

```

1 --Resolver las siguientes consultas utilizando la sintaxis SQL
2 --DATABASE ventas
3 --1. Devuelve un listado con todos los pedidos que se han realizado. Los pedidos
  deben estar ordenados por la fecha de realización, mostrando en primer lugar los
  pedidos más recientes.
4 SELECT * FROM ventas.pedidos ORDER BY fecha desc
5
6 --2. Devuelve todos los datos de los dos pedidos de mayor valor.
7 SELECT * FROM ventas.pedidos ORDER BY cantidad desc limit 2
8
9 --3. Devuelve un listado con los identificadores de los clientes que han realizado
  algún pedido. Tenga en cuenta que no debe mostrar identificadores que estén
  repetidos.
10 SELECT DISTINCT clientes.id AS clientes_id FROM ventas.clientes INNER JOIN
   ventas.pedidos ON ventas.clientes.id=ventas.pedidos.id_cliente
11
12 --4. Devuelve un listado de todos los pedidos que se realizaron durante el año 2022,
   cuya cantidad total sea superior a $500.
13 SELECT * FROM ventas.pedidos WHERE YEAR(fecha)=2022 AND cantidad>500
14
15 --5. Devuelve un listado con el nombre y apellido de los vendedores que tienen una
   comisión entre 0.05 y 0.11.
16 SELECT vendedores.nombre, vendedores.apellido FROM ventas.vendedores WHERE comision
   BETWEEN 0.05 AND 0.11;
17
18 --6. Devuelve el valor de la comisión de mayor valor que existe en la tabla
   vendedores.
19 SELECT MAX(vendedores.comision) FROM ventas.vendedores
20
21 --7. Devuelve el identificador, nombre y primer apellido de aquellos clientes cuyo
   cuitcuil no es NULL. El listado deberá estar ordenado alfabéticamente por apellido y
   nombre.
22 SELECT clientes.id, clientes.nombre, clientes.apellido FROM ventas.clientes WHERE
   cuitcuil IS NOT NULL ORDER BY clientes.apellido asc, clientes.nombre asc
23
24 --8. Devuelve un listado de los nombres de los clientes que empiezan por "A" y
   terminan por "n" y también los nombres que empiezan por "P". El listado deberá estar
   ordenado alfabéticamente.
25 SELECT clientes.nombre FROM ventas.clientes WHERE (nombre LIKE 'A%' and nombre
   LIKE '%n') OR nombre LIKE 'P%' ORDER BY clientes.apellido asc, clientes.nombre asc
26
27 --9. Devuelve un listado de los nombres de los clientes que no empiezan por "A". El
   listado deberá estar ordenado alfabéticamente.
28 SELECT clientes.nombre FROM ventas.clientes WHERE nombre NOT LIKE 'A%' ORDER BY
   clientes.apellido asc, clientes.nombre asc
29
30 --10.Devuelve un listado con los nombres de los vendedores que terminan por "el" o
   "o". Tenga en cuenta que se deberán eliminar los nombres repetidos.
31 SELECT vendedores.nombre FROM ventas.vendedores WHERE nombre LIKE '%el' OR nombre
   LIKE '%o' ORDER BY vendedores.nombre asc
32
33 --11.Devuelve un listado con el identificador, nombre y apellido de todos los
   clientes que han realizado algún pedido. El listado debe estar ordenado
   alfabéticamente y se deben eliminar los elementos repetidos.
34 SELECT DISTINCT clientes.id AS clientes_id, clientes.nombre, clientes.apellido FROM
   ventas.clientes INNER JOIN ventas.pedidos ON
   ventas.clientes.id=ventas.pedidos.id_cliente ORDER BY clientes.apellido asc,
   clientes.nombre asc
35

```

```

36 --12.Devuelve un listado que muestre todos los pedidos que ha realizado cada cliente.
    El resultado debe mostrar todos los datos de los pedidos y del cliente. El listado
    debe mostrar los datos de los clientes ordenados alfabéticamente.
37 SELECT pedidos.id AS pedidos_id FROM ventas.pedidos INNER JOIN ventas.clientes ON
    ventas.pedidos.id=ventas.clientes.id
38
39 --13.Devuelve un listado que muestre todos los pedidos en los que ha participado un
    vendedor. El resultado debe mostrar todos los datos de los pedidos y de los
    vendedores. El listado debe mostrar los datos de los vendedores ordenados
    alfabéticamente.
40 SELECT * FROM ventas.pedidos INNER JOIN ventas.vendedores ON
    ventas.pedidos.id_vendedor=ventas.vendedores.id ORDER BY vendedores.apellido asc,
    vendedores.nombre asc
41
42 --14.Devuelve un listado que muestre todos los clientes, con todos los pedidos que
    han realizado y con los datos de los vendedores asociados a cada pedido.
43 SELECT * FROM ventas.pedidos
44 INNER JOIN ventas.clientes ON ventas.clientes.id=ventas.pedidos.id_cliente
45 INNER JOIN ventas.vendedores ON ventas.pedidos.id_vendedor=ventas.vendedores.id
46
47 --15.Devuelve un listado de todos los clientes que realizaron un pedido durante el
    año 2022, cuyo monto esté entre $300 y $1000.
48 SELECT * FROM ventas.pedidos
49 INNER JOIN ventas.clientes ON ventas.clientes.id=ventas.pedidos.id_cliente
50 WHERE YEAR(fecha)=2022 AND cantidad BETWEEN 300 AND 1000;
51
52
53 --16.Devuelve el nombre y apellido de todos los vendedores que han participado en
    algún pedido realizado por María Santana.
54 SELECT DISTINCT vendedores.nombre, vendedores.apellido FROM ventas.pedidos
55 INNER JOIN ventas.clientes ON ventas.clientes.id=ventas.pedidos.id_cliente
56 INNER JOIN ventas.vendedores ON ventas.pedidos.id_vendedor=ventas.vendedores.id
57 WHERE clientes.nombre='María' and clientes.apellido='Santana'
58
59 --17.Devuelve el nombre de todos los clientes que han realizado algún pedido con el
    vendedor Daniel Sáez.
60 SELECT DISTINCT clientes.nombre FROM ventas.pedidos
61 INNER JOIN ventas.clientes ON ventas.clientes.id=ventas.pedidos.id_cliente
62 INNER JOIN ventas.vendedores ON ventas.pedidos.id_vendedor=ventas.vendedores.id
63 WHERE vendedores.nombre='Daniel' and vendedores.apellido='Sáez'
64
65 --18.Devuelve un listado con todos los clientes junto con los datos de los pedidos
    que han realizado. Este listado también debe incluir los clientes que no han
    realizado ningún pedido. El listado debe estar ordenado alfabéticamente por el
    apellido y nombre de los clientes.
66 SELECT DISTINCT * FROM ventas.clientes
67 LEFT JOIN ventas.pedidos ON ventas.clientes.id=ventas.pedidos.id_cliente
68 ORDER BY clientes.apellido asc, clientes.nombre asc
69
70 --19.Devuelve un listado con todos los vendedores junto con los datos de los pedidos
    que han realizado. Este listado también debe incluir los vendedores que no han
    realizado ningún pedido. El listado debe estar ordenado alfabéticamente por el
    apellido y nombre de los vendedores.
71 SELECT DISTINCT * FROM ventas.vendedores
72 LEFT JOIN ventas.pedidos ON ventas.vendedores.id=ventas.pedidos.id_vendedor
73 ORDER BY vendedores.apellido asc, vendedores.nombre asc
74
75 --20.Devuelve un listado que solamente muestre los clientes que no han realizado
    ningún pedido.
76 SELECT * FROM ventas.clientes

```

```

77 left JOIN ventas.pedidos ON ventas.clientes.id=ventas.pedidos.id_cliente
78 where ventas.pedidos.id_cliente is null
79
80 --21.Devuelve un listado que solamente muestre los vendedores que no han realizado
    ningún pedido.
81 SELECT * FROM ventas.vendedores
82 left JOIN ventas.pedidos ON ventas.vendedores.id=ventas.pedidos.id_vendedor
83 where ventas.pedidos.id_vendedor is null
84
85 --segunda solución
86 SELECT * FROM ventas.pedidos
87 right JOIN ventas.vendedores ON ventas.pedidos.id_vendedor=ventas.vendedores.id where
    pedidos.id is null
88
89 --22.Devuelve un listado con los clientes que no han realizado ningún pedido y de los
    vendedores que no han participado en ningún pedido. Ordene el listado alfabéticamente
    por el apellido y el nombre. En el listado deberá diferenciar de algún modo los
    clientes y los vendedores.
90 SELECT vendedores.nombre, vendedores.apellido, '' as cliente_id, vendedores.id as
    vendedor_id FROM ventas.pedidos
91 right JOIN ventas.vendedores ON ventas.pedidos.id_vendedor=ventas.vendedores.id
92 where ventas.pedidos.id_vendedor is null
93 union
94 SELECT clientes.nombre, clientes.apellido, clientes.id as cliente_id, '' FROM
    ventas.pedidos
95 right JOIN ventas.clientes ON ventas.pedidos.id_cliente=ventas.clientes.id
96 where ventas.pedidos.id_cliente is null
97 ORDER BY apellido asc, nombre asc
98
99 --23.Calcula la cantidad total que suman todos los pedidos que aparecen en la tabla
    pedido.
100 SELECT sum(cantidad) as "Total Cantidad" FROM ventas.pedidos
101
102 --24.Calcula la cantidad media de todos los pedidos que aparecen en la tabla pedido.
103 SELECT AVG(cantidad) as "Cantidad Media" FROM ventas.pedidos
104
105 --25.Calcula el número total de vendedores distintos que aparecen en la tabla pedido.
106 SELECT count(distinct pedidos.id_vendedor) as Nro_vendedores FROM ventas.pedidos
107
108 --26.Calcula el número total de clientes que aparecen en la tabla cliente.
109 SELECT count(ventas.clientes.id) FROM ventas.clientes
110
111 --27.Calcula cuál es la mayor cantidad que aparece en la tabla pedido.
112 SELECT max(ventas.pedidos.cantidad) FROM ventas.pedidos
113
114 --28.Calcula cuál es la menor cantidad que aparece en la tabla pedido.
115 SELECT min(ventas.pedidos.cantidad) FROM ventas.pedidos
116
117 --29.Calcula cuál es el valor máximo de categoría para cada una de las ciudades que
    aparece en la tabla cliente.
118 SELECT max(clientes.categoría), clientes.ciudad from ventas.clientes group BY ciudad
119
120 --30.Calcula cuál es el máximo valor de los pedidos realizados durante el mismo día
    para cada uno de los clientes. Es decir, el mismo cliente puede haber realizado
    varios pedidos de diferentes cantidades el mismo día. Se pide que se calcule cuál es
    el pedido de máximo valor para cada uno de los días en los que un cliente ha
    realizado un pedido. Muestra el identificador del cliente, nombre, apellido, la fecha
    y el valor de la cantidad.
121 SELECT max(pedidos.cantidad) as Max_Pedido, fecha, id_cliente, clientes.nombre,
    clientes.apellido from ventas.pedidos

```

```

122 inner join ventas.clientes ON ventas.clientes.id=ventas.pedidos.id_cliente
123 group by ventas.pedidos.id_cliente, ventas.pedidos.fecha
124
125 --31.Calcula cuál es el máximo valor de los pedidos realizados durante el mismo día
    para cada uno de los clientes, teniendo en cuenta que sólo queremos mostrar aquellos
    pedidos que superen la cantidad de $2000 .
126 SELECT max(pedidos.cantidad) as Max_Pedido, fecha, id_cliente, clientes.nombre,
    clientes.apellido from ventas.pedidos
127 inner join ventas.clientes ON ventas.clientes.id=ventas.pedidos.id_cliente where
    pedidos.cantidad>2000
128 group by ventas.pedidos.id_cliente, ventas.pedidos.fecha
129
130 --32.Calcula el máximo valor de los pedidos realizados para cada uno de los
    vendedores durante la fecha 2021-08-17. Muestra el identificador del vendedor,
    nombre, apellido y total.
131 SELECT pedidos.fecha Fecha, max(pedidos.cantidad) as Max_Pedido_día,
    sum(pedidos.cantidad) as Total_pedidos_día, vendedores.id as
    Id_Vendedor,vendedores.apellido Apellido, vendedores.nombre as Nombre from
    ventas.pedidos
132 inner join ventas.vendedores ON ventas.pedidos.id_vendedor=ventas.vendedores.id
133 where ventas.pedidos.fecha='2021-08-17' group by ventas.pedidos.id_vendedor
134
135 --33.Devuelve un listado con el identificador de cliente, nombre y apellido y el
    número total de pedidos que ha realizado cada uno de los clientes. Tenga en cuenta
    que pueden existir clientes que no han realizado ningún pedido. Estos clientes
    también deben aparecer en el listado indicando que el número de pedidos realizados es
    0.
136 SELECT clientes.id, clientes.nombre, clientes.apellido,count(pedidos.id) as
    Cant_pedidos FROM ventas.clientes
137 LEFT JOIN ventas.pedidos ON ventas.clientes.id=ventas.pedidos.id_cliente group by
    clientes.id order by Cant_pedidos desc
138
139 --34.Devuelve un listado con el identificador de cliente, nombre, apellido y el
    número total de pedidos que ha realizado cada uno de clientes durante el año 2020.
140 SELECT clientes.id as Id_Cliente, clientes.nombre,
    clientes.apellido,count(pedidos.id) as Cant_pedidos FROM ventas.clientes
141 LEFT JOIN ventas.pedidos ON ventas.clientes.id=ventas.pedidos.id_cliente
142 where YEAR(fecha)=2020
143 group by clientes.id order by Cant_pedidos desc
144
145 --35.Devuelve cuál ha sido el pedido de máximo valor que se ha realizado cada año.
146 SELECT YEAR(pedidos.fecha) Año, max(pedidos.cantidad) as Pedido_Max
147 FROM ventas.pedidos
148 group by YEAR(fecha)
149
150 --36.Devuelve el número total de pedidos que se han realizado cada año.
151 SELECT YEAR(pedidos.fecha) Año, count(pedidos.id) as Cant_Pedidos
152 FROM ventas.pedidos
153 group by YEAR(fecha)
154
155 --37.Devuelve un listado con todos los pedidos que ha realizado Adela Salas. (Sin
    utilizar INNER JOIN).
156 SELECT pedidos.id as Id_pedido_Adela
157 FROM ventas.pedidos
158 cross join ventas.clientes
159 where pedidos.id_cliente=clientes.id and clientes.nombre='Adela' and
    clientes.apellido='Salas'
160
161 --38.Devuelve el número de pedidos en los que ha participado el vendedor Daniel Sáez.
    (Sin utilizar INNER JOIN)

```

```

162 SELECT pedidos.id,pedidos.fecha, vendedores.nombre, vendedores.apellido as
    Id_pedido_Adela
163 FROM ventas.pedidos
164 cross join ventas.vendedores
165 where pedidos.id_vendedor=vendedores.id and vendedores.nombre='Daniel' and
    vendedores.apellido='Sáez'
166
167 --39.Devuelve los datos del cliente que realizó el pedido más caro en el año 2020.
    (Sin utilizar INNER JOIN)
168 SELECT clientes.id, max(pedidos.cantidad), YEAR(pedidos.fecha), clientes.nombre,
    clientes.apellido
169 FROM ventas.pedidos
170 cross join ventas.clientes
171 where pedidos.id_cliente=clientes.id and YEAR(fecha)=2020
172
173 --40.Devuelve la fecha y la cantidad del pedido de menor valor realizado por el
    cliente Pepe Ruiz Santana.
174 SELECT clientes.id Cliente_id, min(pedidos.cantidad) Menor_pedido,
    YEAR(pedidos.fecha), clientes.nombre, clientes.apellido
175 FROM ventas.pedidos
176 cross join ventas.clientes
177 where pedidos.id_cliente=clientes.id and clientes.nombre='Pepe' AND
    clientes.apellido='Ruiz'
178
179 --41.Devuelve un listado de los clientes que no han realizado ningún pedido.
    (Utilizando IN o NOT IN).
180 SELECT * FROM ventas.clientes
181 left JOIN ventas.pedidos ON ventas.clientes.id=ventas.pedidos.id_cliente
182 where coalesce(ventas.pedidos.id_cliente,'') IN (')
183
184 --42.Devuelve un listado de los vendedores que no han realizado ningún pedido.
    (Utilizando IN o NOT IN).
185 SELECT * FROM ventas.vendedores
186 left JOIN ventas.pedidos ON ventas.vendedores.id=ventas.pedidos.id_vendedor
187 where coalesce(ventas.pedidos.id_vendedor,'') IN (')
188
189 --43.Devuelve un listado de los clientes que no han realizado ningún pedido.
    (Utilizando EXISTS o NOT EXISTS).
190 SELECT * FROM ventas.clientes
191 left JOIN ventas.pedidos ON ventas.clientes.id=ventas.pedidos.id_cliente
192 WHERE EXISTS
193 (SELECT * FROM ventas.clientes WHERE ventas.pedidos.id_cliente is null);
194
195
196 --44.Devuelve un listado de los vendedores que no han realizado ningún pedido.
    (Utilizando EXISTS o NOT EXISTS).
197 SELECT * FROM ventas.vendedores
198 left JOIN ventas.pedidos ON ventas.vendedores.id=ventas.pedidos.id_vendedor
199 WHERE EXISTS
200 (SELECT * FROM ventas.vendedores WHERE ventas.pedidos.id_vendedor is null);

```