CREATE TABLE user (

user\_id INT PRIMARY KEY AUTO\_INCREMENT,

customer\_id INT,

name VARCHAR(255) NOT NULL,

email VARCHAR(255) NOT NULL UNIQUE,

mobile\_number VARCHAR(15) NOT NULL,

password VARCHAR(255) NOT NULL,

state VARCHAR(255) NOT NULL,

city VARCHAR(255) NOT NULL,

image\_name VARCHAR(255) NOT NULL DEFAULT 'default.jpg',

pincode VARCHAR(20) NOT NULL,

user\_status VARCHAR(255) DEFAULT NULL,

registered\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP INVISIBLE,

last\_modified\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP INVISIBLE,

-- Hidden columns for future use

extra\_col1 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col2 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col3 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col4 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col5 VARCHAR(255) DEFAULT NULL INVISIBLE

);

CREATE TABLE customer (

customer\_id INT PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(255) NOT NULL,

email VARCHAR(255) NOT NULL UNIQUE,

mobile\_number VARCHAR(10) NOT NULL,

password VARCHAR(255) NOT NULL,

state VARCHAR(255) NOT NULL,

city VARCHAR(255) NOT NULL,

image\_name VARCHAR(50) NOT NULL DEFAULT 'default.jpg',

pincode VARCHAR(20) NOT NULL,

customer\_status VARCHAR(255) DEFAULT NULL,

registered\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP INVISIBLE,

last\_modified\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP INVISIBLE,

-- Hidden columns for future use

extra\_col1 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col2 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col3 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col4 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col5 VARCHAR(255) DEFAULT NULL INVISIBLE

);

CREATE TABLE Admin (

admin\_id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(255) NOT NULL,

email VARCHAR(255) NOT NULL UNIQUE,

password VARCHAR(255) NOT NULL,

is\_active INT NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP ,

extra\_col1 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col2 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col3 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col4 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col5 VARCHAR(255) DEFAULT NULL INVISIBLE

);

CREATE TABLE product\_category (

category\_id INT PRIMARY KEY AUTO\_INCREMENT,

category\_name VARCHAR(255) NOT NULL,

category\_description VARCHAR(255) NOT NULL,

category\_image\_name VARCHAR(255) DEFAULT NULL,

registered\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP INVISIBLE,

last\_modified\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP INVISIBLE,

-- Hidden columns for future use

extra\_col1 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col2 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col3 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col4 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col5 VARCHAR(255) DEFAULT NULL INVISIBLE

);

CREATE TABLE ProductSubCategory (

subcategory\_id INT PRIMARY KEY AUTO\_INCREMENT,

category\_id INT NOT NULL,

subcategory\_name VARCHAR(255) NOT NULL,

price FLOAT DEFAULT 0,

subcategory\_image\_path VARCHAR(512) DEFAULT NULL,

category\_description VARCHAR(255) NOT NULL,

registered\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP INVISIBLE,

last\_modified\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP INVISIBLE,

-- Hidden columns for future use

extra\_col1 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col2 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col3 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col4 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col5 VARCHAR(255) DEFAULT NULL INVISIBLE,

-- Optional: Foreign key constraint (if category\_id references product\_category)

CONSTRAINT fk\_subcategory\_category FOREIGN KEY (category\_id) REFERENCES product\_category(category\_id)

);

CREATE TABLE deliveryAddress (

delivery\_id INT PRIMARY KEY AUTO\_INCREMENT,

user\_id INT NOT NULL,

user\_name VARCHAR(255) NOT NULL,

state VARCHAR(100) NOT NULL,

city VARCHAR(100) NOT NULL,

pincode VARCHAR(10) NOT NULL,

street\_name VARCHAR(255),

landmark VARCHAR(255),

-- Logging timestamps

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP INVISIBLE,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP INVISIBLE,

-- Extra columns

extra\_col1 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col2 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col3 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col4 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col5 VARCHAR(255) DEFAULT NULL INVISIBLE,

-- Foreign key constraint

FOREIGN KEY (user\_id) REFERENCES user(user\_id)

);

CREATE TABLE userOrders (

order\_id INT AUTO\_INCREMENT PRIMARY KEY,

-- Buyer (user placing order)

user\_id INT NOT NULL,

-- Seller (shopowner / customer)

customer\_id INT NOT NULL,

-- Delivery details

delivery\_id INT,

-- Order metadata

order\_date DATE NOT NULL DEFAULT (CURRENT\_DATE),

order\_time TIME NOT NULL DEFAULT (CURRENT\_TIME),

order\_status ENUM('pending', 'completed', 'cancelled') DEFAULT 'pending',

-- Logs

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP INVISIBLE,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP INVISIBLE,

-- Hidden columns for future use

extra\_col1 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col2 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col3 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col4 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col5 VARCHAR(255) DEFAULT NULL INVISIBLE,

-- Foreign keys

FOREIGN KEY (user\_id) REFERENCES user(user\_id),

FOREIGN KEY (customer\_id) REFERENCES customer(customer\_id),

FOREIGN KEY (delivery\_id) REFERENCES deliveryAddress(delivery\_id) ON DELETE SET NULL

);

CREATE TABLE userOrderItems (

item\_id INT AUTO\_INCREMENT PRIMARY KEY,

order\_id INT NOT NULL,

product\_id INT,

quantity INT NOT NULL DEFAULT 1,

price DECIMAL(10, 2) NOT NULL,

total\_price DECIMAL(10, 2) GENERATED ALWAYS AS (quantity \* price) STORED,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP INVISIBLE,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP INVISIBLE,

extra\_col1 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col2 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col3 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col4 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col5 VARCHAR(255) DEFAULT NULL INVISIBLE,

FOREIGN KEY (order\_id) REFERENCES userOrders(order\_id) ON DELETE CASCADE,

FOREIGN KEY (product\_id) REFERENCES product\_category (category\_id) ON DELETE CASCADE

);

CREATE TABLE userCart (

item\_id INT PRIMARY KEY AUTO\_INCREMENT,

subcategory\_id INT,

user\_id INT NOT NULL,

-- Quantity must be greater than 1 Kg

quantity\_kg FLOAT NOT NULL CHECK (quantity\_kg > 1),

-- Hidden columns for future use

extra\_col1 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col2 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col3 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col4 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col5 VARCHAR(255) DEFAULT NULL INVISIBLE,

-- Foreign keys

FOREIGN KEY (subcategory\_id) REFERENCES ProductSubCategory(subcategory\_id) ON DELETE SET NULL,

FOREIGN KEY (user\_id) REFERENCES user(user\_id)

);

CREATE TABLE serviceZones (

pincode VARCHAR(10) PRIMARY KEY,

state VARCHAR(100) NOT NULL,

city VARCHAR(100) NOT NULL,

district VARCHAR(100) NOT NULL,

service\_type ENUM('Delivery', 'Pickup', 'Both') NOT NULL,

-- Extra column for future use

extra\_col1 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col2 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col3 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col4 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col5 VARCHAR(255) DEFAULT NULL INVISIBLE

);

CREATE TABLE customerorder (

    orderId INT NOT NULL AUTO\_INCREMENT,

    customerId INT NOT NULL,

    userId INT NOT NULL,

    orderDate DATETIME NOT NULL,

    orderStatus VARCHAR(50) NOT NULL,

    totalAmount DECIMAL(10, 2) NOT NULL,

    deliveryAddress VARCHAR(255) NOT NULL,

    PRIMARY KEY (orderId),

    FOREIGN KEY (customerId) REFERENCES customer(customer\_Id),

    FOREIGN KEY (userId) REFERENCES user(user\_Id)

) ;

CREATE TABLE pickupAddress (

pickup\_id INT PRIMARY KEY AUTO\_INCREMENT,

user\_id INT,

user\_name VARCHAR(255) NOT NULL,

state VARCHAR(100) NOT NULL,

city VARCHAR(100) NOT NULL,

pincode VARCHAR(10) NOT NULL,

street\_name VARCHAR(255),

landmark VARCHAR(255),

-- Maintaining the logs of Operations

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP INVISIBLE,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP INVISIBLE,

-- Extra column for future use

extra\_col1 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col2 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col3 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col4 VARCHAR(255) DEFAULT NULL INVISIBLE,

extra\_col5 VARCHAR(255) DEFAULT NULL INVISIBLE,

-- Foreign key

FOREIGN KEY (user\_id) REFERENCES user(user\_id)

);

DELIMITER //

CREATE PROCEDURE register\_user(

IN p\_name VARCHAR(255),

IN p\_email VARCHAR(255),

IN p\_mobile\_number VARCHAR(15),

IN p\_password VARCHAR(255),

IN p\_state VARCHAR(255),

IN p\_city VARCHAR(255),

IN p\_pincode VARCHAR(20)

)

BEGIN

DECLARE new\_user\_id INT;

-- Start transaction to ensure atomicity

START TRANSACTION;

-- Insert into user table

INSERT INTO user (

name,

email,

mobile\_number,

password,

state,

city,

pincode

) VALUES (

p\_name,

p\_email,

p\_mobile\_number,

p\_password,

p\_state,

p\_city,

p\_pincode

);

-- Retrieve the new user's ID

SET new\_user\_id = LAST\_INSERT\_ID();

-- Insert into deliveryAddress with minimal details; user\_name = p\_name

INSERT INTO deliveryAddress (

user\_id,

user\_name,

state,

city,

pincode,

street\_name,

landmark

) VALUES (

new\_user\_id,

p\_name,

p\_state,

p\_city,

p\_pincode,

'', -- Empty street\_name

'' -- Empty landmark

);

-- Commit transaction if all successful

COMMIT;

END //

DELIMITER ;

*DELIMITER //*

*CREATE PROCEDURE create\_user\_order\_from\_cart(*

*IN p\_user\_id INT,*

*IN p\_delivery\_id INT*

*)*

*BEGIN*

*DECLARE new\_order\_id INT;*

*-- Start transaction*

*START TRANSACTION;*

*-- Insert a new order for this user*

*INSERT INTO userOrders (user\_id, delivery\_id)*

*VALUES (p\_user\_id, p\_delivery\_id);*

*-- Get the new order ID*

*SET new\_order\_id = LAST\_INSERT\_ID();*

*-- Transfer cart items for this user to order items*

*INSERT INTO userOrderItem (order\_id, product\_id, quantity, price)*

*SELECT*

*new\_order\_id,*

*subcategory\_id, -- Assuming subcategory\_id represents product\_id or adjust accordingly*

*quantity\_kg,*

*0.0 -- Set price to 0.0 or calculate if you have pricing info elsewhere*

*FROM userCart*

*WHERE user\_id = p\_user\_id;*

*-- Delete cart items only for this user (not truncate the whole table)*

*DELETE FROM userCart WHERE user\_id = p\_user\_id;*

*-- Commit the transaction*

*COMMIT;*

*END //*

*DELIMITER ;*