Shoe Store Management System database

Step 01: Entity Sets for Shoe Store Management System:

1. Product (Shoes)
2. Category
3. Customer
4. Employee
5. Supplier
6. Inventory
7. Sale
8. Purchase_Order

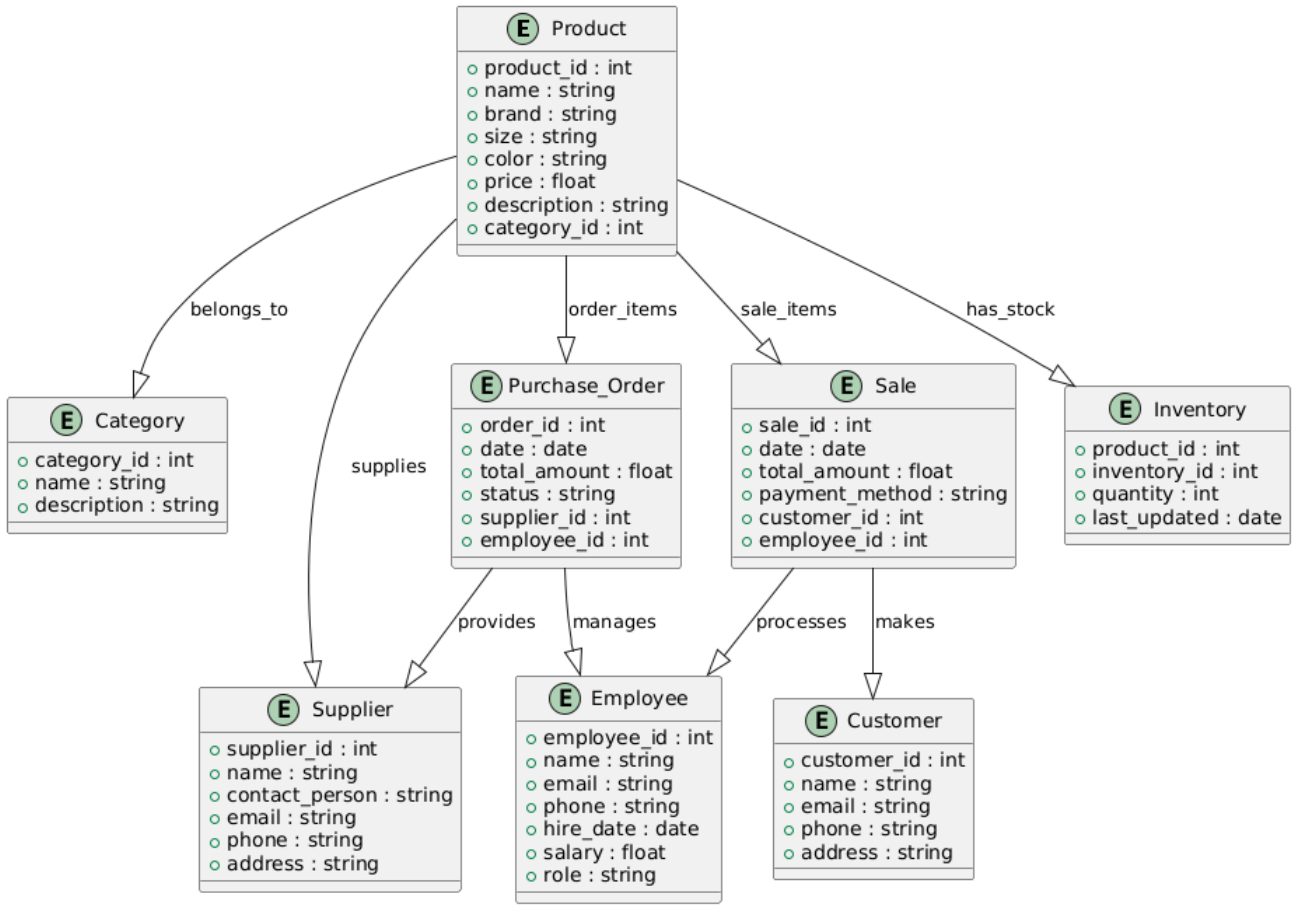
Step 02: Properties/attributes of each entity set:

1. Product(product_id, name, brand, size, color, price, description)
2. Category(category_id, name, description)
3. Customer(customer_id, name, email, phone, address)
4. Employee(employee_id, name, email, phone, hire_date, salary, role)
5. Supplier(supplier_id, name, contact_person, email, phone, address)
6. Inventory(inventory_id, quantity, last_updated)
7. Sale(sale_id, date, total_amount, payment_method)
8. Purchase_Order(order_id, date, total_amount, status)

Step 03: Relations between entity sets:

1. Product-Category (belongs_to): Many-to-One
2. Product-Inventory (has_stock): One-to-One
3. Product-Sale (sale_items): Many-to-Many
4. Product-Purchase_Order (order_items): Many-to-Many
5. Product-Supplier (supplies): Many-to-Many
6. Sale-Customer (makes): Many-to-One
7. Sale-Employee (processes): Many-to-One
8. Purchase_Order-Supplier (provides): Many-to-One
9. Purchase_Order-Employee (manages): Many-to-One

Step 04: E-R diagram of these Entity sets and Relationship Sets is given below:



Step 05: Conversion of the diagram into tables by using the reduction rule.

(a) Entity set 'Category' is connected with Product via relationship set belongs_to. Category side is one. So Category entity set directly converted to Category table with the same attributes.

- `Category(category_id, name, description)`

(b) The entity sets 'Customer', 'Employee', and 'Supplier' are also converted to their respective tables for similar reasons.

- `Customer(customer_id, name, email, phone, address)`
- `Employee(employee_id, name, email, phone, hire_date, salary, role)`
- `Supplier(supplier_id, name, contact_person, email, phone, address)`

(c) Relationship set 'Product_Supplier' is many-many. So create a table for Product_Supplier. The attributes will be the primary keys from the corresponding entity sets and descriptive attribute supply_price.

- `Product_Supplier(product_id, supplier_id, supply_price)`

(d) 'Inventory' is a weak entity set related to Product. So the primary key of the identifying strong entity set (Product) will be part of the weak entity set.

- `Inventory(product_id, inventory_id, quantity, last_updated)`

(e) Relationship sets 'belongs_to', 'makes', 'processes', 'provides', and 'manages' are one-many. So these relationship sets need not be converted to tables. Rather, the entity sets of the many side are converted to tables by adding the primary key of the one side

- Product(product_id, name, brand, size, color, price, description, category_id)

- Sale(sale_id, date, total_amount, payment_method, customer_id, employee_id)

- Purchase_Order(order_id, date, total_amount, status, supplier_id, employee_id)

(f) Relationship sets 'sale_items' and 'order_items' are many-many. So create two tables for these relationships. The attributes will be the primary keys from the corresponding entity sets and descriptive attributes.

- Sale_Items(sale_id, product_id, quantity, unit_price, subtotal)
- Order_Items(order_id, product_id, quantity, unit_price, subtotal)