```
USE amazon_practice_db

-- Question 1:List all customers who have made purchases of more than $80.
SELECT
    U.name,
    U.email,
    SUM(O.total_amount) AS total_spent
FROM
    Users U
JOIN
    Orders O ON U.user_id = O.user_id
GROUP BY
    U.user_id
HAVING
    total_spent > 80; -----------Ouput done

    -- 2. Retrieve all orders placed in the last 30 days along with the
customer name and email.
SELECT
    O.order_id,
    O.order_date,
    U.name,
    U.email,
    O.total_amount
FROM
    Orders O
JOIN
    Users U ON O.user_id = U.user_id
WHERE
    O.order_date >= CURDATE() - INTERVAL 280 DAY;--------------- output
done

    -- 3. Find the average product price for each category.
SELECT
    category,
    AVG(price) AS avg_price
FROM
    Products
GROUP BY
    category;


-- 4. List all customers who have purchased a product from the category
Electronics.
SELECT DISTINCT
    U.name,
    U.email
FROM
    Users U
JOIN
    Orders O ON U.user_id = O.user_id
JOIN
    OrderDetails OD ON O.order_id = OD.order_id
JOIN
```

```
    Products P ON OD.product_id = P.product_id
WHERE
    P.category = 'Electronics';

-- 5.Find the total number of products sold and the total revenue
generated for each product.
SELECT
    P.name AS product_name,
    SUM(OD.quantity) AS total_quantity_sold,
    SUM(OD.quantity * P.price) AS total_revenue
FROM
    OrderDetails OD
JOIN
    Products P ON OD.product_id = P.product_id
GROUP BY
    P.product_id;

-- 6. Update the price of all products in the Books category, increasing
it by 10%.

SET SQL_SAFE_UPDATES = 0;

UPDATE
    Products
SET
    price = price * 1.10
WHERE
    category = 'Books';
-- 7. Remove all orders that were placed before 2020.
DELETE FROM
    Orders
WHERE
    order_date < '2024-01-01';


-  8. Fetch the order details, including customer name, product name, and
   quantity, for orders placed on 2024-11-01.

SELECT
    O.order_id,
    U.name AS customer_name,
    P.name AS product_name,
    OD.quantity
FROM
    Orders O
JOIN
    Users U ON O.user_id = U.user_id
JOIN
    OrderDetails OD ON O.order_id = OD.order_id
JOIN
    Products P ON OD.product_id = P.product_id
WHERE
    O.order_date = '2024-5-01';
```

```sql
-- 9. Fetch all customers and the total number of orders they have placed.

SELECT
    U.name AS customer_name,
    U.email,
    COUNT(O.order_id) AS total_orders
FROM
    Users U
LEFT JOIN
    Orders O ON U.user_id = O.user_id
GROUP BY
    U.user_id;

-- 10.  List all customers who purchased more than 3 units of any product,
including the product name and total quantity purchased.
SELECT
    U.name AS customer_name,
    U.email,
    P.name AS product_name,
    SUM(OD.quantity) AS total_quantity
FROM
    Users U
JOIN
    Orders O ON U.user_id = O.user_id
JOIN
    OrderDetails OD ON O.order_id = OD.order_id
JOIN
    Products P ON OD.product_id = P.product_id
GROUP BY
    U.user_id, P.product_id
HAVING
    total_quantity > 1;

-- 11. Find the total revenue generated by each category along with the
category name.
SELECT
    P.category,
    SUM(OD.quantity * P.price) AS total_revenue
FROM
    Products P
JOIN
    OrderDetails OD ON P.product_id = OD.product_id
GROUP BY
    P.category;
```