**PBH Systems – Developer Assessment Assignment**

Here's a well-structured relational schema for a Warehouse Management System (WMS) with the core modules: Inventory, Orders, and Clients — followed by 3 sample SQL queries using JOINs and aggregations:

Clients (

client\_id INT PRIMARY KEY,

name VARCHAR(100),

email VARCHAR(100),

address TEXT

)

Products (

product\_id INT PRIMARY KEY,

product\_name VARCHAR(100),

sku VARCHAR(50),

category VARCHAR(50)

)

Inventory (

inventory\_id INT PRIMARY KEY,

product\_id INT,

quantity INT,

location VARCHAR(50),

updated\_at DATETIME,

FOREIGN KEY (product\_id) REFERENCES Products(product\_id)

)

Orders (

order\_id INT PRIMARY KEY,

client\_id INT,

order\_date DATE,

status VARCHAR(50),

FOREIGN KEY (client\_id) REFERENCES Clients(client\_id)

)

Order\_Items (

order\_item\_id INT PRIMARY KEY,

order\_id INT,

product\_id INT,

quantity INT,

FOREIGN KEY (order\_id) REFERENCES Orders(order\_id),

FOREIGN KEY (product\_id) REFERENCES Products(product\_id)

)

**Sample SQL Queries:**

**1. Get all orders with client name and total quantity of items ordered**

SELECT

o.order\_id,

c.name AS client\_name,

SUM(oi.quantity) AS total\_items

FROM Orders o

JOIN Clients c ON o.client\_id = c.client\_id

JOIN Order\_Items oi ON o.order\_id = oi.order\_id

GROUP BY o.order\_id, c.name;

**2. List products with available quantity in each inventory location**

CopyEdit

SELECT

p.product\_name,

i.location,

i.quantity

FROM Inventory i

JOIN Products p ON i.product\_id = p.product\_id

ORDER BY p.product\_name, i.location;

**3. Find top 3 most ordered products**

CopyEdit

SELECT

p.product\_name,

SUM(oi.quantity) AS total\_ordered

FROM Order\_Items oi

JOIN Products p ON oi.product\_id = p.product\_id

GROUP BY p.product\_id, p.product\_name

ORDER BY total\_ordered DESC

LIMIT 3;