```sql
-- SELECT
-- Write a query that returns everything in the customer table.

SELECT *
FROM customer;

-- Write a query that displays all of the columns and 10 rows from the customer table, sorted by customer_last_name, then customer_first_ name.

SELECT *
FROM customer
ORDER BY customer_last_name, customer_first_name
LIMIT 10;

--WHERE
--Write a query that returns all customer purchases of product IDs 4 and 9.

SELECT *
FROM customer_purchases
WHERE product_id IN (4,9);

/* Write a query that returns all customer purchases and a new calculated column 'price' (quantity * cost_to_customer_per_qty), filtered by vendor IDs between 8 and 10 (inclusive) using either:
    two conditions using AND
    one condition using BETWEEN */

SELECT *,
        (quantity * cost_to_customer_per_qty) as price
FROM customer_purchases
WHERE vendor_id BETWEEN 8 AND 10;

-- CASE
/* Products can be sold by the individual unit or by bulk measures like lbs. or oz.
Using the product table, write a query that outputs the product_id and product_name columns and add a column called prod_qty_type_condensed that displays the word “unit” if the product_qty_type is “unit,” and otherwise displays the word “bulk.” */

/*We want to flag all of the different types of pepper products that are sold at the market.
Add a column to the previous query called pepper_flag that outputs a 1 if the product_name contains the word “pepper” (regardless of capitalization), and otherwise outputs 0. */

SELECT  product_id,
        product_name,
        CASE
            WHEN product_qty_type = 'unit' THEN 'unit'
            WHEN product_qty_type = 'lbs'  THEN 'bulk'
        END as prod_qty_type_condensed,
        CASE
            WHEN lower (product_name) LIKE '%pepper%' then 1
            ELSE 0
        END as pepper_flag
FROM product;

--JOIN
/* Write a query that INNER JOINs the vendor table to the vendor_booth_assignments table on the vendor_id field they both have in common, and sorts the result by vendor_name, then market_date. */

SELECT v.vendor_id,
    v.vendor_name,
    vba.market_date,
    vba.booth_number

```

```
FROM vendor as v
INNER JOIN vendor_booth_assignments as vba
    ON v.vendor_id = vba.vendor_id
ORDER BY vendor_name, market_date;

```