The SQL code above creates multiple tables for an e-commerce database. Each table is described below:

1. admin: This table stores information about the site administrators. The columns in this table are:
    a. adminID: an integer that serves as a primary key and is automatically incremented with each new entry.
    b. username: a varchar(10) that stores the username of the administrator.
    c. pass: a varchar(20) that stores the password of the administrator.

2. cart: This table stores information about items in a user's cart. The columns in this table are:
    a. userID: an integer that stores the user's id.
    b. total_cost: an integer that stores the total cost of items in the cart. This column has a default value of 0.
    c. productID: an integer that stores the product id of the items in the cart.
    d. quantity: an integer that stores the quantity of the items in the cart. This column has a default value of 1.

The table also has multiple foreign keys and constraints to ensure the validity of data.

3. category: This table stores information about different categories of products. The columns in this table are:
    a. categoryID: an integer that serves as a primary key and is automatically incremented with each new entry.
    b. category_name: a varchar(50) that stores the name of the category.
    c. info: a varchar(50) that stores general information about the category. This column has a default value of null.

4. coupons: This table stores information about the different coupons available on the site. The columns in this table are:
    a. couponID: a char(10) that serves as a primary key and stores the id of the coupon.
    b. discount: an integer that stores the discount offered by the coupon. This column has a default value of 0.
    c. expiry: a date that stores the expiration date of the coupon.
    d. is_used: a tinyint(1) that stores whether the coupon has been used or not. This column has a default value of 0.

The table also has a constraint to ensure the discount is greater than 0.

5. my_orders: This table stores information about items in a user's order. The columns in this table are:
    a. orderID: an integer that stores the order id.

   b. productID: an integer that stores the product id of the items in the order.
   c. quantity: an integer that stores the quantity of the items in the order. This column has a default value of 1.
   d. cost: an integer that stores the cost of the items in the order. This column has a default value of 0.

6. order: This table store the details regarding order.
   a. orderID is the primary key and auto-increment.
   b. paymentID is a unique key and has a foreign key relationship with the "payments" table.
   c. couponID is a foreign key relationship with the "coupons" table.
   d. userId is a foreign key relationship with the "user" table.
   e. order_value has a constraint "value_check" to ensure the value is always positive.
7. payments
   a. paymentID is the primary key and auto-increment.
   b. payment_mode can only take values "cod", "netbanking", and "upi".
   c. paymentID should be less than 8 digits.
   d. payment_address is set as "same as shipment" by default.

8. product
   a. productID is the primary key and auto-increment.
   b. price and quantity have a default value of 0.

9. user
   a. userID: This is an integer column and acts as the primary key for the table. The value is set to automatically increment and cannot be null.
   b. first_name: This is a varchar column with a length of 20 characters. The value cannot be null.
   c. last_name: This is a varchar column with a length of 20 characters and can be null.
   d. user_address: This is a varchar column with a length of 50 characters. The value cannot be null.
   e. email_id: This is a varchar column with a length of 20 characters. The value cannot be null.
   f. phone_number: This is a bigint column and cannot be null.
   g. pass: This is a varchar column with a length of 20 characters and cannot be null.
   h. privilege_status: This is a varchar column with a length of 10 characters. The default value is set to 'normal' and cannot be null. The comment in the code states that the value can either be 'normal' or 'pro'.
   i. The table has 3 unique keys:

i. unique_phone: This is a unique key on the 'phone_number' column to ensure that the same phone number is not used by multiple users.
ii. unique_value: This is a unique key on the 'email_id' column to ensure that the same email id is not used by multiple users.
iii. valid_phone: This is a constraint on the 'phone_number' column to ensure that the phone number is between 1000000000 and 10000000000.