-- Create Database

CREATE DATABASE ecommerce;

USE ecommerce;

-- Create Customers Table

CREATE TABLE customers (

    id INT AUTO\_INCREMENT PRIMARY KEY,

    name VARCHAR(100) NOT NULL,

    email VARCHAR(100) UNIQUE NOT NULL,

    address VARCHAR(255)

);

-- Create Products Table

CREATE TABLE products (

    id INT AUTO\_INCREMENT PRIMARY KEY,

    name VARCHAR(100) NOT NULL,

    price DECIMAL(10, 2) NOT NULL,

    description TEXT,

    discount DECIMAL(5, 2) DEFAULT 0.00

);

-- Create Orders Table

CREATE TABLE orders (

    id INT AUTO\_INCREMENT PRIMARY KEY,

    customer\_id INT,

    order\_date DATE NOT NULL,

    total\_amount DECIMAL(10, 2) NOT NULL,

    FOREIGN KEY (customer\_id) REFERENCES customers(id)

);

-- Insert Sample Data into Customers

INSERT INTO customers (name, email, address) VALUES

('Rahul Sharma', 'rahul.sharma@example.com', 'Flat 12, Prestige Apartments, Bangalore, Karnataka'),

('Priya Patel', 'priya.patel@example.com', '45 Gandhi Road, Mumbai, Maharashtra'),

('Amit Kumar', 'amit.kumar@example.com', 'House 7, Green Colony, Delhi'),

('Anjali Reddy', 'anjali.reddy@example.com', '22 Park View, Hyderabad, Telangana'),

('Vikram Singh', 'vikram.singh@example.com', 'Sector 15, Noida, Uttar Pradesh'),

('Deepa Mehta', 'deepa.mehta@example.com', '15 Lake View, Ahmedabad, Gujarat'),

('Rajesh Gupta', 'rajesh.gupta@example.com', 'Apartment 3B, Salt Lake, Kolkata, West Bengal'),

('Shruti Iyer', 'shruti.iyer@example.com', '78 Brahmin Street, Chennai, Tamil Nadu'),

('Arun Nair', 'arun.nair@example.com', 'Villa 22, Kochi, Kerala'),

('Neha Joshi', 'neha.joshi@example.com', '56 University Road, Pune, Maharashtra'),

('Sanjay Desai', 'sanjay.desai@example.com', 'Flat 45, Satellite Area, Ahmedabad'),

('Kavita Bose', 'kavita.bose@example.com', '12 Park Street, Kolkata'),

('Manish Malhotra', 'manish.malhotra@example.com', 'DLF Phase 2, Gurgaon, Haryana'),

('Pooja Khanna', 'pooja.khanna@example.com', 'Sector 42, Chandigarh'),

('Suresh Babu', 'suresh.babu@example.com', '89 Residency Road, Bangalore, Karnataka');

-- Retrieving the inserted sample data from customers

SELECT \* From customers;



-- Insert Sample Data into Products (15 records with real product names)

INSERT INTO products (name, price, description) VALUES

('Apple iPhone 13', 54999.00, 'Powerful smartphone with advanced dual-camera system'),

('Samsung 65" QLED 4K Smart TV', 129999.00, 'Immersive viewing experience with Quantum Dot technology'),

('Kindle Paperwhite', 13999.00, 'High-resolution e-reader with adjustable warm light'),

('Noise ColorFit Ultra Smart Watch', 2999.00, 'Fitness tracker with health monitoring features'),

('Sony WH-1000XM4 Noise Cancelling Headphones', 29999.00, 'Premium wireless headphones with industry-leading noise cancellation'),

('Xiaomi Mi Robot Vacuum', 24999.00, 'Smart robotic vacuum cleaner with advanced navigation'),

('OnePlus Buds Pro', 9999.00, 'Wireless earbuds with active noise cancellation'),

('Dyson V11 Absolute Vacuum Cleaner', 52999.00, 'Powerful cordless vacuum with intelligent suction'),

('LG 1.5 Ton Inverter Split AC', 44999.00, 'Energy-efficient air conditioner with cooling technology'),

('Fitbit Versa 3 Smartwatch', 18999.00, 'Advanced fitness and health smartwatch'),

('Canon EOS R6 Mirrorless Camera', 159999.00, 'Professional-grade full-frame mirrorless camera'),

('Bose SoundLink Revolve+ Bluetooth Speaker', 19999.00, 'Portable 360-degree sound speaker'),

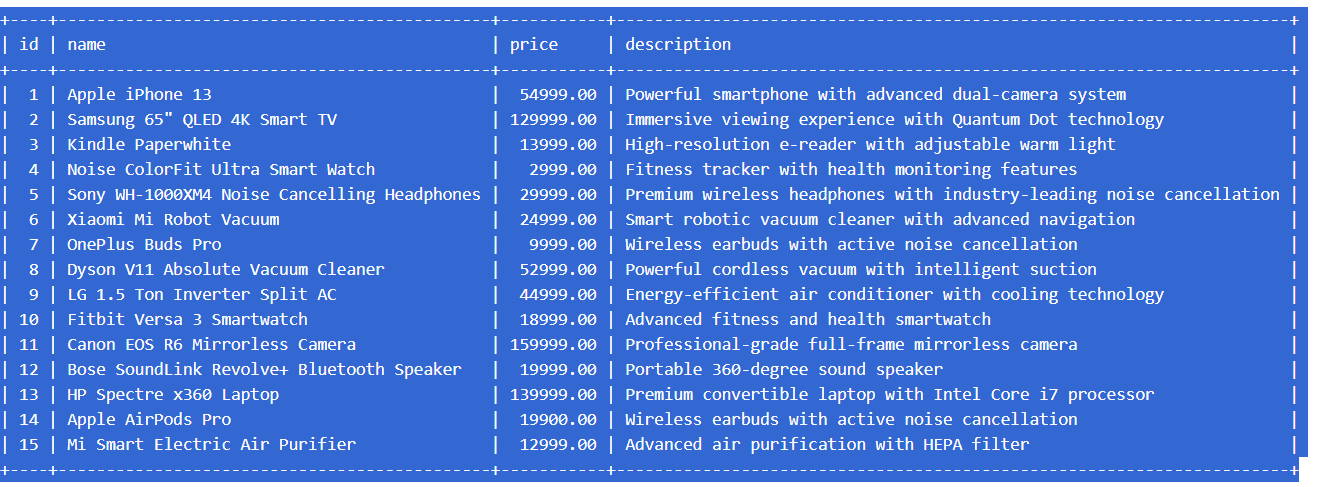
('HP Spectre x360 Laptop', 139999.00, 'Premium convertible laptop with Intel Core i7 processor'),

('Apple AirPods Pro', 19900.00, 'Wireless earbuds with active noise cancellation'),

('Mi Smart Electric Air Purifier', 12999.00, 'Advanced air purification with HEPA filter');

-- Retrieving the inserted sample data from products

SELECT \* From products;



-- Insert Sample Data into Orders

INSERT INTO orders (customer\_id, order\_date, total\_amount) VALUES

(1, '2024-06-10', 54999.00),

(2, '2016-04-15', 159999.00),

(3, '2024-03-20', 44999.00),

(4, '2024-01-25', 52999.00),

(5, '2023-03-30', 29999.00),

(6, '2019-02-05', 19999.00),

(7, '2024-09-10', 13999.00),

(8, '2023-07-15', 139999.00),

(9, '2024-07-20', 24999.00),

(10, '2024-07-25', 18999.00),

(11, '2023-11-30', 9999.00),

(12, '2024-08-05', 12999.00),

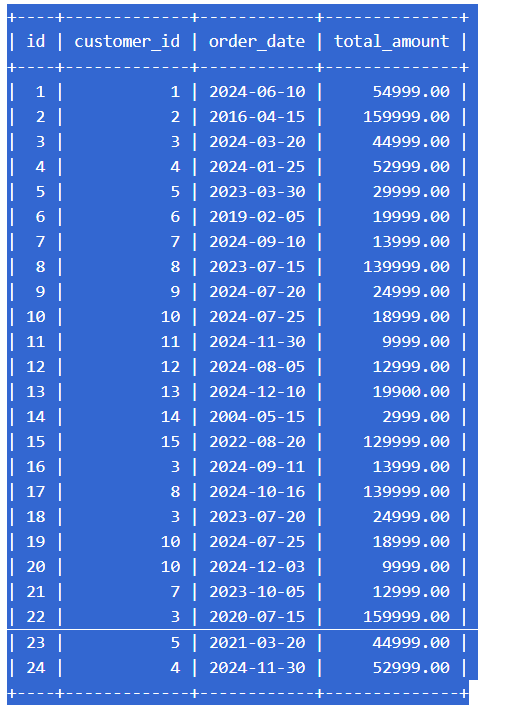
(13, '2014-12-10', 19900.00),

(14, '2004-05-15', 2999.00),

(15, '2022-08-20', 129999.00);

-- Retrieving the inserted sample data from orders

SELECT \* From orders;



-- Query 1: Retrieve customers who placed an order in the last 30 days

SELECT DISTINCT c.id, c.name, c.email

FROM customers c

JOIN orders o ON c.id = o.customer\_id

WHERE o.order\_date >= DATE(NOW()) - INTERVAL 30 DAY;



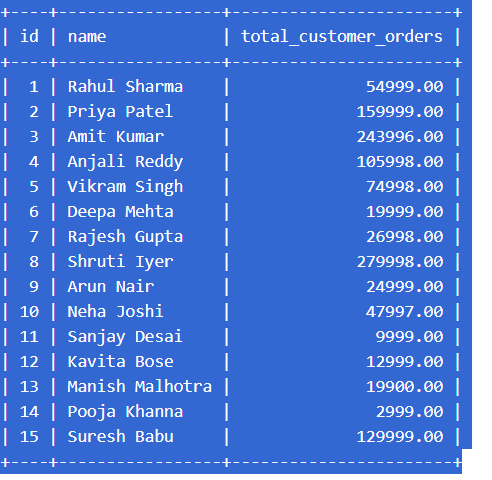
-- Query 2: Total amount of orders placed by each customer

SELECT c.id, c.name, SUM(o.total\_amount) AS total\_customer\_orders

FROM customers c

LEFT JOIN orders o ON c.id = o.customer\_id

GROUP BY c.id;



-- Query 3: Update price of a specific product (e.g., OnePlus Buds Pro )

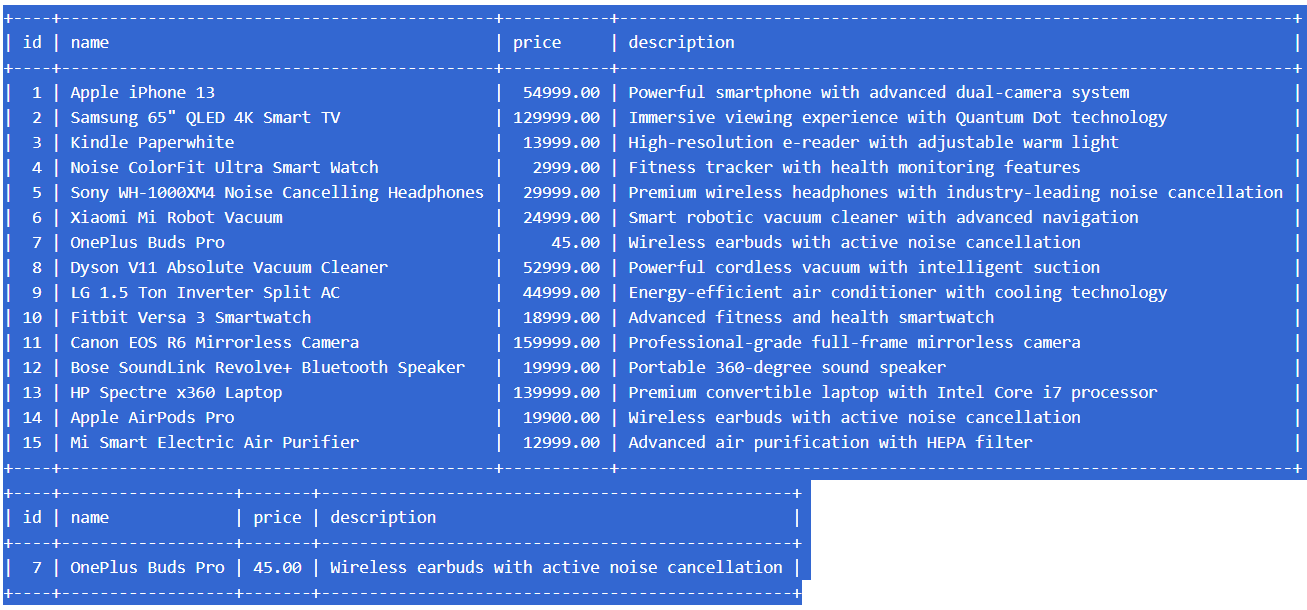
UPDATE products

SET price = 45.00

WHERE name = 'OnePlus Buds Pro';

SELECT \* FROM products;

SELECT \* FROM products WHERE name = 'OnePlus Buds Pro';

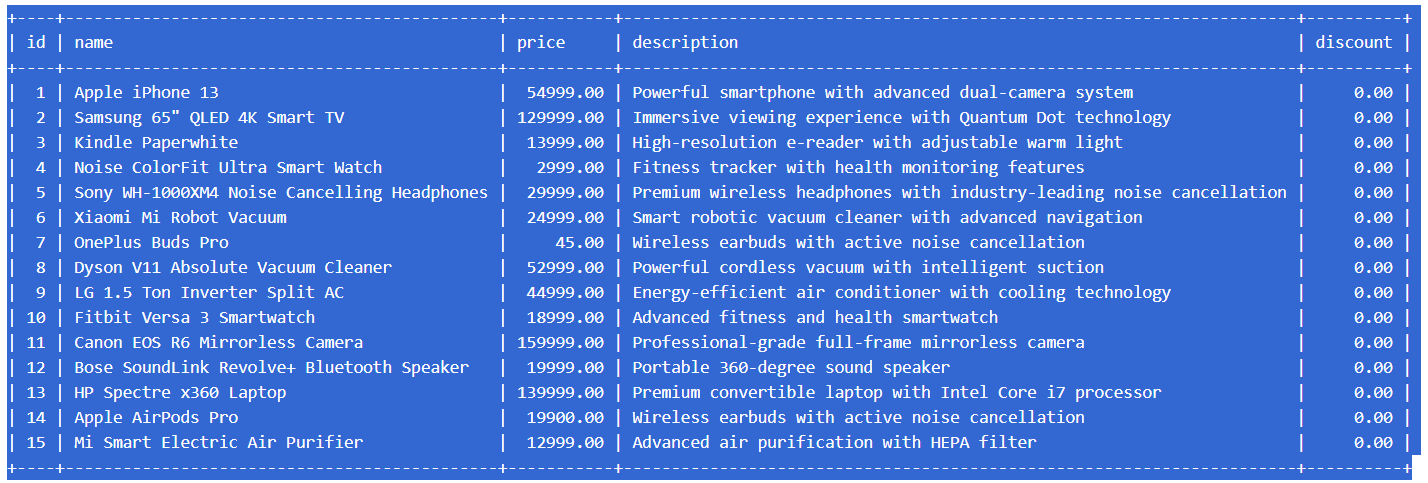


-- Query 4: Add a new column "discount" to the products table

ALTER TABLE products

ADD COLUMN discount DECIMAL(5, 2) DEFAULT 0.00;

SELECT \* FROM products;



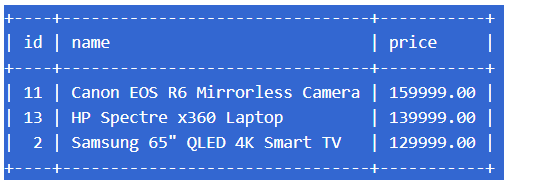
-- Query 5: Top 3 products with highest price

SELECT id, name, price

FROM products

ORDER BY price DESC

LIMIT 3;



-- Query 9: Normalize the database by creating an order\_items table

CREATE TABLE order\_items (

id INT AUTO\_INCREMENT PRIMARY KEY,

order\_id INT NOT NULL,

product\_id INT NOT NULL,

quantity INT NOT NULL,

FOREIGN KEY (order\_id) REFERENCES orders(id),

FOREIGN KEY (product\_id) REFERENCES products(id)

);

-- Insert Sample Data into Order Items

INSERT INTO order\_items (order\_id, product\_id, quantity) VALUES

(1, 1, 1),

(2, 11, 1),

(3, 9, 1),

(4, 8, 1),

(5, 5, 1),

(6, 12, 1),

(7, 3, 1),

(8, 13, 1),

(9, 6, 1),

(10, 10, 1),

(11, 7, 1),

(12, 15, 1),

(13, 14, 1),

(14, 4, 1),

(15, 2, 1),

(16, 3, 1),

(17, 13, 1),

(18, 6, 1),

(19, 10, 1),

(20, 7, 1),

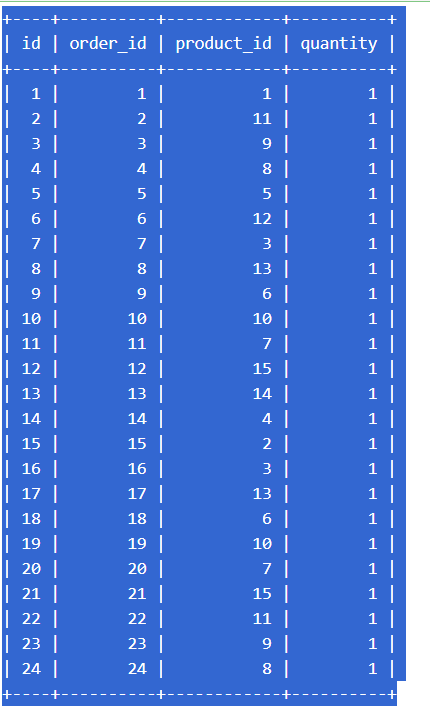
(21, 15, 1),

(22, 11, 1),

(23, 9, 1),

(24, 8, 1);

SELECT \* FROM order\_items;



-- Query 6: Names of customers who ordered a specific product (e.g., LG 1.5 Ton Inverter Split AC)

SELECT DISTINCT c.name

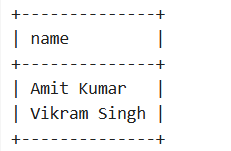
FROM customers c

JOIN orders o ON c.id = o.customer\_id

JOIN order\_items oi ON o.id = oi.order\_id

JOIN products p ON oi.product\_id = p.id

WHERE p.name = 'LG 1.5 Ton Inverter Split AC';

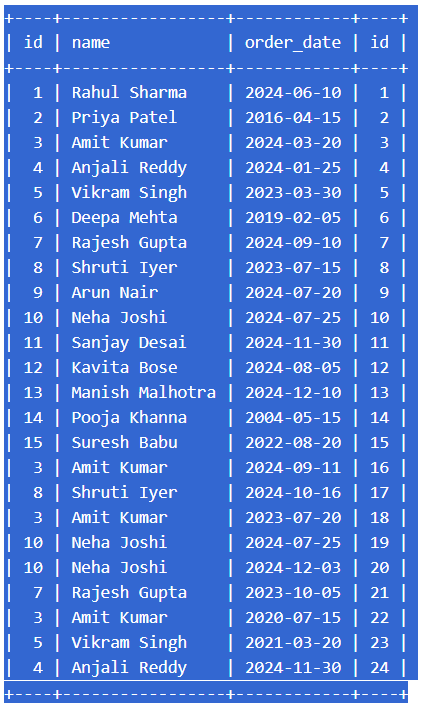


-- Query 7: Join orders and customers to get customer name and order date

SELECT c.id, c.name, o.order\_date, o.id

FROM orders o

JOIN customers c ON o.customer\_id = c.id;

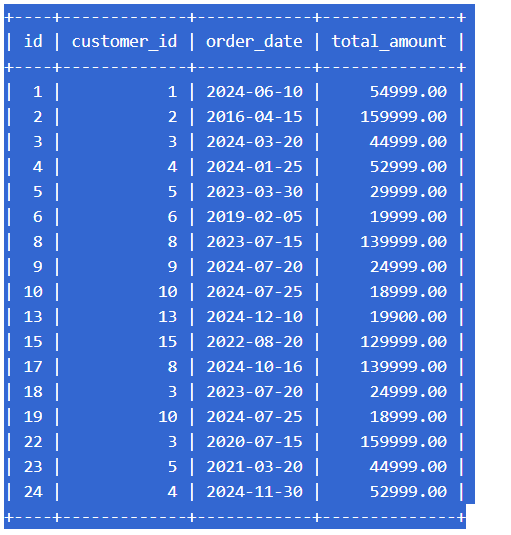


-- Query 8: Orders with total amount greater than 15000

SELECT \*

FROM orders

WHERE total\_amount > 15000;



-- Query 9: Normalize the database by creating an order\_items table

CREATE TABLE order\_items (

id INT AUTO\_INCREMENT PRIMARY KEY,

order\_id INT NOT NULL,

product\_id INT NOT NULL,

quantity INT NOT NULL,

FOREIGN KEY (order\_id) REFERENCES orders(id),

FOREIGN KEY (product\_id) REFERENCES products(id)

);

-- Insert Sample Data into Order Items

INSERT INTO order\_items (order\_id, product\_id, quantity) VALUES

(1, 1, 1),

(2, 11, 1),

(3, 9, 1),

(4, 8, 1),

(5, 5, 1),

(6, 12, 1),

(7, 3, 1),

(8, 13, 1),

(9, 6, 1),

(10, 10, 1),

(11, 7, 1),

(12, 15, 1),

(13, 14, 1),

(14, 4, 1),

(15, 2, 1),

(16, 3, 1),

(17, 13, 1),

(18, 6, 1),

(19, 10, 1),

(20, 7, 1),

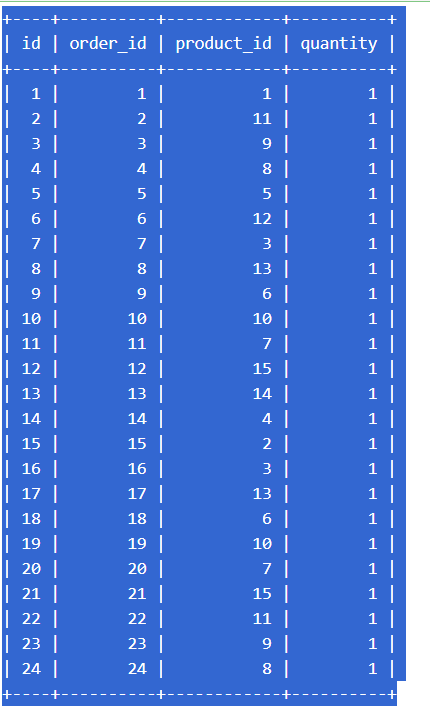
(21, 15, 1),

(22, 11, 1),

(23, 9, 1),

(24, 8, 1);

SELECT \* FROM order\_items;



-- Query 10: Average total of all orders

SELECT ROUND(AVG(total\_amount), 2) AS order\_total\_avg

FROM orders;

