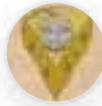


WUOLAH



QuesoViejo_
www.wuolah.com/student/QuesoViejo_

★ 128980

EJERCICIOS-RESUELTOS-Algebra-y-Calculo-Relacional.pdf

Ejercicios Resueltos Teoría



2º Ficheros y Bases de Datos



Grado en Ingeniería Informática



**Escuela Politécnica Superior
Universidad Carlos III de Madrid**

**Como aún estás en la portada, es
momento de redes sociales.
Cotilléanos y luego a estudiar.**



WUOLAH

MÁS

EJERCICIOS RESUELTOS ÁLGEBRA Y CÁLCULO RELACIONAL

Centro preparador y examinador de títulos oficiales

Simulacros de examen todos los viernes, OXFORD, PET, FIRST, CAE, TOELF, IELTS, TOEIC, gratis.

4 horas de speaking gratis a la semana

Grupos reducidos

Flexibilidad horaria

Clases de prueba gratis y sin compromiso



Centros en Moncloa y Leganés
www.thatsfun.es

📍 Ordóñez, 38, Leganés

☎ 911 03 58 00

📞 609 52 40 92

📍 Gaztambide 61, 1º 4 Madrid

☎ 911 27 32 04

📞 911 27 32 04

📧 info@thatsfun.es

That's fun!
Inglés LOW COST

Ejercicios de Exámenes de Años Anteriores

1. Si tenemos tres relaciones $r(R)$, $s(S)$ y $t(T)$. ¿Qué condiciones deben cumplir sus esquemas para que sea posible realizar la siguiente operación? Justifique su respuesta (0,75 puntos).

$$(T \times S) / \sigma_{\text{edad} > 17}(R)$$

La relación R debe tener un atributo "edad"

La relación resultante del producto cartesiano $(T \times S)$ debe contener los atributos de la relación R y alguno mas, de lo contrario daría error.

1. Si tenemos tres relaciones $r(R)$, $s(S)$ y $t(T)$. ¿Qué condiciones deben cumplir sus esquemas para que sea posible realizar la siguiente operación? Justifique su respuesta (0,75 puntos).

$$\Pi_{\text{nombre,apellidos}}(R \cap (S \bowtie T))$$

R debe tener el mismo grado y sus atributos los mismos dominios que el resultado de $S \bowtie T$ y como los nombres de los atributos de $(R \cap (S \bowtie T))$ son los de R , entonces R debe tener los atributos "nombre" y "apellidos"

* Nota: Si S y T no tienen ningún campo en común, el resultado de $S \bowtie T$ sería el producto cartesiano de

esas relaciones, por lo que no habría error.

1. Si tenemos tres relaciones $r(R)$, $s(S)$ y $t(T)$. ¿Qué condiciones deben cumplir sus esquemas para que sea posible realizar la siguiente operación? Justifique su respuesta (0,75 puntos).

$$\sigma_{\text{universidad} = \text{"UCA"}}(\Pi_{\text{dni}}(R) \times (S \cap T))$$

S y T deben tener el mismo grado y sus atributos deben tener los mismos dominios (esquemas compatibles) para que se pueda realizar la intersección.

Además, puesto que los atributos de $S \cap T$ van a tener los mismos nombres que los de S , S debe tener un atributo universidad.

R debe tener un atributo dni

3. Sean tres relaciones $r(R)$, $s(S)$ y $t(T)$. ¿Qué condiciones deben cumplir sus esquemas para que sea posible realizar la siguiente operación? Responda en función de los atributos de los esquemas de las tres relaciones, NO de las resultantes de las operaciones intermedias. Justifique su respuesta (0,4 puntos).

$$\sigma_{\text{univ}="UCA"}(R) \bowtie (\Pi_{\text{dni}}(\sigma_{\text{nombre}="Rubius"}(S \cap T)))$$

La relación R debe tener un atributo "universidad"

La relación S debe el mismo grado y sus atributos los mismos dominios que T

La relación T debe el mismo grado y sus atributos los mismos dominios que T

Además, $S \cap T$ tendrá los mismos nombres en los atributos que S, por lo que S debe tener los atributos "nombre" y "dni"

1. Sean tres relaciones $r(R)$, $s(S)$ y $t(T)$. ¿Qué condiciones deben cumplir sus esquemas para que sea posible realizar la siguiente operación? Justifique su respuesta (0,75 puntos). Indique la respuesta en función de los esquemas de las tres relaciones, de las resultantes de operaciones intermedias.

$$(\sigma_{\text{universidad}="UCA"}(R)) \cap (\Pi_{\text{dni}}(S) \bowtie T)$$

R debe tener un atributo "universidad".

Tanto R como el resultado de $(\Pi_{\text{dni}}(S) \bowtie T)$ deben tener el mismo grado y sus atributos el mismo dominio, para que pueda realizarse la intersección

S debe tener un atributo "dni"

3. Sean tres relaciones $r(R)$, $s(S)$ y $t(T)$. ¿Qué condiciones deben cumplir sus esquemas para que sea posible realizar la siguiente operación? Indique la respuesta en función de los esquemas de las tres relaciones, NO de las resultantes de operaciones intermedias. Justifique su respuesta (0,5 puntos).

$$\sigma_{\text{universidad}="UCA"}(R / (\Pi_{\text{dni}}(S \cap T)))$$

La relación R debe _____

La relación S debe _____

La relación T debe _____

S y T deben tener el mismo grado y sus atributos deben tener los mismos dominios para que se pueda realizar la intersección.

Además, puesto que los atributos de $S \cap T$ van a tener los mismos nombres que los de S, S debe tener un atributo dni

R debe tener un atributo "dni" para que se pueda realizar la división y un atributo "universidad" para la selección (puede tener más atributos además de esos 2)

3. Sean tres relaciones $r(R)$, $s(S)$ y $t(T)$. ¿Qué condiciones deben cumplir sus esquemas para que sea posible realizar la siguiente operación? Indique la respuesta en función de los esquemas de las tres relaciones, NO de las resultantes de operaciones intermedias. Justifique su respuesta (0,4 puntos).

$$\sigma_{\text{universidad}="UCA"}(R) / (\Pi_{\text{dni}}(S \cap \sigma_{\text{nombre}="Esther"}(T)))$$

La relación R debe _____

La relación S debe _____

La relación T debe _____

S y T deben tener el mismo grado y sus atributos deben tener los mismos dominios para que se pueda realizar la intersección.

Además, puesto que los atributos de $S \cap T$ van a tener los mismos nombres que los de S, S debe tener un atributo dni

T debe tener un atributo "nombre" para que se pueda realizar la selección.

R debe tener un atributo "dni" para que se pueda realizar la división y un atributo "universidad" para la selección (puede tener más atributos además de esos 2)

1. Sean tres relaciones $r(R)$, $s(S)$ y $t(T)$. ¿Qué condiciones deben cumplir sus esquemas para que sea posible realizar la siguiente operación? Justifique su respuesta (0,75 puntos). Indique la respuesta en función de los esquemas de las tres relaciones, de las resultantes de operaciones intermedias.

$$(\sigma_{\text{universidad}="UCA"}(R)) \bowtie (\Pi_{\text{dni}}(S-T))$$

S y T deben tener el mismo grado y sus atributos deben tener los mismos dominios para que se pueda realizar la diferencia.

Además, puesto que los atributos de $S-T$ van a tener los mismos nombres que los de S , S debe tener un atributo dni .

R debe tener un atributo universidad .

3. Sean tres relaciones $r(R)$, $s(S)$ y $t(T)$. ¿Qué condiciones deben cumplir sus esquemas para que sea posible realizar la siguiente operación? Indique la respuesta en función de los esquemas de las tres relaciones, NO de las resultantes de operaciones intermedias. Justifique su respuesta (0,4 puntos).

$$(\Pi_{\text{dni, país}}(R \bowtie \sigma_{\text{país}="URSS"}(S))) / T$$

La relación R debe _____

La relación S debe _____

La relación T debe _____

S debe tener un atributo país .

El resultado de $(R \bowtie \sigma_{\text{país}="URSS"}(S))$ debe tener

los atributos "dni" y "país", por lo que esos atributos deben estar en al menos una de las relaciones R y S.

T solo puede tener los atributos dni y país (uno de ellos, si tuviera los dos daría error) y no puede tener ningún otro atributo.

3. Dados los esquemas relacionales siguientes en los que los atributos pertenecientes a la clave primaria están subrayados y los que tienen igual nombre en dos relaciones distintas son claves foráneas:

Piragüistas(cod_piragüista, nombre_p, sexo, mano, año_nacim)

Equipos(cod_equipo, cod_torneo, cod_director, país)

Selecciones(cod_piragüista, cod_equipo)

Directores(cod_director, nombre_d, teléfono, web)

Torneos(cod_torneo, fecha, país_celebración)

Expresar las siguientes consultas (cada solución correcta vale 0,25 puntos y cada explicación de la consulta 0,25 puntos):

a) Obtener en álgebra relacional el nombre y teléfono de los directores que tienen página web y dirigieron equipos italianos en torneos celebrados en Ucrania donde jugó algún piragüista que usara la mano izquierda.

b) Obtener en álgebra relacional el nombre de los directores que han dirigido algún equipo en todos los torneos del siglo pasado.

c) Obtener en cálculo relacional de tuplas la lista de equipos en los que haya piragüistas zurdos y diestros y tengan al menos diez años de diferencia de edad entre ellos.

d) Obtener en cálculo relacional de dominios la lista de países donde se celebraron torneos en la década de los 90 y en los que participó una mujer piragüista representando a Reino Unido.

$$a) A \leftarrow \pi_{\text{cod_equipo}, \text{cod_director}} (\sigma_{\text{país} = \text{Italia}} (\text{Equipos}) \bowtie \sigma_{\text{país_celeb} = \text{Ucrania}} (\text{Torneos}))$$

$$B \leftarrow \pi_{\text{cod_piragüista}} (\sigma_{\text{mano} = \text{izquierda}} (\text{Piragüistas}))$$

$$C \leftarrow (\text{Directores} \bowtie (\text{Selecciones} \bowtie A \bowtie B))$$

$$\pi_{\text{nombre_d}, \text{teléfono}} (\sigma_{\text{web} \neq \text{NULL}} (C))$$

b) $\forall \text{cod-torneo} \in \text{Torneos} \left(\text{fecha} \geq 31/3/3900 \text{ AND } \text{fecha} \leq 31/12/3999 \right)$

$\forall \text{nombre-d} \in \text{Directores} \times (\text{Equipos} / A)$

c) $\{ t \mid t \in \text{Equipos} \wedge \exists p_1 (p_1 \in \text{Piragüistas}) \wedge$

$\exists p_2 (p_2 \in \text{piragüistas}) \wedge (p_1.\text{mano} \neq p_2.\text{mano}) \wedge$

$(\text{ABS}(p_1.\text{año-nacim} - p_2.\text{año-nacim}) \geq 10) \wedge$

$\exists s_1 (s_1 \in \text{Selecciones})$

$\wedge \exists s_2 (s_2 \in \text{Selecciones}) \wedge (s_1.\text{cod-piragüista} = p_1.\text{cod-piragüista})$

$\wedge (s_2.\text{cod-piragüista} = p_2.\text{cod-piragüista})$

$\wedge (s_1.\text{cod-equipo} = s_2.\text{cod-equipo})$

$\wedge (s_1.\text{cod-equipo} = t.\text{cod-equipo})$

$\{$

d) $\{ p \mid \text{Torneos}(c \mid p) \wedge \text{Equipos}(e \mid c - -)$

$\wedge \text{Piragüistas}(k - \text{"mujer"} - -) \wedge \text{Selecciones}(k \mid e) \wedge$

$(p \geq 31/3/3900) \wedge p(\leq 31/12/3999) \{$

3. Dados los esquemas relacionales siguientes en los que los atributos pertenecientes a la clave primaria están subrayados y los que tienen igual nombre en dos relaciones distintas son claves foráneas:

Familias(cod_familia, apellidos, contacto, núm_miembros, país)

Acogidas(cod_gato, cod_familia, fecha_ini, fecha_fin, causa_fin)

Gatos(cod_gato, nombre, sexo, raza, año_nacim, cod_gato_madre, cod_gato_padre) →

cod_gato_madre y cod_gato_padre son, respectivamente, los códigos de la madre y padre del gato/a

Expresar las siguientes consultas (cada solución correcta vale 0,3 puntos y cada explicación de la consulta 0,2 puntos):

a) Obtener en álgebra relacional la cantidad de familias que nunca han acogido a un gato de raza siberiana.

b) Obtener en álgebra relacional el nombre de las gatas (sexo femenino) que han estado de acogida en todas las familias de Andorra.

c) Obtener en cálculo relacional de tuplas la lista de familias que hayan acogido dos gatos en menos de un mes.

d) Obtener en cálculo relacional de dominios la lista de gatos (machos) que siempre han tenido descendencia con la misma gata.

$$a) A \leftarrow \text{Acogidas} \bowtie \sigma_{\text{raza} = \text{siberiana}} (\text{Gatos})$$

$$B \leftarrow \pi_{\text{cod-familia}} (\text{Familias}) - \pi_{\text{cod-familia}} (A)$$

$$\pi_{\text{COUNT cod-familias}} (\sigma_{\text{COUNT cod-familias}} (B))$$

b)

$$A \leftarrow \pi_{\text{cod-gato, cod-familia}} (\sigma_{\text{país} = \text{Andorra}} (\text{Acogidas})) \mid \pi_{\text{cod-gato}} (\sigma_{\text{sexo} = \text{femenino}} (\text{Gatos}))$$

$$\pi_{\text{nombre}} (\text{Gatos} \bowtie A)$$

$$\begin{aligned}
 c) \quad & \exists f \mid f \in \text{Familias} \wedge \exists g_1 (g_1 \in \text{Gatos}) \wedge \exists g_2 (g_2 \in \text{Gatos}) \\
 & \wedge \exists a_1 (a_1 \in \text{Acogidas}) \wedge \exists a_2 (a_2 \in \text{Acogidas}) \wedge \\
 & (a_1.\text{cod-gato} = g_1.\text{cod-gato}) \wedge (a_2.\text{cod-gato} = g_2.\text{cod-gato}) \\
 & \wedge (a_1.\text{cod-familia} = a_2.\text{cod-familia}) \\
 & \wedge (a_1.\text{cod-familia} = f.\text{cod-familia}) \\
 & \wedge (ABS(a_1.\text{fecha-ini} - a_2.\text{fecha-ini}) \leq 30) \\
 & \{ \text{Valor absoluto}
 \end{aligned}$$

* Doy por hecho que 3 mes tiene 30 días y que se pueden restar 2 fechas.

$$\begin{aligned}
 d) \quad & \exists p \exists x \exists y \mid (7EC) \quad \text{Gatos}(c \text{ --- } mp) \wedge \\
 & \text{Gatos}(p \text{ "macho" } x \text{ y ---}) \wedge \text{Gatos}(m \text{ "hembra" ---})
 \end{aligned}$$

"No existe ningún gato que para el padre p no tenga a la madre m"

3. Dados los esquemas relacionales siguientes en los que los atributos pertenecientes a la clave primaria están subrayados y los que tienen igual nombre en dos relaciones distintas son claves foráneas:

Ovejas(cod_oveja, sexo, raza, año_nacim, cod_rebaño)

Rebaños(cod_rebaño, cod_granja)

Pastores(cod_pastor, nombre, teléfono)

Granjas(cod_granja, localidad)

Vigila(cod_pastor, cod_rebaño, fecha, horas)



Expresar las siguientes consultas (cada solución correcta vale 0,25 puntos y cada explicación de la consulta 0,25 puntos):

a) Obtener en álgebra relacional el código de los pastores que vigilaron rebaños de Montequaje con ovejas de raza payoya durante el mes de agosto del año 2016.

b) Obtener en álgebra relacional el nombre y teléfono de los pastores que han vigilado a ovejas de todas las razas.

c) Obtener en cálculo relacional de tuplas la lista de rebaños "puros", es decir aquellos que tienen todas sus ovejas del mismo tipo.

d) Obtener en cálculo relacional de dominios la lista de granjas donde han nacido ovejas en el año 2017.

a) $A \leftarrow \pi_{\text{cod_rebaño}} (\sigma_{\text{raza} = \text{payoya}} (\text{Ovejas}) \bowtie \text{Rebaño})$

$\pi_{\text{cod_pastor}} \left(\sigma_{\substack{\text{fecha} \geq 7/31/8/2016 \text{ AND} \\ \text{fecha} \leq 31/8/2016}}^{(\text{Vigila})} \bowtie A \right)$

b) $A \leftarrow \pi_{\text{cod_pastor}} \left(\pi_{\text{raza}} (\text{Ovejas} \bowtie \text{Vigila}) \right)$

$B \leftarrow A / \pi_{\text{raza}} (\text{Ovejas})$

$\pi_{\text{nombre}, \text{teléfono}} (B \bowtie \text{Pastores})$

c) $\{ t \mid t \in \text{Rebaño} \wedge \forall p_1 (p_1 \in \text{Ovejas} \rightarrow \exists p_2 (p_2 \in \text{Ovejas}) ((p_1.\text{cod-rebaño} = t.\text{cod-rebaño}) \wedge (p_2.\text{cod-rebaño} = p_1.\text{cod-rebaño}) \wedge (p_1.\text{raza} = p_2.\text{raza})))$

d) $\{ g \mid \exists c \text{ Ovejas}(c) \wedge \exists r \text{ Rebaños}(r) \wedge \exists g \text{ Granjas}(g) \}$

3. Dados los esquemas relacionales siguientes en los que los atributos pertenecientes a la clave primaria están subrayados y los que tienen igual nombre en dos relaciones distintas son claves foráneas:

Futbolistas(cod_futbolista, nombre_f, sexo, nacionalidad, año_nacim, cod_equipo, año)

Equipos(cod_equipo, cod_liga, país) → es el país del equipo, no de la liga. Por ejemplo, los equipos de Andorra tienen que jugar las ligas de otra federación)

Árbitros(cod_árbitro, nombre_a, teléfono, licencia) → licencia puede ser regional, nacional o internacional

Partidos(cod_partido, cod_árbitro, fecha, cod_liga)

Ligas(cod_liga, federación, profesional) → profesional es un booleano que vale TRUE si la liga es profesional y FALSE si es amateur.

Expresar las siguientes consultas (cada solución correcta vale 0,25 puntos y cada explicación de la consulta 0,25 puntos):

a) Obtener en álgebra relacional el nombre y teléfono de los árbitros que arbitraron partidos en las ligas de Kazajistán (se almacena en el campo *federación*) donde jugó algún futbolista español durante el mes de agosto del año 2017.

b) Obtener en álgebra relacional el nombre de las futbolistas que han jugado en todos los equipos de Turquía.

c) Obtener en cálculo relacional de tuplas la lista de equipos "mixtos", es decir aquellos que tienen al menos un jugador de cada sexo.

d) Obtener en cálculo relacional de dominios la lista de partidos de ligas *amateur* en el año 2017 en que pitaron árbitros con licencia nacional o internacional.

a)

$A \leftarrow \Pi_{\text{cod_arbitros}, \text{cod_liga}, \text{fecha}} (\text{Partidos}) \bowtie \sigma_{\text{federación} = \text{"Kazajistán"}} (\text{Ligas})$

$B \leftarrow \sigma_{\text{nacionalidad} = \text{España}} (\text{Futbolista}) \bowtie \text{Equipos}$

$\Pi_{\text{nombre}, \text{teléfono}} (\sigma_{\text{fecha} \geq 1/8/2017 \text{ AND } \text{fecha} \leq 31/8/2017} (A \bowtie B \bowtie \text{Arbitro}))$

b) $A \leftarrow \prod_{\text{equipo}} (\sigma_{\text{pais} = \text{Turquía}} (\text{Equipos}))$

$\prod_{\text{nombre.g}} (\sigma_{\text{sexo} = \text{Mujer}} (\text{Futbolistas}) / A)$

c) $\{ e \mid e \in \text{Equipos} \wedge \exists g (g \in \text{Futbolistas})$
 $\wedge \exists g (g \in \text{Futbolistas}) \wedge g.\text{sexo} = \text{Hombre} \wedge$
 $g.\text{sexo} = \text{Mujer} \wedge e.\text{cod-equipo} = g.\text{cod-equipo} \wedge$
 $e.\text{cod-equipo} = g.\text{cod-equipo} \}$

d) $\{ ck gq \mid \text{Partidos}(ck gq) \wedge \text{Ligas}(q - \text{false})$
 $\wedge g \geq 31/3/2037 \wedge g \leq 31/12/2037 \wedge$
 $\text{Arbitro}(k - 1) \wedge (l = \text{"Nacional"} \vee l = \text{"Internacional"}) \}$

Ejemplo: Dar 2 tuplas válidas y 2 no válidas en Partidos

C.P.

Válidas: $\langle 1, 1, "23/04/2039", 1 \rangle$

$\langle 1, 2, "23/04/2039", 2 \rangle$

No válidas: (suponiendo que las anteriores ya están insertadas)

$\langle 1, 1, "23/04/2039", 1 \rangle$ (CP repetida)

$\langle \text{null}, 2, "23/04/2039", 1 \rangle$ (CP nula)

3. Dados los esquemas relacionales siguientes en los que los atributos pertenecientes a la clave primaria están subrayados y los que tienen igual nombre en dos relaciones distintas son claves foráneas:

Músicos(cod_músico, nombre_m, sexo, nacionalidad, instrumento, cod_grupo)

Grupos(cod_grupo, nombre, año_fundación)

Apoderado(cod_apoderado, nombre_m, sexo, teléfono)

Representaciones(cod_apoderado, cod_grupo, comisión)

Conciertos(cod_concierto, cod_grupo, fecha, país_concierto)

Expresar las siguientes consultas (cada solución correcta vale 0,25 puntos y cada explicación de la consulta 0,25 puntos):

a) Obtener en álgebra relacional el país y fecha de los conciertos que el apoderado R18 organizó con grupos fundados en el siglo XXI en los que una mujer tocara el arpa.

b) Obtener en álgebra relacional si existe algún apoderado representante a todos los músicos que tocan el chelo.

c) Obtener en cálculo relacional de tuplas la lista de grupos formados sólo por hombres pero representados por una mujer.

d) Obtener en cálculo relacional de dominios la lista de conciertos donde tocaran grupos con algún músico ruso y su apoderados se llevara una comisión de al menos 1000 euros.

a)

$$A \leftarrow \pi_{\text{cod_grupo}}(\sigma_{\text{cod_apoderado} = R18}(\text{Representaciones}))$$

$$B \leftarrow A \bowtie \pi_{\text{cod_grupo}}(\sigma_{\text{año_fundación} \geq 2000 \wedge \text{año_fundación} < 2100}(\text{Grupos}))$$

$$C \leftarrow B \bowtie \pi_{\text{cod_grupo}}(\sigma_{\text{sexo} = \text{mujer} \wedge \text{instrumento} = \text{Arpa}}(\text{Músicos}))$$

$$\pi_{\text{país}, \text{fecha}}(C \bowtie \text{Conciertos})$$

b)

$A \leftarrow \prod_{\text{cod-grupo}} (\text{Instrumento} = \text{chelo (Musico)})$

$B \leftarrow \prod_{\text{cod-apoder}} (\text{Representaciones} / A)$

Apoderado \bowtie B

c) $\} \in / \in \text{Grupos} \wedge \forall m (m \in \text{Musicos}) m.\text{cod-grupo} =$

$t.\text{cod-grupo} \wedge m.\text{sexo} = \text{hombre} \wedge \exists r (r \in \text{Representaciones})$

$r.\text{cod-grupo} = t.\text{cod-grupo} \wedge \exists a (a \in \text{Apoderado}) \wedge$

$a.\text{cod-apoderado} = r.\text{cod-apoderado} \wedge a.\text{sexo} = \text{mujer} \{$

d)

$\} \in / \text{Conciertos (cg --)} \wedge \text{Musicos (m -- 'Rusia'-g)}$

$\wedge \text{representaciones (a gd)} \wedge d > 1000 \{$

1) Expresar en álgebra relacional y obtener la instancia resultante de los siguientes apartados, usando las relaciones adjuntas.

- Los apellidos de todos los autores.
- Los datos del autor cuyo apellido es *Ullman*.
- El nombre y el apellido de los autores con código mayor que 1.
- Los datos de los socios y de los libros.
- Los datos de todos los autores y de todos los editores.
- Los datos de todos los autores que no sean, a su vez editores.
- Los datos de los autores que también son editores.
- Los datos de los socios, cuyo apellido sea *Manrique*, y de los libros.
- Los datos de los libros que tienen en préstamo cada socio.
- Los datos de los socios que tienen en préstamo un ejemplar de todos los libros.

Autores

<i>Cod_Aut</i>	<i>Nombre</i>	<i>Apellido</i>
1	C. J.	Date
2	A.	De Miguel
3	D. J.	Ullman

Editores

<i>Cod_Edit</i>	<i>Nombre</i>	<i>Apellido</i>
1	S.	Ceri
2	A.	De Miguel
3	P.	Chen

Socios

<i>Cod_Soc</i>	<i>Nombre</i>	<i>Apellido</i>
1	E.	Manrique
2	M.	García

Libros

<i>Cod_Lib</i>	<i>Nombre</i>	<i>Cod_Edit</i>	<i>Cod_Aut</i>
1	Introducción a los Sistemas de BD	1	1
2	Fundamentos y modelos de BD	2	2
3	Sistema de BD	3	3

Préstamos

<i>Cod_Lib</i>	<i>Cod_Soc</i>	<i>Fecha</i>
1	1	10-04-98
2	1	18-02-99
3	1	23-07-99
1	2	01-12-97

1) Álgebra: $\Pi_{\text{apellido}}(\text{autores})$

SQL: `select * from autores`

TABLA:

<u>Apellido</u>
De Miguel
Ullman

2) Álgebra: $\sigma_{\text{apellido} = \text{'Ullman'}}(\text{autores})$

SQL: `select apellido from autores where apellido = 'Ullman'`

TABLA:

<u>Cod. aut</u>	<u>Nombre</u>	<u>Apellido</u>
2	D.J.	Ullman

3) Álgebra: $\Pi_{\text{nombre, apellido}}(\sigma_{\text{cod. aut} > 1}(\text{Autores}))$

SQL: `select nombre, apellido from autores where cod > 1`

TABLA:

<u>nombre</u>	<u>apellido</u>
A.	De Miguel
D.J.	Ullman

* Si nos piden todas las combinaciones

47 Álgebra: Socios x Libros

SQL: Select * from Socios, Libros

TABLA:

Cod Soc	N	Ap	CL	N	C. Edit	C. Aut
1	E	Manrique	1	Intro...	1	1
1	E	Manrique	2	Fun...	2	2
1	E	Manrique	3	Sis...	3	3
2	M	Gar	1	Intro...	1	1
2	M	Gar	2	Fun...	2	2
2	M	Gar	3	Sis...	3	3

* En este caso, mejor la OUTERUNION

47 Álgebra: Socios OUTERUNION Libros

TABLA:

Cod-socio	Cod-lib	Nombre	Apellido	Cod-Edit	Cod.Aut
1	w	E.	Manrique	w	w
2	w	M.	García	w	w
w	1	Introducción...	w	1	1
w	2	Fundamentos...	w	2	2
w	3	Sistema de...	w	3	3

No se puede hacer una unión normal (U) porque no tienen esquemas compatibles.

*Estos sí tienen el mismo esquema. (entero, cadena, cadena)

5) Álgebra: Autores U Editores

TABLA:

Cod-Aut	Nombre	Apellido
3	C.J.	Date
2	A.	De Miguel
3	D.J.	Ullman
3	S.	Ceri
2	A.	De Miguel
3	P.	Chen

6) Álgebra: Autores - Editores

SQL: $\text{Select } * \text{ from Autores where código NOT IN}$
 $\text{Select código from Editores}$

* Suponemos que el código coinciden en ambos (de Miguel con código 2 en ambos)

TABLA:

Cod-Aut	Nombre	Apellido
3	C.J.	Date
3	D.J.	Ullman

7) Álgebra: Autores \cap Editores

SQL: Select * from Autores where código IN
Select código from Editores

Select * from ⁰autores INTERSECT Select * from Editores

TABLA:

Cod-Aut	Nombre	Apellido
2	A.	De Miguel

Equivalente a Primero Selección luego Producto cartesiano

8) Álgebra: Socios \bowtie Libros, θ = Apellido = Manrique

SQL: Select * from Socios, Libros where
Socios.apellido = "Manrique"

Cod.Soc	N	Ap	CL	N	C. Edit	C. Aut
1	E	Manrique	1	Intro...	1	1
1	E	Manrique	2	Fun...	2	2
1	E	Manrique	3	Sis...	3	3

97 Álgebra: Libros \bowtie Prestamo

SQL: select Libros.* , Prestamos.* $\stackrel{*}{=}$ from Libros L
 Prestamos P where L.cod-lib = P.cod-lib

Cod-Libro	Nom	Cod-Edit	Cod-Aut	Cod-Soc	Fecha
1	.	.	.	1	.
1	.	.	.	2	.
2	.	.	.	1	.
3	.	.	.	1	.

10) Alg: Socios \bowtie ($\Pi_{\text{cod-lib, cod-soc}}^{\text{(Prestamos)}} / \Pi_{\text{cod-lib}} (\text{Libros})$)

Cod-soc	Nombre	Apellido
1	E.	Manrique

Alg: Socios \bowtie (Prestamos / $\Pi_{\text{cod-lib}} (\text{Libros})$)

Cod-soc	Nombre	Apellido
1	E.	Manrique

3) Dado el siguiente esquema relacional¹

Cliente (id_cliente, nom_cliente, renta_anual)

Embarque (num_embarque, id_cliente, peso, id_camión, destino)

expresar en álgebra relacional las siguientes consultas:

a) ¿Qué clientes tienen una renta anual que excede de 30.000 €?

b) ¿Cuál es el nombre del cliente nº 433?

c) ¿Cuál es la ciudad destino del embarque número 3244?

a) $\sigma_{\text{renta_anual} > 30000}(\text{Cliente})$

b) $\pi_{\text{nombre}}(\sigma_{\text{id_cliente} = 433}(\text{Cliente}))$

c) $\pi_{\text{destino}}(\sigma_{\text{num-embarque} = 3244}(\text{Embarque}))$

d) ¿Qué camiones han transportado paquetes con un peso por encima de los 100kg?

e) ¿Cuáles son los nombres de los clientes que han enviado paquetes a la ciudad de Vigo?

f) A qué destinos han enviado paquetes los clientes con renta anual inferior a 6.000€?

d) $\pi_{\text{id-camion}}(\sigma_{\text{peso} > 100}(\text{Embarque}))$

e) $\pi_{\text{nom-cliente}}(\text{Cliente} \bowtie \sigma_{\text{destino} = \text{"Vigo"}}(\text{Embarque}))$

f) $\pi_{\text{destino}}(\text{Embarque} \bowtie \sigma_{\text{renta-anual} < 6000}(\text{Cliente}))$

5) Tenemos el siguiente esquema relacional²

<i>Cliente</i>	(<u>id_cliente</u> , nom_cliente, renta_anual)
<i>Embarque</i>	(<u>num_embarque</u> , id_cliente, peso, id_camión, destino)
<i>Camión</i>	(<u>id_camión</u> , nom_chófer)
<i>Ciudad</i>	(<u>nomb_ciudad</u> , población)

expresar en álgebra relacional las siguientes consultas:

- Una lista con los números de los embarques que pesan más de 20kg.
- Los nombres de los clientes con más de 60.000 € de renta anual.
- El chófer del camión nº 45.
- Los nombres de las ciudades que han recibido envíos que pesan más de 100kg.

- $\pi_{\text{num-embarque}}(\sigma_{\text{peso} > 20}(\text{Embarque}))$
- $\pi_{\text{nom-cliente}}(\sigma_{\text{renta-anual} > 60000}(\text{cliente}))$
- $\pi_{\text{nom-chófer}}(\sigma_{\text{id-camión} = 45}(\text{Camión}))$
- $\pi_{\text{destino}}(\sigma_{\text{peso} > 100})$

- Los nombres y la renta anual de los clientes que han enviado paquetes que pesan más de 100kg.
- El número de los camiones que han transportado embarques que pesan más de 100kg.
- Los nombres de los chóferes que han distribuido envíos que pesan más de 100kg.
- Las ciudades que han recibido embarques de clientes que tienen una renta anual por encima de los 90.000 €.

- $\pi_{\text{nom-cliente}, \text{renta-anual}}(\text{Cliente} \bowtie (\pi_{\text{id-cliente}}(\sigma_{\text{peso} > 100}(\text{Embarque}))))$
Otra opción: $\pi_{\text{nom-cliente}, \text{renta-anual}}(\sigma_{\text{peso} > 100}(\text{Cliente} \bowtie \text{Embarque}))$
- $\pi_{\text{id-camión}}(\sigma_{\text{peso} > 100}(\text{Embarque}))$
- $\pi_{\text{nom-chófer}}(\text{Camión} \bowtie (\pi_{\text{id-camión}}(\sigma_{\text{peso} > 100}(\text{Embarque}))))$
- $\pi_{\text{destino}}(\text{Embarque} \bowtie (\pi_{\text{id-cliente}}(\sigma_{\text{renta-anual} > 90000}(\text{Clientes}))))$

- i) Los clientes que tienen una renta anual por encima de los 90.000 € que han enviado paquetes con peso menor de un kilo.
- j) Los clientes que tienen una renta anual por encima de los 90.000 € que han enviado paquetes con peso menor de un kilo o han enviado embarques a la ciudad de Madrid.
- k) Los clientes cuyos envíos han sido distribuidos por el chófer Juan.
- l) Los conductores que han distribuido envíos de clientes con renta anual por encima de los 120.000 € a ciudades con población superior a un millón de habitantes.

i) $\Pi_{\text{nom-cliente}} (\sigma_{\text{renta_anual} > 90000} (\text{Cliente} \bowtie (\Pi_{\text{id-cliente}} (\sigma_{\text{peso} < 1} (\text{Embarque}))))$

j) $\Pi_{\text{nom-cliente}} (\sigma_{\text{renta_anual} > 90000} (\text{Cliente} \bowtie \Pi_{\text{id-cliente}} (\sigma_{\text{peso} < 1 \text{ OR } \text{destino} = 'Madrid'})))$

k) $\Pi_{\text{nom-cliente}} (\text{Cliente} \bowtie \Pi_{\text{id-cliente}} (\text{Embarque} \bowtie (\sigma_{\text{nom-chofer} = 'Juan'} (\text{Camion}))))$

l) $\text{Embarque}(\dots, \text{nomciudad}) \leftarrow \text{Embarque}(\dots, \text{destino})$

$\Pi_{\text{nom-chofer}} (\text{Camion} \bowtie (\sigma_{\text{renta_anual} > 120000} (\text{Cliente} \bowtie (\text{Embarque} \bowtie (\sigma_{\text{población} > 1000000} (\text{Ciudad}))))))$

- m) Clientes que han recibido envíos distribuidos por todos los chóferes.
- n) Ciudades que han recibido embarques de todos los clientes.
- ñ) Chóferes que han distribuido envíos a todas las ciudades.
- o) Clientes que han enviado embarques a toda ciudad con población mayor de 500.000 habitantes³.

$$m) \text{ Cliente } \bowtie \left(\pi_{\text{id-cliente}, \text{id-camion}}^{(\text{Embarque})} / \pi_{\text{id-camion}}^{(\text{Camion})} \right)$$

* Nota: Suponemos que no hay 2 camiones con el mismo choffer.

$$n) \pi_{\text{destino}} (\text{Embarque} / \pi_{\text{id-cliente}} (\text{cliente}))$$

$$\tilde{n}) \text{ Embarque } (\dots, \text{nomb-ciudad}) \leftarrow \text{Embarque } (\dots, \text{destino})$$

$$A \leftarrow \left(\pi_{\text{id-camion}, \text{nomb-ciudad}} (\text{Embarque}) / \pi_{\text{nomb-ciudad}} (\text{Ciudad}) \right)$$

$$\pi_{\text{nomb-choffer}} (\text{Camion } \bowtie A)$$

* Cuando las expresiones son muy largas conviene usar variables intermedias.

$$o) \text{ Embarque } (\dots, \text{nomb-ciudad}) \leftarrow \text{Embarque } (\dots, \text{destino})$$

$$A \leftarrow \pi_{\text{nomb-ciudad}} (\sigma_{\text{poblacion} > 500000} (\text{Ciudad}))$$

$$\text{Cliente } \bowtie \left(\pi_{\text{id-cliente}, \text{nomb-ciudad}} (\text{Embarque}) / A \right)$$

6) Expresar en cálculo relacional las consultas del ejercicio 5.

<i>Cliente</i>	(<u>id_cliente</u> , nom_cliente, renta_anual)
<i>Embarque</i>	(<u>num_embarque</u> , id_cliente, peso, id_camión, destino)
<i>Camión</i>	(<u>id_camión</u> , nom_chófer)
<i>Ciudad</i>	(<u>nomb_ciudad</u> , población)

- Una lista con los números de los embarques que pesan más de 20kg.
- Los nombres de los clientes con más de 60.000 € de renta anual.
- El chófer del camión nº 45.
- Los nombres de las ciudades que han recibido envíos que pesan más de 100kg.

a)

Tuplas: $\{ t.\text{num-embarque} \mid t \in \text{Embarque} \wedge$
 $t.\text{peso} > 20 \}$

Dominio: $\{ n \mid (\exists i)(\exists p)(\exists k)(\exists d) (\text{Embarque}(ni pkd)) \wedge p > 20 \}$

* La notación de Access que usaré más adelante es más sencilla y no requiere poner $(\exists i)(\exists p) \dots$. Se pueden usar ambas para el cálculo de dominios.

b)

Tuplas: $\{ c.\text{nom_cliente} \mid c \in \text{Cliente} \wedge$
 $c.\text{renta-anual} > 60000 \}$

Dominios $\{ n \mid (\exists i)(\exists r) (\text{Cliente}(inr)) \wedge r > 60000 \}$

c)

Tuplas: $\{ c \cdot \text{nom-chófer} \mid c \in \text{Camion} \wedge c.\text{id-camion} = 45 \}$

Dominios: $\{ n \mid (\exists i) (\text{Camión}(i, n) \wedge i = 45) \}$

d)

Tuplas: $\{ e.\text{destino} \mid (e \in \text{Embarque}) \wedge (e.\text{peso} > 100) \}$

Dominios: $\{ e \mid (\exists a)(\exists b)(\exists c)(\exists d)(\exists e) \text{Embarque}(a, b, c, d, e) \wedge c > 100 \}$

e) Los nombres y la renta anual de los clientes que han enviado paquetes que pesan más de 100kg.

f) El número de los camiones que han transportado embarques que pesan más de 100kg.

g) Los nombres de los chóferes que han distribuido envíos que pesan más de 100kg.

h) Las ciudades que han recibido embarques de clientes que tienen una renta anual por encima de los 90.000 €.

e)

Tuplas: $\{ c.\text{nom-cliente}, c.\text{renta-anual} \mid c \in \text{Cliente} \wedge \exists e$
 $(e \in \text{Embarque}) \wedge e.\text{id-cliente} = c.\text{id-cliente} \wedge$
 $e.\text{peso} > 100 \}$

Dominios $\{ n, r \mid (\exists c)(\exists y)(\exists p)(\exists x)(\exists d) \quad (\text{Cliente}(c, n, r)$
 $\text{Embarque}(m, y, p, x, d) \wedge (i = y) \wedge p > 100 \}$

8)

Tuplas: $\{ e.id_camion \mid e \in Embarque \wedge e.peso > 500 \}$

Dominios: $\{ d \mid \overset{(\exists a)(\exists b)(\exists c)(\exists d)(\exists e)}{Embarque(a,b,c,d,e)} \wedge c > 500 \}$

9)

Tuplas: $\{ c.nom_choffer \mid c \in Camion \wedge \exists e (e \in Embarque) \wedge c.id_camion = e.id_camion \wedge e.peso > 500 \}$

Dominios: $\{ n \mid \exists i \exists a \exists b \exists c \exists d \exists e Camion(i, n) Embarque(a, b, c, d, e) \wedge c = d \wedge c > 500 \}$

h) $\{ e.destino \mid e \in Embarque \wedge \exists c (c \in clientes) \wedge e.id_cliente = c.id_cliente \wedge c.renta_anual > 90000 \}$

D: $\{ g \mid \overset{(\exists a)(\exists b)(\exists c)(\exists d)(\exists e)}{Embarque(a,b,c,d,e)} \wedge \overset{(\exists a)(\exists b)(\exists c)(\exists d)(\exists e)}{Cliente(g,h)} \wedge b = g \wedge h > 90000 \}$

- i) Los clientes que tienen una renta anual por encima de los 90.000 € que han enviado paquetes con peso menor de un kilo.
- j) Los clientes que tienen una renta anual por encima de los 90.000 € que han enviado paquetes con peso menor de un kilo o han enviado embarques a la ciudad de Madrid.
- k) Los clientes cuyos envíos han sido distribuidos por el chófer Juan.
- l) Los conductores que han distribuido envíos de clientes con renta anual por encima de los 120.000 € a ciudades con población superior a un millón de habitantes.

i)

$$T \} c.\text{nom_cliente} \mid c \in \text{Cliente} \wedge \exists e (e \in \text{Embarque}) \wedge$$

$$c.\text{id_cliente} = e.\text{id_cliente} \wedge e.\text{peso} < 1 \wedge$$

$$c.\text{renta_anual} > 90000 \}$$

D: } b $\mid \exists a \exists b \dots$ $\text{Cliente}(a b c) \wedge \text{Embarco}(d e f g h)$

$$a = e \wedge c > 90000 \wedge g < 1 \}$$

j)
 Tuplas: } $c.\text{nom_cliente} \mid c \in \text{Cliente} \wedge \exists e (e \in \text{Embarco})$

$$\wedge c.\text{id_cliente} = e.\text{id_cliente} \wedge c.\text{renta_anual} > 9000$$

$$\wedge (e.\text{peso} < 1 \vee e.\text{destino} = \text{"Madrid"}) \}$$

Domínios:

} a $\mid \exists \dots$ $\text{Cliente}(a b c) \wedge \text{Embarco}(d e f g h)$

$$a = e \wedge c > 90000 \wedge (g < 1 \vee h = \text{"Madrid"}) \}$$

* A partir de aquí uso notación de Access en cálculo de k) dominios.

* supongamos que cada chofer conduce un único camión

Tuplas: $\{ t \mid t \in \text{Clientes} \wedge \exists e (e \in \text{Embarco}) \wedge$

$t.\text{id-cliente} = e.\text{id-cliente} \wedge \exists c (c \in \text{Camion}) \wedge$

$c.\text{nom-chofer} = \text{"Juan"} \wedge e.\text{id-camion} = c.\text{id-camion} \}$

Dominios: $\{ nr \mid \text{Clientes}(nr), \text{Embarque}(m, i, -, d, -),$
 $\text{Camion}(d, \text{"Juan"}) \}$

1)

Tuplas: $\{ t.\text{nom-chofer} \mid t \in \text{Camion} \wedge \exists c (c \in \text{Ciudad})$

$c.\text{población} > 3000000 \wedge \exists a (a \in \text{Clientes}) \wedge$

$a.\text{renta-anual} > 320000 \wedge \exists e (e \in \text{Embarque})$

$\wedge e.\text{id-cliente} = a.\text{id-cliente} \wedge e.\text{destino} = c.\text{nomb-ciudad}$

$\wedge e.\text{id-camion} = t.\text{id-camion} \}$

Dominios $\{ n \mid \text{Camion}(i, n), \text{Ciudad}(m, p),$
 $\text{Clientes}(j, r) \wedge \text{Embarque}(\text{x}, j, i, m) \wedge$
 $p > 3000000 \wedge r > 320000 \}$

Normalmente las claves primarias, aunque no se usen, se ponen con una letra ya que toda tupla tiene clave primaria.

m)

Tuplas:

$$\{ t \in \text{clientes} \wedge (\forall c)(\exists e) (c \in \text{camion}) (e \in \text{Embarque}) \\ (e.\text{id-camion} = c.\text{id-camion}) \}$$

* Nota: Suponemos que no hay 2 camiones con el mismo chofer.

Dominios: $\{ i \in r \mid (\forall j) \text{Clientes}(i, r) \wedge \text{Camion}(j, -) \\ \wedge \text{Embarque}(x, i, j, -) \}$

n)

Tuplas: $\{ t \in \text{Ciudad} \wedge (\forall c)(\exists e)(c \in \text{cliente}) (e \in \text{Embarque}) \\ (t.\text{nomb-ciudad} = e.\text{destino} \wedge e.\text{id-cliente} = c.\text{id-cliente}) \}$

Dominios: $\{ n \in p \mid (\forall i) \text{Ciudad}(n, p) \wedge \text{cliente}(i, -) \\ \wedge \text{Embarque}(x, i, -, n) \}$

$\tilde{n})$

Tuplas: $\{ t.\text{nom-chóffer} \mid t \in \text{Camion} \wedge (\forall c)(\exists e)$
 $(c \in \text{Ciudad}) (e \in \text{Embarque}) (t.\text{id-camion} = e.\text{id-camion}$
 $\wedge e.\text{destino} = c.\text{nomb-ciudad}) \}$

Dominios: $\{ n \mid (\forall m) \text{Camion} (i n), \text{Ciudad} (m -)$
 $\wedge \text{Embarque} (x i - - m) \}$

o)

Tuplas: $\{ t \mid t \in \text{Clientes} \wedge (\forall c)(\exists e) (c \in \text{Ciudad})$
 $(e \in \text{Embarque}) (e.\text{id-cliente} = t.\text{id-cliente} \wedge$
 $(c.\text{poblacion} \leq 500\,000 \vee e.\text{destino} = c.\text{nomb-ciudad})$
 $\}$

Dominios: $\{ nr \mid (\forall m) \text{Clientes} (i nr) \wedge \text{Ciudad} (m, p)$
 $\wedge \text{Embarque} (x i - - m) \wedge p > 500\,000 \}$

8) Tenemos la base de datos siguiente:

<i>Empleado</i>	(<u>nombre-empleado</u> , calle, ciudad)
<i>Trabaja</i>	(<u>nombre-empleado</u> , nombre-empresa, sueldo)
<i>Empresa</i>	(<u>nombre-empresa</u> , ciudad)
<i>Jefe</i>	(<u>nombre-empleado</u> , nombre-jefe)

encontrar una expresión en álgebra relacional para las siguientes cuestiones:

- Averiguar los nombres de todos los empleados que trabajan para el Banco Crédito.
- Obtener el nombre y la ciudad de residencia de todos los empleados que trabajan para el Banco Crédito.
- Obtener una lista con el nombre, la calle y la ciudad de residencia de todos los empleados que trabajan para el Banco Crédito y ganan más de 12.000 € anuales.
- Sacar un listado que contenga el nombre de todos los empleados de esta base de datos que viven en la misma ciudad que la compañía para la que trabajan.

a) $\pi_{\text{nombre-empleado}} \left(\sigma_{\text{nombre-empresa} = \text{"Banco Crédito"}} (\text{Trabaja}) \right)$

b) $A \leftarrow \pi_{\text{nombre-empleado}} \left(\sigma_{\text{nombre-empresa} = \text{"Banco Crédito"}} (\text{Trabaja}) \right)$

$\pi_{\text{nombre-empleado, ciudad}} (\text{Empleado} \bowtie A)$

c) $A \leftarrow \pi_{\text{nombre-empleado}} \left(\sigma_{\substack{\text{nombre-empresa} = \text{"Banco Crédito"} \\ \text{sueldo} > 12000}} (\text{trabaja}) \right)$

$(\text{Empleado} \bowtie A)$

d) $\pi_{\text{nombre-empleado}} (\underbrace{\text{Empleado} \bowtie \text{Trabaja}} \bowtie \text{Empresa})$

El primero que se hace

Producto natural con 2 campos:
nombre-empresa y ciudad.

- e) Averiguar el nombre de todos los empleados que viven en la misma ciudad y en la misma calle que sus jefes.
- f) Obtener una lista con el nombre de todos los empleados de esta base de datos que no trabajan para el Banco Crédito.
- g) Averiguar el nombre de todos los empleados que ganan más que cualquier empleado del Banco Pequeño.

e)

$A \leftarrow \pi_{\text{nombre-empleado, ciudad, nombre-jefe}} (\text{Empleado} \bowtie \text{Jefe})$

$\text{Empleado}(\text{nombre-jefe}, \dots) \leftarrow \text{Empleado}(\text{nombre-empleado}, \dots)$

$B \leftarrow \pi_{\text{nombre-jefe, ciudad}} (\text{Empleado} \bowtie \text{Jefe})$

$\pi_{\text{nombre-empleado}} (\underbrace{A \bowtie B})$

2 campos en común: ciudad y nombre-jefe

8)

$A \leftarrow \pi_{\text{nombre-empleado}} (\sigma_{\text{nombre-empresa} = \text{"Banco Crédito"}} (\text{Trabaja}))$

En A están todos los que sí trabajan

$\pi_{\text{nombre-empleado}} (\text{Empleados}) - A$

"Todos los de la tabla Empleados que no están en A"

9)

$A \leftarrow \sigma_{\text{nombre-empresa} = \text{"Banco Pequeño"}} (\text{Trabaja})$

$B \leftarrow \text{suelto } \rho_{\text{MAXIMO sueldo}} (A)$

B tiene una única columna: "MAXIMO sueldo" con una sola tupla

$C \leftarrow \text{Empleados} \times B$

A cada tupla de C se le añade la columna "MAXIMO sueldo"

$\pi_{\text{nombre-empleado}} (\sigma_{\text{suelto} > \text{MAXIMO sueldo}} (C))$

* Hay que tener en cuenta que aunque en B solo haya un valor no puedo poner: ~~$\sigma_{\text{suelto} > B}$~~ porque B es una tabla.

9) Encontrar las expresiones en cálculo relacional que satisfagan los apartados del ejercicio 8.

8) Tenemos la base de datos siguiente:

<i>Empleado</i>	(<u>nombre-empleado</u> , calle, ciudad)
<i>Trabaja</i>	(<u>nombre-empleado</u> , nombre-empresa, sueldo)
<i>Empresa</i>	(<u>nombre-empresa</u> , ciudad)
<i>Jefe</i>	(<u>nombre-empleado</u> , nombre-jefe)

- a) Averiguar los nombres de todos los empleados que trabajan para el Banco Crédito.
- b) Obtener el nombre y la ciudad de residencia de todos los empleados que trabajan para el Banco Crédito.
- c) Obtener una lista con el nombre, la calle y la ciudad de residencia de todos los empleados que trabajan para el Banco Crédito y ganan más de 12.000 € anuales.
- d) Sacar un listado que contenga el nombre de todos los empleados de esta base de datos que viven en la misma ciudad que la compañía para la que trabajan.

a)

Tuplas: $\{ t.\text{nombre-empleado} \mid t \in \text{Trabaja} \wedge t.\text{nombre-empresa} = \text{"Banco Crédito"} \}$

Domínios: $\{ n \mid \text{Trabaja} (n \text{ "Banco Crédito"} -) \}$

b)

Tuplas: $\{ e.\text{nombre-empleado}, e.\text{ciudad} \mid e \in \text{Empleado} \wedge \exists t (t \in \text{Trabaja}) \wedge t.\text{nombre-empresa} = \text{"Banco Crédito"} \wedge t.\text{nombre-empleado} = e.\text{nombre-empleado} \}$

Dominios: $\{ n \in \text{Empleado} \mid (n, c) \in \text{Trabaja} \mid c = \text{"Banco Credito"} \}$

c)

Tuplas: $\{ e \mid e \in \text{Empleado} \wedge \exists t (t \in \text{Trabaja} \wedge$

$t.\text{nombre-empresa} = \text{"Banco Credito"} \wedge$

$t.\text{suelo} > \$2000 \wedge e.\text{nombre-empresa} = t.\text{nombre-empresa} \}$

Dominios: $\{ n \in K \mid \text{Empleado}(n, k) \wedge \text{Trabaja}(n,$

$\text{"Banco Credito"}, s) \wedge s > \$2000 \}$

d)

Tuplas $\{ e.\text{nombre-empresa} \mid e \in \text{Empleado} \wedge$

$\exists t (t \in \text{Trabaja}) \wedge e.\text{nombre-empresa} = t.\text{nombre-empresa}$

$\wedge \exists x (x \in \text{Empresa}) \wedge x.\text{nombre-empresa} = t.\text{nombre-empresa}$

$\wedge x.\text{ciudad} = e.\text{ciudad} \}$

Dominios: $\{n \mid \text{Empleado}(n-c) \wedge \text{Trabaja}(n-e-)$
 $\wedge \text{Empresa}(e-c) \}$

- e) Averiguar el nombre de todos los empleados que viven en la misma ciudad y en la misma calle que sus jefes.
- f) Obtener una lista con el nombre de todos los empleados de esta base de datos que no trabajan para el Banco Crédito.
- g) Averiguar el nombre de todos los empleados que ganan más que cualquier empleado del Banco Pequeño.

e)

Tuplas: $\{e.\text{nombre} \mid e \in \text{Empleado} \wedge \exists j (j \in \text{Empleado})$
 $\wedge \exists j (j \in \text{Jefe}) \wedge e.\text{nombre-empleado} = j.\text{nombre-empleado}$
 $\wedge j.\text{nombre-empleado} = j.\text{nombre-jefe}$
 $\wedge e.\text{ciudad} = j.\text{ciudad} \}$

Dominios $\{n \mid \text{Empleado}(n-c) \wedge \text{Empleado}(m-c)$
 $\wedge \text{Jefe}(n-m) \wedge n \neq m \}$

g)

Tuplas: $\{e.\text{nombre-empleado} \mid e \in \text{Empleado}$

$\wedge \exists t (t \in \text{Trabaja}) \wedge e.\text{nombre-empleado} = t.\text{nombre-empleado}$
 $\wedge e.\text{nombre-empresa} \neq \text{"Banco Crédito"} \}$

Domínios: $\{ n \mid \text{Empleado } (n - -) \wedge \text{Trabaja } (n e -) \wedge e \neq \text{"Banco Crédito"} \}$

g) No hemos visto cómo hacer el máximo en cálculo:

- Agregación en cálculo
 - En cálculo se pueden incluir de manera bastante "natural" los operadores de agregación de SQL: sumatorio, media, mínimo y **máximo** y contar
 - **No entran en el examen** (al menos este curso ;)



11) Las siguientes tablas están almacenadas en un SGBD relacional:

<i>Hotel</i>	(<u>n^ohotel</u> , nombre, dirección)
<i>Habitación</i>	(<u>n^ohabitación</u> , n ^o hotel, tipo, precio)
<i>Registro</i>	(<u>n^ohotel</u> , <u>n^ohuesped</u> , fecha-entrada, fecha-salida, n ^o habitación)
<i>Huesped</i>	(<u>n^ohuesped</u> , nombre, dirección)

donde *Hotel* contiene los datos del hotel⁴, *Habitación* contiene los datos de las habitaciones de cada hotel, *Registro* contiene los datos de los registros realizados y *Huesped* contiene los datos de los huéspedes que se hospedan en los diferentes hoteles.

Generar las expresiones correspondientes en álgebra relacional para las siguientes consultas:

⁴La clave primaria de las tablas está formada por los atributos subrayados en cada una de ellas.

- Lista de todos los hoteles. (supondré que se refiere a los nombres)
- Lista de todas las habitaciones individuales con un precio menor de 100 €.
- Obtener los nombres y direcciones de todos los huéspedes.

a) $\Pi_{\text{nombre}}(\text{Hotel})$

b) * Supondré que quiere: $\frac{\text{Nombre-hotel}}{\text{N}^{\circ} \text{habitación}}$

Ya podría decir claramente lo que quiere...

$\Pi_{\text{nombre}, \text{n}^{\circ} \text{habitación}}(\text{Hotel} \bowtie \sigma_{\text{precio} < 100 \wedge \text{tipo} = \text{"individual"}}(\text{Habitación}))$

c) $\Pi_{\text{nombre}, \text{dirección}}(\text{Huesped})$

- d) Listar el precio y el tipo de todas las habitaciones del Hotel Atlantico.
- e) Obtener una lista de todos los huéspedes actuales del Hotel Atlantico.
- f) Obtener un listado con los datos de todas las habitaciones del Hotel Atlantico, incluyendo el nombre del huésped actual de la habitación, si es que está ocupada.

d) $\Pi_{\text{precio, tipo}} \left(\Pi_{\text{n}^\circ \text{hotel}}^{(\text{Hotel})} (\sigma_{\text{nombre} = \text{"Hotel Atlántico"}}) \bowtie \text{Habitación} \right)$

e) $\text{Huésped} \bowtie \left(\Pi_{\text{n}^\circ \text{hotel}}^{(\text{Hotel})} (\sigma_{\text{nombre} = \text{"Hotel Atlántico"}}) \bowtie \text{Registro} \right)$

g) $A \leftarrow \text{Habitación} \bowtie \Pi_{\text{n}^\circ \text{hotel}}^{(\text{Hotel})} (\sigma_{\text{nombre} = \text{"Hotel Atlántico"}})$

En A tengo todas las habitaciones de ese hotel.

$B \leftarrow \Pi_{\text{n}^\circ \text{hotel}, \text{n}^\circ \text{huésped}, \text{n}^\circ \text{habitación}} (A \bowtie \text{Registro})$

En B tengo el n° de huésped asociado a cada habitación del hotel Atlántico (si es que hay huésped asociado)

$C \leftarrow \Pi_{\text{nombre}, \text{n}^\circ \text{hotel}, \text{n}^\circ \text{habitación}} (B \bowtie \text{Huésped})$

En C meto el nombre del huésped en vez del número

$A \bowtie C$

Unión externa por la izquierda. Hace el producto natural y añade las filas de A que no se correspondan con ninguna de C, es decir, las habitaciones sin huésped.

12) Dadas dos relaciones R_1 y R_2 , donde R_1 contiene N_1 tuplas, R_2 contiene N_2 tuplas, y $N_2 > N_1 > 0$, dar el mínimo y el máximo número de tuplas de la relación resultante producida por cada una de las siguientes expresiones algebraicas⁵:

a) $R_1 \cup R_2$

b) $R_1 \cap R_2$

c) $R_1 - R_2$

d) $R_1 \times R_2$

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

f) $\Pi_a(R_1)$

g) $R_1 \div R_2$

* Supondremos que R_1 y R_2 tienen el mismo esquema para que se puedan realizar operaciones como la unión, intersección...

a)

Max: $N_1 + N_2$ (todas las tuplas son distintas)

Min: $\max(N_1, N_2)$ (todas las tuplas de uno están entre las del otro)

b)

Max: $\min(N_1, N_2)$ (todas las tuplas de uno están entre las del otro)

Min: 0 (Ninguna tupla repetida)

c)

Max: N_1 (R_2 no tiene ninguna tupla de R_1)

Min: 0 (Todas las tuplas de R_1 están en la relación R_2)

d)

$$\text{Max: } N_1 \cdot N_2$$

(Cada tupla de R_1 con cada tupla de R_2 en ambos casos)

$$\text{Min: } N_1 \cdot N_2$$

e)

$$\text{Max: } N_1 \quad (\text{En todas las tuplas de } R_1 \text{ "a" es 5})$$

$$\text{Min: } 0 \quad (\text{En ninguna tupla de } R_1 \text{ "a" es 5})$$

f)

$$\text{Max: } N_1 \quad (\text{Cada tupla de } R_1 \text{ tiene un valor distinto en "a"})$$

$$\text{Min: } 1 \quad (\text{Todas las tuplas de } R_1 \text{ tienen el mismo valor en "a"})$$

* Supondremos ahora que N_1 tiene todos los atributos de N_2 y algún atributo propio

g)

$$\text{Max: } N_1 / N_2 \quad (\text{Toda tupla de } R_2 \text{ está en } R_1 \text{ con cada uno de los valores de los atributos propios de } R_1)$$

$$\text{Min: } 0 \quad (\text{Ninguna combinación de valores de atributos propios de } R_1 \text{ está con todas las tuplas de } R_2 \text{ en la relación } R_1)$$

13) Dados los esquemas de relaciones siguientes:

<i>Alumno</i>	(<u>DNI</u> , Nombre, Domicilio)
<i>Matrícula</i>	(DNI, <u>Código-asignatura</u>)
<i>Asignatura</i>	(<u>Código-asignatura</u> , Nombre, Código-departamento)
<i>Departamento</i>	(<u>Código-departamento</u> , Nombre)

expresar en álgebra relacional las siguientes consultas:

- Nombre de los alumnos matriculados en alguna asignatura del departamento de Matemáticas.
- Alumnos matriculados en todas las asignaturas del departamento de Matemáticas.
- Alumnos que no está matriculados en asignaturas del departamento de Matemáticas.

a)

$A \leftarrow \pi_{\text{Código-departamento}} (\sigma_{\text{Nombre} = \text{"Matemáticas"}} (\text{Departamento}))$

En "A" tengo el código del departamento de Matemáticas.

$B \leftarrow \pi_{\text{Código-asignatura}} (\text{Asignatura} \bowtie A)$

En "B" tengo las asignaturas del departamento de Matemáticas.

$\pi_{\text{Nombre}} (\text{Alumno} \bowtie \pi_{\text{DNI}} (\text{Matrícula} \bowtie B))$

Saco los DNIs de los alumnos que van a las asignaturas de B y lo que muestro son sus nombres.

b)

$A \leftarrow \Pi_{\text{código-departamento}} (\sigma_{\text{nombre} = \text{"Matemáticas"}} (\text{Departamento}))$

En "A" tengo el código del departamento de Matemáticas

$B \leftarrow \Pi_{\text{código-asignatura}} (\text{Asignatura} \bowtie A)$

En "B" tengo las asignaturas del departamento de Matemáticas

$\Pi_{\text{nombre}} (\text{Alumno} \bowtie \Pi_{\text{DNI}} (\text{Matrícula} / B))$

Saco los DNIs de los alumnos que van a **TODAS** asignaturas de B y lo que muestro son sus nombres.

c) * Podría haber hecho desde el principio \neq "Matemáticas"

$A \leftarrow \pi_{\text{código-departamento}} (\sigma_{\text{nombre} = \text{"Matemáticas"}} (\text{Departamento}))$

En "A" tengo el código del departamento de Matemáticas

$B \leftarrow \pi_{\text{código-asignatura}} (\sigma_{\text{asignatura} = A})$

En "B" tengo las asignaturas del departamento de Matemáticas

$C \leftarrow \pi_{\text{DNI}} (\sigma_{\text{Matrícula} \in B})$

En C meto los DNIs de los alumnos que van a las asignaturas de B

$\pi_{\text{nombre}} (\text{Alumno} \bowtie (\pi_{\text{DNI}} (\text{Matrícula}) - C))$

Saco los DNIs que no están en C y muestro sus nombres.

14) Expresar en cálculo relacional de tuplas las consultas del ejercicio 13.

13) Dados los esquemas de relaciones siguientes:

<i>Alumno</i>	(<u>DNI</u> , Nombre, Domicilio)
<i>Matrícula</i>	(DNI, <u>Código-asignatura</u>)
<i>Asignatura</i>	(<u>Código-asignatura</u> , Nombre, <u>Código-departamento</u>)
<i>Departamento</i>	(<u>Código-departamento</u> , Nombre)

expresar en álgebra relacional las siguientes consultas:

- Nombre de los alumnos matriculados en alguna asignatura del departamento de Matemáticas.
- Alumnos matriculados en todas las asignaturas del departamento de Matemáticas.
- Alumnos que no está matriculados en asignaturas del departamento de Matemáticas.

a) $\{ t.Nombre \mid t \in Alumno \wedge \exists a (a \in Asignatura) \wedge \exists d (d \in Departamento) \wedge \exists m (m \in Matrícula) \wedge d.Nombre = "Matemáticas" \wedge d.Código-departamento = a.Código-departamento \wedge a.Código-asignatura = m.Código-asignatura \wedge m.DNI = t.DNI \}$

b) $\} t \mid t \in \text{Alumno} \wedge$

$\wedge (\exists d)(d \in \text{Departamento}) \wedge d.\text{Nombre} = \text{"Matemáticas"}$

$\wedge (\forall a)(\exists m) (a \in \text{Asignatura}) \wedge (m \in \text{Matrícula})$

$(d.\text{Código-departamento} \neq a.\text{Código-departamento} \vee$

$a.\text{Código-asignatura} = m.\text{Código-asignatura})$

$\wedge m.\text{DNI} = t.\text{DNI} \}$

c) $\} t \mid t \in \text{Alumno} \wedge \exists d (d \in \text{Departamento})$

$\wedge d.\text{Nombre} = \text{"Matemáticas"}$

$\wedge (\forall a)(\exists m) (a \in \text{Asignatura}) (m \in \text{Matrícula})$

$(a.\text{Código-Departamento} = d.\text{Código-Departamento} \vee$

$a.\text{Código-Asignatura} = m.\text{Código-Asignatura})$

$\wedge t.\text{DNI} = m.\text{DNI} \}$

