

-- 1. Give the segmental location of the nodules of patient 2 . Return NodulID and LocSeg.

-- Only two results: two nodules

```
SELECT N.locseg, N.nodulid
FROM nodul N
WHERE N.pacient_ID = 2;
```

-- <> stands for different from patient 2 (not equal to)

```
SELECT N.locseg, N.nodulid
FROM nodul N
WHERE N.pacient_ID <> 2;
```

-- not in (patients from which we DO NOT want information)

-- in (patients from which we ONLY want information)

```
SELECT N.locseg, N.nodulid
FROM nodul N
WHERE N.pacient_ID not in (1, 2, 3);
```

-- 2. Resolution of the Toshiba devices for CT studies. Return device name and resolution.

-- Not device is only named 'Toshiba'

```
SELECT C.aparell, C.resolucioX, C.resolucioY, C.resolucioZ
FROM ctestudi C
WHERE C.aparell = 'Toshiba';
```

-- Devices in 'Aparell' field have a larger name than 'Toshiba'. % is placed for 'more info' we do not know

```
SELECT C.aparell, C.resolucioX, C.resolucioY, C.resolucioZ
FROM ctestudi C
WHERE C.aparell LIKE 'Toshiba%';
```

-- Displays types of 'Aparell' in the column

```
SELECT DISTINCT C.aparell
FROM ctestudi C;
```

-- 3. CT studies with planning done between 01/01/2016 and 01/03/2016. Return CTData, serie label and device field " aparell

-- TO\_DATE(): convert character to data format

```
SELECT C.ctdata, C.seriellabel, C.aparell
FROM ctestudi C
WHERE C.ctdata BETWEEN TO_DATE('01/01/2016', 'DD/MM/YYYY') AND TO_DATE('01/03/2016', 'DD/MM/YYYY');
```

-- 4. Patients with nodules. Return pacient\_ID (use the distinct command to not duplicate patients).

```
SELECT DISTINCT N.pacient_ID
FROM nodul N;
```

-- 5. How old are the patients? Return Pacient\_ID, years and born date

-- months\_between(): returns the number of months difference

-- round(): converts result to integer

```
SELECT P.pacient_ID, P.data_naixement
FROM pacient P;
```

-- Years rounded

```
SELECT P.pacient_ID, P.data_naixement, round(months_between(SYSDATE, P.data_naixement)/12) as
EDAT
FROM pacient P;
```

-- Years not rounded

```
SELECT P.pacient_ID, P.data_naixement, months_between(SYSDATE, P.data_naixement)/12 as EDAT
FROM pacient P;
```

-- 6. Year of each video bronchoscopy exploration. Return name of the file and year.

```
SELECT B.videoarxiu, B.videodata  
FROM broncvideo B;
```

-- We only get the year of the complete date DD/MM/YYYY

```
SELECT B.videoarxiu, extract(year from B.videodata)  
FROM broncvideo B;
```

-- 7. Planned distance (field "DistanciaCarinaPlan") and real distance (field "DistanciaCarinaReal") for the biopsied nodules in 09/02/2016. Return Pacient\_ID, nodulID, DistanciaCarinaPlan and DistanciaCarinaReal.

```
SELECT N.pacient_id, N.nodulid, P.distanciacarinaplan, R.distanciacarinareal  
FROM biopsiplan P, biopsiresult R, broncvideo B, nodul N  
WHERE B.videodata = TO_DATE('09/02/2016', 'DD/MM/YYYY');
```

-- 8. Years of the youngest patient? Return years

```
SELECT MIN(ROUND(MONTHS_BETWEEN(SYSDATE, P.data_naixement)/12)) as EDAT  
FROM pacient P;
```

-- 9. Minimum size (field "diam\_menor") for the biopsied nodules with conclusive diagnosis (diagnosis not null)

```
SELECT C.diam_menor  
FROM ctnodul C, nodul N  
WHERE C.pacient_id = N.pacient_id AND N.diagnostic != 'NULL';
```

-- 10. How many navigation systems (field "navegador") are there?

```
SELECT COUNT(DISTINCT B.navegador)  
FROM biopsiplan B;
```

-- 11. Number of nodules at more than 100 mm from the carina (distanciaCarinaReal).

```
SELECT COUNT(B.nodulid)  
FROM biopsiresult B  
WHERE B.distanciacarinareal > 100;
```

-- 12. Average weight of ex-smokers that have a planned exploration (table BiopsiPlan).

```
SELECT AVG(P.pes)  
FROM pacient P, biopsiplan BP  
WHERE P.tabac = 'EX'  
AND P.pacient_id = BP.pacient_id;
```

-- 13. Total weight of ex-smokers that have a planned exploration (table BiopsiPlan).

```
SELECT SUM(P.pes)  
FROM pacient P, biopsiplan BP  
WHERE P.tabac = 'EX'  
AND P.pacient_id = BP.pacient_id;
```

-- 14. Number of nodules biopsied (CTNodul) in February 2016 (CTEstudi) for ex smoker patients (Pacient)

```
SELECT COUNT(*)  
FROM nodul N, pacient P, ctestudi CE, ctnodul CN  
WHERE CE.ct_id = CN.ct_id  
AND CN.nodulid = N.nodulid  
AND CN.pacient_id = N.pacient_id  
AND P.pacient_id = N.pacient_id  
AND P.tabac = 'EX'  
AND TO_CHAR(CE.ctdata, 'MM/YYYY') = '02/2016';
```

-- 15. How many nodules share diagnostic obtained with the following sampling instruments: basdiagnostic, raspalextdiagnostic, raspalldiagnostic.

```
SELECT COUNT(*)
FROM biopsireult BR
WHERE BR.basdiagnostic != 'NULL'
      AND BR.raspalextdiagnostic != 'NULL'
      AND BR.raspalldiagnostic != 'NULL';
```

-- 16. How many CT scanners systems (field "aparell") are there?

```
SELECT COUNT(DISTINCT aparell) as CTSCANNERS
FROM ctestudi;
```

-- 17. Maximum size (field "diam\_major" in table CTNodul) for the nodules that have a planned exploration. Return maximum size.

```
SELECT MAX(CN.diam_major) as MAXDIAM
FROM ctnodul CN, biopsiplan BP
WHERE CN.nodulid = BP.nodulid
      AND CN.ct_id = BP.ct_id
      AND CN.pacient_id = BP.pacient_id;
```

-- 18. Number of videobronchoscopies per date. Return date and number of biopsies.

```
SELECT TO_CHAR(videodata, 'DD/MM/YYYY'), COUNT(*)
FROM broncovidéo
GROUP BY TO_CHAR(videodata, 'DD/MM/YYYY')
ORDER BY 1, 2;
```

-- 19. Number of CT studies per date. Return date and number of studies.

```
SELECT TO_CHAR(ctdata, 'DD/MM/YYYY'), COUNT(*)
FROM ctestudi
GROUP BY TO_CHAR(ctdata, 'DD/MM/YYYY')
ORDER BY 1, 2;
```

-- 20. Number of males and females. Return sex ("home", "dona") and number of patients.

```
SELECT sexo, COUNT(*)
FROM pacient
GROUP BY sexo
ORDER BY 1, 2;
```

-- 21. Number of patients for each age. Return age in years and number of patients.

-- CEIL() returns the smallest integer value that is bigger than or equal to a number. Often used with aggregate functions (COUNT(), MAX(), MIN(), SUM(), AVG()) to group the result-set by one or more columns

-- GROUP BY() groups rows that have the same values into summary rows, like "find the number of customers in each country"

```
SELECT ROUND(CEIL(((SYSDATE - P.data_naixement)/365))) as AGE, COUNT(*)
FROM pacient P
GROUP BY ROUND(CEIL(((SYSDATE - P.data_naixement)/365)))
ORDER BY 1, 2;
```

-- 22. Average distance (distanciaccarinareal) for each patient. Return pacient\_ID and average distance.

```
SELECT P.pacient_id, AVG(BR.distanciaccarinareal)
FROM pacient P, biopsireult BR
WHERE P.pacient_id = BR.pacient_id
GROUP BY P.pacient_id
ORDER BY 1, 2;
```

-- 23. Number of CT studies for each device (field "aparell" in table CTEstudi) for those nodules that have been biopsied with non-conclusive (NULL) diagnosis. Return device and number of studies.

```
SELECT CE.aparell, COUNT(*)
```

```

FROM ctestudi CE, biopsiresult BR, nodul N
WHERE BR.pacient_id = N.pacient_id
      AND BR.nodulid = N.nodulid
      AND BR.ct_id = CE.ct_id
      AND N.diagnostic is NULL
GROUP BY CE.aparell
ORDER BY 1, 2;

```

-- 24. Number of patients with conclusive diagnosis (Nodul) according to its smoking condition. Return smoking condition (value of field "tabac") and number of patients.

```

SELECT P.tabac, COUNT(*)
FROM pacient P, nodul N
WHERE P.pacient_id = N.pacient_id
      AND N.diagnostic != 'NULL'
GROUP BY P.tabac
ORDER BY 1, 2;

```

-- 25. Number of planned nodes for each navigation system. Return navigation system and number of nodes.

```

SELECT navegador, COUNT(*)
FROM biopsiplan
GROUP BY navegador
ORDER BY 1, 2;

```

-- 26. Minimum nodule size (diam\_menor) (CTNodul) for each CT device during July of 2015 (CTData, CTESTUDI). Return device (CTEstudi) and minimum nodule size.

```

SELECT CE.aparell, MIN(CN.diam_menor)
FROM ctestudi CE, ctnodul CN
WHERE CE.ct_id = CN.ct_id
      AND TO_CHAR(CE.ctdata, 'MM/YYYY') = '07/2015'
GROUP BY CE.aparell
ORDER BY 1, 2;

```

-- 27. Segmental locations (field "locseg") with more than 2 nodules. Return segmental location.

```

SELECT DISTINCT N.locseg
FROM nodul N
GROUP BY N.locseg
HAVING COUNT(*) > 2 -- appears > 2 in locseg column
ORDER BY 1;

```

-- 28. Bronchoscopes that have biopsied with exit (diagnostic is not null) more than 1 nodule. Return the bronchoscope (field "broncoscopi").

```

SELECT BV.broncoscopi
FROM nodul N, biopsiresult BR, broncovideo BV
WHERE N.diagnostic != 'NULL'
      AND BR.pacient_id = N.pacient_id
      AND BR.nodulid = N.nodulid
      AND BR.video_arxiu = BV.videoarxiu
GROUP BY BV.broncoscopi
HAVING COUNT(*) > 1
ORDER BY 1;

```

-- 29. Segmental location with more than 1 nodule (>1) with diameter < 40 mm (fie Id "diam\_menor") that have been biopsied. Return segmental location

```

SELECT N.locseg
FROM nodul N, ctnodul CN, biopsiresult BR
WHERE N.pacient_id = CN.pacient_id
      AND N.nodulid = CN.nodulid
      AND BR.pacient_id = N.pacient_id
      AND BR.nodulid = N.nodulid
      AND CN.diam_menor < 40

```

```
GROUP BY N.locseg
HAVING COUNT (N.locseg) > 1
ORDER BY 1;
```

```
-- 30. List of patients with more than one nodule (>1). Return patient ID (field "pacient_ID").
SELECT P.pacient_id
FROM pacient P, nodul N
WHERE P.pacient_id = N.pacient_id
GROUP BY P.pacient_id
HAVING COUNT(N.nodulid) > 1
ORDER BY 1;
```

```
-- 31. List of patients with more than one CT study (>1). Return patient ID (field "pacient_ID") and number of
CTs.
SELECT pacient_id, COUNT(ct_id)
FROM ctnodul
GROUP BY pacient_id
HAVING COUNT(*)>1
ORDER BY 1, 2;
```

```
-- 32. Navigation systems with more than 1 biopsy with conclusive diagnosis. Return navigation system (field
"navegador") and number of biopsies.
SELECT BP.navegador, COUNT(*)
FROM biopsiplan BP, nodul N
WHERE N.diagnostic != 'NULL'
      AND N.nodulid = BP.nodulid
      AND N.pacient_id = BP.pacient_id
GROUP BY BP.navegador
HAVING COUNT(*) > 1
ORDER BY 1, 2;
```

```
-- 33. Bronchoscopes with more than 2 biopsies with conclusive diagnosis where the nodule was at <120mm
of distance (field "distanciaCarinaReal"). Return bronchoscope and number of biopsies.
SELECT BV.broncoscopi, COUNT(*)
FROM broncovidéo BV, nodul N, biopsiresult BR
WHERE N.diagnostic != 'NULL'
      AND N.pacient_id = BR.pacient_id
      AND N.nodulid = BR.nodulid
      AND BR.video_arxiu = BV.videoarxiu
      AND BR.distanciacarinareal < 120
GROUP BY BV.broncoscopi
HAVING COUNT(*) > 2
ORDER BY 1, 2;
```

```
-- 34. The youngest patient. Return patient ID
SELECT P.pacient_id
FROM pacient P
WHERE (SYSDATE-P.data_naixement)/365 = (SELECT MIN(SYSDATE-P.data_naixement)/365 FROM
pacient P)
ORDER BY 1;
```

```
-- 35. Heaviest patient. Return patient ID.
SELECT pacient_id
FROM pacient P
WHERE P.pes = (SELECT MAX(P.pes) FROM pacient P)
ORDER BY 1;
```

```
-- 36. Segmental location (nodulid) and final diagnostic (nodulid) for the smallest nodule (field "diam_major")
(ctnodul). Return segmental location and diagnostic.
```

```
SELECT MIN(CN.diam_major)
FROM ctnodul CN;
```

```
SELECT N.locseg, N.diagnostic
FROM nodul N, ctnodul CN
WHERE CN.diam_major = (SELECT MIN(CN.diam_major) FROM ctnodul CN)
  AND N.pacient_id = CN.pacient_id
  AND N.nodulid = CN.nodulid
ORDER BY 1, 2;
```

-- 37. Minimum nodule sizes (field "diam\_menor") (ctnodul) for nodules at maximal distance (field "distanciaccarinareal") (biopsiresult) with conclusive diagnosis (nodul: diagnostic is not null). Return size (field "diam\_menor").

-- Huge subquery is equal to max(distanciaccarinareal)

```
SELECT CN.diam_menor
FROM nodul N, ctnodul CN, biopsiresult BR
WHERE N.diagnostic != 'null'
  AND BR.ct_id = CN.ct_id
  AND BR.pacient_id = CN.pacient_id
  AND CN.pacient_id = N.pacient_id
  AND BR.nodulid = CN.nodulid
  AND CN.nodulid = N.nodulid
  AND BR.distanciaccarinareal = (SELECT MAX(BR.distanciaccarinareal)
                                FROM biopsiresult BR, nodul N, ctnodul CN
                                WHERE N.diagnostic != 'null'
                                  AND BR.ct_id = CN.ct_id
                                  AND BR.pacient_id = CN.pacient_id
                                  AND CN.pacient_id = N.pacient_id
                                  AND BR.nodulid = CN.nodulid
                                  AND CN.nodulid = N.nodulid)
```

```
ORDER BY 1;
```

-- 38. Oldest patient. Return patient ID.

```
SELECT P.pacient_id
FROM patient P
WHERE ((SYSDATE-P.data_naixement)/365) = (SELECT MAX((SYSDATE-P.data_naixement)/365) FROM
patient P)
ORDER BY 1;
```

-- 39. Patients with maximal discrepancy between the planned distance (field "distanciaccarinaplan", biopsiplan) and true distance (field "distanciaccarinareal" in table biopsiresult). Return the patient ID.

```
SELECT BP.pacient_id
FROM biopsiresult BR, biopsiplan BP
WHERE BR.cami_id = BP.cami_id
  AND BR.ct_id = BP.ct_id
  AND BR.nodulid = BP.nodulid
  AND BR.pacient_id = BP.pacient_id
  AND (BP.distanciaccarinaplan - BR.distanciaccarinareal) = (SELECT MAX(BP.distanciaccarinaplan -
```

```
BR.distanciaccarinareal)
FROM biopsiresult BR, biopsiplan BP
WHERE BR.cami_id = BP.cami_id
  AND BR.ct_id = BP.ct_id
  AND BR.nodulid = BP.nodulid
  AND BR.pacient_id = BP.pacient_id)
```

```
ORDER BY 1;
```

-- 40. Patient with the highest number of nodules. Return Patient ID.

```
SELECT MAX(COUNT(*))
FROM nodul
GROUP BY patient_id;
```

```
SELECT N.pacient_id
```

```

FROM nodul N
GROUP BY N.pacient_id
HAVING COUNT(*) = (SELECT MAX(COUNT(*))
                    FROM nodul
                    GROUP BY pacient_id)
ORDER BY 1;

```

-- 41. Broncoscope with the highest number of biopsied nodules with conclusive diagnosis (field "diagnostic" is not null). Return broncoscope (field "broncoscopi") and number of biopsies.

-- Highest number of biopsied nodules with conclusive diagnosis

```

SELECT MAX(COUNT(*))
FROM biopsiresult BR, nodul N, broncvideo BV
WHERE N.diagnostic != 'NULL'
    AND N.pacient_id = BR.pacient_id
    AND N.nodulid = BR.nodulid
    AND BR.video_arxiu = BV.videoarxiu
GROUP BY BV.broncoscopi;

```

-- Broncoscope with the highest number of biopsied nodules with conclusive diagnosis

```

SELECT BV.broncoscopi, COUNT(*)
FROM broncvideo BV, biopsiresult BR, nodul N
WHERE N.diagnostic != 'NULL'
    AND N.pacient_id = BR.pacient_id
    AND N.nodulid = BR.nodulid
    AND BR.video_arxiu = BV.videoarxiu
GROUP BY BV.broncoscopi
HAVING COUNT(*) = (SELECT MAX(COUNT(*))
                    FROM biopsiresult BR, nodul N, broncvideo BV
                    WHERE N.diagnostic != 'NULL'
                        AND N.pacient_id = BR.pacient_id
                        AND N.nodulid = BR.nodulid
                        AND BR.video_arxiu = BV.videoarxiu
                    GROUP BY BV.broncoscopi)
ORDER BY 1, 2;

```

-- 42. Segmental location with the highest number of biopsied nodules with conclusive diagnosis (field "diagnostic" is not null). Return segmental location (field "locseg") and number of biopsies

-- Highest number of biopsied nodules with conclusive diagnosis

```

SELECT MAX(COUNT(*))
FROM biopsiresult BR, nodul N
WHERE N.diagnostic != 'NULL'
    AND N.pacient_id = BR.pacient_id
    AND N.nodulid = BR.nodulid
GROUP BY N.locseg;

```

-- Segmental location with the highest number of biopsied nodules with conclusive diagnosis

```

SELECT N.locseg, COUNT(*)
FROM biopsiresult BR, nodul N
WHERE N.diagnostic != 'NULL'
    AND N.pacient_id = BR.pacient_id
    AND N.nodulid = BR.nodulid
GROUP BY N.locseg
HAVING COUNT(*) = (SELECT MAX(COUNT(*))
                    FROM biopsiresult BR, nodul N
                    WHERE N.diagnostic != 'NULL'
                        AND N.pacient_id = BR.pacient_id
                        AND N.nodulid = BR.nodulid
                    GROUP BY N.locseg)
ORDER BY 1, 2;

```

-- 43. CT device with the least number of scans done in 2015. Return CT device (field "aparell")

```

SELECT aparell
FROM ctestudi

```

```

WHERE TO_CHAR(ctdata, 'YYYY') = '2015'
GROUP BY aparell
HAVING COUNT(*) = (SELECT MIN(COUNT(*))
                    FROM ctestudi
                    WHERE TO_CHAR(ctdata, 'YYYY') = '2015'
                    GROUP BY aparell)
ORDER BY 1;

```

-- 44. Most frequent (HAVING COUNT) nodule type. Return type of the nodule (field "tipus").  
 -- COUNT() function returns the number of rows that matches a specified criterion  
 -- HAVING COUNT() to count frequencies

```

SELECT N.tipus
FROM nodul N
GROUP BY N.tipus
HAVING COUNT(*) = (SELECT MAX(COUNT(*))
                  FROM nodul N
                  GROUP BY N.tipus)
ORDER BY 1;

```

-- 45. Patients with the least number of successfully biopsied nodules (conclusive diagnosis) Return patient ID and number of successfully biopsied nodules

```

SELECT MIN(COUNT(*))
FROM nodul N, biopsiresult BR
WHERE N.diagnostic != 'NULL'
      AND N.pacient_id = BR.pacient_id
      AND N.nodulid = BR.nodulid
GROUP BY N.pacient_id;

```

```

SELECT N.pacient_id, COUNT(*)
FROM nodul N, biopsiresult BR
WHERE N.diagnostic != 'NULL'
      AND N.pacient_id = BR.pacient_id
      AND N.nodulid = BR.nodulid
GROUP BY N.pacient_id
HAVING COUNT(*) = (SELECT MIN(COUNT(*))
                  FROM nodul N, biopsiresult BR
                  WHERE N.diagnostic != 'NULL'
                        AND N.pacient_id = BR.pacient_id
                        AND N.nodulid = BR.nodulid
                  GROUP BY N.pacient_id)
ORDER BY 1, 2;

```

-- 46. FileNames (field "videoarxiu") for bronchoscopies without biopsy. Return filenames.

```

SELECT videoarxiu
FROM broncvideo
MINUS
SELECT video_arxiu
FROM biopsiresult
ORDER BY 1;

```

-- 47. Nodules that have CT but do not have a biopsy. Return the nodul ID.

```

SELECT nodulid
FROM ctnodul
MINUS
SELECT nodulid
FROM biopsiresult
ORDER BY 1;

```

-- 48. CT Devices that do not have a nodule associated with diameter equal to 4mm (field "diam\_menor"). Return device (field "aparell").

```

SELECT aparell

```



```

FROM ctestudi
MINUS
SELECT aparell
FROM ctestudi CE, ctnodul CN
WHERE CE.ct_id = CN.ct_id
      AND CN.diam_menor = 4
ORDER BY 1;

```

-- 49. Nodules for the segmental location "LSE3" that do not have a biopsy, yet. Return patient ID and nodule ID.

```

SELECT pacient_id, nodulid
FROM nodul
WHERE locseg = 'LSE3'
MINUS
SELECT pacient_id, nodulid
FROM biopsiresult
ORDER BY 1, 2;

```

-- !!! 50. Average of number of nodules per patient according to smoking condition (numerator is number of nodules per smoking condition, divisor is number of patients per smoking condition)

```

SELECT tabac, COUNT(*)
FROM pacient
GROUP BY tabac;

SELECT T1.tabac, AVG((T1.total_tabac/T2.total) * 100)
FROM (SELECT P.tabac, COUNT(*) as total_tabac
      FROM pacient P, nodul N
      WHERE P.pacient_id = N.pacient_id
      GROUP BY P.tabac) T1,
      (SELECT P.tabac, COUNT(*) as total
      FROM pacient P
      GROUP BY P.tabac) T2
GROUP BY T1.tabac;

```

-- 54. Percentage of males

```

SELECT (100 * COUNT(*))/(SELECT COUNT(*) FROM pacient)
FROM pacient P
WHERE P.sexo = 'HOME'
GROUP BY P.sexo
ORDER BY 1;

```

```

SELECT (T1.quantity_males/T2.total)*100
FROM (SELECT COUNT(*) as quantity_males FROM pacient WHERE sexo = 'HOME') T1, (SELECT
COUNT(*) as total FROM pacient) T2;

```