SQL Assignment: WHERE, AND, OR (PostgreSQL)

Schema Summary

Tables: suppliers, warehouses, customers, purchase_orders, purchase_order_items, inventory, sales_orders, sales_order_items, shipments.

Key Columns

suppliers(supplier_id, supplier_name, contact_person, phone_number, email, country, created_at); warehouses(warehouse_id, warehouse_name, city, state, country, capacity_units, created_at); customers(customer_id, customer_name, phone_number, email, city, country, created_at); purchase_orders(po_id, supplier_id, po_date, expected_date, status, total_amount, created_at); purchase_order_items(po_item_id, po_id, product_id, quantity, unit_price, created_at); inventory(inventory_id, warehouse_id, product_id, quantity_on_hand, reorder_level, last_updated); sales_orders(sales_order_id, customer_id, order_date, status, total_amount, created_at); sales_order_items(so_item_id, sales_order_id, product_id, quantity, unit_price, created_at); shipments(shipment_id, sales_order_id, warehouse_id, shipment_date, delivery_date, shipment_status, tracking_number, created_at).

Instructions

Write PostgreSQL SELECT queries using the WHERE clause with AND/OR operators. Do not modify data. Do not include answers in your submission. Use table aliases where helpful.

Assignment Questions (30)

1   List all suppliers located in India.

2   Find suppliers where supplier_name contains 'Tech' and country is not 'USA'.

3   Show suppliers created after 2024-01-01 and before 2024-12-31 (inclusive).

4   Return suppliers where contact_person IS NULL OR email IS NULL.

5   Get suppliers whose phone_number starts with '+91' AND email ends with '.com'.

6   List warehouses in the state 'Maharashtra' OR city = 'Pune'.

7   Find warehouses in India AND capacity_units >= 10000.

8   Show warehouses where warehouse_name ILIKE '%central%' AND country = 'India'.

9   Return warehouses created in the last 30 days (use current_date) OR capacity_units < 1000.

10 Get warehouses where state IS NULL OR city IS NULL (missing location details).

11 List customers with email ending in '@gmail.com' AND country = 'India'.

12 Find customers whose city IN ('Mumbai', 'Pune', 'Delhi') OR phone_number LIKE '022%'.

13 Show customers created_at between two dates, e.g., 2025-01-01 AND 2025-06-30.

14 Return customers where customer_name ILIKE '%store%' OR ILIKE '%mart%'.

15 Get customers with missing phone_number OR invalid phone_number (NOT LIKE '+%').

16 List purchase orders where status = 'Pending' OR status = 'Approved'.

17 Find purchase orders for supplier_id = 5 AND total_amount > 100000.

18 Show purchase orders expected_date < po_date (late entry) OR expected_date IS NULL.

19 Return purchase orders in 2025 (po_date between 2025-01-01 and 2025-12-31).

20 Get purchase orders with total_amount BETWEEN 5000 AND 20000 AND status <> 'Cancelled'.

21 List PO items where quantity >= 100 AND unit_price <= 50.

22 Find PO items for a given po_id (e.g., 101) OR product_id IN (10, 20, 30).

23 Show PO items created_at IS NOT NULL AND quantity % 2 = 0 (even quantities).

24 Return PO items where unit_price * quantity > 5000 OR unit_price IS NULL.

25 Get PO items with quantity BETWEEN 1 AND 10 AND NOT (unit_price > 100).

26 List inventory rows where quantity_on_hand < reorder_level OR quantity_on_hand = 0.

27 Find inventory for warehouse_id = 3 AND product_id = 1001.

28 Show inventory last_updated > now() - interval '7 days' OR quantity_on_hand > 1000.

29 Return inventory with product_id IN (SELECT product_id FROM purchase_order_items) AND quantity_on_hand > 0.

30 Get inventory records where warehouse_id IN (SELECT warehouse_id FROM warehouses WHERE country = 'India') AND reorder_level >= 50.

31 List sales orders where status IN ('Pending','Shipped') AND total_amount > 20000.

32 Find sales order items for sales_order_id = 200 OR quantity >= 50.

33 Show shipments where shipment_status = 'Delivered' AND delivery_date BETWEEN '2025-01-01' AND '2025-12-31'.

Note: Use WHERE with AND/OR. You may also use operators like IN, BETWEEN, LIKE/ILIKE, IS NULL, NOT, and arithmetic expressions when appropriate.