

UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE CIENCIAS EXACTAS Y NATURALES
DEPARTAMENTO DE COMPUTACIÓN



BASES DE DATOS
Lenguajes de Consulta

GUÍA DE EJERCICIOS

1 AR y CRT

1.1. Dadas las relaciones R, S y T calcular:

- (a) $R \cup S$
- (b) $R - S$
- (c) $R \times S$
- (d) $R \bowtie S$
- (e) $\pi_B(R)$
- (f) $\sigma_{A=C}(R \times S)$
- (g) $S \div TS$
- (h) $R \bowtie_{R.B < S.C} S$

R:	A	B	S:	B	C	T:	C
	a	b		b	c		c
	b	c		e	a		d
	c	b		b	d		
	d	e					

1.2. Considerando el siguiente esquema de una base de datos:

FRECUENTA(Persona, Bar)
 SIRVE (Bar, Cerveza)
 GUSTA(Persona, Cerveza).

Expresar las siguientes consultas usando álgebra relacional (AR):

- (a) Bares que sirven alguna cerveza que le guste a “Juan K.”
 - (b) Personas que frecuentan al menos un bar que sirve alguna cerveza que les guste.
 - (c) Personas que no frecuenten ningún bar que sirva una cerveza que les guste.
 - (d) Personas que frecuentan todos los bares. (Asumir que todos los bares sirven al menos una cerveza).
- 1.3. Dado $R(a)$ un esquema de relación. Expresar en AR la consulta que devuelve las tuplas $t \in r(R)$ tal que $t[a] \leq t'[a] \forall t' \in r(R)$ (o sea, el mínimo a para todas las tuplas de la relación).
- 1.4. Dado $R(a, b)$ un esquema de relación. Expresar en AR la consulta que devuelve las tuplas $t \in r(R)$ tal que $t[b] \leq t'[b] \forall t' \in r(R)/t[a] = t'[a]$ (o sea, los mínimos b por cada uno de los grupos de tuplas de la relación que cumplen que el valor para el atributo a es el mismo).
- 1.5. Dado $R(a, b)$ un esquema de relación. Expresar en AR la consulta que devuelve las tuplas $t \in r(R)$ tal que $\exists t', t'' \in r(R)/t[a] = t'[a] \wedge t[b] \neq t'[b]$.
- 1.6. Dado $R(a, b)$ un esquema de relación. Expresar en AR la consulta que devuelve las tuplas $t \in r(R)$ tal que $\exists t', t'' \in r(R)/t[a] = t'[a] \wedge t[b] \neq t'[b]$ y que además $\neg \exists t''/t[a] = t''[a] \wedge t[b] \neq t''[b] \wedge t'[b] \neq t''[b]$.

2 AR, CRT y SQL

2.1. Considerando el siguiente esquema de una base de datos:

ALUMNO (NroLU, Nombre, Edad, Sexo, FecIngreso)

MATERIA (IdMateria, Nombre)

CURSA (NroLU, IdMateria)

CARRERA (IdCarrera, Nombre)

DOCENTE (Legajo, Nombre)

DICTA (Legajo, IdMateria)

PLAN(IdPlan, Nombre, IdCarrera)

PLAN_DE_ESTUDIO(IdPlan, IdMateria)

Se pide:

- (a) Mediante SQL, AR y CRT listar los nombres y edades de los alumnos mujeres que ingresaron en el año 2009.
- (b) Mediante SQL, AR y CRT listar para cada alumno las materias que cursa. Se deberá mostrar el nombre del alumno y el nombre de la materia.
- (c) Mediante SQL, AR y CRT listar para cada alumno el nombre de la carrera que cursa.
- (d) Mediante AR y CRT obtener los nombres de las materias que tienen más de 1 alumno hombre mayor de 30 años.
- (e) Mediante SQL obtener los nombres de las materias que tienen más de 50 alumnos hombres mayores de 30 años. ¿Es posible resolver esta consulta usando AR y CRT? Justifique la respuesta.
- (f) Mediante SQL indicar cuántas materias tiene cada Plan de estudios de la carrera Ciencias de la Computación.
- (g) Mediante SQL y AR listar los nombres de los alumnos mayores de 25 años que cursan todas las materias de la carrera "Ciencias de la Computación" que no son dictadas por el profesor "Jeff Ullman"
- (h) Mediante SQL indicar para cada materia en cuántos planes de estudio se requiere cursar dicha materia.
- (i) Mediante SQL resolver el punto b) incluyendo en el listado los alumnos que no están cursando ninguna materia.
- (j) Mediante SQL listar los nombres de las materias que tienen más de 30 alumnos inscriptos

NOTA: Los nombres y las descripciones no se repiten. No hay valores nulos. Sexo puede ser 'M' o 'F'. Una materia puede ser dictada por más de un docente

2.2. Sea el siguiente esquema de relación:

Empleado(empleadoID, nombre, apellido, reportaA),

En donde *reportaA* es el *empleadoID* que corresponde a su jefe.

Se pide resolver en SQL:

- (a) Listar todos los empleados junto a su jefe. Las tuplas resultantes tendrán la siguiente forma: (nombre empleado, apellido de empleado, nombre jefe, apellido de jefe)
- (b) Resolver el ítem anterior pero que no falte ningún empleado en el listado.

2.3. Considerando el siguiente esquema de una base de datos:

FRECUENTA(Persona, Bar)

SIRVE (Bar, Cerveza)

GUSTA(Persona, Cerveza).

Se pide en SQL obtener:

- (a) Bares que sirven alguna cerveza que le guste a “Juan K.”
- (b) Personas que frecuentan al menos un bar que sirve alguna cerveza que les guste.
- (c) Personas que no frecuenten ningún bar que sirva una cerveza que les guste.
- (d) Personas que frecuentan todos los bares. (Asumir que todos los bares sirven al menos una cerveza).
- (e) Definir una vista que devuelva una relación de la forma (p, c, b) de tal manera que a la persona p le gusta la cerveza c, el bar b sirve la cerveza c y la persona p frecuenta el bar b.

2.4. Suponga que al esquema del ejercicio anterior le agregamos la siguiente relación

VENDE (Bar, Cerveza, Cantidad)

Se pide

- (a) Obtener en SQL el promedio de cerveza vendida por bar
- (b) Obtener en SQL el total vendido de cada cerveza por bar, excluyendo a los que no venden la cerveza.
- (c) Obtener en SQL la máxima cantidad de cerveza vendida, tomando en cuenta que una cerveza puede ser vendida por más de un Bar.

2.5. Sea el siguiente esquema de relación:

PERSONA (Nombre, Progenitor),

AMIGO(Nombre1, Nombre2).

El atributo *Progenitor* indica para cada PERSONA el NOMBRE de su padre y puede tener valor NULL.

- (a) Suponga que la relación AMIGO es tal que $(a, a) \notin AMIGO$ para ningún a y que si $(a, b) \in AMIGO$ entonces $(b, a) \notin AMIGO$.
 - i. Expresar en el álgebra relacional y en cálculo relacional de tuplas la consulta que devuelve como resultado a aquellas personas cuyos padres no tienen amigos.
 - ii. Expresar en el álgebra relacional y en cálculo relacional de tuplas la consulta que devuelve como resultado a aquellas personas cuyos amigos son amigos de sus padres.
 - iii. Expresar una consulta SQL que devuelva como resultado a las personas que tengan mas de 8 amigos
- (b) Suponga que se ha descuidado el control de las tuplas que se ingresan a la base de datos, y que la restricción de antisimetría expresada en a) ya no es válida. Escriba la consulta en SQL que en esta situación resuelve lo pedido en el punto (a)iii.

2.6. Considerando el siguiente esquema de una base de datos:SUCURSAL (codsuc, nomsuc, zona)

PRODUCTO (codprod, nomprod, stockcritico)
 MOVIMIENTO (nromov, fechahora, codsuc, signomov)
 MOV_PROD (nromov, codprod, cantidad)

Obs: fechahora es diferente para cada movimiento. signomov es +1 si es un depósito ó -1 si es un retiro.

Obs: stockcritico es la cantidad mínima deseada para TODO el depósito (NO para cada sucursal).

Se pide:

- (a) En AR y SQL, obtener el código y el nombre de las sucursales que el mes de abril del 2010 hayan enviado a depósito TODOS los productos que comercializa la cadena (se encuentran en PRODUCTO) y que también hayan realizado retiros de TODOS los productos.
 Obs: Los productos no necesariamente se han depositado en un único movimiento sino en varios. Lo mismo vale para los retiros.
- (b) En SQL, obtener los códigos y los nombres de los productos que algún retiro del mes de abril del 2010 haya dejado con un total almacenado en el depósito por debajo de su stock crítico.
 Obs: El stock de un producto en un momento dado es igual a la suma (con signo) de todos los movimientos de ese producto realizados hasta dicho momento.
- (c) En AR (sin funciones de agregación) y SQL (sin vistas ni sub-queries en el FROM) obtener el código y el nombre de los productos tales que, durante el mes de abril del 2010: toda sucursal haya retirado en al menos dos oportunidades y no haya realizado ningún depósito (las dos condiciones deben ser satisfechas por cada sucursal).
- (d) En SQL (sin vistas ni sub-queries en el FROM), obtener el número de movimiento, el código y nombre de la sucursal asociada de aquellos depósitos del mes de abril del 2010 tales que: la cantidad depositada de cada uno de los ítems que lo componen es superior al stock del producto correspondiente para la sucursal asociada.
 Por ejemplo: si la sucursal S en el movimiento X depositó los productos A, B y C; entonces la cantidad depositada de A es superior al stock hasta ese momento del producto A para la sucursal S (dicho stock es: la suma con signo de todos los movimientos del producto A para S), ídem para B e ídem para C.

2.7. Considerando el siguiente esquema de una base de datos que guarda información sobre series:

ACTOR(idActor, nombreActor, edad)
 SERIE(idSerie, nombreSerie, idGénero, añoInicio, añoFin)
 GÉNERO(idGénero, nombreGénero)
 CANAL(idCanal, nombreCanal)
 PARTICIPA_EN(idActor, idSerie)
 TRANSMITE(idCanal, idSerie)

Se pide en SQL:

- (a) Listar los nombres de los actores mayores de 30 años que participan en la serie "Friends".
- (b) Listar los nombres de los canales que transmiten todas las series de comedia.

- (c) Listar los nombres de los actores mayores de 30 años que participan en la serie “Friends” y que además participaron alguna vez en series que comenzaron luego del año 2000
- (d) Listar los nombres de los actores mayores de 30 años que participan en la serie “Friends” y que además nunca participaron en series que comenzaron luego del año 2000.
- (e) Listar los ids de las series que comenzaron más recientemente.
- (f) Listar los ids de los actores que participaron en al menos 2 series.
- (g) Suponiendo que puede haber nombres de serie repetidos, listar los nombres que hayan sido usados en dos o más series
- (h) Considerando lo mismo del ejercicio anterior, listar todos los canales que transmiten alguna serie que tiene el nombre repetido.
- (i) Obtener la serie con el promedio de edad (de los actores) mas alto.
- (j) Obtener la genero donde participa el actor más joven.

2.8. Considerando el siguiente esquema de una base de datos con los partidos de fútbol de Primera A:

PARTIDO (equipo1, equipo2, torneo, goles1, goles2, puntos1, puntos2)
JUGADOR (nombre, edad, equipo)

- Cada partido se registra una sola vez. Por ejemplo si Boca le ganó a River en el torneo “Apertura 2002” por 2 a 1, se registra únicamente la tupla (“Boca”, “River”, “Apertura 2002”, 2, 1, 3, 0)
- Los candidatos a campeones de un torneo se determinan sumando los puntos obtenidos por cada equipo, seleccionando los que suman el máximo valor. Al haber más de un candidato, se resuelve el campeonato tomando en cuenta la diferencia de goles.
- Para simplificar se asume que ningún jugador cambió de equipo y los nombres no se repiten.

Se pide:

- (a) Expresar en AR las siguientes consultas:
 - i. Listar los equipos que jugaron exactamente en dos torneos.
 - ii. Listar el jugador más joven y el más viejo de cada uno de los equipos que finalizaron invictos (sin perder ningún partido) en al menos dos torneos – Las tuplas del resultado son de la forma (equipo, nombre1, nombre2)
- (b) Dada la misma BD del ejercicio anterior pero ahora considerando que cada partido se registra dos veces. Por ejemplo si Boca le ganó a River en el torneo “Apertura 2002” por 2 a 1, se registran dos tuplas
(“Boca”, “River”, “Apertura 2002”, 2, 1, 3, 0)
(“River”, “Boca”, “Apertura 2002”, 1, 2, 0, 3)

Expresar en SQL las siguientes consultas:

- i. Listar los equipos que hayan finalizado invictos en al menos dos torneos.
- ii. Listar los torneos que finalizaron con más de un candidato a campeón.

2.9. Considerando el siguiente esquema de una base de datos de recetas médicas:

ELABORA (laboratorio, droga)
PRODUCE (laboratorio, farmaco)
VENDE (farmacia, farmaco, precio)
COMPUESTO_POR (farmaco, droga)
RECETA (recetaId, fecha, medico, paciente)
RECETA_FARMACO (recetaId, farmaco)

ELABORA indica las drogas que elabora cada laboratorio. PRODUCE indica los fármacos que produce cada laboratorio. VENDE indica los fármacos que vende cada farmacia y su precio correspondiente (un mismo fármaco puede tener diferentes precios en distintas farmacias). COMPUESTO_POR indica las drogas que componen cada fármaco. RECETA tiene la cabecera de cada receta confeccionada. RECETA_FARMACO tiene los fármacos prescritos en cada receta.

- (a) Expresar en AR y CRT las siguientes consultas:
- Listar los laboratorios que no elaboran ninguna de las drogas de los fármacos que producen.
 - Listar las drogas que son componentes de todos los fármacos que fueron prescritos al menos dos veces antes del “01/10/2010”
- .
- (b) Expresar en SQL las siguientes consultas:
- Listar las farmacias que venden todos los fármacos que fueron prescritos más de 5 veces después del “30/09/2010”.
 - Listar los fármacos, cuyos precios promedio son mayores a los de todos los prescritos por los médicos que más fármacos recetaron.
 - Obtener los nombres de los fármacos que son equivalentes (en forma de pares, fármaco1, fármaco2), es decir que están compuestos por las mismas drogas.

2.10. Considerando el siguiente esquema de una base de datos

PERSONA(Nombre, Sexo, Fecha_Nac, Nombre_Madre, Nombre_Padre),
MATRIMONIO(Nombre_Mujer, Nombre_Hombre, Fecha_Realización, Fecha_Fin)

Se asume que dos personas pueden contraer matrimonio entre sí más de una vez.

Se pide

- Listar en SQL las personas que hayan tenido algún hijo en su primer matrimonio.
- Expresar en cálculo relacional de tuplas la consulta que devuelve como resultado los nombres de los hijos extramatrimoniales.
- Expresar en álgebra relacional la consulta que devuelve como resultado los nombres de los primogénitos de cada matrimonio. Recordar que en álgebra relacional no se dispone de funciones de agregación.

2.11. En el siguiente esquema tenemos dos relaciones, una de ellas representa ítems con su nombre y precio actual mientras la otra guarda información histórica. Cada vez que hay una variación en el precio se guarda en la tabla histórica. Tomar en cuenta que los ítems se guardan solamente cuando cambia el precio y no todos los días.

Historia(idItem, fecha_guardado, precio)
Items (idItem, nombre, precio_actual, categoríaId)

- (a) Realizar en SQL: Una consulta que devuelva el precio histórico promedio de los ítems de categoríaId=1 al 1/7/1999
Nota: no usar SELECT en el FROM, vistas ni tablas auxiliares.
- (b) Escribir en Algebra Relacional una consulta que devuelva los nombres de los ítems que no tengan precio histórico guardado.
- (c) Escribir en CRT una consulta que devuelva el ítem con mayor precio actual junto al mayor precio histórico del mismo ítem.

2.12. Dados los siguientes esquemas de relación (donde la relación Amigo es simétrica.)

Miembro (nombrePersona, nombreGrupo)

Amigo (nombrePersona1, nombrePersona2)

Persona (nombrePersona, edad, genero)

Grupo (nombreGrupo, fechaInicio)

Se pide:

- (a) Obtener, en SQL, para cada persona los grupos en los cuales el no es miembro pero todos sus amigos si son miembros. La consulta debe devolver tuplas con (nombrePersona, nombreGrupo) donde nombreGrupo es el nombre del grupo que cumple con lo que se pide.
- (b) Realizar en Algebra Relacional una consulta que devuelva las personas que sólo tienen amigos de genero diferente y mayores de 18 años (ej si el genero es masculino solo tendría de genero femenino mayores de 18 y viceversa)
- (c) Realizar en Calculo Relacional de Tuplas una consulta que devuelva las personas para las que todos sus amigos son miembros de al menos un grupo con fecha de inicio mayor al 1/12/2001.