

Code, Learn & Practice (Base de datos (Trabajo con expresiones regulares y funciones matemáticas"))

Descripción

En la siguiente tarea se premia el uso de expresiones regulares en BBDD.

Realiza la lectura de la base de datos a través del fichero base datos de clientes.

```
-- Crear la tabla de Clientes
CREATE TABLE IF NOT EXISTS Clientes (
    id INTEGER PRIMARY KEY,
    nombre TEXT NOT NULL,
    email TEXT UNIQUE
);

-- Crear la tabla de Productos
CREATE TABLE IF NOT EXISTS Productos (
    id INTEGER PRIMARY KEY,
    nombre TEXT NOT NULL,
    precio REAL
);

-- Crear la tabla de Pedidos
CREATE TABLE IF NOT EXISTS Pedidos (
    id_pedido INTEGER PRIMARY KEY,
    id_cliente INTEGER,
    id_producto INTEGER,
    cantidad INTEGER,
    fecha_pedido DATE,
    FOREIGN KEY (id_cliente) REFERENCES Clientes(id),
    FOREIGN KEY (id_producto) REFERENCES Productos(id)
);

INSERT INTO Clientes (nombre, email) VALUES
    ('Juan Pérez', 'juan@example.com'),
    ('María Gómez', 'maria@example.com'),
    ('Carlos López', 'carlos@example.com'),
    ('Ana Rodríguez', 'ana@example.com'),
    ('Luisa Martínez', 'luisa@example.com'),
    ('Pedro Sánchez', 'pedro@example.com'),
    ('Laura García', 'laura@example.com'),
    ('Miguel Martín', 'miguel@example.com'),
    ('Elena González', 'elena@example.com'),
    ('David Torres', 'david@example.com'),
    ('Sofía Ruiz', 'sofia@example.com'),
    ('Javier López', 'javier@example.com'),
    ('Carmen Vargas', 'carmen@example.com'),
    ('Daniel Muñoz', 'daniel@example.com'),
    ('Isabel Serrano', 'isabel@example.com'),
```

```
( 'Alejandro Muñoz', 'alejandro@example.com'),
( 'Raquel Herrera', 'raquel@example.com'),
( 'Francisco Mora', 'francisco@example.com'),
( 'Marina Díaz', 'marina@example.com'),
( 'Antonio Jiménez', 'antonio@example.com'),
( 'Beatriz Romero', 'beatriz@example.com'),
( 'Carlos Gómez', 'carlos.gomez@example.com'),
( 'Clara Sánchez', 'clara.sanchez@example.com'),
( 'Andrés Martínez', 'andres@example.com'),
( 'Lucía Díaz', 'lucia@example.com'),
( 'Mario Serrano', 'mario@example.com'),
( 'Eva Torres', 'eva.torres@example.com'),
( 'Roberto Ruiz', 'roberto@example.com'),
( 'Celia García', 'celia@example.com');
```

INSERT INTO productos (nombre, precio) VALUES

```
( 'Laptop', 1200.00),
( 'Smartphone', 699.99),
( 'TV LED', 799.50),
( 'Tablet', 299.99),
( 'Auriculares Bluetooth', 79.99),
( 'Impresora', 199.99),
( 'Cámara Digital', 499.99),
( 'Reproductor de Audio', 149.99),
( 'Altavoces Inalámbricos', 129.99),
( 'Reloj Inteligente', 249.99),
( 'Teclado Inalámbrico', 59.99),
( 'Ratón Óptico', 29.99),
( 'Monitor LED', 349.99),
( 'Mochila para Portátil', 49.99),
( 'Disco Duro Externo', 89.99),
( 'Router Wi-Fi', 69.99),
( 'Lámpara LED', 39.99),
( 'Batería Externa', 19.99),
( 'Estuche para Auriculares', 14.99),
( 'Tarjeta de Memoria', 24.99),
( 'Cargador Inalámbrico', 34.99),
( 'Kit de Limpieza para Computadoras', 9.99),
( 'Funda para Tablet', 19.99),
( 'Soporte para Teléfono', 14.99),
( 'Hub USB', 29.99),
( 'Webcam HD', 59.99),
( 'Funda para Laptop', 29.99),
( 'Adaptador HDMI', 12.99);
```

INSERT INTO Pedidos (id_cliente, id_producto, cantidad, fecha_pedido)
VALUES

```
(1, 1, 2, '2024-02-01'),
(2, 2, 1, '2024-02-02'),
(3, 3, 3, '2024-02-03'),
(4, 4, 1, '2024-02-04'),
(5, 5, 2, '2024-02-05'),
(6, 6, 1, '2024-02-06'),
(7, 7, 3, '2024-02-07'),
(8, 8, 2, '2024-02-08'),
```

```

(9, 9, 1, '2024-02-09'),
(10, 10, 2, '2024-02-10'),
(11, 11, 1, '2024-02-11'),
(12, 12, 3, '2024-02-12'),
(13, 13, 1, '2024-02-13'),
(14, 14, 2, '2024-02-14'),
(15, 15, 1, '2024-02-15'),
(16, 16, 3, '2024-02-16'),
(17, 17, 2, '2024-02-17'),
(18, 18, 1, '2024-02-18'),
(19, 19, 2, '2024-02-19'),
(20, 20, 1, '2024-02-20'),
(21, 21, 3, '2024-02-21'),
(22, 22, 1, '2024-02-22'),
(23, 23, 2, '2024-02-23'),
(24, 24, 1, '2024-02-24'),
(25, 25, 3, '2024-02-25'),
(26, 26, 2, '2024-02-26'),
(27, 27, 1, '2024-02-27'),
(28, 28, 2, '2024-02-28'),
(29, 29, 1, '2024-02-29'),
(30, 30, 3, '2024-03-01');
    
```

Realiza el listado de consultas que se encuentran en el fichero consultas-bd.

Obtener todos los clientes.

```

sqlite> select * from clientes;
+----+-----+-----+
| id | nombre | email |
+----+-----+-----+
| 1 | Juan Pérez | juan@example.com |
| 2 | María Gómez | maria@example.com |
| 3 | Carlos López | carlos@example.com |
| 4 | Ana Rodríguez | ana@example.com |
| 5 | Luisa Martínez | luisa@example.com |
| 6 | Pedro Sánchez | pedro@example.com |
| 7 | Laura García | laura@example.com |
| 8 | Miguel Martín | miguel@example.com |
| 9 | Elena González | elena@example.com |
| 10 | David Torres | david@example.com |
| 11 | Sofía Ruiz | sofia@example.com |
| 12 | Javier López | javier@example.com |
| 13 | Carmen Vargas | carmen@example.com |
| 14 | Daniel Muñoz | daniel@example.com |
| 15 | Isabel Serrano | isabel@example.com |
| 16 | Alejandro Muñoz | alejandro@example.com |
| 17 | Raquel Herrera | raquel@example.com |
| 18 | Francisco Mora | francisco@example.com |
| 19 | Marina Díaz | marina@example.com |
    
```

20	Antonio Jiménez	antonio@example.com	
21	Beatriz Romero	beatriz@example.com	
22	Carlos Gómez	carlos.gomez@example.com	
23	Clara Sánchez	clara.sanchez@example.com	
24	Andrés Martínez	andres@example.com	
25	Lucía Díaz	lucia@example.com	
26	Mario Serrano	mario@example.com	
27	Eva Torres	eva.torres@example.com	
28	Roberto Ruiz	roberto@example.com	
29	Celia García	celia@example.com	

Obtener la cantidad total de productos en todos los pedidos.

```
sqlite> select sum(cantidad) as Suma_total_productos from pedidos;
+-----+
| Suma_total_productos |
+-----+
| 54                    |
+-----+
```

Obtener el precio promedio de los productos.

```
sqlite> select avg(precio)as precio_promedio_productos from productos;
+-----+
| precio_promedio_productos |
+-----+
| 188.294285714286          |
+-----+
```

Obtener los clientes que tienen un email válido (contiene '@').

```
sqlite> select email from clientes
...> where email regexp '@';
+-----+
| email |
+-----+
| alejandro@example.com |
| ana@example.com       |
| andres@example.com    |
| antonio@example.com   |
| beatriz@example.com   |
| carlos.gomez@example.com |
| carlos@example.com    |
| carmen@example.com    |
| celia@example.com     |
```

clara.sanchez@example.com	
daniel@example.com	
david@example.com	
elena@example.com	
eva.torres@example.com	
francisco@example.com	
isabel@example.com	
javier@example.com	
juan@example.com	
laura@example.com	
lucia@example.com	
luisa@example.com	
maria@example.com	
marina@example.com	
mario@example.com	
miguel@example.com	
pedro@example.com	
raquel@example.com	
roberto@example.com	
sofia@example.com	
+-----+	

Obtener el producto más caro.

sqlite> select max(precio)as precio_mas_alto from productos;		
+-----+		
precio_mas_alto		
+-----+		
1200.0		
+-----+		
sqlite> select nombre, max(precio)as precio_mas_caro from productos;		
+-----+-----+		
nombre	precio_mas_caro	
+-----+-----+		
Laptop	1200.0	
+-----+-----+		

Obtener los pedidos realizados en febrero de 2024.

FROM pedidos		
WHERE fecha_pedido BETWEEN '2024-02-01' AND '2024-02-29';		
+-----+-----+		
id_pedido	fecha_pedido	
+-----+-----+		
1	2024-02-01	
2	2024-02-02	
3	2024-02-03	
4	2024-02-04	

	5		2024-02-05	
	6		2024-02-06	
	7		2024-02-07	
	8		2024-02-08	
	9		2024-02-09	
	10		2024-02-10	
	11		2024-02-11	
	12		2024-02-12	
	13		2024-02-13	
	14		2024-02-14	
	15		2024-02-15	
	16		2024-02-16	
	17		2024-02-17	
	18		2024-02-18	
	19		2024-02-19	
	20		2024-02-20	
	21		2024-02-21	
	22		2024-02-22	
	23		2024-02-23	
	24		2024-02-24	
	25		2024-02-25	
	26		2024-02-26	
	27		2024-02-27	
	28		2024-02-28	
	29		2024-02-29	
+	-----	+	-----	+

Obtener la cantidad total de productos en todos los pedidos por producto.

sqlite>	SELECT	pr.nombre,	SUM(p.cantidad)	AS	cantidad_total
	FROM	pedidos	p		
	INNER JOIN	productos	pr	ON	pr.id = p.id_producto
	GROUP BY	pr.id,	pr.nombre		
	ORDER BY	cantidad_total	DESC;		
+	-----	+	-----	+	
	nombre		cantidad_total		
+	-----	+	-----	+	
	TV LED		3		
	Cámara Digital		3		
	Ratón Óptico		3		
	Router Wi-Fi		3		
	Cargador Inalámbrico		3		
	Hub USB		3		
	Laptop		2		
	Auriculares Bluetooth		2		
	Reproductor de Audio		2		
	Reloj Inteligente		2		
	Mochila para Portátil		2		
	Lámpara LED		2		
	Estuche para Auriculares		2		
	Funda para Tablet		2		

Webcam HD	2	
Adaptador HDMI	2	
Smartphone	1	
Tablet	1	
Impresora	1	
Altavoces Inalámbricos	1	
Teclado Inalámbrico	1	
Monitor LED	1	
Disco Duro Externo	1	
Batería Externa	1	
Tarjeta de Memoria	1	
Kit de Limpieza para Computadoras	1	
Soporte para Teléfono	1	
Funda para Laptop	1	
+-----+-----+	+-----+	+

Obtener los clientes que han realizado más de un pedido.

```
sqlite> SELECT id_cliente
FROM pedidos
GROUP BY id_cliente
HAVING COUNT(*) > 1;

** ningun cliente ha hecho mas de un pedido.
```

Obtener los productos que tienen un precio registrado.

```
sqlite> SELECT *
FROM productos
WHERE precio IS NOT NULL;
```

id	nombre	precio
1	Laptop	1200.0
2	Smartphone	699.99
3	TV LED	799.5
4	Tablet	299.99
5	Auriculares Bluetooth	79.99
6	Impresora	199.99
7	Cámara Digital	499.99
8	Reproductor de Audio	149.99
9	Altavoces Inalámbricos	129.99
10	Reloj Inteligente	249.99
11	Teclado Inalámbrico	59.99
12	Ratón Óptico	29.99
13	Monitor LED	349.99
14	Mochila para Portátil	49.99
15	Disco Duro Externo	89.99

	16		Router Wi-Fi		69.99	
	17		Lámpara LED		39.99	
	18		Batería Externa		19.99	
	19		Estuche para Auriculares		14.99	
	20		Tarjeta de Memoria		24.99	
	21		Cargador Inalámbrico		34.99	
	22		Kit de Limpieza para Computadoras		9.99	
	23		Funda para Tablet		19.99	
	24		Soporte para Teléfono		14.99	
	25		Hub USB		29.99	
	26		Webcam HD		59.99	
	27		Funda para Laptop		29.99	
	28		Adaptador HDMI		12.99	
+-----+-----+-----+-----+-----+						

Obtener la fecha del primer pedido realizado.

```
sqlite> select p.id_pedido, p.fecha_pedido from pedidos p
...> order by p.fecha_pedido asc
...> limit 1;
+-----+-----+
| id_pedido | fecha_pedido |
+-----+-----+
| 1         | 2024-02-01   |
+-----+-----+
```

Obtener los productos cuyos nombres comienzan con 'A' o 'B'.

```
sqlite> select pr.nombre from productos pr
...> where nombre regexp '^a|^b|^A|^B';
+-----+
| nombre |
+-----+
| Auriculares Bluetooth |
| Altavoces Inalámbricos |
| Batería Externa |
| Adaptador HDMI |
+-----+
```

Obtener la cantidad total de productos en todos los pedidos por cliente ordenado por cliente.

```
sqlite> select id_cliente, sum(cantidad)as cantidad_total from pedidos
...> group by id_cliente;
+-----+-----+
| id_cliente | cantidad_total |
+-----+-----+
```


+-----+-----+		
1	2	
2	1	
3	3	
4	1	
5	2	
6	1	
7	3	
8	2	
9	1	
10	2	
11	1	
12	3	
13	1	
14	2	
15	1	
16	3	
17	2	
18	1	
19	2	
20	1	
21	3	
22	1	
23	2	
24	1	
25	3	
26	2	
27	1	
28	2	
29	1	
30	3	
+-----+-----+		

Obtener los clientes que han realizado más de un pedido en febrero de 2024.

sql

Obtener los productos con precio entre 100 y 500.

sql

Obtener la cantidad total de productos en todos los pedidos por cliente ordenado por cantidad descendente.

sql

Obtener los clientes que tienen una 'a' en cualquier posición de su nombre.

sql

Obtener el precio máximo de los productos.

sql

Obtener los pedidos realizados por el cliente con ID 1 en febrero de 2024.

sql

Obtener la cantidad total de productos en todos los pedidos por producto ordenado por total de productos descendente.

sql

Obtener los productos que no tienen un precio registrado.

sql

Obtener la fecha del último pedido realizado.

sql

Obtener los clientes cuyo nombre tiene al menos 5 caracteres.

sql

Obtener los productos que tienen la letra 'o' en cualquier posición del nombre.

sql

Obtener la cantidad total de productos en todos los pedidos por cliente ordenado por cliente.

sql

Obtener los clientes cuyos nombres no contienen la letra 'i'.

sql

Obtener