ **Combinación externa o exterior**
 - ▶ para conservar en el resultado de la reunión de R y S
 - ▶ todas las tuplas que estén en R , en S o en ambas,
 - ▶ ya sea que existan o **no** tuplas coincidentes en ambas relaciones
 - ▶ Hay tres tipos: izquierda, derecha y completa
- ▶ La **combinación exterior izquierda** $R \sqcup S$
 - ▶ conserva todas las tuplas del esquema de relación R combinadas con
 - ▶ las tuplas coincidentes de S .
 - ▶ Si para una tupla de R no se encuentra una tupla coincidente en S
 - ▶ los atributos de S se “completan” con valores nulos

Extensión del algebra relacional

- ▶ La **combinación exterior derecha** $R \bowtie \sqsubset S$
 - ▶ conserva todas las tuplas del esquema de relación S combinadas con
 - ▶ las tuplas coincidentes de R
 - ▶ Si para una tupla de S no se encuentra una tupla coincidente en R
 - ▶ los atributos de R se “completan” con valores nulos
- ▶ La **combinación exterior completa** $R \boxtimes \sqsubset S$
 - ▶ conserva todas las tuplas de ambos esquemas de relación S
 - ▶ Si para una tupla de una de las relaciones
 - ▶ no se encuentra una tupla coincidente en la otra relación
 - ▶ los atributos de la otra relación se “completan” con valores nulos

Extensión del algebra relacional: Ejemplos

3274623R	Antonio Lopez	Retama 14
27347234T	Marta Sánchez	Loreto 134
85647456W	Alberto San Gil	Plaza Tornero 23
37562365F	María Puente	Travesía del Reloj 1
34126455Y	Juan Panero	Viviana 123

PR1	27347234T	20
PR1	34126455Y	10
PR2	27347234T	25
PR3	37562365F	45
PR2	37562365F	10

Empleados \bowtie Distribucion

3274623R	Antonio Lopez	Retama 14	NULL	NULL	NULL
27347234T	Marta Sánchez	Loreto 134	PR2	27347234T	25
85647456W	Alberto San Gil	Plaza Tornero 23	NULL	NULL	NULL
37562365F	María Puente	Travesía del Reloj 1	PR2	37562365F	10
37562365F	María Puente	Travesía del Reloj 1	PR3	37562365F	45
34126455Y	Juan Panero	Viviana 123	PR1	34126455Y	10

Operación División

- ▶ Dada la relación/tabla: Tarjetas: T (NumT, TipoT, Organización)
- ▶ Consulta:
 - ▶ Quiero las Organizaciones que tienen tarjetas de todos los tipos
- ▶ Necesito: operación División
 - Tipos de tarjetas que existen $\text{Tipos} \leftarrow \Pi_{(\text{TipoT})} (\text{Tarjetas})$
 - Organizaciones (y NumT) con todos los Tipos $\text{Tarjetas} \div \text{Tipos}$
 - Solo las organizaciones $\Pi_{(\text{Organizacion})} (\text{Tarjetas} \div \text{Tipos})$
- ▶ Uso:
 - ▶ Busco filas de una relación que tengan un atributo que
 - ▶ Toma todos los posibles valores de otro atributo

Operación División: Formal

- ▶ Se utilizan cuando se busca que
 - ▶ algún atributo de una relación tome
 - ▶ todos los valores de otro atributo en otra relación
- ▶ Sean $R(A_1, \dots, A_n)$ y $S(B_1, \dots, B_m)$ dos esquemas de relación
 - ▶ con $\{B_1, \dots, B_m\} \subset \{A_1, \dots, A_n\}$
- ▶ La **operación división** $R \div S$ produce un esquema de relación con:
 - ▶ Atributos: $\{C_1, \dots, C_j\} = \{A_1, \dots, A_n\} - \{B_1, \dots, B_m\}$
 - ▶ Instancias:
 - ▶ Dada r instancia válida de R ,
 - ▶ s instancia válida de S ,
 - ▶ una tupla t esta en $R \div S$ cuando
 - para todo q de S ,
 - la tupla que se obtiene al unir los valores de t y q esta en R

División

Determinar los datos personales de los empleados que trabajan en todos los proyectos que trabaja el empleado con DNI 8967866R

► $\text{ProyEmp} \leftarrow \Pi_{\text{CodigoPr}}(\sigma_{(\text{DNIEmp}=8967866R)}(\text{Distribucion}))$

$\text{DNIProyecto} \leftarrow \Pi_{(\text{CodigoPr}, \text{DNIEmp})}(\text{Distribucion})$

► $\text{DNIBuscados} \leftarrow \text{DNIProyecto} \div \text{ProyEmp}$

► $\text{DatosBuscados} \leftarrow \rho_{(\text{DNI})}(\text{DNIBuscados}) \bowtie \text{EMP}$

División

DNIProyecto

PR1	27347234T
PR1	34126455Y
PR2	27347234T
PR3	37562365F
PR3	27347234T

÷

ProyEmp

PR1
PR2
PR3



DNIBuscados

27347234T

Valores Nulos

- ▶ Los valores nulos afectan a las operaciones del álgebra relacional
- ▶ Cualquier comparación ($=$, \neq , $<$, \leq , $>$, \geq) que incluya un valor nulo,
 - ▶ tiene como resultado un valor **desconocido**
- ▶ Los valores nulos también aparecen en operaciones booleanas
 - $\text{NULO} \wedge \text{cierto} = \text{desconocido}$
 - $\text{NULO} \wedge \text{falso} = \text{falso}$
 - $\text{NULO} \wedge \text{NULO} = \text{desconocido}$
 - $\text{NULO} \vee \text{cierto} = \text{cierto}$
 - $\text{NULO} \vee \text{falso} = \text{desconocido}$
 - $\text{NULO} \vee \text{NULO} = \text{desconocido}$
 - $\neg \text{NULO} = \text{desconocido}$

Valores Nulos

- ▶ Selección
 - ▶ Si la evaluación del predicado sobre una tupla devuelve cierto,
 - ▶ la tupla se incluye en el resultado
 - ▶ en otro caso (falso o desconocido), no se incluye.
- ▶ Proyección
 - ▶ Trata los nulos como cualquier otro valor a la hora de eliminar duplicados
 - ▶ Si una parte de valores de las tuplas coinciden
 - ▶ y las dos tienen valores nulos en el resto: se tratan como duplicados
- ▶ Unión, intersección y diferencia
 - ▶ Estas operaciones tratan los valores nulos como las proyecciones.

Valores Nulos

- ▶ Reunión
 - ▶ Se utiliza la equivalencia
 - ▶ *reunión = selección (producto cartesiano)*
 - ▶ Por tanto dos tuplas con valor nulo en el atributo común **no** coinciden
- ▶ Reunión externa
 - ▶ Funciona de manera similar para aquellas tuplas que
 - ▶ cumplan las condiciones de reunión
 - ▶ Para el resto de tuplas
 - ▶ dependiendo del tipo de reunión externa
 - se incluirán rellenando con valores nulos.

Optimizaciones del Algebra Relacional

- ▶ Objetivo: mejorar el rendimiento de las operaciones, mediante
 - ▶ la reducción del volumen de datos obtenido en una operación para
 - ▶ Mejorar el rendimiento en las operaciones posteriores
- ▶ Orden de las operaciones encadenadas en una operación compuesta:
 1. Selección
 2. Proyección
 3. Cada Unión entre R_1 y R_2 que use más tuplas
 - ▶ según las cardinalidades de R_1 y R_2
 4. Unión Natural mejor que prod. Cart con una Selección posterior