```
CREATE TABLE countries (
    countryID INT PRIMARY KEY,
    countryName VARCHAR(75)
);

CREATE TABLE cities (
    cityID INT PRIMARY KEY,
    countryID INT,
    cityName VARCHAR(100),
    FOREIGN KEY (countryID) REFERENCES countries(countryID)
);

CREATE TABLE towns (
    townID INT PRIMARY KEY,
    cityID INT,
    townName VARCHAR(100),
    FOREIGN KEY (cityID) REFERENCES cities(cityID)
);

CREATE TABLE districts (
    districtID INT PRIMARY KEY,
    townID INT,
    districtName VARCHAR(100),
    FOREIGN KEY (townID) REFERENCES towns(townID)
);

//

CREATE TABLE address (
    addressID INT PRIMARY KEY,
    customerID INT,
    countryID INT,
    cityID INT,
    townID INT,
    districtID INT,
    postalCode VARCHAR(15),
    addressText VARCHAR(500),
    FOREIGN KEY (countryID) REFERENCES countries(countryID),
    FOREIGN KEY (cityID) REFERENCES cities(cityID),
    FOREIGN KEY (townID) REFERENCES towns(townID),
    FOREIGN KEY (districtID) REFERENCES districts(districtID)
);

CREATE TABLE customers (
    customerID INT PRIMARY KEY,
    customerName VARCHAR(50),
    customerSurname VARCHAR(50),
    email VARCHAR(100),
    gender CHAR(1),
    birthDate DATE,
    phoneNumber VARCHAR(20),
    addressID INT,
    FOREIGN KEY (addressID) REFERENCES address(addressID)
);
```

```
//

CREATE TABLE product_categories (
    categoryID INT PRIMARY KEY,
    categoryName VARCHAR(100)
);

CREATE TABLE measure (
    measureID INT PRIMARY KEY,
    measureName VARCHAR(50)
);

CREATE TABLE suppliers (
    supplierID INT PRIMARY KEY,
    supplierName VARCHAR(100),
    phoneNumber VARCHAR(20)
);

CREATE TABLE products (
    productID INT PRIMARY KEY,
    categoryID INT,
    supplierID INT,
    productName VARCHAR(100),
    price FLOAT,
    stockAmount INT,
    reorderLevel INT,
    measureID INT,
    FOREIGN KEY (categoryID) REFERENCES product_categories(categoryID),
    FOREIGN KEY (supplierID) REFERENCES suppliers(supplierID),
    FOREIGN KEY (measureID) REFERENCES measure(measureID)
);

CREATE TABLE supplier_products (
    supplierProductID INT PRIMARY KEY,
    productID INT,
    supplierID INT,
    FOREIGN KEY (productID) REFERENCES products(productID),
    FOREIGN KEY (supplierID) REFERENCES suppliers(supplierID)
);

//

CREATE TABLE order_status (
    statusID INT PRIMARY KEY,
    orderStatus VARCHAR(25)
);

CREATE TABLE payment_type (
    paymentID INT PRIMARY KEY,
    typeName VARCHAR(25)
);

CREATE TABLE orders (
```

```
    orderID INT PRIMARY KEY,
    customerID INT,
    productID INT,
    addressID INT,
    orderDate DATE,
    statusID INT,
    totalPrice FLOAT,
    FOREIGN KEY (customerID) REFERENCES customers(customerID),
    FOREIGN KEY (productID) REFERENCES products(productID),
    FOREIGN KEY (addressID) REFERENCES address(addressID),
    FOREIGN KEY (statusID) REFERENCES order_status(statusID)
);

CREATE TABLE sales (
    saleID INT PRIMARY KEY,
    customerID INT,
    paymentID INT,
    orderID INT,
    saleDate DATE,
    quantity INT,
    totalPrice FLOAT,
    saleType VARCHAR(25),
    FOREIGN KEY (customerID) REFERENCES customers(customerID),
    FOREIGN KEY (paymentID) REFERENCES payment_type(paymentID),
    FOREIGN KEY (orderID) REFERENCES orders(orderID)
);

//

CREATE TABLE credit_status (
    statusID INT PRIMARY KEY,
    creditStatus VARCHAR(50)
);

CREATE TABLE credit_sales (
    creditID INT PRIMARY KEY,
    saleID INT,
    statusID INT,
    FOREIGN KEY (saleID) REFERENCES sales(saleID),
    FOREIGN KEY (statusID) REFERENCES credit_status(statusID)
);

CREATE TABLE crypto_sales (
    cryptoID INT PRIMARY KEY,
    saleID INT,
    FOREIGN KEY (saleID) REFERENCES sales(saleID)
);
```