**Introduction:**

**Introduction and database description.**

Alyp Key is our e-commerce database system specially designed for online clothing store. This database is responsible for storing and organizing all important information about products, sales, orders and customers in our store.

The Category table serves as a central location for all product categories, including whether the product is for women or men, and is offered in the store. The Product table stores information about all products available for purchase, including their names, descriptions, prices. The Product\_sale and Sale tables store details of any current or past sales, including start and end dates, discounts offered.

The Product\_quantity and Need\_to\_buy tables track inventory levels, ensuring that the store has enough inventory to meet customer demand. The Property\_variation table stores information about product parameters such as size and color.

The Shopping\_cart and Paid\_cart tables allow customers to select and purchase items in the store, while the payments table records all financial transactions associated with those purchases. The Delivery and Address tables store information about delivery addresses and, accordingly, ensure that products are delivered to the right place.

The Payment table is an important part of the online store database system and contains information related to the payment process. The Payment table stores details such as the payment status (e.g., processing, paid, canceled). When a customer makes a purchase, the payment information is recorded in the Payment table, and the payment status is updated accordingly as the transaction progresses. This allows the store to keep track of all financial transactions and ensures that payments are processed correctly.

The Account table stores information about customers, including their login details

Overall, we think that our online store database system is a comprehensive and efficient system that simplifies online store management while providing smooth and satisfactory customer service.

**Entity Relationship Design:**



**Normalization:**

* Category table:
* The table has a potential key - Category\_ID. So the table is already in 1NF.
* The Parented\_ID field is a foreign key that links the Category\_ID to other rows in the table. Therefore, the table is in 2NF.
* All non-key are fully functionally dependent on the primary key. Therefore, the table is in 3NF.
* Product table:
* The table already has a potential key - Product\_ID. So the table is already in 1NF.
* Non-key columns are entirely dependent on the primary key. Therefore, the table is in 2NF.
* The table has no transitive dependencies, so it is also in 3NF.
* Sale table:
* The table has a potential key - Sale\_ID. So the table is already in 1NF.
* All non-key columns are completely dependent on the primary key, so the table is in 2NF.
* The table has no transitive dependencies, so it is also in 3NF.
* Property\_variation
* The table has a potential key - Property\_ID. So the table is already in 1NF.
* The Size and Color fields are non-key attributes that are fully functionally dependent on the primary key. So the table is in 2NF.
* The table has no other dependencies other than primary key dependencies. Therefore, the table is also in 3NF.
* Product\_quantity
* The table has a potential key - Product\_Qty\_ID. So the table is already in 1NF.
* The Property\_variation\_ID field is a foreign key that links the Product\_Qty\_ID to the Property\_variation table. Therefore, the table is in 2NF.
* The Quantity field is fully functionally dependent on the primary key, i.e. from each unique combination of Product\_Qty\_ID and Property\_Variation\_ID. So the table is not in 3NF.
* Need\_to\_buy(Subclass of Product\_quantity table)
* The table has a potential key - Buy\_ID. So the table is already in 1NF.
* The Product\_Qty\_ID field is a foreign key that links Buy\_ID to the Product\_Quantity table. Therefore, the table is in 2NF.
* The Buy\_Quantity field is fully functionally dependent on the primary key, i.e. from each unique combination of Buy\_ID and Product\_Qty\_ID. So the table is in 3NF.
* Shopping\_cart
* The table has a potential key – Shopping\_cart\_ID. So the table is already in 1NF.
* The Payment\_ID field is a foreign key that links the Shopping\_cart\_ID to the Payment table. Therefore, the table is in 2NF.
* The Total\_Cost field is fully functionally dependent on the primary key, i.e. from Cart\_ID. So the table is in 3NF.
* Paid\_cart (Subclass of Shopping\_cart table)
* The table has a potential key - Paid\_cart\_ID. So the table is already in 1NF.
* The Paid\_cart\_ID field is a foreign key that links the Paid\_Cart\_ID to the Shopping\_Cart table. Therefore, the table is in 2NF. The Delivery\_ID field is a foreign key that links the Paid\_cart\_ID to the Delivery table. Therefore, the table is in 2NF.
* The Date field and Payment\_Method are fully functionally dependent on the primary key, i.e. from Paid\_cart\_ID. So the table is in 3NF.
* Payment
* The table has a potential key - Payment\_ID. So the table is already in 1NF.
* The Payment\_Status field is an attribute that is independent of other attributes in the table. So the table is in 2NF.
* The table has no other dependencies other than primary key dependencies. Therefore, the table is also in 3NF.
* Account
* The table has a potential key - Account\_ID. So the table is already in 1NF.
* The fields Username, Fullname, Password and Phone\_number do not depend on other attributes in the table, but only on the primary key. So the table is in 2NF.
* The table has no other dependencies other than primary key dependencies. Therefore, the table is also in 3NF.
* Address
* The table has a potential key - Address\_ID. So the table is already in 1NF.
* The fields Address and City do not depend on other attributes in the table, but only on the primary key. So the table is in 2NF.
* The table has no other dependencies other than primary key dependencies. Therefore, the table is also in 3NF.
* Delivery
* The table has a potential key - Delivery\_ID. So the table is already in 1NF.
* The Arrival\_Date field is an attribute that is independent of other attributes in the table. So the table is in 2NF.
* The Address\_ID field is a foreign key linking the table to the Address table. So the table is in 3NF.