

Nombre:.....

Legajo:

Recuperatorio del Primer Parcial de Base de Datos

Ejercicio 1	Ejercicio 2	Ejercicio 3	Ejercicio 4	Nota
/ 2.1	/ 4.4	/1.2	/2.3	

Condición Mínima de Aprobación: Sumar 4 puntos en total, 3 de los cuales deben ser de [Ejercicio 1+ Ejercicio 2]

Ejercicio 1

Se dice que dos consultas SQL son equivalentes si al ejecutarlas para cualquier instancia, producen el mismo resultado (con la misma cantidad de tuplas y valores), salvo quizás el orden de aparición de las tuplas resultantes, que es lo único en lo que podrían diferir).

Para cada par de consultas se pide indicar si son o no equivalentes. Si lo son, hay que expresar en forma “coloquial” lo que se está buscando obtener con la consulta. Si no son equivalentes, se debe mostrar un contraejemplo, o sea, una instancia que evidencie la diferencia, indicando los resultados que generarían Q1 y Q2. Si el problema es que alguna de las consultas ni siquiera compila, señalarlo claramente explicando su error sintáctico.

Todas las consultas se aplican sobre los siguientes esquemas de relación:

PELICULA(titulo, director, year)
 INTERPRETE(titulo, actor, salario)
 CRITICA(titulo, critico, estrellas)

La tabla PELICULA contiene nombre de película, quien la dirigió y en que año se estrenó. La tabla INTERPRETE permite conocer los actores que participaron en las películas y los sueldos que percibieron por sus participaciones en cada una de ellas. Por último la tabla CRÍTICA muestra la cantidad de estrellas que cada crítico del espectáculo le asignó a cada película.

Tener en cuenta que la equivalencia entre consultas debe ser válida para cualquier instancia de las tablas (vacías, con una sola tupla, con muchas tuplas, etc.)

1.1)

Q1:

```
SELECT director, AVG(estrellas)
FROM PELICULA p1 , CRITICA
WHERE p1.titulo = CRITICA.titulo AND
director IN
  (SELECT director
   FROM PELICULA p2, INTERPRETE
   WHERE actor = 'Pitt'
   AND p1.titulo=INTERPRETE.titulo
  )
GROUP BY director
```

Q2:

```
SELECT director, AVG(estrellas)
FROM PELICULA , CRITICA
WHERE EXISTS
  (SELECT *
   FROM INTERPRETE
   WHERE actor= 'Pitt'
   AND PELICULA.titulo=INTERPRETE.titulo
  )
  AND PELICULA.titulo= CRITICA.titulo
GROUP BY director
```

1.2)

Q3:

```
SELECT PELICULA.titulo
FROM PELICULA, INTERPRETE
WHERE PELICULA.titulo= INTERPRETE.titulo
      AND PELICULA.year=2004
GROUP BY INTERPRETE.titulo
HAVING COUNT(INTERPRETE.actor)>5
```

Q4:

```
SELECT PELICULA.titulo
FROM PELICULA
WHERE PELICULA.year=2004
      AND PELICULA.titulo IN
      (SELECT actor
      FROM INTERPRETE
      WHERE interprete.titulo=pelicula.titulo
      GROUP BY actor
      HAVING COUNT(*)>5)
```

1.3)

Q5:

```
SELECT DISTINCT critico
FROM (SELECT *
      FROM CRITICA C1
      WHERE NOT EXISTS
            (SELECT *
            FROM CRITICA C2
            WHERE C2.critico=C1.critico
            AND C2.estrellas>c1.estrellas )
      ) P
WHERE estrellas=3
ORDER BY critico;
```

Q6:

```
SELECT DISTINCT C1.critico
FROM CRITICA C1
WHERE C1.estrellas=3
AND NOT EXISTS
      (SELECT *
      FROM CRITICA C2
      WHERE C2.critico = C1.critico AND
            C2.estrellas>3)
ORDER BY critico;
```

Ejercicio 2

Sea el siguiente esquema de base de datos relacional que se refiere a una empresa que terceriza las encuestas de satisfacción de sus clientes sobre algunos de sus productos, con ciertos encuestadores:

PRODUCTO(ProdID, descripcion)
CLIENTE(ClientID, nombre)
ENCUESTADOR(EncID, nombre)
LOG(ProdID, ClientID, EncID, fecha, satisfaccion)

La tabla ENCUESTADOR guarda información de los códigos y nombre de la empresa encuestadora (externa a la organización). La tabla PRODUCTO guarda información de los códigos y descripciones de los productos que vende la empresa. La tabla CLIENTE guarda información de los códigos y nombres de los clientes de la empresa. La tabla LOG guarda información sobre la satisfacción de los clientes con algunos de los productos, según los datos provistos por los encuestadores. Para ello registran: código del producto sobre el que se preguntó, código del cliente encuestado, código del encuestador que realizó esa encuesta, fecha de realización (sólo se guarda dd/mm/yyyy), satisfacción el cual se representa por un número entero entre 1 y 10 (1 baja satisfacción y 10 máxima satisfacción). Si el cliente no quiere indicar respuesta sobre el producto (no sabe/no contesta) se representa con NULL.

Ejemplo:

Log				
ProdID	ClientID	EncID	Fecha	Satisfaccion
10	921	102	'12/10/2004'	5
30	921	102	'12/10/2004'	5
50	921	165	'16/02/2004'	5
10	3420	165	'16/02/2004'	3
20	3420	165	'18/02/2004'	7
30	3420	165	'18/02/2004'	7
40	3420	220	'12/10/2004'	9
50	3420	220	'12/10/2004'	9
10	200	102	'15/02/2004'	2
30	200	102	'15/02/2004'	NULL
50	933	220	'15/02/2004'	8
60	971	370	'15/02/2004'	7

Producto	
ProdID	Descripcion
10	'PYT'
20	'BUH'
30	'CRE'
40	'GIR'
50	'TTO'
60	'ABC'

Cliente	
ClientID	Nombre
921	'Rial'
1001	'Ray'
3420	'Stewart'
200	'Shrek'
933	'Felicitas'
971	'Dumas'

Encuestador	
EncID	Nombre
102	'Andrade'
165	'Hedn'
220	'IANNYS'
370	'Loprano'

Muy Importante

Todas las consultas que se solicitan a continuación deben ser resueltas sin crear explícitamente otras tablas temporarias, ni vistas auxiliares, pero sí se permiten consultas anidadas.

2.1)

Encontrar el nombre del producto más encuestado junto con la cantidad de personas encuestadas sobre él. No contabilizar a las personas que si bien fueron encuestadas sobre dicho producto, respondieron con un “no sabe/no contesta”. Para las tablas del ejemplo, se debería obtener:

Descripción	Cantidad
PYT	3
TTO	3

2.2)

Imprimir el nombre de los clientes que fueron encuestados sobre todos los productos antes de septiembre del 2004. Para las tablas del ejemplo no se deben obtener tuplas.

2.3)

Mostrar los nombres de los productos que por lo menos fueron encuestados 3 veces, acompañado por un cartel que indique el grado promedio de satisfacción obtenido. El cartel debe responder a la siguiente convención: entre 1 y 3 MALO, entre 4 y 5 REGULAR, entre 6 y 7 BUENO, entre 8 y 9 MUY BUENO y 10 EXCELENTE. Tener en cuenta que se quiere que aquellos productos que tuvieron alguna encuesta que arrojó un NO SABE/NO CONTESTA, intervengan en el promedio con valor -5.

Para las tablas del ejemplo, se debería obtener:

Descripción	Cartel
PYT	'Regular'
CRE	'Malo'
TTO	'Muy Bueno'

2.4)

Mostrar el nombre de los encuestadores que nunca hayan encuestado a clientes encuestados alguna vez por otros encuestadores, es decir, los encuestadores que tengan sólo clientes exclusivos. Para las tablas del ejemplo se debe obtener:

Nombre
Loprano

Ejercicio 3

En el modelo relacional, se tiene un esquema de relación **R(a, b, c)**. Se dispone de una instancia **r1** de **R** con **N1** tuplas y otra instancia **r2** de **R** con **N2** tuplas. Se sabe que **N2 > N1 > 0**.

Completar el siguiente cuadro, indicando cuál es la mínima y máxima cantidad de tuplas que puede obtenerse en cada relación resultante, al aplicar los siguientes operadores del Álgebra Relacional:

Relación Resultante	Mínima cantidad de tuplas	Máxima cantidad de tuplas
$\sigma_{a=3 \text{ and } b=8}(r1)$		
$\pi_{a,c}(r1)$		
$r1 \times r2$		
$r1 \cup r2$		
$r1 \cap r2$		

Ejercicio 4

Sea un sistema con páginas de 4 KB, punteros de 4 bytes y RID de 4 bytes.

Se crea la tabla DATOS con dos campos C1 (cadena fija de 46 bytes) y C2 (cadena fija de 44 bytes cada uno) y las siguientes características:

- Header a nivel bloque de 96 bytes
- Header a nivel tupla de 10 bytes.
- PCTFREE = 60
- PCTUSED = 20

Sobre el campo C1 se crea un índice B+Tree denso, con las siguientes características:

- Header a nivel bloque de 96 bytes
- PCTFREE = 40
- PCTUSED = 10

Calcular la mínima cantidad de bloques que llega a tener la tabla DATOS en el instante en que su índice alcanza altura 2 (la raíz se considera nivel 0). Justificar debidamente, detallando cada cálculo realizado.