The goal is to design an Entity-Relationship Diagram (ERD) for a database that captures information related to pet products, their manufacturers, customers, transactions, and shipments to various Walmart locations.

Below is a step-by-step breakdown of the design:

Step 1: Identifying Entities

Based on the requirements, we can identify the following entities:

1. **Product**: Represents general information about a product.
2. **PetFood**: A specialized type of Product with specific attributes.
3. **PetToy**: A specialized type of Product with specific attributes.
4. **PetApparel**: A specialized type of Product with specific attributes.
5. **Animal**: Represents different types of animals associated with products.
6. **Manufacturer**: Represents manufacturers associated with products.
7. **Customer**: Represents customers who purchase products.
8. **Transaction**: Represents a purchase made by a customer.
9. **Location**: Represents Walmart store locations.
10. **Shipment**: Represents a shipment from one location to another.

Step 2: Define Relationships Between Entities

- **Product and Manufacturer:** Each product is associated with one manufacturer. (**Many-to-One**)
- **Product and Animal:** Each product can be associated with one or more animals. (**Many-to-Many**)
- **Customer and Transaction:** Each customer can make multiple transactions. (**One-to-Many**)
- **Transaction and Product:** Each transaction can include multiple products, and each product can be part of multiple transactions. (**Many-to-Many**)
- **Location and Shipment**: Each shipment involves an origin and a destination location. (**One-to-Many for both origin and destination**)
- **Shipment and Product**: Each shipment includes a collection of products, each with an associated quantity. (**Many-to-Many**)

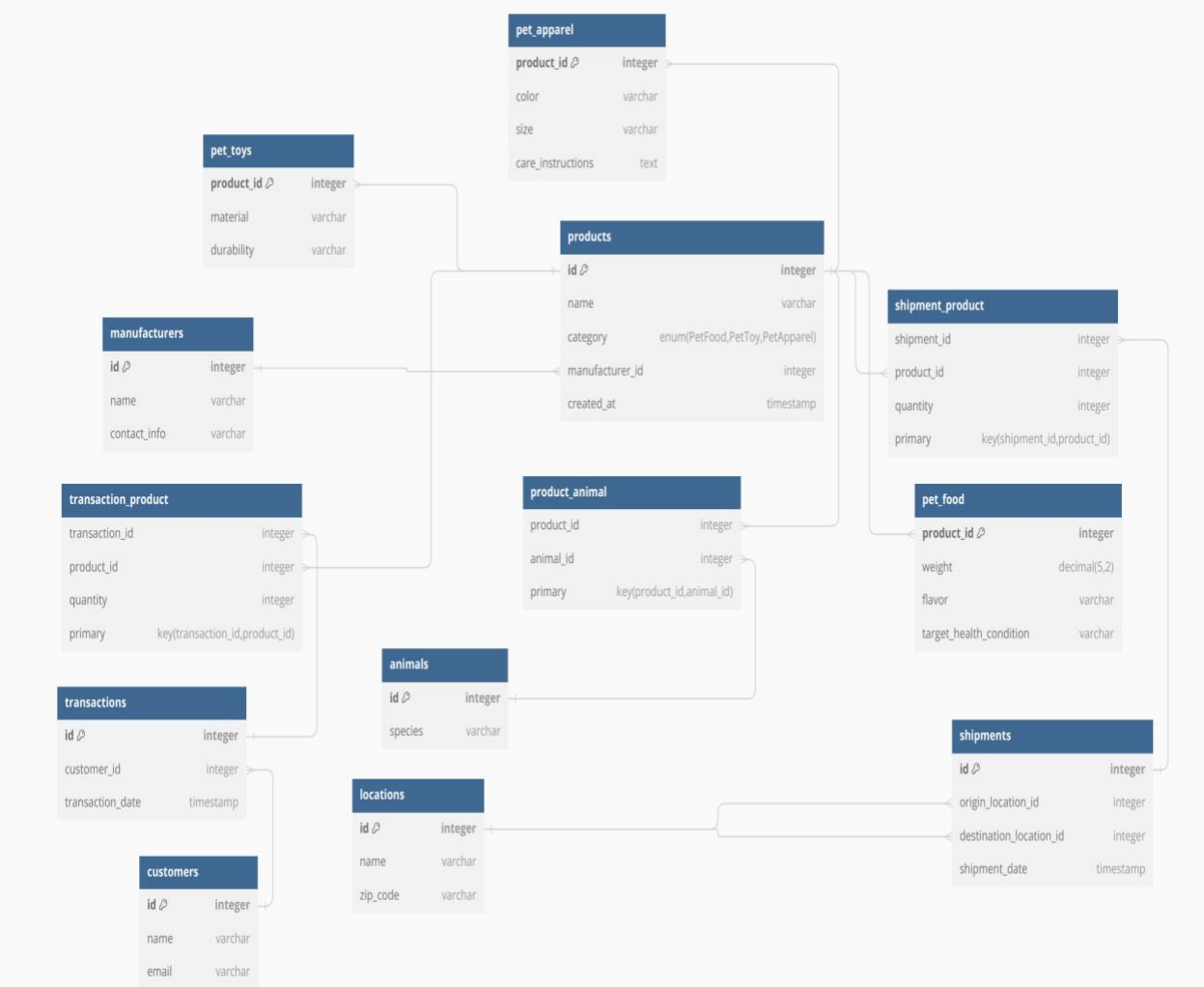
Step 3: Define Attributes for Each Entity

1. Product
    - ProductID (Primary Key)
    - Name
    - Category (e.g., PetFood, PetToy, PetApparel)
    - ManufacturerID (Foreign Key to Manufacturer)
2. PetFood (inherits from Product)
    - Weight
    - Flavor

    - TargetHealthCondition
3. PetToy (inherits from Product)
    - Material
    - Durability
4. PetApparel (inherits from Product)
    - Color
    - Size
    - CareInstructions
5. Animal
    - AnimalID (Primary Key)
    - Species
6. Manufacturer
    - ManufacturerID (Primary Key)
    - Name
    - ContactInfo
7. Customer
    - CustomerID (Primary Key)
    - Name
    - Email
8. Transaction
    - TransactionID (Primary Key)
    - CustomerID (Foreign Key to Customer)
    - TransactionDate
9. Location
    - LocationID (Primary Key)
    - Name
    - ZipCode
10. Shipment
    - ShipmentID (Primary Key)
    - OriginLocationID (Foreign Key to Location)
    - DestinationLocationID (Foreign Key to Location)
    - ShipmentDate

Step 4: Implement Join Tables for Many-to-Many Relationships

1. Product_Animal
    - ProductID (Foreign Key to Product)
    - AnimalID (Foreign Key to Animal)
2. Transaction_Product
    - TransactionID (Foreign Key to Transaction)
    - ProductID (Foreign Key to Product)
    - Quantity
3. Shipment_Product
    - ShipmentID (Foreign Key to Shipment)
    - ProductID (Foreign Key to Product)
    - Quantity


pet_apparel
product_id integer
color varchar
size varchar
care_instructions text
pet_toys
product_id integer
material varchar
durability varchar
products
id integer
name varchar
category enum(PetFood,PetToy,PetApparel)
manufacturer_id integer
created_at timestamp
shipment_product
shipment_id integer
product_id integer
quantity integer
primary key(shipment_id,product_id)
manufacturers
id integer
name varchar
contact_info varchar
transaction_product
transaction_id integer
product_id integer
quantity integer
primary key(transaction_id,product_id)
product_animal
product_id integer
animal_id integer
primary key(product_id,animal_id)
pet_food
product_id integer
weight decimal(5,2)
flavor varchar
target_health_condition varchar
animals
id integer
species varchar
transactions
id integer
customer_id integer
transaction_date timestamp
shipments
id integer
origin_location_id integer
destination_location_id integer
shipment_date timestamp
locations
id integer
name varchar
zip_code varchar
customers
id integer
name varchar
email varchar