CICLO: DESARROLLO APLICACIONES WEB-DISTANCIA

**MÓDULO: BASE DE DATOS** 

**TAREA: BD04** 

**ALUMNO: DAVID MEDINA GARCIA** 

\*La tabla que nos diste no detectaba las tildes, asi que la he modificado para que recoja las tildes. La adjunto en el archivo .rar

#### **ENUNCIADO 1:**

Listar por las ventas de medicamentos con receta realizadas por los médicos de cada centro de salud. Las columnas que visualizaremos son: población del médico, centro de salud del médico, nombre medicamento, nombre laboratorio, fecha venta, unidades, precio unitario, total venta (unidades\*precio unitario). Sólo listaremos aquellos medicamentos cuyas ventas se hayan realizado entre las fecha: 01/10/2021 y 31/12/2021, y que el nombre de medicamento contenga las palabras: medicamento antibiótico en cualquier lugar. Se ordenará por población, dentro de ésta por centro salud, por nombre medicamento y por fecha venta.

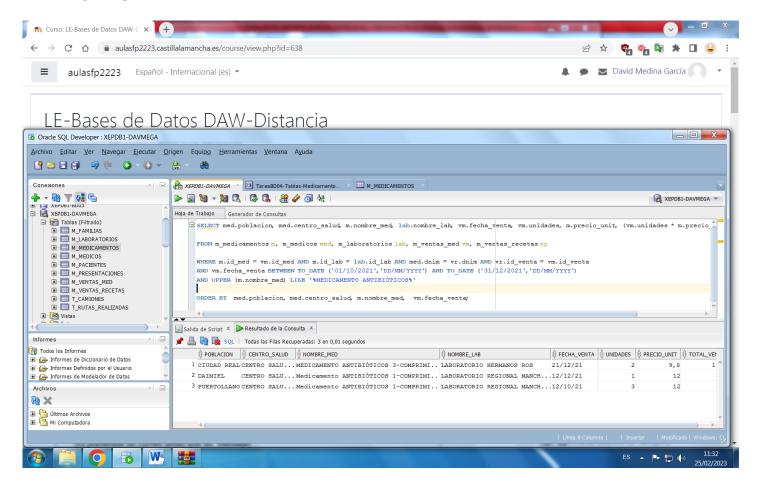
#### **SENTENCIA SQL:**

SELECT med.poblacion, med.centro\_salud, m.nombre\_med, lab.nombre\_lab, vm.fecha\_venta, vm.unidades, m.precio\_unit, (vm.unidades \* m.precio\_unit) as TOTAL VENTA

FROM m\_medicamentos m, m\_medicos med, m\_laboratorios lab, m\_ventas\_med vm, m\_ventas\_recetas vr

WHERE m.id\_med = vm.id\_med AND m.id\_lab = lab.id\_lab AND med.dnim = vr.dnim AND vr.id\_venta = vm.id\_venta AND vm.fecha\_venta BETWEEN TO\_DATE ('01/10/2021', 'DD/MM/YYYY') AND TO\_DATE ('31/12/2021', 'DD/MM/YYYY') AND UPPER (m.nombre\_med) LIKE '%MEDICAMENTO ANTIBIÓTICO%'

ORDER BY med.poblacion, med.centro\_salud, m.nombre\_med, vm.fecha\_venta;



#### **ENUNCIADO 2:**

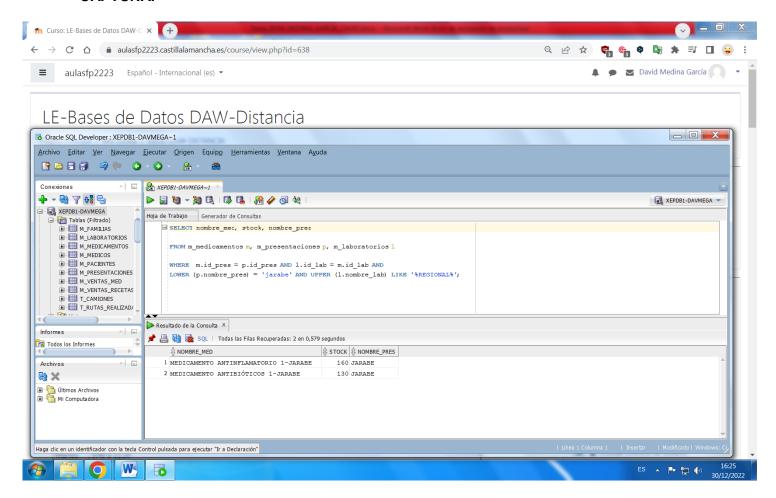
**VERSIÓN 1:** Listado del nombre de los medicamentos y las cantidades en stock de aquellos medicamentos cuya presentación sea jarabe y el nombre de su laboratorio contenga la palabra regional.

#### **SENTENCIA SQL:**

SELECT m.nombre\_med, m.stock, p.nombre\_pres

FROM m\_medicamentos m, m\_presentaciones p, m\_laboratorios l

WHERE m.id\_pres = p.id\_pres AND l.id\_lab = m.id\_lab AND LOWER (p.nombre\_pres) = 'jarabe' AND UPPER (l.nombre\_lab) LIKE '%REGIONAL%';



## **ENUNCIADO 2:**

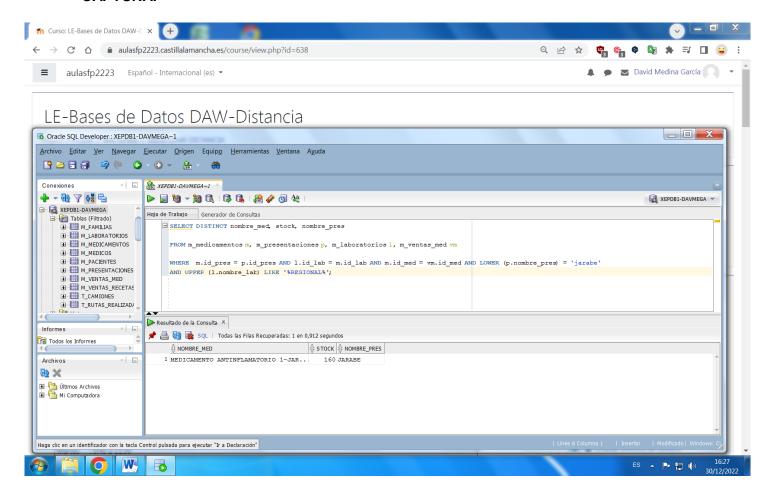
VERSIÓN 2: Sólo saldrán los medicamentos de los que hayan realizado más de una venta.

## **SENTENCIA SQL:**

SELECT m.nombre\_med, m.stock, p.nombre\_pres

FROM m\_medicamentos m JOIN m\_presentaciones p ON m.id\_pres = p.id\_pres JOIN m\_laboratorios I ON l.id\_lab = m.id\_lab

WHERE LOWER (p.nombre\_pres) = 'jarabe' AND UPPER (l.nombre\_lab) LIKE '%REGIONAL%' AND m.id\_med IN (SELECT vm.id\_med FROM m\_ventas\_med vm GROUP BY vm.id\_med HAVING COUNT (vm.id\_med) >1);



# **ENUNCIADO 3:**

**VERSIÓN 1**: Se quiere visualizar el nombre de cada familia, el número de medicamentos vendidos y el total de las ventas (unidades \* precio unitario) de esa familia. Ordenado por nombre de la familia

## **SENTENCIA SQL:**

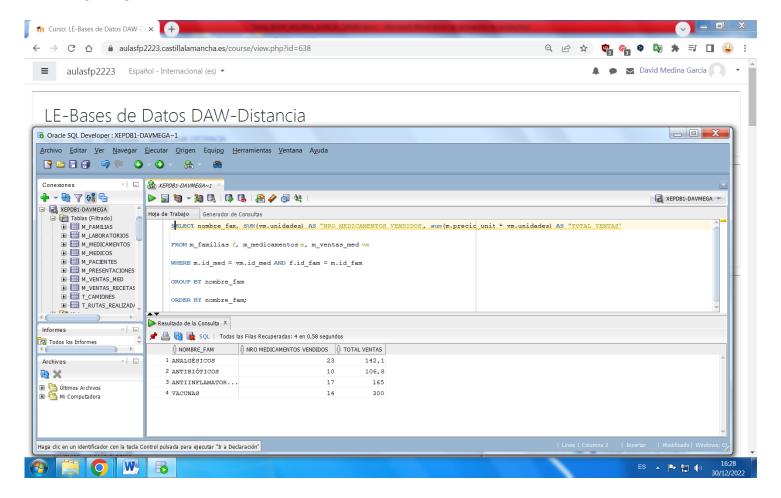
SELECT nombre\_fam, SUM (vm.unidades) AS "NRO MEDICAMENTOS VENDIDOS", sum (m.precio\_unit \* vm.unidades) AS "TOTAL VENTAS"

FROM m\_familias f, m\_medicamentos m, m\_ventas\_med vm

WHERE m.id med = vm.id med AND f.id fam = m.id fam

**GROUP BY nombre\_fam** 

**ORDER BY nombre\_fam;** 



#### **ENUNCIADO 3:**

**VERSIÓN 2**: Que sólo salgan las familias en el que el nº total de medicamentos vendidos sea mayor de 15.

#### **SENTENCIA SQL:**

SELECT nombre\_fam, SUM (vm.unidades) AS "NRO MEDICAMENTOS VENDIDOS", sum(m.precio unit \* vm.unidades) AS "TOTAL VENTAS"

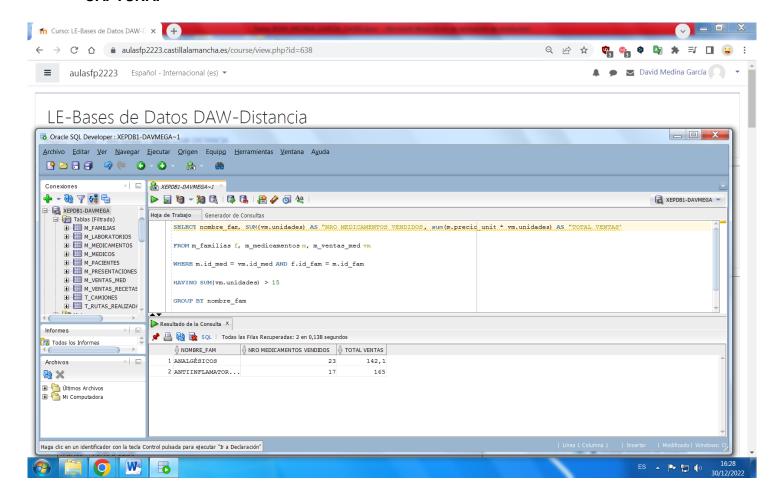
FROM m\_familias f, m\_medicamentos m, m\_ventas\_med vm

WHERE m.id\_med = vm.id\_med AND f.id\_fam = m.id\_fam

**HAVING SUM (vm.unidades) > 15** 

**GROUP BY nombre\_fam** 

**ORDER BY nombre\_fam;** 



## **ENUNCIADO 3:**

**VERSIÓN 3:** Que sólo salgan las familias en las que la media de sus ventas sean mayor que la media de todas las ventas de todos los medicamentos.

## **SENTENCIA SQL:**

```
SELECT nombre_fam, SUM (unidades) as "NRO MEDICAMENTOS VENDIDOS", SUM (m.precio_unit * vm.unidades) AS "TOTAL VENTAS"
```

FROM m\_familias f, m\_medicamentos m,m\_ventas\_med vm

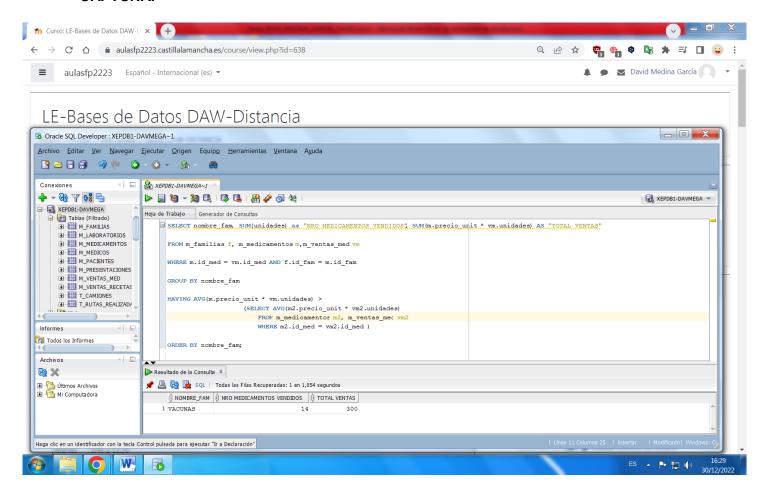
WHERE m.id\_med = vm.id\_med AND f.id\_fam = m.id\_fam

**GROUP BY nombre\_fam** 

HAVING AVG (m.precio\_unit \* vm.unidades) >

(SELECT AVG (m2.precio\_unit \* vm2.unidades)
FROM m\_medicamentos m2, m\_ventas\_med vm2
WHERE m2.id\_med = vm2.id\_med)

**ORDER BY nombre fam;** 



#### **ENUNCIADO 4:**

Visualizar de cada familia: nombre de la familia, nombre del medicamento dentro de cada familia del que haya menor número de unidades en stock y el stock. Ordenado por el nombre de la familia.

## **SENTENCIA SQL:**

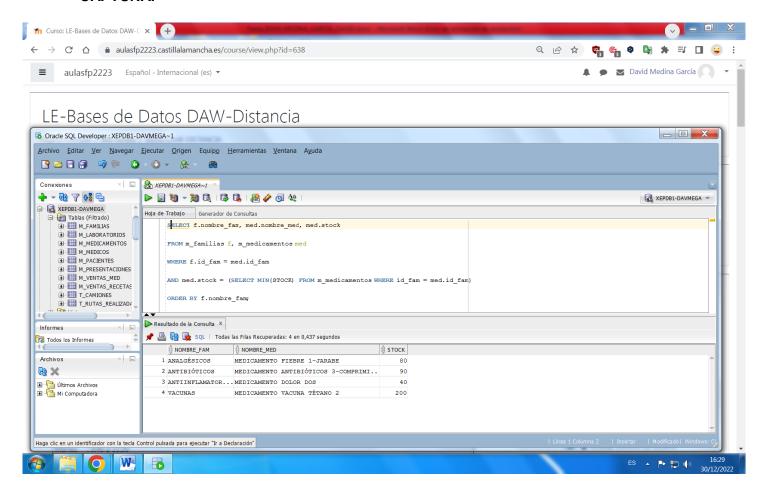
SELECT f.nombre\_fam, med.nombre\_med, med.stock

FROM m familias f, m medicamentos med

WHERE f.id\_fam = med.id\_fam

AND med.stock IN (SELECT MIN (STOCK) FROM m\_medicamentos WHERE id\_fam = f.id\_fam)

**ORDER BY f.nombre\_fam;** 



#### **ENUNCIADO 5:**

Se listará el apellido y nombre del paciente, población, la suma de las unidades vendidas de medicamentos, total de sus ventas (unidades \* precio venta) y fecha de la última venta. Sólo se tendrán en cuenta aquellos medicamentos que se hayan vendido más de 2 veces.

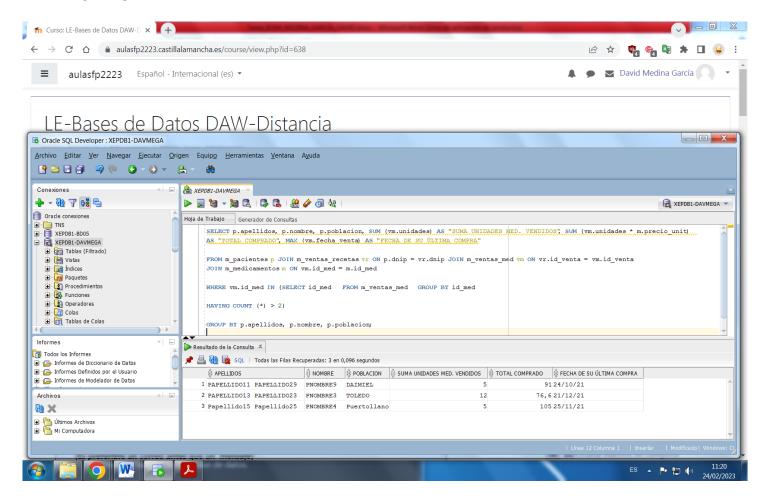
# **SENTENCIA SQL:**

SELECT p.apellidos, p.nombre, p.poblacion, SUM (vm.unidades) AS "SUMA UNIDADES MED. VENDIDOS", SUM (vm.unidades \* m.precio\_unit) AS "TOTAL COMPRADO", MAX (vm.fecha\_venta) AS "FECHA DE SU ÚLTIMA COMPRA"

FROM m\_pacientes p JOIN m\_ventas\_recetas vr ON p.dnip = vr.dnip JOIN m\_ventas\_med vm ON vr.id\_venta = vm.id\_venta JOIN m\_medicamentos m ON vm.id\_med = m.id\_med

WHERE vm.id\_med IN (SELECT id\_med FROM m\_ventas\_med GROUP BY id\_med HAVING COUNT (\*) > 2)

# **GROUP BY p.apellidos, p.nombre, p.poblacion;** CAPTURA:



#### **ENUNCIADO 6:**

Realiza un listado que nos indique el nº de productos que nos vende cada laboratorio. Saldrán todos los laboratorios, aunque no nos venda ninguno.

## **SENTENCIA SQL:**

SELECT 'DESDE ' || I.poblacion || ' EL ' ||I.nombre\_lab||' NOS VENDE ' || COUNT (m. id\_med) ||' MEDICAMENTOS.' AS "LABORATORIOS"

FROM m laboratorios I LEFT JOIN m medicamentos m ON l.id lab = m.id lab

**GROUP BY I.poblacion, I.nombre lab;** 

