1.- Mostrar, ordenadamente, los clientes que cumplen que todas sus facturas tienen un descuento que sea mayor que la media del descuento de sus líneas. (0.75 puntos)

2.- Mostrar, ordenadamente, los clientes junto con los artículos que solicitaron durante el primer semestre del año 95 y no solicitaron esos mismos artículos en el segundo semestre del mismo año. (0.75 puntos)

(2.25 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;
```

1.- Mostrar el coste total de los artículos que tenemos en stock cuyo precio por unidad sea mayor de 2000 pesetas, de los que se ha solicitado en el último mes menos cantidad que su stock mínimo. (0.75 puntos)

2.- Mostrar, ordenadamente, los clientes que han comprado un mismo artículo en más de una factura. (0.75 puntos)

(2.25 puntos)

```
SELECT art.descrip
  FROM articulos art
  WHERE NOT EXISTS
     ( SELECT *
          FROM clientes cli, pueblos pue, provincias prv
                    prv.nombre = 'CASTELLON'
          WHERE
                    prv.codpro = pue.codpro
pue.codpue = cli.codpue
             AND
             AND
                     NOT EXISTS
             AND
               ( SELECT
                     FROM facturas fac, lineas_fac lin
                     WHERE fac.codfac = lin.codfac
                               fac.codcli = cli.codcli
                       AND
                       AND
                               lin.codart = art.codart ) );
  ORDER BY 1;
```

1.- Mostrar, ordenadamente, los clientes que han comprado en el año 1996, todos los articulos que tienen un stock menor que el stock mínimo. (0.75 puntos)

2.- Mostrar, ordenamente, los clientes que han solicitado más de dos clases de tornillos durante el mes de Diciembre de 1996.

(2.25 puntos)

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

3° I.I.

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)

(2.25 puntos)

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

1.- Mostrar, ordenadamente por provincias, el nombre de la provincia y el número de clientes que durante el año pasado han facturado más de 100000 ptas, sin tener en cuenta el descuento y el iva. (0.75 puntos)

2.- Mostrar, ordenadamente, los clientes que nunca han realizado más de una compra en una fecha determinada. (0.75 puntos)

(2.25 puntos)

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

1.- Mostrar, ordenadamente, los clientes que han realizado la devolución de un artículo más de una semana después de la fecha de compra. Se asume que la fecha de compra coincide con la de la factura más proxima en la que aparece el artículo vendido. (0.75 puntos)

2.- Mostrar, ordenadamente, los artículos que durante el año pasado sufrieron todos los tipos de IVA, {0 ó NULL, 7, 16}, y que cumple que el precio actual es menor que todos los aplicados durante el año pasado. (0.75 puntos)

(2.25 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
             fac1.codcli = cli1.codcli AND cli1.codpue = pue1.codpue
     AND
     AND
             pue1.codpro = prv1.codpro
            UPPER (prv1.nombre) = 'CASTELLON'
     AND
            MONTHS_BETWEEN (fac1.fecha, SYSDATE) < 1
     AND
  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
                    fac2.codcli = cli2.codcli AND cli2.codpue = pue2.codpue
             AND
             AND
                    pue2.codpro = prv2.codpro
                    UPPER (prv2.nombre) <> 'CASTELLON'
             AND
                    MONTHS_BETWEEN (fac2.fecha, SYSDATE) < 1
             AND
          GROUP BY prv2.codpro )
  ORDER BY 1
```

EJERCICIO 3	(2.25 puntos)

1.- Mostrar, ordenadamente, los clientes que durante el año pasado han comprado todos los artículos cuya descripción incluye el texto BASE. (0.75 puntos)

 Mostrar, ordenadamente, los meses del año pasado en los que se realizó una venta menor que la media mensual de dicho año. (No tener en cuenta los descuentos ni el IVA) (0.75 puntos)

(2.25 puntos)

```
SELECT pro.nombre, COUNT (*)
  FROM provincias pro, pueblos pue, clientes cli, facturas fac
             pro.codpro= pue.codpro AND pue.codpue = cli.codpue
  WHERE
     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
          SELECT pro.nombre, 0
FROM provincias pro
WHERE UPPER (pro.nombre) IN
('CASTELLON','VALENCIA','ALICANTE')
UNION
                ÀND
                        NOT EXISTS
                   ( SELECT *
                        FROM pueblos pue, clientes cli, facturas fac
                         WHERE
                                   pro.codpro= pue.codpro
                                   pue.codpue = cli.codpue
                           AND
                           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;
```

1.- Mostrar las provincias de la comunidad valenciana ordenadas respecto del número de pueblos de cada provincia con un nombre compuesto, es decir con un carácter blanco en su nombre. El resultado debe mostrar una fila para cada provincia. (Hay que considerar que una provincia puede no tener ningún pueblo con nombre compuesto) (0.75 puntos)

2.- Mostrar, ordenadamente, los clientes que han realizado compras durante este año que superan el 80% de las compras realizadas por el resto de clientes durante este mes. (No tener en cuenta los descuentos ni el IVA) (0.75 puntos)

(2.25 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
             cli.codcli = fac.codcli AND fac.codfac = lin.codfac
  WHERE
             TO_CHAR (fac.fecha,'YY') = TO_CHAR (SYSDATE,'YY')
     AND
  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')
                     fac.codfac, cli.nombre, fac.fecha, fac.iva,
UNION
          SELECT
                     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
             TO CHAR (fac.fecha, 'YY') = TO CHAR (SYSDATE, 'YY')
     AND
  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
                     fac.codfac = lin.codfac
           WHERE
                     TO_CHAR (fac.fecha,'YY') =
             AND
                        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)

(2.25 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
                      fac.codfac = lin.codfac
               AND
               AND
                      TO_CHAR(fac.fecha,'MMYYYY') = '121998')
     AND EXISTS
       ( SELECT *
            FROM lineas_fac lin, facturas fac
                      lin.codart = art.codart
            WHERE
                      fac.codfac = lin.codfac
               AND
                      TO CHAR(fac.fecha,'MMYYYY') = '121998' )
               AND
     AND
            art.precio >
       ( SELECT NVL(AVG(lin.precio),0)
            FROM lineas fac lin, facturas fac
                      lin.codart = art.codart
            WHERE
               AND
                      fac.codfac = lin.codfac
               AND
                      TO_CHAR(fac.fecha,'MMYYYY') = '111998')
     AND EXISTS
       ( SELECT *
            FROM lineas fac lin, facturas fac
                      lin.codart = art.codart
            WHERE
               AND
                      fac.codfac = lin.codfac
               AND
                      TO_CHAR(fac.fecha,'MMYYYY') = '111998')
  ORDER BY 1;
```

EJERCICIO 3	(2 puntos)

1.- Listar, ordenadamete, la descripción de todos los artículos junto con el número de días del año pasado en el que se realizaron ventas de dicho artículo. (0.75 puntos)

2.- Mostrar el mes del año pasado en el que se realizaron más ventas (sin tener en cuenta descuentos e iva). (0.75 puntos)

(2 puntos)

```
SELECT cli.nombre
  FROM clientes cli, facturas fac, lineas_fac lin
             cli.codcli = fac.codcli AND fac.codfac = lin.codfac
  WHERE
             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;
```

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)

(2 puntos)

```
SELECT cli.nombre
  FROM clientes cli, facturas fac, lineas fac lin
              cli.codcli = fac.codcli AND fac.codfac = lin.codfac
  WHERE
              TO_NUMBER(TO_CHAR(fac.fecha,'YYYY')) =
     AND
              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
                      cli.codcli = fac.codcli AND fac.codfac = lin.codfac
           WHERE
              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;
```

1.- Mostrar, ordenadamente, los clientes que sólo han hecho compras en un único mes, en todos los años en los que aparece como cliente, debiéndose visualizar también el mes en cuestión y el rango de los años. (0.75 puntos)

2.- Mostrar, ordenadamente, los artículos en el último año que SÓLO han sido comprados en la provincia de Castellón. (0.75 puntos)

(2 puntos)

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

FICHEROS Y BASES DE DATOS

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 provoco que el stock estuviera por debajo del stock mínimo. Hay que tener en cuenta que dicha factura no tiene porque 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)

(2 puntos)

```
SELECT TO_CHAR (fac.fecha,'MONTH')
  FROM facturas fac, lineas fac lin
               fac.codfac = lin.codfac
  WHERE
               TO_NUMBER(TO_CHAR (fac.fecha,'YYYY')) =
     AND
            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
                    fac2.codfac = lin2.codfac
       WHERE
          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;
```

01/09/2000

EJERCICIO 3 (2 puntos)

Mostrar, ordenadamente, el nombre de las provincias que durante el año pasado tienen un volumen de ventas superior a la media de las ventas de las provincias que han tenido alguna venta dicho año (no tener en cuenta descuentos e iva). (0.75 puntos)

2.-Mostrar, ordenadamente, los clientes a los que durante el año pasado se les aplicaron todos los diferentes iva que se aplicaron durante ese año. (0.75 puntos)

(2 puntos)