# Ecommerce Database Management System for shop, witch sells Skin care products.

1.Introduction

## Team

This project was made by Manap Zhassulan.

Project Description

This project includes the development of a database management system for an online skin care store. My main goal was to create a relational database solution that could effectively manage various e-commerce entities.The system will also include various procedures, functions, and triggers to enhance the functionality of the system. This document provides a comprehensive overview of our project, which includes an introduction, an Object Relationship Diagram (ERD), a relational schema, a detailed explanation of structure normal forms, and an overview of the SQL implementation.

### 2. Relational Database Schema

3. ERD

### 4.Tables

### Table 1. Entities and their Attributes

|  |  |
| --- | --- |
| **Table Name** | **Attributes** |
| Customers | c\_id, F\_name, L\_name, Email, Phone |
| Address | Address\_id, Unit\_num, Street\_num, Address\_name, City, Region, Postal\_code |
| Customer\_address | Customer\_id, Address\_id, Is\_default |
| Product\_size | Size\_id, Product\_Size |
| Ingredients | Ingredients\_id, Name |
| Skin\_concern | Skin\_concern\_id, Skin\_concern\_Type |
| Brand | Brand\_id, Name |
| Category | Category\_id, Name |
| Products | Product\_id, Product\_name, Description, Category\_id |
| Product\_entry | Product\_entry\_id, Product\_id, Size\_id, Ingredients\_id, Skin\_concern\_id, Brand\_id, Quantity\_stock |
| Shopping\_cart | Cart\_id, Product\_entry\_id, Quantity, Customer\_id |
| Payment\_type | Payment\_type\_id, Payment\_name |
| Payment\_method | Payment\_id, Customer\_id, Payment\_type\_id, Provider, Account\_number, Expire\_date, Is\_default |
| Shipping\_method | Shipping\_id, Name, price |
| Order\_status | Order\_status\_id, Status |
| Shop\_order | Shop\_order\_id, Customer\_id, Order\_date, Payment\_method\_id, Shipping\_method\_id, Shipping\_address\_id, Order\_total, Order\_status |

## **Table 2: Relationships between entities**

|  |  |  |
| --- | --- | --- |
| **Entity** | **Related Entity** | **Relationship** |
| Customers | Customer\_address | one-to-many |
| Customers | Shopping\_cart | one-to-many |
| Customers | Shop\_order | one-to-many |
| Address | Customer\_address | one-to-many |
| Address | Shop\_order | one-to-many |
| Customer\_address | Customers | many-to-one |
| Customer\_address | Address | many-to-one |
| Products | Category | many-to-one |
| Products | Product\_entry | one-to-many |
| Product\_size | Product\_entry | one-to-many |
| Ingredients | Product\_entry | one-to-many |
| Skin\_concern | Product\_entry | one-to-many |
| Brand | Product\_entry | one-to-many |
| Product\_entry | Products | many-to-one |
| Product\_entry | Shopping\_cart | one-to-many |
| Shopping\_cart | Product\_entry | many-to-one |
| Shopping\_cart | Customers | many-to-one |
| Payment\_type | Payment\_method | one-to-many |
| Payment\_method | Customers | many-to-one |
| Payment\_method | Payment\_type | many-to-one |
| Payment\_method | Shop\_order | one-to-many |
| Shipping\_method | Shop\_order | one-to-many |
| Order\_status | Shop\_order | one-to-many |

5. Explanation of normal forms

Customers, Address, Customer\_address tables:

1st Normal Form (1NF): All columns in these tables hold atomic values. Each column has a single value, and there are no repeating groups or arrays of values.

2nd Normal Form (2NF): Each of these tables has a primary key defined (Customers.c\_id, Address.Address\_id, and the composite key Customer\_address(Customer\_id, Address\_id)). All non-key attributes depend on the entire primary key, and there is no partial dependency.

3rd Normal Form (3NF): There is no transitive dependency in these tables. There are no attributes that depend on non-key attributes.

Therefore, the Customers, Address, and Customer\_address tables are in 3NF. Product\_size, Ingredients, Skin\_concern, Brand, and Category tables:

1st Normal Form (1NF): All columns in these tables hold atomic values. Each column has a single value, and there are no repeating groups or arrays of values.

2nd Normal Form (2NF): Each of these tables has a primary key defined (Size\_id, Ingredients\_id, Skin\_concern\_id, Brand\_id, and Category\_id). There is no partial dependency, as all non-key attributes depend on the entire primary key.

3rd Normal Form (3NF): There is no transitive dependency in these tables. There are no attributes that depend on non-key attributes.

Therefore, the Product\_size, Ingredients, Skin\_concern, Brand, and Category tables are in 3NF. Products, Product\_entry, Shopping\_cart, Payment\_type, Payment\_method, Shipping\_method, Order\_status, and Shop\_order tables:

1st Normal Form (1NF): All columns in these tables hold atomic values. Each column has a single value, and there are no repeating groups or arrays of values.

2nd Normal Form (2NF): Each of these tables has a primary key defined (Product\_id, Product\_entry\_id, Cart\_id, Payment\_type\_id, Payment\_id, Shipping\_id, Order\_status\_id, and Shop\_order\_id). All non-key attributes depend on the entire primary key, and there is no partial dependency.

3rd Normal Form (3NF): There is no transitive dependency in these tables. There are no attributes that depend on non-key attributes. Therefore, the Products, Product\_entry, Shopping\_cart, Payment\_type, Payment\_method, Shipping\_method, Order\_status, and Shop\_order tables are in 3NF.