

SQL Schema Sistem Inventory Walet

```
CREATE TABLE roles (  
    id BIGSERIAL PRIMARY KEY,  
    code VARCHAR(50) NOT NULL UNIQUE,  
    name VARCHAR(100) NOT NULL,  
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()  
);  
  
CREATE TABLE users (  
    id BIGSERIAL PRIMARY KEY,  
    role_id BIGINT NOT NULL REFERENCES roles(id),  
    name VARCHAR(150) NOT NULL,  
    email VARCHAR(150),  
    phone VARCHAR(50),  
    status VARCHAR(20) NOT NULL DEFAULT 'ACTIVE',  
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()  
);  
  
CREATE TABLE suppliers (  
    id BIGSERIAL PRIMARY KEY,  
    code VARCHAR(50) NOT NULL UNIQUE,  
    name VARCHAR(150) NOT NULL,  
    phone VARCHAR(50),  
    email VARCHAR(150),  
    bank_name VARCHAR(100),  
    bank_account_number VARCHAR(100),  
    address TEXT,  
    status VARCHAR(20) NOT NULL DEFAULT 'ACTIVE',  
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()  
);  
  
CREATE TABLE customers (  
    id BIGSERIAL PRIMARY KEY,  
    code VARCHAR(50) NOT NULL UNIQUE,  
    name VARCHAR(150) NOT NULL,  
    phone VARCHAR(50),  
    email VARCHAR(150),  
    bank_name VARCHAR(100),  
    bank_account_number VARCHAR(100),  
    address TEXT,  
    status VARCHAR(20) NOT NULL DEFAULT 'ACTIVE',  
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()  
);  
  
CREATE TABLE units (  
    id BIGSERIAL PRIMARY KEY,  
    code VARCHAR(20) NOT NULL UNIQUE,  
    name VARCHAR(50) NOT NULL,  
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()  
);  
  
CREATE TABLE item_types (  
    id BIGSERIAL PRIMARY KEY,  
    code VARCHAR(50) NOT NULL UNIQUE,  
    name VARCHAR(100) NOT NULL,  
    description TEXT,  
    status VARCHAR(20) NOT NULL DEFAULT 'ACTIVE',  
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()
```

```
);
```

```
CREATE TABLE item_grades (  
    id BIGSERIAL PRIMARY KEY,  
    code VARCHAR(50) NOT NULL UNIQUE,  
    name VARCHAR(100) NOT NULL,  
    rank_order INT,  
    description TEXT,  
    status VARCHAR(20) NOT NULL DEFAULT 'ACTIVE',  
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()  
);
```

```
CREATE TABLE warehouses (  
    id BIGSERIAL PRIMARY KEY,  
    code VARCHAR(50) NOT NULL UNIQUE,  
    name VARCHAR(100) NOT NULL,  
    address TEXT,  
    status VARCHAR(20) NOT NULL DEFAULT 'ACTIVE',  
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()  
);
```

```
CREATE TABLE warehouse_locations (  
    id BIGSERIAL PRIMARY KEY,  
    warehouse_id BIGINT NOT NULL REFERENCES warehouses(id),  
    code VARCHAR(50) NOT NULL,  
    name VARCHAR(100) NOT NULL,  
    location_type VARCHAR(50),  
    status VARCHAR(20) NOT NULL DEFAULT 'ACTIVE',  
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    UNIQUE (warehouse_id, code)  
);
```

```
CREATE TABLE adjustment_reasons (  
    id BIGSERIAL PRIMARY KEY,  
    code VARCHAR(50) NOT NULL UNIQUE,  
    name VARCHAR(100) NOT NULL,  
    category VARCHAR(50) NOT NULL,  
    status VARCHAR(20) NOT NULL DEFAULT 'ACTIVE',  
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()  
);
```

```
CREATE TABLE purchases (  
    id BIGSERIAL PRIMARY KEY,  
    purchase_no VARCHAR(50) NOT NULL UNIQUE,  
    supplier_id BIGINT NOT NULL REFERENCES suppliers(id),  
    purchase_date DATE NOT NULL,  
    supplier_invoice_no VARCHAR(100),  
    status VARCHAR(20) NOT NULL DEFAULT 'DRAFT',  
    notes TEXT,  
    created_by BIGINT NOT NULL REFERENCES users(id),  
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()  
);
```

```
CREATE TABLE purchase_lines (  
    id BIGSERIAL PRIMARY KEY,  
    purchase_id BIGINT NOT NULL REFERENCES purchases(id) ON DELETE CASCADE,  
    item_type_id BIGINT NOT NULL REFERENCES item_types(id),  
    item_grade_id BIGINT REFERENCES item_grades(id),  
    qty_ordered NUMERIC(18,3) NOT NULL,  
    unit_id BIGINT NOT NULL REFERENCES units(id),  
    unit_price NUMERIC(18,2) NOT NULL,  
    subtotal NUMERIC(18,2) NOT NULL,
```

```

classification_status VARCHAR(20) NOT NULL DEFAULT 'FINAL',
notes TEXT,
created_at TIMESTAMP NOT NULL DEFAULT NOW(),
updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

```

```

CREATE TABLE receipts (
  id BIGSERIAL PRIMARY KEY,
  receipt_no VARCHAR(50) NOT NULL UNIQUE,
  purchase_id BIGINT NOT NULL REFERENCES purchases(id),
  supplier_id BIGINT NOT NULL REFERENCES suppliers(id),
  receipt_date DATE NOT NULL,
  status VARCHAR(20) NOT NULL DEFAULT 'DRAFT',
  notes TEXT,
  received_by BIGINT NOT NULL REFERENCES users(id),
  created_at TIMESTAMP NOT NULL DEFAULT NOW(),
  updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

```

```

CREATE TABLE receipt_lines (
  id BIGSERIAL PRIMARY KEY,
  receipt_id BIGINT NOT NULL REFERENCES receipts(id) ON DELETE CASCADE,
  purchase_line_id BIGINT NOT NULL REFERENCES purchase_lines(id),
  item_type_id BIGINT NOT NULL REFERENCES item_types(id),
  item_grade_id BIGINT REFERENCES item_grades(id),
  qty_received NUMERIC(18,3) NOT NULL,
  qty_accepted NUMERIC(18,3) NOT NULL,
  qty_rejected NUMERIC(18,3) NOT NULL DEFAULT 0,
  unit_id BIGINT NOT NULL REFERENCES units(id),
  unit_cost NUMERIC(18,2) NOT NULL,
  notes TEXT,
  created_at TIMESTAMP NOT NULL DEFAULT NOW(),
  updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

```

```

CREATE TABLE inventory_lots (
  id BIGSERIAL PRIMARY KEY,
  lot_code VARCHAR(100) NOT NULL UNIQUE,
  parent_lot_id BIGINT REFERENCES inventory_lots(id),
  source_type VARCHAR(30) NOT NULL,
  source_ref_id BIGINT,
  supplier_id BIGINT REFERENCES suppliers(id),
  purchase_id BIGINT REFERENCES purchases(id),
  purchase_line_id BIGINT REFERENCES purchase_lines(id),
  receipt_id BIGINT REFERENCES receipts(id),
  receipt_line_id BIGINT REFERENCES receipt_lines(id),
  item_type_id BIGINT NOT NULL REFERENCES item_types(id),
  item_grade_id BIGINT NOT NULL REFERENCES item_grades(id),
  warehouse_id BIGINT NOT NULL REFERENCES warehouses(id),
  warehouse_location_id BIGINT REFERENCES warehouse_locations(id),
  original_qty NUMERIC(18,3) NOT NULL,
  available_qty NUMERIC(18,3) NOT NULL,
  reserved_qty NUMERIC(18,3) NOT NULL DEFAULT 0,
  damaged_qty NUMERIC(18,3) NOT NULL DEFAULT 0,
  shrinkage_qty NUMERIC(18,3) NOT NULL DEFAULT 0,
  unit_id BIGINT NOT NULL REFERENCES units(id),
  unit_cost NUMERIC(18,2) NOT NULL,
  received_at TIMESTAMP NOT NULL,
  status VARCHAR(20) NOT NULL DEFAULT 'ACTIVE',
  qr_code_value VARCHAR(255),
  barcode_value VARCHAR(255),
  notes TEXT,
  created_at TIMESTAMP NOT NULL DEFAULT NOW(),
  updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

```

```

CREATE INDEX idx_inventory_lots_item_grade_wh ON inventory_lots(item_type_id, item_grade_id,

```

```

warehouse_id);
CREATE INDEX idx_inventory_lots_supplier_item_grade ON inventory_lots(supplier_id, item_type_id,
item_grade_id);

CREATE TABLE sorting_sessions (
  id BIGSERIAL PRIMARY KEY,
  sorting_no VARCHAR(50) NOT NULL UNIQUE,
  source_lot_id BIGINT NOT NULL REFERENCES inventory_lots(id),
  sorting_date TIMESTAMP NOT NULL,
  input_qty NUMERIC(18,3) NOT NULL,
  output_qty NUMERIC(18,3) NOT NULL,
  shrinkage_qty NUMERIC(18,3) NOT NULL DEFAULT 0,
  notes TEXT,
  sorted_by BIGINT NOT NULL REFERENCES users(id),
  created_at TIMESTAMP NOT NULL DEFAULT NOW(),
  updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

CREATE TABLE sorting_results (
  id BIGSERIAL PRIMARY KEY,
  sorting_session_id BIGINT NOT NULL REFERENCES sorting_sessions(id) ON DELETE CASCADE,
  result_lot_id BIGINT NOT NULL REFERENCES inventory_lots(id),
  item_type_id BIGINT NOT NULL REFERENCES item_types(id),
  item_grade_id BIGINT NOT NULL REFERENCES item_grades(id),
  qty_result NUMERIC(18,3) NOT NULL,
  unit_cost NUMERIC(18,2) NOT NULL,
  notes TEXT,
  created_at TIMESTAMP NOT NULL DEFAULT NOW(),
  updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

CREATE TABLE sales (
  id BIGSERIAL PRIMARY KEY,
  sales_no VARCHAR(50) NOT NULL UNIQUE,
  customer_id BIGINT NOT NULL REFERENCES customers(id),
  sales_date DATE NOT NULL,
  status VARCHAR(20) NOT NULL DEFAULT 'DRAFT',
  notes TEXT,
  created_by BIGINT NOT NULL REFERENCES users(id),
  created_at TIMESTAMP NOT NULL DEFAULT NOW(),
  updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

CREATE TABLE sales_lines (
  id BIGSERIAL PRIMARY KEY,
  sales_id BIGINT NOT NULL REFERENCES sales(id) ON DELETE CASCADE,
  item_type_id BIGINT NOT NULL REFERENCES item_types(id),
  item_grade_id BIGINT NOT NULL REFERENCES item_grades(id),
  qty_sold NUMERIC(18,3) NOT NULL,
  unit_id BIGINT NOT NULL REFERENCES units(id),
  selling_price NUMERIC(18,2) NOT NULL,
  subtotal NUMERIC(18,2) NOT NULL,
  costing_total NUMERIC(18,2) NOT NULL DEFAULT 0,
  gross_margin NUMERIC(18,2) NOT NULL DEFAULT 0,
  notes TEXT,
  created_at TIMESTAMP NOT NULL DEFAULT NOW(),
  updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

CREATE TABLE sales_allocations (
  id BIGSERIAL PRIMARY KEY,
  sales_line_id BIGINT NOT NULL REFERENCES sales_lines(id) ON DELETE CASCADE,
  inventory_lot_id BIGINT NOT NULL REFERENCES inventory_lots(id),
  qty_allocated NUMERIC(18,3) NOT NULL,
  unit_cost NUMERIC(18,2) NOT NULL,
  total_cost NUMERIC(18,2) NOT NULL,
  allocated_at TIMESTAMP NOT NULL,

```

```

    allocated_by BIGINT NOT NULL REFERENCES users(id),
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

```

```

CREATE TABLE inventory_movements (
    id BIGSERIAL PRIMARY KEY,
    movement_no VARCHAR(50) NOT NULL UNIQUE,
    movement_type VARCHAR(30) NOT NULL,
    inventory_lot_id BIGINT NOT NULL REFERENCES inventory_lots(id),
    related_lot_id BIGINT REFERENCES inventory_lots(id),
    warehouse_id BIGINT NOT NULL REFERENCES warehouses(id),
    warehouse_location_id BIGINT REFERENCES warehouse_locations(id),
    qty_in NUMERIC(18,3) NOT NULL DEFAULT 0,
    qty_out NUMERIC(18,3) NOT NULL DEFAULT 0,
    balance_after NUMERIC(18,3) NOT NULL,
    unit_cost NUMERIC(18,2) NOT NULL,
    reference_type VARCHAR(30) NOT NULL,
    reference_id BIGINT NOT NULL,
    reason_id BIGINT REFERENCES adjustment_reasons(id),
    movement_date TIMESTAMP NOT NULL,
    created_by BIGINT NOT NULL REFERENCES users(id),
    notes TEXT,
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

```

```

CREATE TABLE sales_returns (
    id BIGSERIAL PRIMARY KEY,
    sales_id BIGINT NOT NULL REFERENCES sales(id),
    customer_id BIGINT NOT NULL REFERENCES customers(id),
    return_date DATE NOT NULL,
    status VARCHAR(20) NOT NULL DEFAULT 'DRAFT',
    notes TEXT,
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

```

```

CREATE TABLE sales_return_lines (
    id BIGSERIAL PRIMARY KEY,
    sales_return_id BIGINT NOT NULL REFERENCES sales_returns(id) ON DELETE CASCADE,
    sales_line_id BIGINT NOT NULL REFERENCES sales_lines(id),
    inventory_lot_id BIGINT REFERENCES inventory_lots(id),
    item_type_id BIGINT NOT NULL REFERENCES item_types(id),
    item_grade_id BIGINT NOT NULL REFERENCES item_grades(id),
    qty_returned NUMERIC(18,3) NOT NULL,
    return_condition VARCHAR(50),
    resolution VARCHAR(50),
    notes TEXT,
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

```

```

CREATE TABLE purchase_returns (
    id BIGSERIAL PRIMARY KEY,
    purchase_id BIGINT NOT NULL REFERENCES purchases(id),
    supplier_id BIGINT NOT NULL REFERENCES suppliers(id),
    return_date DATE NOT NULL,
    status VARCHAR(20) NOT NULL DEFAULT 'DRAFT',
    notes TEXT,
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

```

```

CREATE TABLE purchase_return_lines (
    id BIGSERIAL PRIMARY KEY,
    purchase_return_id BIGINT NOT NULL REFERENCES purchase_returns(id) ON DELETE CASCADE,

```

```
inventory_lot_id BIGINT NOT NULL REFERENCES inventory_lots(id),
qty_returned NUMERIC(18,3) NOT NULL,
unit_cost NUMERIC(18,2) NOT NULL,
notes TEXT,
created_at TIMESTAMP NOT NULL DEFAULT NOW(),
updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);
```

```
CREATE TABLE stock_adjustments (
  id BIGSERIAL PRIMARY KEY,
  adjustment_no VARCHAR(50) NOT NULL UNIQUE,
  inventory_lot_id BIGINT NOT NULL REFERENCES inventory_lots(id),
  adjustment_type VARCHAR(30) NOT NULL,
  reason_id BIGINT NOT NULL REFERENCES adjustment_reasons(id),
  qty_before NUMERIC(18,3) NOT NULL,
  qty_change NUMERIC(18,3) NOT NULL,
  qty_after NUMERIC(18,3) NOT NULL,
  cost_impact NUMERIC(18,2) NOT NULL DEFAULT 0,
  adjustment_date TIMESTAMP NOT NULL,
  notes TEXT,
  created_by BIGINT NOT NULL REFERENCES users(id),
  created_at TIMESTAMP NOT NULL DEFAULT NOW(),
  updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);
```

```
CREATE TABLE lot_labels (
  id BIGSERIAL PRIMARY KEY,
  inventory_lot_id BIGINT NOT NULL REFERENCES inventory_lots(id) ON DELETE CASCADE,
  label_type VARCHAR(20) NOT NULL,
  label_value VARCHAR(255) NOT NULL,
  printed_at TIMESTAMP,
  printed_by BIGINT REFERENCES users(id),
  print_count INT NOT NULL DEFAULT 0,
  status VARCHAR(20) NOT NULL DEFAULT 'ACTIVE',
  created_at TIMESTAMP NOT NULL DEFAULT NOW(),
  updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);
```