

Ejemplo de División:

ALUMNOS			DEPARTAMENTOS		NOTAS			
Padron	Apellido	Nombre	Depto	Denominación	Padron	Depto	Codigo	Nota
70000	ARIAS	ANIBAL	71	Gestión	70000	75	05	7
75000	BONO	BEATRIZ	75	Computación	70000	75	15	7
80000	CARELA	CARLOS			80000	71	15	5
85000	DELUCA	DAMIAN			80000	75	06	5
90000	ESTEVEZ	ELISA			80000	75	15	5
					85000	75	06	10
					90000	71	15	8
					90000	75	06	6
					90000	75	15	9

MATERIAS		
Depto	Codigo	Nombre
71	15	MODELOS 2
75	06	ORG.DE DATOS
75	15	BASE DE DATOS

Buscar el padrón de los alumnos que aprobaron **todas las materias**. (Aclaración: en la tabla notas están solo los alumnos que aprobaron). Si un alumno esta en la tabla Notas, es porque aprobó esa materia.

Para resolver la división debemos encontrar:

- El numerador: Conjunto A -> Alumnos
- El denominador: Conjunto B -> Materias
- El vínculo que los relaciona -> del alumno que aprobó la materia

$$\pi_{\text{padron, depto, codigo}}(\text{Notas}) \% \pi_{\text{depto, codigo}}(\text{Materias})$$

NOTAS			%	MATERIAS		=		
Padron	Depto	Codigo		Depto	Codigo		Padron	
70000	75	05		71	15		80000	
70000	75	15		75	06		90000	
80000	71	15		75	15			
80000	75	06						
80000	75	15						
85000	75	06						
90000	71	15						
90000	75	06						
90000	75	15						

Haciendo esta división, me va a quedar solo el padrón, porque el depto y código son comunes a A y B (como que se cancelan). **Para ello, deben estar ordenados según el orden de las columnas**

Los marcados son los padrones que cumplen con la condición de que están relacionados con todas las materias.

Ejercicios del taller:

b) Mostrar el nombre y apellido de directores catalogados como de 'Sci-Fi' (ciencia ficción) con una probabilidad mayor igual a 0.5.

ID_DIRECTORES_SCIFI = $\pi_{\text{director_id, directors_genres}}(\sigma(\text{directors_genres.genre} = \text{'Sci-Fi'} \wedge \text{directors_genres.prob} \geq 0.5) \text{directors_genres})$

π directors.first_name, directors.last_name, directors_genres.prob(ID_DIRECTORES_SCIFI \bowtie director_id=directors.id directors)

directors.first_name	directors.last_name	directors_genres.prob
'James (I)'	'Cameron'	0.5
'Richard (II)'	'Kelly'	0.5

c) Mostrar los nombres de las películas filmadas por James(I) Cameron que figuren en la base.

ID_JAMES_CAM = π id (σ (directors.first_name='James (I)' \wedge directors.last_name = 'Cameron')directors)

J_M_MOVIES_ID = π movie_id(ID_JAMES_CAM \bowtie directors.id=movies_directors.director_id movies_directors)

J_M_MOVIES_NAMES = π name(J_M_MOVIES_ID \bowtie movies_directors.movie_id = movies.id movies)

J_M_MOVIES_NAMES

movies.name
'Aliens'
'Terminator 2: Judgment Day'

d) mostrar los nombres y apellidos de las actrices que trabajaron en la película 'Judgment at Nuremberg'

MOVIE_ID = π id, name(σ (movies.name='Judgment at Nuremberg')movies)

ACTORS_ID_MOVIE = (MOVIE_ID \bowtie movies.id = roles.movie_id roles)

ACTORS_NAMES = π first_name,last_name(σ actors.gender='F'(ACTORS_ID_MOVIE \bowtie roles.actor_id = actors.id actors))

ACTORS_NAMES

actors.first_name	actors.last_name
'Sheila'	'Bromley'
'Virginia'	'Christine'
'Marlene'	'Dietrich'
'Olga'	'Fabian'
'Bess'	'Flowers'
'Judy (I)'	'Garland'
'Jana'	'Taylor'

- e) Muestre los actores que trabajaron en todas las películas de Woody Allen de la base.
- f) Directores que abarcaron, al menos, los mismos géneros que Welles (géneros en directores).
- g) Actores que filmaron más de una película en algún año a partir de 1999.
- h) Listar las películas del último año.
- i) Películas del director Spielberg en las que actuó Harrison (I) Ford.
- j) Películas del director Spielberg en las que no actuó Harrison (I) Ford.
- k) Películas en las que actuó Harrison (I) Ford que no dirigió Spielberg.
- l) Directores que filmaron películas de más de tres géneros distintos, uno de los cuales sea 'Film-Noir

ver clase aprox 1h 50 para saber como obtener el mayor con un autojoin, y una resta.

Para hacer el autojoin en relax hay que cambiarle los nombres a los esquemas.

“mas de 3 generos” autojoin de 3 elementos

Ejercicios de parcial

A)

Utilizando la siguiente notación para representar las operaciones del álgebra relacional: π , σ , ρ , \times , \cup , $-$, \cap , \bowtie , \div , resuelva la siguiente consulta:

Obtener el nombre del hotel (u hoteles) de 4 estrellas o más que ofrece (u ofrecen) habitaciones con todos los tipos de equipamiento registrados.

- hoteles(cod hotel, nombre, estrellas, direccion, ciudad, provincia)
// (1, 'Horizontes del Pasador', 4, 'Av. Rivadavia 500', 'La Falda', 'Córdoba')
- habitaciones(cod hotel, numero, max huespedes)
// (1, 100, 3)
- equipamientos(cod hotel, numero, tipo equipamiento, cantidad)
// (1, 100, 'AIRE ACOND', 1)
- reservas(cod hotel, numero, fecha, tipo doc, nro doc, nombre)
// (1, 100, '2022-01-01', 'DNI', 28900555, 'Juan Bandiola')

1. Saco todos los tipos de equipamientos que hay, de la columna tipo_equipamiento de la tabla equipamientos

Antes de hacer la división, conviene hacer la selección, porque conviene achicar lo mas posible antes de hacer una división/join, etc.

2. Busco los hoteles los de 4 estrellas o mas haciendo una selección de la tabla de hoteles, de la columna estrellas mayor o igual a 4
3. Hago una proyección de esa selección y me quedo con el nombre y el código de hotel
4. Me quedo con las habitaciones de los hoteles de 4 o mas estrellas haciendo un join de hoteles de 4 estrellas y habitaciones
5. Hago una división de las habitaciones con los tipos de equipamiento que proyecté. Porque necesito las habitaciones que estén relacionados con **todos** los elementos de esa columna, osea que cuenten con todos los tipos de equipamiento.

```
HOTEL_4_MAS <-  $\pi$  nombre, cod_hotel ( $\sigma$  estrellas  $\geq$  4 (hoteles))
HABITACIONES_4_MAS <-  $\pi$  nombre, cod_hotel, numero (HOTEL_4_MAS
 $\bowtie$  habitaciones)
EQ_TODOS <-  $\pi$  tipo_equipamiento (equipamientos)
FIN <-  $\pi$  nombre, cod_hotel, numero, tipo_equipamiento
(HABITACIONES_4_MAS  $\bowtie$  equipamientos)  $\div$  EQ_TODOS
 $\pi$  nombre (FIN)
```

B)

Utilizando la siguiente notación para representar las operaciones del álgebra relacional: π , σ , ρ , \times , \cup , $-$, \cap , \bowtie , \div , resuelva la siguiente consulta:

Obtener el nombre y fecha de reserva de aquellos huéspedes que hayan hecho reservas que hayan sido durante el 2022 y sean en habitaciones de capacidad máxima. (Con la capacidad máxima nos referimos a las habitaciones de máxima capacidad entre todos los hoteles, no en cada uno en particular)

```
CAP_NO_MAX <-  $\pi$  1.max_huespedes (habitaciones  $\bowtie$  1.max_huespedes <
2.max_huespedes habitaciones)
CAP_MAX <-  $\pi$  max_huespedes (habitaciones) - CAP_NO_MAX
HAB_CAP_MAX <- CAP_MAX  $\bowtie$  max_huespedes habitaciones
RESERVAS_2022 <-  $\sigma$  fecha<="31/12/2022" and fecha>="01/01/2022"
(reservas)
RESERVAS_2022_MAX <- HAB_CAP_MAX  $\bowtie$  RESERVAS_2022
 $\pi$  nombre, fecha (RESERVAS_2022_MAX)
```

C)

Utilizando la siguiente notación para representar las operaciones del álgebra relacional: π , σ , ρ , \times , \cup , $-$, \cap , \bowtie , \div , resuelva la siguiente consulta:

Para aquellas reservas de 2022 que fueron en alguna de las habitaciones (en el mismo hotel) que ocupó 'Messi' durante 2021, devuelva los siguientes atributos: cod hotel, nombre hotel, numero habitación, ciudad hotel, DNI, nombre huésped

```

R2022 <- σ fecha >= '2022-01-01' ^ fecha <= '2022-12-31' (reservas)
M2021 <- π cod_hotel, numero ( σ fecha >= '2021-01-01' ^ fecha <= '2021-12-31' ^ nombre = 'Lionel Messi' (reservas))
MISMAS <- R2022 ⋈ cod_hotel, numero M2021
FINAL <- hoteles ⋈ cod_hotel MISMAS
π cod_hotel, nombre_hotel, numero, ciudad, nro_doc, nombre FINAL

```

D)

Obtener para cada hotel, la fecha de reserva y nombre de aquellos huéspedes que hayan ocupado la/s habitacion/es de número máximo.

```

NUMS <- cod_hotel, numero(habitaciones)
NOT_MAX <- π 1.cod_hotel, 1.numero (NUMS ⋈ 1.cod_hotel 2.cod_hotel <- 1.numero < 2.numero NUMS)
MAX_NUMS <- NUMS - NOT_MAX
fecha, nombre, cod_hotel, numero (reservas ⋈ cod_hotel numero MAX_NUMS)

```