

Vistas

Vista **view_user_orders**

Descripción: Muestra todos los pedidos realizados por los usuarios, junto con detalles del producto y el estado de entrega.

Objetivo: Facilitar la consulta de pedidos por parte del equipo de ventas o administración.

Tablas involucradas: users, orders, order_products, products.

```
1 view_user_ordersCREATE VIEW view_user_orders AS
2 SELECT
3     u.id AS user_id,
4     u.first_name,
5     u.last_name,
6     o.id AS order_id,
7     o.total,
8     o.delivery_status,
9     p.title AS product_title,
10    op.quantity,
11    op.price
12 FROM users u
13 JOIN orders o ON u.id = o.customer_id
14 JOIN order_products op ON o.id = op.order_id
15 JOIN products p ON p.id = op.product_id;
16
```

Vista: **view_active_offers**

Descripción: Muestra las ofertas activas, incluyendo el nombre del producto y el porcentaje de descuento.

Objetivo: Ofrecer una consulta rápida de las ofertas actualmente activas en la tienda.

Tablas involucradas: offers, products.

```
18 CREATE VIEW view_active_offers AS
19 SELECT
20     p.title AS product_title,
21     o.discount_percentage,
22     o.start_date,
23     o.end_date
24 FROM offers o
25 JOIN products p ON p.id = o.product_id
26 WHERE o.active = TRUE;
27
```

Funciones

Función: **fn_calculate_order_total**

Descripción: Calcula el total de un pedido sumando los precios de los productos multiplicados por la cantidad.

Objetivo: Calcular dinámicamente el total de un pedido, basado en los productos seleccionados y sus cantidades.

Tablas involucradas: order_products.

```
1 DELIMITER $$
2
3 CREATE FUNCTION fn_calculate_order_total(order_id INT)
4 RETURNS DECIMAL(10,2)
5 DETERMINISTIC
6 BEGIN
7     DECLARE total DECIMAL(10,2);
8
9     SELECT SUM(price * quantity) INTO total
10    FROM order_products
11    WHERE order_products.order_id = order_id;
12
13     RETURN IFNULL(total, 0);
14
15 END $$
16
17 DELIMITER ;
18
```

Función: **fn_user_full_name**

Descripción: Devuelve el nombre completo de un usuario.

Objetivo: Simplificar la obtención del nombre completo de un usuario.

Tablas involucradas: users.

```
1 DELIMITER $$
2
3 CREATE FUNCTION fn_user_full_name(user_id INT)
4 RETURNS VARCHAR(255)
5 DETERMINISTIC
6 READS SQL DATA
7 BEGIN
8     DECLARE full_name VARCHAR(255);
9     SELECT CONCAT(first_name, ' ', last_name) INTO full_name
10    FROM users
11    WHERE id = user_id;
12     RETURN full_name;
13 END$$
14
15 DELIMITER ;
16
```

Stored Procedure

Función: **fn_calculate_order_total**

Descripción: Activa una oferta para un producto en específico.

Objetivo: Automatizar la activación de ofertas según fechas y condiciones.

Tablas involucradas: offers.

```
1 DELIMITER //
2
3 CREATE PROCEDURE sp_activate_offer(IN offer_id INT)
4 BEGIN
5     UPDATE offers
6     SET active = TRUE
7     WHERE id = offer_id
8     AND start_date <= NOW()
9     AND end_date >= NOW();
10 END //
11
12 DELIMITER ;
13
```