```sql
CREATE TABLE users (
  user_id INT AUTO_INCREMENT PRIMARY KEY,
  username VARCHAR(50) UNIQUE NOT NULL,
  full_name VARCHAR(100),
  role ENUM('admin','manager','staff') DEFAULT 'staff',
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

CREATE TABLE categories (
  category_id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  description TEXT
);

CREATE TABLE products (
  product_id INT AUTO_INCREMENT PRIMARY KEY,
  sku VARCHAR(50) UNIQUE NOT NULL,
  name VARCHAR(200) NOT NULL,
  category_id INT,
  unit VARCHAR(30),
  cost_price DECIMAL(12,2),
  selling_price DECIMAL(12,2),
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  FOREIGN KEY (category_id) REFERENCES categories(category_id)
);

CREATE TABLE suppliers (
  supplier_id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(200) NOT NULL,
  contact_name VARCHAR(100),
  phone VARCHAR(30),
  email VARCHAR(100),
  address TEXT
);

CREATE TABLE warehouses (
  warehouse_id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  location VARCHAR(200),
  phone VARCHAR(30)
);

CREATE TABLE inventory (
  inventory_id INT AUTO_INCREMENT PRIMARY KEY,
  product_id INT NOT NULL,
  warehouse_id INT NOT NULL,
  quantity DECIMAL(14,3) DEFAULT 0,
  reorder_point DECIMAL(14,3) DEFAULT 0,
  last_updated TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  UNIQUE KEY uq_inventory_product_warehouse (product_id, warehouse_id),
  FOREIGN KEY (product_id) REFERENCES products(product_id),
  FOREIGN KEY (warehouse_id) REFERENCES warehouses(warehouse_id)
);

CREATE TABLE purchase_orders (
  po_id INT AUTO_INCREMENT PRIMARY KEY,
  po_number VARCHAR(50) UNIQUE NOT NULL,
  supplier_id INT NOT NULL,
  order_date DATE NOT NULL,
  expected_date DATE,
  status ENUM('pending','received','partially_received','cancelled') DEFAULT 'pending',
  total_amount DECIMAL(14,2) DEFAULT 0,
  created_by INT,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  FOREIGN KEY (supplier_id) REFERENCES suppliers(supplier_id),
  FOREIGN KEY (created_by) REFERENCES users(user_id)
);

CREATE TABLE purchase_lines (
  po_line_id INT AUTO_INCREMENT PRIMARY KEY,
  po_id INT NOT NULL,
  product_id INT NOT NULL,
```

```sql
    quantity_ordered DECIMAL(14,3) NOT NULL,
    quantity_received DECIMAL(14,3) DEFAULT 0,
    unit_price DECIMAL(12,2) NOT NULL,
    line_total DECIMAL(14,2) AS (quantity_ordered * unit_price) STORED,
    FOREIGN KEY (po_id) REFERENCES purchase_orders(po_id) ON DELETE CASCADE,
    FOREIGN KEY (product_id) REFERENCES products(product_id)
);

CREATE TABLE sales_orders (
    so_id INT AUTO_INCREMENT PRIMARY KEY,
    so_number VARCHAR(50) UNIQUE NOT NULL,
    customer_name VARCHAR(200),
    order_date DATE NOT NULL,
    status ENUM('pending','shipped','completed','cancelled') DEFAULT 'pending',
    total_amount DECIMAL(14,2) DEFAULT 0,
    created_by INT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (created_by) REFERENCES users(user_id)
);

CREATE TABLE sales_lines (
    so_line_id INT AUTO_INCREMENT PRIMARY KEY,
    so_id INT NOT NULL,
    product_id INT NOT NULL,
    warehouse_id INT NOT NULL,
    quantity DECIMAL(14,3) NOT NULL,
    unit_price DECIMAL(12,2) NOT NULL,
    line_total DECIMAL(14,2) AS (quantity * unit_price) STORED,
    FOREIGN KEY (so_id) REFERENCES sales_orders(so_id) ON DELETE CASCADE,
    FOREIGN KEY (product_id) REFERENCES products(product_id),
    FOREIGN KEY (warehouse_id) REFERENCES warehouses(warehouse_id)
);

CREATE TABLE stock_movements (
    movement_id INT AUTO_INCREMENT PRIMARY KEY,
    product_id INT NOT NULL,
    warehouse_id INT NOT NULL,
    change DECIMAL(14,3) NOT NULL,
    movement_type ENUM('purchase','sale','adjustment','transfer_in','transfer_out') NOT NULL,
    reference_id INT,
    reference_line_id INT,
    reason TEXT,
    moved_by INT,
    moved_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (product_id) REFERENCES products(product_id),
    FOREIGN KEY (warehouse_id) REFERENCES warehouses(warehouse_id),
    FOREIGN KEY (moved_by) REFERENCES users(user_id)
);

CREATE TABLE supplier_products (
    supplier_product_id INT AUTO_INCREMENT PRIMARY KEY,
    supplier_id INT NOT NULL,
    product_id INT NOT NULL,
    supplier_sku VARCHAR(100),
    price DECIMAL(12,2),
    lead_time_days INT,
    FOREIGN KEY (supplier_id) REFERENCES suppliers(supplier_id),
    FOREIGN KEY (product_id) REFERENCES products(product_id)
);

INSERT INTO users (username,full_name,role) VALUES
('admin','Admin User','admin'),
('store1','Store Staff','staff');

INSERT INTO categories (name,description) VALUES
('Electronics','Electronic items'),
('Stationery','Office supplies');

INSERT INTO products (sku,name,category_id,unit,cost_price,selling_price) VALUES
('SKU-1001','USB Cable 1m',1,'pcs',50.00,120.00),
('SKU-2001','A4 Notebook',2,'pcs',20.00,45.00);
INSERT INTO suppliers (name,contact_name,phone,email,address) VALUES
```

```sql
('Fast Supplies Pvt Ltd','Ramesh','9999999999','ramesh@fastsup.com','Mumbai'),
('PaperWorld','Suresh','8888888888','sales@paperworld.com','Delhi');

INSERT INTO warehouses (name,location,phone) VALUES
('Main Warehouse','Pune','1234567890'),
('Outlet Store','Pune MG Road','0987654321');

INSERT INTO inventory (product_id,warehouse_id,quantity,reorder_point) VALUES
(1,1,100,20),
(1,2,20,10),
(2,1,300,50);
```