

Primer Parcial de Base de Datos I

Ejercicio 1	Ejercicio 2	Ejercicio 3	Ejercicio 4	Nota
/5	/1.2	/2.8	/1	/10

Muy Importante

- Condición Mínima de Aprobación:
Sumar no menos de 4 y tener por lo menos 4 ejercicios bien de los siguientes ítems: 1.1, 1.2, 1.3, 1.4, 2.1, 2.2, 3.1, 3.2
- Se debe colocar nombre y legajo en la primer página del enunciado y en todas las páginas auxiliares
- La duración del examen es de 2,5 hs

Ejercicio 1

El siguiente esquema de base de datos almacena los alquileres de autos de sus clientes.

CLIENTE		
<u>DNI</u>	NOMBRE	CIUDAD

AUTO			
<u>PATENTE</u>	MARCA	MODELO	COLOR

ALQUILA						
DNI	Patente	FechaR	CiudadR	FechaD	CiudadD	km

Las tablas CLIENTE y AUTO no pueden tener ningún atributo con valor null.

La tabla ALQUILA representa el alquiler de un auto a un cliente. Tiene dos claves: *DNI+FechaR* y *Patente+FechaR*. Sus características son:

DNI es clave foránea a CLIENTE

Patente es clave foránea a AUTO.

CiudadR es la ciudad en la cliente retira el auto. No puede ser null.

FechaR:es la fecha en la que el cliente retira el auto. No puede ser null.

CiudadD es la ciudad en la que el cliente devuelve el auto (si todavía no lo devolvió es null).

FechaD es la fecha en que el cliente devuelve el auto (si todavía no lo devolvió es null). Los autos no se alquilan por menos de un día y, como en los hoteles, la fecha en que queda registrada la devolución es mayor a la Fecha de Retiro (*FechaR*). Es decir, si para una tupla, *FechaR* = 12/10/2003 y *FechaD*= 13/10/2003, significa que dicho auto fue alquilado por un día. Además, no pueden registrarse alquileres mayores a 30 días.

Km es la cantidad de km recorridos por ese auto en ese alquiler. Si el auto todavía no fue devuelto, el valor es null.

Un ejemplo de esta base puede ser:

CLIENTE			AUTO			
DNI	NOMBRE	CIUDAD	PATENTE	MARCA	MODELO	COLOR
44	Anabella Luzuriaga	USHUAIA	fyi654	Chevrolet	Corsa	gris
32	Manuel Ruiz	CABA	omg987	Ford	Fiesta	azul
27	Santiago López	LA PLATA	vip195	VW	Vento	gris
35	Luciana Martínez	ROSARIO	aka654	VW	Suran	beige
76	José Pelman	ROSARIO	lol123	Honda	Civic	beige
12	Greta Báez	USHUAIA	mib111	Honda	Civic	azul

ALQUILA						
DNI	Patente	FechaR	CiudadR	FechaD	CiudadD	km
44	vip195	15/03/2015	BARILOCHE			
35	fyi654	10/03/2015	CABA	11/03/2015	ROSARIO	660
32	fyi654	14/01/2015	CABA	18/01/2015	CABA	990
32	aka654	14/02/2015	VIEDMA	15/02/2015	VIEDMA	200
27	omg987	20/03/2015	ROSARIO	22/03/2015	LA PLATA	600
35	lol123	02/02/2015	MENDOZA	09/02/2015	CORDOBA	1300
32	lol123	05/01/2015	CORDOBA	08/01/2015	MENDOZA	330
27	fyi654	04/02/2015	CABA	07/02/2015	CABA	850
27	lol123	13/02/2015	CORDOBA	20/02/2015	CABA	1200
27	fyi654	10/02/2015	CABA	12/02/2015	CABA	700
12	mib111	08/04/2015	USHUAIA	09/04/2015	USHUAIA	300
12	mib111	15/04/2015	CORDOBA	17/04/2015	USHUAIA	1200
12	mib111	20/04/2015	USHUAIA			

En este ejemplo, el cliente con dni 44 alquiló el auto vip195 en la ciudad de Bariloche el 15/03/2015 pero todavía no lo ha devuelto, por eso los atributos, FechaD, CiudadD y km están en null.

Se pide escribir las consultas en SQL2. No se pueden crear tablas temporarias persistentes auxiliares para resolverlas.

1.1)

Mostrar cada DNI acompañado por la cantidad de colores distintos de autos que alquiló sólo para aquellos clientes que nunca han alquilado un auto azul (sin importar si lo devolvieron o no)

Se pide que los clientes que nunca alquilaron nada no aparezcan en el resultado.

Con los datos del ejemplo se obtiene:

DNI	Cantidad
32	2
35	2
44	1

Explicación del resultado:

El cliente con DNI 76 no aparece en el resultado porque nunca alquiló un auto.

El cliente con DNI 27 no aparece en el resultado porque alquiló un auto azul.

El cliente con DNI 32 aparece en resultado porque alquiló 3 autos: el fyi654, color gris, el aka854, color beige y el lol123, color beige, y ninguno de ellos es color azul. El número 2 en la columna **Cantidad** se debe a que alquiló autos de color gris y beige.

Análogamente se calculan los otros clientes.

1.2)

Listar las patentes de los autos que sólo fueron devueltos en la ciudad de residencia del conductor. Es decir, que nunca fueron devueltos en una ciudad distinta a la de residencia del conductor.

Realizar este cálculo excluyendo del análisis las tuplas en las cuales los autos todavía no fueron devueltos.

Con los datos del ejemplo se obtiene:

Patente
omg987
mib111

Explicación del resultado:

El auto con patente omg987 fue alquilado una sola vez y fue devuelto en La Plata por el cliente con dni 27 cuya ciudad de residencia es justamente La Plata.

El auto con patente fyi654 fue alquilado el 10/03/2015 por el cliente con dni 35 y devuelto en Rosario, que es su ciudad de residencia. También, el 14/01/2015, por el cliente con dni 32 y devuelto en CABA, que es su ciudad de residencia. Sin embargo, no aparece en el resultado porque también fue alquilado el día 04/02/2015 por el cliente con dni 27 cuya residencia es en La Plata y fue devuelto en CABA.

Análogamente se calculan las otras patentes.

1.3)

Crear la vista KILOMETROS que contenga para cada cliente, la suma de los km realizados con sus autos alquilados, sólo para los clientes que no superan la suma de 1600 kms.

Para este cálculo deben tomarse las personas de la tabla Cliente. Los autos no devueltos se consideran para este cálculo con recorrido 0(cero). Los clientes que no alquilan autos se consideran para este cálculo con recorrido 0(cero).

Si la vista se consultara de la siguiente forma:

SELECT * FROM kilometros

se obtendría:

KILOMETROS	
DNI	Kms
12	1500
32	1520
44	0
76	0

1.4)

Para obtener el promedio de kilómetros recorridos por día por un auto, se calcula la suma de sus kilómetros recorridos en todos sus alquileres y se lo divide por la suma de la cantidad de días en que el auto estuvo alquilado.

Tomando como base este promedio, se dice que un auto pertenece a:

Categoría 1 si el promedio de kilómetros recorridos por día es menor que 200, es decir, en el intervalo [0,200),
Categoría 2 si el promedio de kilómetros recorridos por día está en el intervalo [200, 300),
Categoría 3 si el promedio de kilómetros recorridos por día es mayor o igual a 300, es decir, en el intervalo [300, ∞)

No se deben tener en cuenta los autos que aún no han sido devueltos.

Se pide obtener la suma total de los kilómetros recorridos por los autos alquilados, correspondientes a las categorías antes mencionadas.

En el resultado se debe mostrar Categoría y Cantidad de kilómetros

Con los datos del ejemplo se obtendría:

Categoría	Kms
Cat 1	2830
Cat 2	200
Cat 3	5300

Explicación del resultado:

Para calcular las categorías se deben obtener los promedios de km/día de los autos alquilados.

El promedio de kilómetros por día del auto con patente *fyi654* es 320 y se obtiene de la suma de sus kilómetros recorridos (660+990+850+700) dividido por la cantidad de días (1+4+3+2) : $3200/10=320$

Por lo tanto este auto corresponde a la **categoría 3**.

El promedio de kilómetros por día del auto *omg987* es 300 que surge de sumar sus km recorridos, 600, y dividirlos por los día alquilados, 2: $600/2=300$.

Este auto corresponde entonces a la **categoría 3**.

El promedio de kilómetros por día del auto *mib111* es 1500 que surge de sumar sus km recorridos, (300+1200), y dividirlos por los día alquilados, (1+2): $1500/3=500$.

Este auto corresponde entonces a la **categoría 3**.

Estos son los únicos autos de esa categoría. La suma de los kilómetros recorridos por estos autos es $3200+600+1500 = 5300$.

Análogamente se obtienen el resto de las categorizaciones y las sumas.

Ejercicio 2

Se dice que dos consultas SQL son equivalentes si al ejecutarlas para cualquier instancia, producen el mismo resultado (salvo quizás el orden de aparición de las tuplas resultantes que es lo único en lo que podrían diferir). Para que dos consultas SQL se consideren equivalente sus resultados deberían coincidir hasta en la cantidad de tuplas.

Dado el siguiente esquema:

R(A,B)

S(D, E, F)

Para el cual:

en R:

A es clave primaria

en S:

DE es clave primaria

D es clave foránea que referencia a R(A).

Y todos los atributos son de tipo entero.

2.1)

Dada la consulta:

```
SELECT DISTINCT A, B
FROM R JOIN S ON R.A=S.D
WHERE S.F = 9;
```

Escribir una consulta SQL equivalente que no utilice el DISTINCT.

2.2)

Dada la consulta:

```
SELECT MAX(F)
FROM S;
```

Escribir una consulta SQL equivalente que no utilice funciones de agregación.

Ejercicio 3

Para los siguientes ejercicios se utiliza el esquema de relación del **ejercicio 1**.

3.1) Dada la siguiente consulta expresada en Cálculo relacional de tuplas

$\{T | (\exists C)(\text{cliente}(C) \wedge \neg (\exists R) (\text{auto}(R) \wedge R[\text{color}] = \text{'beige'} \wedge \neg (\exists S) (\text{alquila}(S) \wedge S[\text{patente}] = R[\text{patente}] \wedge S[\text{dni}] = C[\text{dni}]) \wedge T[\text{nombre}] = C[\text{nombre}])))\}$

Escribir una consulta en Algebra Relacional que devuelva el mismo resultado.

3.2) Dada la siguiente consulta expresada en Algebra Relacional

$Auxi \leftarrow \pi_{dni} (\sigma_{a1.dni = a2.dni \text{ and } a1.patente \neq a2.patente} (\rho_{a1} (\sigma_{fechar \geq \text{'01/02/2015'} \text{ and } fechar \leq \text{'28/02/2015'}}(\text{alquila}))) \times (\rho_{a2} (\sigma_{fechar \geq \text{'01/02/2015'} \text{ and } fechar \leq \text{'28/02/2015'}}(\text{alquila}))))$

$Resultado \leftarrow \pi_{nombre} ((\pi_{dni} (\sigma_{fechar \geq \text{'01/02/2015'} \text{ and } fechar \leq \text{'28/02/2015'}}(\text{alquila})) - auxi) \bowtie cliente)$

Escribir una consulta en Calculo Relacional por Dominios que devuelva el mismo resultado.

Ejercicio 4

Ya se tienen creadas las tablas correspondientes a las entidades A, B, C y D. Escribir la sentencia SQL que genere la tabla correspondiente a la relación R. No omitir las claves candidatas.

