A) Dadas las siguientes relaciones:

Clientes (id\_cliente, nombre, apellido, teléfono, localidad)

Compras (id\_compra, factura, id\_cliente(\*), fecha)

Artículos (id\_articulo, descripción, stock, rubro, localidad)

Detalles (id\_compra(\*), id\_articulo(\*), importe, id\_detalle)

MARCAR CLAVES AJENAS (\*) Y PRIMARIAS (subrayan)

1) Mostrar el nombre y apellido de todos los clientes que compraron tornillos

SELECT c.nombre , c.apellido FROM clientes as c , artículos as a , compras as comp , detalles as d WHERE c.id\_cliente = comp.id\_cliente AND comp.id\_compra = d.id\_compra AND a.id\_articulo = d.id\_articulo AND a.descripcion = “tornillos”

2) Mostrar todos los distintos nombres de artículos que fueron vendidos por lo menos una vez ordenados de forma descendente por nombre.

SELECT distinct descripcion

FROM articulos

WHERE id\_articulo IN

(SELECT id\_articulo

FROM detalles)

ORDER BY descripcion desc

//utilizando inner join

SELECT distinct a.descripcion

FROM articulos as a , detalles as d

WHERE a.id\_articulo = d.id\_articulo

ORDER BY a.descripcion desc

3) Mostrar la cantidad de compras cuya factura comienza con los caracteres “ab” y contiene alguna “c” y además se realizaron entre 01/01/18 y el 30/03/18

SELECT \* FROM compras

WHERE factura LIKE 'ab%c%’

AND fecha >= '01/01/18' AND fecha<= '30/03/18'

// otra forma

SELECT \* FROM compras

WHERE factura LIKE 'ab%'

AND factura LIKE '%c%

AND fecha BEETWEEN '01/01/18' AND '30/03/18'

4) Mostar el nombre del artículo y el importe total de ventas de los mismos

SELECT a.descripcion , SUM(d.importe)

FROM articulos as a , detalles as d

WHERE a.id\_articulo = d.id\_articulo

GROUP BY a.descripcion

5) Mostrar los nombres de los artículos no vendidos

SELECT DISTINCT descripcion

FROM articulos as a

WHERE id\_articulo NOT IN

(SELECT id\_articulo

FROM DETALLES)

6) Mostrar las facturas de compras con más de un detalle

SELECT factura

FROM compras

WHERE id\_compra IN

(SELECT id\_compra

FROM detalles

group by id\_compra

HAVING count(\*)>1 )

B) Enumere y explique brevemente las características de un sistema de BD.

* Independencia de datos respecto de los sistemas: si se modifican los datos o info no afecta al sistema.
* Posibilidad de disminuir la redundancia: técnica de normalización para que los datos no se repitan en varios lados
* Naturaleza auto descriptiva de los sistemas de las BD: no solo contiene los objetos y contenido sino que puedo consultar todo , es decir el “catalogo(diccionario”
* Manejo múltiple visto de los datos: cada usuario ve lo que necesita acotando tablas mediante vistas
* Posibilidad de aplicar restricciones de seguridad: otorgar permisos, que puede hacer y que no, etc.
* Posibilidad de compartir datos : la info está disponible para varios usuarios
* Compacto , rápido y actual:
* Auditoria: viene deshabilitado por defecto y es un registro y que registrado quien hace tal cosa, a que día y que usuario.
* Recupero: recovery ante una caída y se debe hacer backup para recuperar
* Relacionabilidad: diferentes caminos de acceso para utilizar y explorar datos

C) Enumere y explique los 3 niveles en que se divide la arquitectura de un SGBD:

* Externo: usuario indivual , para él será o bien un lenguaje de consulta o algún lenguaje de aplicación especial. Se ocupa de la forma como los usuarios individuales perciben los datos
* Conceptual: representación de toda la información contenida en la BD , es la vista comunitaria de los usuarios o sea un nivel de mediación
* Interno: es el que se ocupa de la forma como se almacenan físicamente los datos. Es una representación de bajo nivel

D) Explique diferencia entre modelo jerárquico y modelo plex : la diferencia fundamental que tiene es que en el modelo jerarquico no se permite que un hijo tiene varios padres en cambio en el modelo plex cada uno puede tener más de un padre