

EJERCICIO 3 (cont)**(2.25 puntos)**

- 3.- ¿A que consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.75 puntos)

```
SELECT cli.nombre
FROM clientes cli, facturas fac
WHERE cli.codcli = fac.codcli
AND EXISTS
( SELECT *
  FROM lineas_fac lin
  WHERE fac.codfac = lin.codfac
  GROUP BY lin.codfac
  HAVING ( SUM(lin.precio * lin.cant * NVL(lin.dto,0) / 100) *
          (1 + NVL(fac.dto,0)) * (1 + NVL(fac.iva,0))) > 10000 )
GROUP BY cli.codcli, cli.nombre
HAVING COUNT(*) > 3
ORDER BY 1;
```


EJERCICIO 3 (cont)**(2.25 puntos)**

- 3.- ¿A que consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.75 puntos)

```
SELECT  art.descrip
FROM    articulos art
WHERE   NOT EXISTS
      ( SELECT  *
        FROM    clientes cli, pueblos pue, provincias prv
        WHERE   prv.nombre = 'CASTELLON'
              AND prv.codpro = pue.codpro
              AND pue.codpue = cli.codpue
              AND NOT EXISTS
                    ( SELECT  *
                      FROM    facturas fac, lineas_fac lin
                      WHERE   fac.codfac = lin.codfac
                            AND fac.codcli = cli.codcli
                            AND lin.codart = art.codart  ) );

ORDER BY 1;
```


EJERCICIO 3 (cont)**(2.25 puntos)**

- 3.- ¿A que consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.75 puntos)

```
SELECT cli.nombre, art.descrip
FROM clientes cli, articulos art
WHERE cli.codcli IN
    ( SELECT fac.codcli
      FROM facturas fac, lineas_fac lin
      WHERE fac.codfac = lin.codfac
      GROUP BY fac.codcli
      HAVING SUM(lin.cant) >= ALL
        ( SELECT SUM(lin.cant)
          FROM facturas fac, lineas_fac lin
          WHERE fac.codfac = lin.codfac
          GROUP BY fac.codcli ) )
AND art.codart IN
    ( SELECT lin.codart
      FROM facturas fac, lineas_fac lin
      WHERE fac.codfac = lin.codfac
      AND fac.codcli = cli.codcli
      GROUP BY lin.codart
      HAVING SUM(lin.cant) >= ALL
        ( SELECT SUM(lin.cant)
          FROM facturas fac, lineas_fac lin
          WHERE fac.codfac = lin.codfac
          AND fac.codcli = cli.codcli
          GROUP BY lin.codart ) )
ORDER BY 1, 2 ;
```

- 1.- Mostrar, ordenadamente, los nombres de las provincias que tienen el mayor número de pueblos. (0.75 puntos)
- 2.- Mostrar, ordenadamente, los artículos que siempre aparecen solos en una factura, es decir, no aparece ningún otro artículo en ninguna de sus líneas de las facturas en las que aparece dicho artículo. (0.75 puntos)

EJERCICIO 3 (cont)**(2.25 puntos)**

- 3.- ¿A que consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.75 puntos)

```
SELECT  art1.descrip, art2.descrip
FROM    articulos art1, articulos art2
WHERE   art1.codart < art2.codart
AND     NOT EXISTS
      ( SELECT  lin.codfac
        FROM    lineas_fac lin
        WHERE   lin.codart = art1.codart
      MINUS
      SELECT  lin.codfac
        FROM    lineas_fac lin
        WHERE   lin.codart = art2.codart )
AND     NOT EXISTS
      ( SELECT  lin.codfac
        FROM    lineas_fac lin
        WHERE   lin.codart = art2.codart
      MINUS
      SELECT  lin.codfac
        FROM    lineas_fac lin
        WHERE   lin.codart = art1.codart )
AND     art1.codart IN
      ( SELECT  lin.codart  FROM lineas_fac lin)
ORDER BY 1, 2;
```


EJERCICIO 3 (cont)**(2.25 puntos)**

- 3.- ¿A que consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.75 puntos)

```
SELECT cli.nombre
FROM clientes cli, facturas fac
WHERE cli.codcli = fac.codcli
AND TO_CHAR(fac.fecha,'YYQ') = '964'
AND cli.codcli NOT IN
( SELECT cli.codcli
  FROM clientes cli, facturas fac
  WHERE cli.codcli = fac.codcli
    AND TO_CHAR(fac.fecha,'YYQ') = '964'
MINUS
SELECT cli.codcli
  FROM clientes cli, facturas fac
  WHERE cli.codcli = fac.codcli
    AND TO_CHAR(fac.fecha,'YYQ') = '963' )
ORDER BY 1;
```


EJERCICIO 3 (cont)**(2.25 puntos)**

- 3.- ¿A que consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.75 puntos)

```
SELECT  art.descrip
FROM    articulos art, lineas_fac lin1, facturas fac1, clientes cli1, pueblos pue1,
        provincias prv1
WHERE   art.codart = lin1.codart AND lin1.codfac = fac1.codfac
        AND fac1.codcli = cli1.codcli AND cli1.codpue = pue1.codpue
        AND pue1.codpro = prv1.codpro
        AND UPPER (prv1.nombre) = 'CASTELLON'
        AND MONTHS_BETWEEN (fac1.fecha, SYSDATE) < 1
GROUP BY art.codart, art.descrip
HAVING  SUM(lin1.cant) < ANY
        ( SELECT  SUM(lin2.cant)
          FROM    lineas_fac lin2, facturas fac2, clientes cli2, pueblos pue2,
                provincias prv2
          WHERE   art.codart = lin2.codart AND lin2.codfac = fac2.codfac
                AND fac2.codcli = cli2.codcli AND cli2.codpue = pue2.codpue
                AND pue2.codpro = prv2.codpro
                AND UPPER (prv2.nombre) <> 'CASTELLON'
                AND MONTHS_BETWEEN (fac2.fecha, SYSDATE) < 1
          GROUP BY prv2.codpro )
ORDER BY 1
```


EJERCICIO 3 (cont)**(2.25 puntos)**

- 3.- ¿A que consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.75 puntos)

```
SELECT  pro.nombre, COUNT (*)
FROM    provincias pro, pueblos pue, clientes cli, facturas fac
WHERE   pro.codpro= pue.codpro AND pue.codpue = cli.codpue
        AND cli.codcli = fac.codcli
        AND UPPER (pro.nombre) IN
            ('CASTELLON','VALENCIA','ALICANTE')
        AND 10000 <
            ( SELECT  SUM (lin.precio * lin.cant)
              FROM    lineas_fac lin
              WHERE   fac.codfac = lin.codfac )
GROUP BY pro.codpro, pro.nombre
UNION   SELECT  pro.nombre, 0
        FROM    provincias pro
        WHERE   UPPER (pro.nombre) IN
            ('CASTELLON','VALENCIA','ALICANTE')
        AND    NOT EXISTS
            ( SELECT *
              FROM  pueblos pue, clientes cli, facturas fac
              WHERE pro.codpro= pue.codpro
                AND  pue.codpue = cli.codpue
                AND  cli.codcli = fac.codcli
                AND  10000 <
                    ( SELECT  SUM (lin.precio * lin.cant)
                      FROM    lineas_fac lin
                      WHERE   fac.codfac = lin.codfac ) )
GROUP BY pro.codpro, pro.nombre
ORDER BY 1;
```


EJERCICIO 3 (cont)

(2.25 puntos)

- 3.- ¿A que consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.75 puntos)

```

SELECT fac.codfac, cli.nombre, fac.fecha, fac.iva,
SUM(lin.precio * lin.cant), SUM (lin.cant)
FROM clientes cli, facturas fac, lineas_fac lin
WHERE cli.codcli = fac.codcli AND fac.codfac = lin.codfac
AND TO_CHAR (fac.fecha,'YY')= TO_CHAR (SYSDATE,'YY')
GROUP BY fac.codfac, cli.codcli, cli.nombre, fac.fecha, fac.iva
HAVING SUM (lin.precio * lin.cant) =
( SELECT MAX(SUM (lin.precio * lin.cant))
FROM facturas fac, lineas_fac lin
WHERE fac.codfac = lin.codfac
AND TO_CHAR (fac.fecha,'YY')=
TO_CHAR (SYSDATE,'YY') )
UNION SELECT fac.codfac, cli.nombre, fac.fecha, fac.iva,
SUM(lin.precio * lin.cant), SUM (lin.cant)
FROM clientes cli, facturas fac, lineas_fac lin
WHERE cli.codcli = fac.codcli AND fac.codfac = lin.codfac
AND TO_CHAR (fac.fecha,'YY')= TO_CHAR (SYSDATE,'YY')
GROUP BY fac.codfac, cli.codcli, cli.nombre, fac.fecha, fac.iva
HAVING SUM (lin.cant) =
( SELECT MAX(SUM (lin.cant))
FROM facturas fac, lineas_fac lin
WHERE fac.codfac = lin.codfac
AND TO_CHAR (fac.fecha,'YY')=
TO_CHAR (SYSDATE,'YY') )
ORDER BY 1;

```


- 1.- Mostrar, ordenadamente, los pueblos donde la suma de las compras de sus clientes en el último año superó en un 5% las compras de hace dos años. (No tener en cuenta los descuentos ni el IVA) (0.75 puntos)
- 2.- Mostrar, ordenadamente, los artículos que cumplen que nunca se les aplica descuento cuando se venden a los clientes que no son de CASTELLÓN y el descuento a los clientes de CASTELLÓN siempre es mayor que 20. (0.75 puntos)

EJERCICIO 3 (cont)**(2.25 puntos)**

- 3.- ¿A que consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.75 puntos)

```
SELECT  art.descrip
FROM    articulos art
WHERE   art.precio <
        ( SELECT  NVL(AVG(lin.precio),0)
          FROM    lineas_fac lin, facturas fac
          WHERE   lin.codart = art.codart
                AND fac.codfac = lin.codfac
                AND TO_CHAR(fac.fecha,'MMYYYY') = '121998' )
AND EXISTS
        ( SELECT  *
          FROM    lineas_fac lin, facturas fac
          WHERE   lin.codart = art.codart
                AND fac.codfac = lin.codfac
                AND TO_CHAR(fac.fecha,'MMYYYY') = '121998' )
AND     art.precio >
        ( SELECT  NVL(AVG(lin.precio),0)
          FROM    lineas_fac lin, facturas fac
          WHERE   lin.codart = art.codart
                AND fac.codfac = lin.codfac
                AND TO_CHAR(fac.fecha,'MMYYYY') = '111998' )
AND EXISTS
        ( SELECT  *
          FROM    lineas_fac lin, facturas fac
          WHERE   lin.codart = art.codart
                AND fac.codfac = lin.codfac
                AND TO_CHAR(fac.fecha,'MMYYYY') = '111998' )
ORDER BY 1 ;
```


EJERCICIO 3 (cont)**(2 puntos)**

- 3.- ¿A que consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.5 puntos)

```
SELECT cli.nombre
FROM clientes cli, facturas fac, lineas_fac lin
WHERE cli.codcli = fac.codcli AND fac.codfac = lin.codfac
AND TO_CHAR (fac.fecha,'YYYY') = TO_CHAR (SYSDATE,'YYYY')
GROUP BY cli.codcli, cli.nombre
HAVING SUM (lin.cant) >
( SELECT
SUM (DECODE (cli.codcli,fac.codcli,
DECODE (fac.codfac,lin.codfac,
DECODE (TO_CHAR (fac.fecha,'YYYY'),
TO_CHAR (SYSDATE,'YYYY'),lin.cant,0),0),0))
/
COUNT (DISTINCT
DECODE (cli.codcli,fac.codcli,
DECODE (fac.codfac,lin.codfac,
DECODE (TO_CHAR (fac.fecha,'YYYY'),
TO_CHAR (SYSDATE,'YYYY'),cli.codcli))))
FROM clientes cli, facturas fac, lineas_fac lin
)
ORDER BY 1;
```

- 1.- Mostrar, ordenadamente, las provincias donde al menos 8 clientes han superado las 10.000 pesetas de compras, sin tener en cuenta descuentos ni iva, durante el último año. (0.75 puntos)
- 2.- Mostrar, ordenadamente, los artículos que han sido vendidos a menos de 20 clientes. (0.75 puntos)

EJERCICIO 3 (cont)**(2 puntos)**

- 3.- ¿A qué consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.5 puntos)

```
SELECT cli.nombre
FROM clientes cli, facturas fac, lineas_fac lin
WHERE cli.codcli = fac.codcli AND fac.codfac = lin.codfac
AND TO_NUMBER(TO_CHAR(fac.fecha,'YYYY')) =
TO_NUMBER(TO_CHAR(SYSDATE,'YYYY')) - 1
GROUP BY cli.codcli, cli.nombre
HAVING SUM(lin.precio * lin.cant) * 0,5 > any
( SELECT SUM(lin.precio * lin.cant)
FROM facturas fac, lineas_fac lin
WHERE cli.codcli = fac.codcli AND fac.codfac = lin.codfac
AND TO_NUMBER(TO_CHAR(fac.fecha,'YYYY')) =
TO_NUMBER(TO_CHAR(SYSDATE,'YYYY')) - 1
GROUP BY TO_CHAR(fac.fecha,'MM') )
ORDER BY 1;
```


EJERCICIO 3 (cont)**(2 puntos)**

- 3.- ¿A qué consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.5 puntos)

```
SELECT  art.descrip, MAX(ABS(lin.precio-art.precio)) VENTA_ACTUAL
FROM    articulos art, lineas_fac lin, facturas fac
WHERE   art.codart = lin.codart AND lin.codfac = fac.codfac
        AND    TO_NUMBER(TO_CHAR(fac.fecha,'YYYY')) =
                TO_NUMBER(TO_CHAR(SYSDATE,'YYYY')) - 1
GROUP BY art.codart, art.descrip
HAVING  MAX(ABS(lin.precio-art.precio)) >=
        ( SELECT  MAX(MAX(ABS(lin.precio-art.precio)))
          FROM    articulos art, lineas_fac lin, facturas fac
          WHERE   art.codart = lin.codart AND lin.codfac = fac.codfac
                AND    TO_NUMBER(TO_CHAR(fac.fecha,'YYYY')) =
                        TO_NUMBER(TO_CHAR(SYSDATE,'YYYY')) - 1
          GROUP BY lin.codart )
ORDER BY 1;
```


EJERCICIO 3

(2 puntos)

- 1.- Mostrar, ordenadamente, los artículos que están por debajo de su stock mínimo, junto con la fecha de la factura en la que se produjo la venta que provocó que el stock estuviera por debajo del stock mínimo. Hay que tener en cuenta que dicha factura no tiene que ser la última. (0.75 puntos)
- 2.- Mostrar, ordenadamente, los pueblos, y la provincia a la que pertenece, en los que todos sus clientes han realizado compras superiores a 10000 pesetas, sin tener en cuenta descuentos e iva, durante el año pasado. (0.75 puntos)

EJERCICIO 3 (cont)**(2 puntos)**

- 3.- ¿A qué consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.5 puntos)

```
SELECT TO_CHAR (fac.fecha,'MONTH')
FROM facturas fac, lineas_fac lin
WHERE fac.codfac = lin.codfac
AND TO_NUMBER(TO_CHAR (fac.fecha,'YYYY')) =
TO_NUMBER(TO_CHAR (SYSDATE,'YYYY')) - 1
GROUP BY TO_CHAR (fac.fecha,'MONTH')
HAVING 3 >
( SELECT COUNT(COUNT(*))
FROM facturas fac2, lineas_fac lin2
WHERE fac2.codfac = lin2.codfac
AND TO_NUMBER(TO_CHAR (fac2.fecha,'YYYY')) =
TO_NUMBER(TO_CHAR (SYSDATE,'YYYY')) - 1
GROUP BY TO_CHAR (fac2.fecha,'MONTH')
HAVING SUM(lin.precio * lin.cant) < SUM(lin2.precio * lin2.cant) )
ORDER BY 1;
```


EJERCICIO 3 (cont)**(2 puntos)**

- 3.- ¿A qué consulta corresponde la siguiente sentencia SQL?. Responder sin ambigüedad.
(0.5 puntos)

```
SELECT  art.descrip, TO_CHAR (fac.fecha,'YYYY')
FROM    facturas fac, articulos art, lineas_fac lin
WHERE   fac.codfac = lin.codfac AND lin.codart = art.codart
GROUP BY TO_CHAR (fac.fecha,'YYYY'), art.codart, art.descrip
HAVING
        TO_NUMBER (
            TO_CHAR (
                TO_DATE ('3112'|| TO_CHAR(fac.fecha,'YYYY'),
                    'DDMMYYYY'), 'WW')) =
        COUNT (DISTINCT TO_CHAR(fac.fecha,'WW'))
ORDER BY 1;
```