

Relación 4 - Álgebra Relacional

Problema 1. Tenemos el siguiente esquema relacional de base de datos:

CLIENTE(Num-Cliente, Nombre, Dirección, Teléfono, Población)

PRODUCTO(Cod-Producto, Descripción, Precio)

VENTA(Cod-Producto, NumCliente, Cantidad, Id-Venta)

La tabla **CLIENTE** almacena información sobre cada posible cliente de nuestra empresa. En la tabla **PRODUCTO** se guarda información de los productos almacenados y la tabla **VENTA** relaciona las dos anteriores utilizando el atributo **Cod-Producto** para indicar el producto que se vende, y el atributo **NumCliente** para indicar el cliente al que vendimos el producto.

1. Realizar una consulta que muestre el nombre de los clientes de Palencia.
2. Indicar el código y la descripción de los productos cuyo código coincida con su descripción
3. Obtener el nombre de los clientes junto con el identificador de venta y la cantidad vendida, de aquellas operaciones en las que se vendieron más de 500 unidades.
4. Nombre de los clientes de la tabla **CLIENTE** que no aparecen en la tabla de ventas (clientes que no han comprado nada).
5. Nombre de los clientes que han comprado todos los productos de la empresa.
6. Identificador de las ventas cuya cantidad supera a la cantidad vendida en la venta número 18.
7. Productos que nunca ha comprado un cliente de Palencia.
8. Códigos de los productos que se han vendido tanto en Palencia como en Valladolid (a clientes que vivan allí).
9. Poblaciones donde hemos vendido todos nuestros productos.

Imaginemos ahora que añadimos la tabla de facturas que se relaciona con la de ventas, de modo que a la tabla de ventas le añadimos el número de factura con la que se relaciona. En la tabla **FACTURA** indicamos la fecha, el número y si se pagó o no. Cada factura se corresponde con varias ventas y con un solo cliente, para lo cual se varía el diseño de **VENTA**¹:

FACTURA(Num. Factura, Fecha, Pagada, Num. Cliente)

VENTA(Cod. Producto, Num. Factura, Cantidad, Id. Venta)

10. Obtener el nombre de los clientes que tienen alguna factura sin pagar.
11. Nombre de los clientes que han pagado todas sus facturas.

¹Nótese que estos apartados se pueden hacer a nivel teórico, pero que no hay datos relacionados con los mismos en el archivo **RelaX** proporcionado en la web

Problema 2. Consideremos el siguiente esquema:

SANITARIO(Id-Sanitario, Nombre, Profesión)

PACIENTE(Id-Paciente, Nombre, No-SS, Dirección, Teléfono)

CONSULTA(Id-Sanitario, Id-Paciente, Día, Mes, Año, Comentarios, Id-Consulta)

RECETA(Id-Consulta, Marca, Comentarios)

Los pacientes visitan al personal sanitario y se anota la visita anotando una fila en la tabla de consultas en la que se indica la fecha (separando el día, el mes y el año) el sanitario y el paciente relacionados con la consulta. Los pacientes son los que pertenecen al centro de salud, pero no tienen porque haber hecho ni una sola consulta. En cada consulta se pueden recetar una o más recetas. Con este esquema realizar las siguientes consultas

1. Mostrar el nombre y número de la seguridad social de los pacientes que aún no han hecho ni una sola consulta
2. Mostrar el nombre de los profesionales sanitarios que han tenido consulta con el paciente número 205
3. Pacientes (nombre e identificador) que no han tenido consulta con ningún enfermero o enfermera
4. Pacientes (nombre e identificador) que han tenido consulta con el personal sanitario número 189 y número 230
5. Pacientes (nombre e identificador) a los que nunca se les ha recetado nada
6. Identificador y nombre de los pacientes que han ido a consulta todos los meses en los que ha habido consultas.
7. Nombre e identificador de los pacientes que no han ido a consulta en todo el año 2015

Problema 3. Tenemos el siguiente esquema relacional de base de datos sobre proveedores, componentes, artículos y envíos:

PROVEEDOR(IDP, Nombre, Categoría, Ciudad)

COMPONENTE(IDC, Color, Peso, Ciudad)

ARTICULO(IDT, Nombre, Ciudad)

ENVIO(IDP, IDC, IDT, Cantidad)

1. Obtener el nombre de todos los artículos de Cáceres.
2. Obtener todos los valores de IDP para los proveedores que abastecen el artículo con identificador T1.
3. Obtener los valores de IDP para los proveedores que suministran para el artículo T1 el componente C1.
4. Obtener la lista de pares de atributos (COLOR, CIUDAD) de la tabla componentes eliminando los pares duplicados.
5. Obtener los valores TNOMBRE para los artículos abastecidos por el proveedor P1.

6. Obtener los valores de IDC para los componentes suministrados para cualquier artículo de MADRID.
7. Obtener todos los valores de IDC de los componentes tales que ningún otro componente tenga un valor de peso inferior.
8. Obtener los valores de IDP para los proveedores que suministren los artículos T1 y T2. Recuerda que Mysql no tiene el operador INTERSECT.
9. Obtener los valores de IDP para los proveedores que suministran para un artículo de SEVILLA o MADRID un componente ROJO.
10. Obtener, mediante subconsultas (SELECT anidados), los valores de IDC para los componentes suministrados para algún artículo de SEVILLA por un proveedor de SEVILLA.
11. Obtener los valores de IDT distintos para los artículos que usan al menos un componente que se puede obtener con el proveedor P1.
12. Obtener todas las ternas (CIUDAD1, IDC, CIUDAD2) tales que un proveedor de la CIUDAD1 suministre el componente especificado para un artículo montado en la CIUDAD2.
13. Repetir el ejercicio anterior pero sin recuperar las ternas en los que los dos valores de ciudad sean los mismos.
14. Obtener el número de suministros, el de artículos distintos suministrados y la cantidad total de artículos suministrados por el proveedor P2.
15. Para cada artículo y componente suministrado obtener los valores de IDC, IDT y la cantidad total correspondiente.
16. Obtener los valores de IDT de los artículos abastecidos al menos por un proveedor que no sea de MADRID y que no sea de la misma ciudad en la que se monta el artículo.
17. Obtener los valores de IDP para los proveedores que suministran al menos un componente suministrado al menos por un proveedor que suministra al menos un componente ROJO.
18. Obtener los identificadores de artículos IDT para los que se ha suministrado algún componente del que se haya suministrado una cantidad media superior a 500 artículos.
19. Seleccionar los identificadores de proveedores que hayan realizado algún envío con Cantidad mayor que la media de los envíos realizados para el componente a que corresponda dicho envío.
20. Seleccionar los identificadores de componentes suministrados para el artículo 'T2' por el proveedor 'P2'.
21. Seleccionar todos los datos de los envíos realizados de componentes cuyo color no sea 'ROJO'.

22. Seleccionar los identificadores de componentes que se suministren para los artículos 'T1' y 'T2'.
23. Seleccionar el identificador de proveedor y el número de envíos de componentes de color 'ROJO' llevados a cabo por cada proveedor.
24. Seleccionar los colores de componentes suministrados por el proveedor 'P5'.
25. Seleccionar los datos de envío y nombre de ciudad de aquellos envíos que cumplan que el artículo, proveedor y componente son de la misma ciudad.
26. Seleccionar los nombres de los componentes que son suministrados en una cantidad total superior a 500.
27. Seleccionar los identificadores de proveedores que residan en Sevilla y no suministren más de dos artículos distintos.
28. Seleccionar los identificadores de artículos para los cuales todos sus componentes se fabrican en una misma ciudad.
29. Seleccionar los identificadores de artículos para los que se provean envíos de todos los componentes existentes en la base de datos.
30. Seleccionar los códigos de proveedor y artículo que suministran al menos dos componentes de color 'ROJO'.

Problema 4. Tenemos el siguiente esquema relacional de base de datos sobre aficiones²:

PERSONAS(idPersona, persona)

HOBBIES(hobby)

AFICIONES(idPersona, hobby)

Se pide realizar las siguientes consultas:

1. Relación de hobbies.
2. Lista de nombres de personas.
3. Nombres de personas y sus aficiones.
4. Personas a las que les gusta el Tenis.
5. Personas a las que no les gusta ningún hobby.
6. Personas a las que gustan todos los hobbies (probar a realizarlo de dos formas distintas: con operadores derivados y solo con los básicos).

Problema 5. Se tiene una base de datos que gestiona la información sobre suministradores, piezas, proyectos y suministros. Las relaciones son las siguientes:

²Ejemplo tomado de una práctica del curso 2009/2010 del Departamento de Lenguajes y Sistemas Informáticos (LSI) de nuestra universidad (a quien agradecemos su preparación), que hasta el curso 2023/2024 estaba enlazada, pero que en su nueva web remodelada ya no se encuentra a disposición.

- La relación **S** representa los suministradores, cada uno con un código único (s) y los atributos nombre de proveedor (nom_s), estado ($estado$) y ciudad ($ciudad$).
- La relación **P** representa las piezas, con un código de pieza único (p), nombre de pieza (nom_p), color ($color$), peso ($peso$) y ciudad ($ciudad$).
- La relación **J** representa los proyectos, con un código de proyecto único (j), nombre del proyecto (nom_j) y ciudad ($ciudad$).
- La relación **SPJ** representa los suministros, es decir, la cantidad ($cant$) de cada pieza (p) que cada suministrador (s) suministra a cada proyecto (j). La clave primaria de la relación está formada por el código de suministradores, el código de pieza y el código de proyecto.

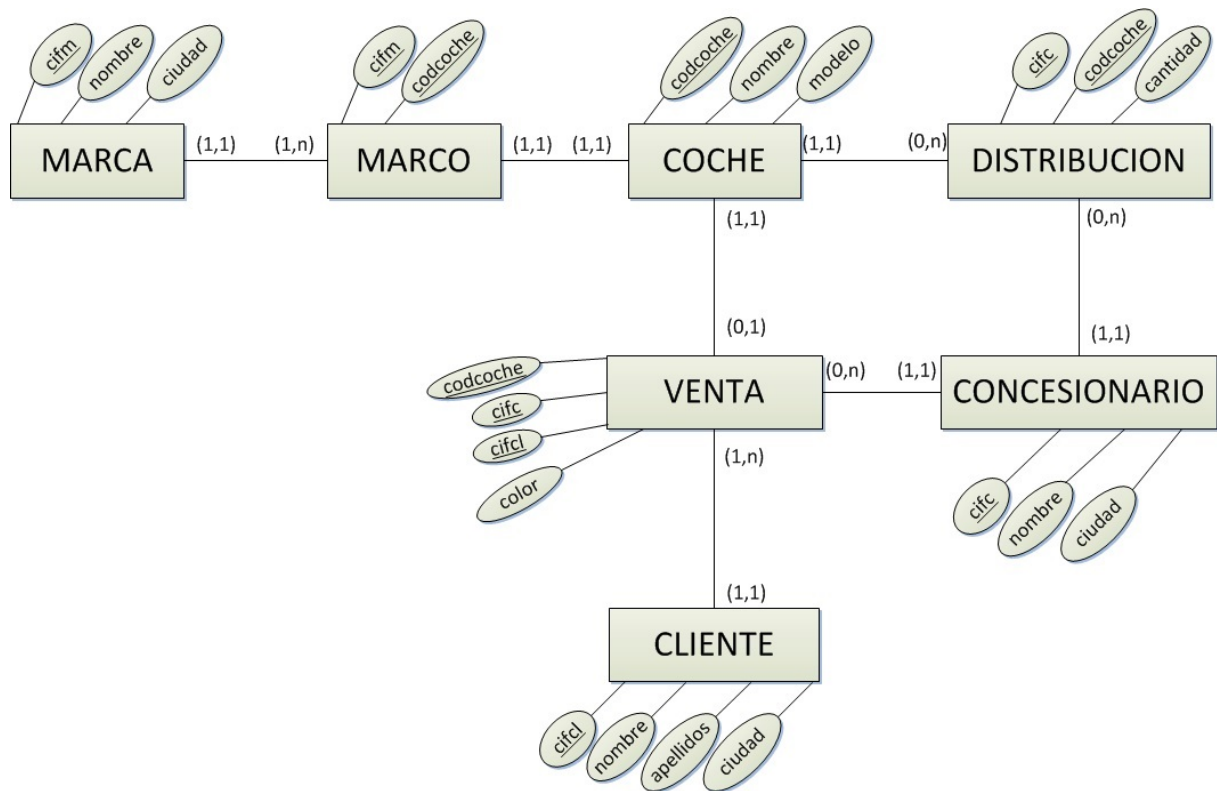
Se pide realizar las siguientes consultas:

1. Obtener todos los atributos de todos los proyectos.
2. Obtener todos los atributos de todos los proyectos en Londres.
3. Obtener los códigos de los suministradores del proyecto J1 ordenados.
4. Obtener los suministros cuya cantidad esté entre 300 y 750 inclusive.
5. Obtener todas las ocurrencias de P.color y P.ciudad eliminando los pares duplicados.
6. Obtener todos los posibles tríos de código de suministrador, de pieza y de proyecto en los que el suministrador, pieza y proyecto estén todos en la misma ciudad.
7. Obtener los códigos de suministrador, de pieza y de proyecto de aquellos suministros en los que el suministrador, pieza y proyecto estén todos en la misma ciudad.
8. Obtener todos los posibles tríos de código de suministrador, de pieza y de proyecto en los que el suministrador, pieza y proyecto no estén todos en la misma ciudad.
9. Obtener todos los posibles tríos de código de suministrador, de pieza y de proyecto en los que el suministrador, pieza y proyecto no estén ninguno en la misma ciudad.
10. Obtener las piezas suministradas por suministradores de Londres.
11. Obtener las piezas suministradas por suministradores de Londres a proyectos en Londres.
12. Obtener todos los pares de nombres de ciudades en las que el suministrador de la primera sirva a un proyecto de la segunda.
13. Obtener códigos de piezas que sean suministradas a un proyecto por un suministrador de la misma ciudad del proyecto.
14. Obtener códigos de proyectos que sean suministrados por un suministrador de una ciudad distinta a la del proyecto.
15. Obtener todos los pares de códigos de piezas suministradas por el mismo suministrador.

16. Obtener el número total de proyectos suministrados por el suministrador S1.
17. Obtener la cantidad total de piezas P1 suministradas por S1.
18. Obtener para cada pieza suministrada a un proyecto, el código de pieza, el código de proyecto y la cantidad total correspondiente.
19. Obtener los códigos de proyectos y los códigos de piezas en los que la cantidad suministrada a algún proyecto sea en una cantidad media superior a 320.
20. Obtener los códigos de suministros en los que la cantidad no sea nula.
21. Obtener los nombres de los proyectos a los que suministra S1.
22. Obtener los colores de las piezas suministradas por S1.
23. Obtener los códigos de las piezas suministradas a cualquier proyecto de Londres.
24. Obtener los códigos de proyectos que usen al menos una pieza suministrada por S1.
25. Obtener los códigos de suministradores que suministren al menos una pieza suministrada por un suministrador que suministre al menos una pieza de color rojo.
26. Obtener los códigos de suministradores con estado menor que S1.
27. Obtener los códigos de los proyectos que usen la pieza P1 en una cantidad media mayor que la mayor cantidad en la que cualquier pieza sea suministrada al proyecto J1.
28. Obtener los códigos de suministradores que suministren a algún proyecto la pieza P1 en una cantidad mayor que la cantidad media en la que se suministra la pieza P1 a dicho proyecto.
29. Obtener los códigos de los proyectos que no reciben ninguna pieza roja suministrada por algún suministrador de Londres.
30. Obtener los códigos de los proyectos suministrados únicamente por S1.
31. Obtener los códigos de piezas suministradas a todos los proyectos en Londres.
32. Obtener los códigos de los suministradores que suministren al menos la misma pieza a todos los proyectos.
33. Obtener los códigos de los proyectos que reciban al menos todas las piezas que suministra S1.
34. Construir una lista ordenada de todas las ciudades en las que al menos resida un suministrador, una pieza o un proyecto.

Problema 6.

Consideremos una base de datos con las siguientes relaciones:



Se pide obtener, para cada uno de los siguientes apartados, una sentencia de álgebra relacional que permita obtener la información pedida:

1. Obtener los datos de todos los clientes de Madrid.
2. Obtener los nombres de todas las marcas de coches.
3. Obtener el identificador de todos los concesionarios cuya cantidad distribuida es mayor que 18.
4. Obtener el identificador de todos los concesionarios cuya cantidad distribuida está comprendida entre 10 y 18, ambos inclusive. Realizarlo de dos maneras distintas.
5. Obtener el identificador de todos los concesionarios cuya cantidad distribuida está comprendida entre 10 y 18, ambos exclusive.
6. Obtener el identificador de los concesionarios que han adquirido más de 10 coches de algún tipo, o menos de 5 (como se indica, de un tipo, no en total).
7. Obtener todas las parejas de identificadores de marcas y clientes que sean de la misma ciudad.
8. Obtener todas las parejas de identificadores de clientes y marcas que no sean de la misma ciudad.
9. Obtener los códigos de los coches distribuidos por algún concesionario de Barcelona.
10. Obtener el código de los coches vendidos a clientes de Madrid.
11. Obtener los códigos de los coches adquiridos por un cliente de Madrid en un concesionario de Madrid.
12. Obtener el código de los coches comprados en un concesionario de la misma ciudad del cliente que lo compra.

13. Obtener los códigos de los coches comprados en un concesionario de distinta ciudad que el cliente que lo compra.
14. Obtener todas las parejas de nombres de marcas que sean de la misma ciudad.
15. Obtener las parejas de modelos de coches cuyo nombre es el mismo y cuya marca es de Bilbao.
16. Obtener el número total de nombres de marcas de coche que son de Madrid.
17. Obtener la media de la cantidad de coches que distribuyen todos los concesionarios.
18. Obtener el identificador de cliente con la numeración más alta de todos los clientes de Madrid.
19. Obtener el identificador de cliente con numeración más baja de los que han comprado un coche blanco.
20. Obtener el identificador de los concesionarios cuyo número de coches en stock (coches que distribuyen) no es nulo.
21. Obtener el identificador de los clientes que han comprado algún coche a un concesionario de Madrid.
22. Obtener el color de los coches vendidos por el concesionario acar.
23. Obtener el código de los coches vendidos por algún concesionario de Madrid.
24. Obtener el nombre y modelo de los coches vendidos por algún concesionario de Barcelona.
25. Obtener todos los nombre de los clientes que hayan adquirido algún coche del concesionario dcar.
26. Obtener el nombre y apellido de los clientes que han adquirido un coche modelo gti de color blanco.
27. Obtener el nombre y apellido de los clientes que han adquirido un automóvil a un concesionario de Madrid que posea actualmente coches en stock del modelo gti.
28. Obtener el nombre y apellido de los clientes con identificador menor que el del cliente Juan Martín.
29. Obtener el nombre y apellido de los clientes cuyo identificador es menor que el de los clientes de Barcelona.
30. Obtener el nombre y apellido de los clientes cuyo nombre empieza por a y cuyo identificador es mayor que el de todos los clientes de Madrid.
31. Obtener el nombre y apellido de los clientes cuyo nombre empieza por a y cuyo identificador es mayor que el de alguno de los clientes de Madrid.
32. Obtener el nombre y apellido de los clientes cuyo nombre empieza por A y cuyo identificador es mayor que el de alguno de los clientes de Madrid, o menor que el de todos los de Valencia.
33. Obtener el nombre y apellido de los clientes que han comprado como mínimo un coche blanco y otro rojo.
34. Obtener el identificador de los clientes cuya ciudad sea la última de la lista alfabética de las ciudades donde hay concesionarios.
35. Obtener la media de los automóviles que cada concesionario tiene actualmente en stock (es decir, los que distribuye).

36. Obtener el identificador del concesionario que no sea de Madrid cuya media de vehículos en stock sea la más alta de todas las medias.
37. Obtener usando exists el identificador de los clientes que hayan adquirido por lo menos alguno de los coches que ha sido vendido por el concesionario cuyo identificador es 1.
38. Obtener los identificadores de los clientes que sólo han comprado coches al concesionario 1.
39. Obtener los nombres de los clientes que no han comprado ningún coche rojo a ningún concesionario de Madrid.
40. Obtener el nombre de los clientes que sólo han comprado en el concesionario 1.
41. Obtener el código de coche de aquellos automóviles comprados por todos los clientes de Madrid.
42. Obtener el identificador de los clientes que han adquirido por lo menos los mismos automóviles que el cliente Luis García.
43. Obtener el identificador de los concesionarios que han vendido el mismo coche a todos los clientes.