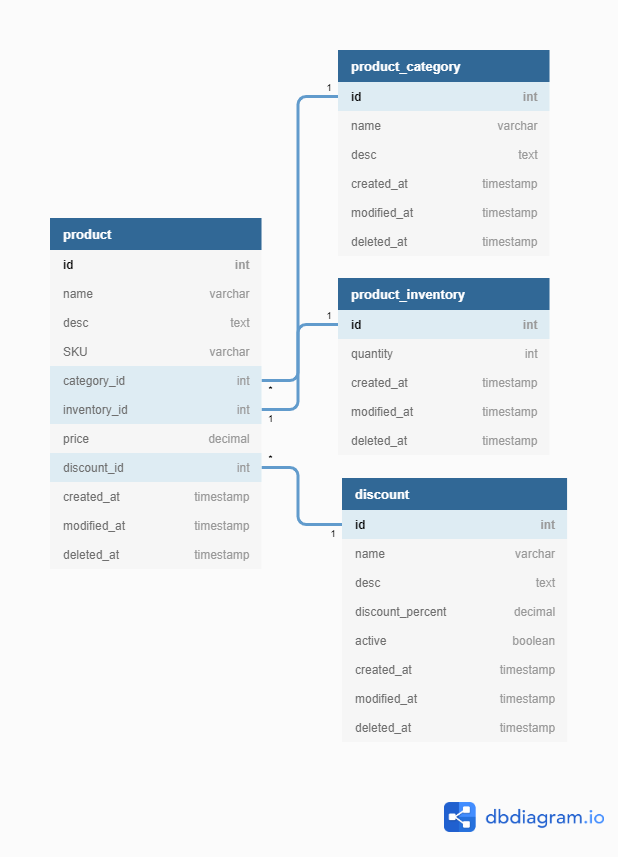
1. Explain the relationship between the “product” and product - Category entities from the below diagram.



SOLUTION:

* Entity Relationship: The "Product" and "Product - Category" entities are interconnected in a one-to-many relationship, where a single product category can accommodate multiple products while each product belongs to only one category.
* Database Structure: The relationship is facilitated by a foreign key constraint within the "Product" table, typically named "category - id", referencing the primary key of the "Product - Category" table. This ensures that each product is associated with a specific category.
* Normalization Benefits: The relationship aids in maintaining a normalized database schema by segregating product categories into a separate entity. This prevents redundancy and ensures data consistency throughout the database.
* Query Efficiency: Through this relationship, querying the database becomes more streamlined. It allows for easy retrieval of products belonging to a particular category, thereby enhancing query performance and database efficiency.
* Data Integrity Maintenance: Establishing this relationship helps uphold data integrity by enabling seamless operations such as updating or deleting categories without causing cascading effects on associated products. This enhances the robustness of the database management system.

2.How could you ensure that each product in the “product” table has a valid category assigned to it.

Solution:

* Foreign Key Constraint : Establish a foreign key constraint within the "Product" table, linking it to the primary key of the "Product - Category" table. This constraint forms a connection between the "Product" and "Product-Category" entities.
* Category ID Field in Product Table : Confirm that the "Product" table includes a field (e.g., "category-id") to hold the identification of the product category associated with each product.
* Ensure Referential Integrity : Set up the foreign key constraint to maintain referential integrity. This ensures that every value in the "category-id" field of the "Product" table corresponds to an existing value in the primary key column of the "Product - Category" table.
* Disallow NULL Values : Optionally, configure the foreign key constraint to prevent NULL values in the "category-id" field of the "Product" table. This guarantees that all products are assigned to a valid category, enhancing data accuracy and consistency.

Certainly! Here's the modified statement for implementing the validation of categories in SQL:

CREATE OR REPLACE FUNCTION check\_category\_exists()

RETURNS TRIGGER AS $$

BEGIN

IF NOT EXISTS (SELECT 1 FROM Product\_Category WHERE category\_id = NEW.category\_id) THEN

RAISE EXCEPTION 'Invalid category assigned to the product';

END IF;

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER before\_insert\_update\_product

BEFORE INSERT OR UPDATE ON Product

FOR EACH ROW

EXECUTE FUNCTION check\_category\_exists();