```
-- Monday Coffee -- Data Analysis

SELECT
    *
FROM
    city;
SELECT
    *
FROM
    products;
SELECT
    *
FROM
    customers;
SELECT
    *
FROM
    sales;

-- Reports & Data Analysis

-- Coffee COnsumers Count
-- How many people in each city are estimated to consume coffee, given
that 25% of the population does?

SELECT
    city_name,
    ROUND((population * 0.25) / 1000000, 2) AS
Cofee_Consumer_in_Millions,
    city_rank
FROM
    city
ORDER BY 2 DESC;

-- -- Q.2
-- Total Revenue from Coffee Sales
-- What is the total revenue generated from coffee sales across all
cities in the last quarter of 2023?

SELECT
    ct.city_name,
    SUM(s.total) AS total_revenue,
    YEAR(s.sale_date),
    QUARTER(s.sale_date)
FROM
    sales AS s
        JOIN
    customers AS cs ON s.customer_id = cs.customer_id
        JOIN
    city AS ct ON ct.city_id = cs.city_id
WHERE
    YEAR(s.sale_date) = 2023
        AND QUARTER(s.sale_date) = 4
GROUP BY 1
;


SELECT
```

```
    ci.city_name, SUM(s.total) AS total_revenue
FROM
    sales AS s
        JOIN
    customers AS c ON s.customer_id = c.customer_id
        JOIN
    city AS ci ON ci.city_id = c.city_id
WHERE
    EXTRACT(YEAR FROM s.sale_date) = 2023
        AND EXTRACT(QUARTER FROM s.sale_date) = 4
GROUP BY 1
ORDER BY 2 DESC;


-- Q.3
-- Sales Count for Each Product
-- How many units of each coffee product have been sold?
SELECT
    p.product_name, COUNT(sale_id) AS total_orders
FROM
    products p
        LEFT JOIN
    sales s ON p.product_id = s.product_id
GROUP BY 1
ORDER BY 2 DESC;

-- Q.4
-- Average Sales Amount per City
-- What is the average sales amount per customer in each city?

SELECT
    COUNT(DISTINCT cs.customer_id) AS total_customer,
    ct.city_name,
    AVG(s.total) AS avg_sales,
    AVG(s.total) / COUNT(DISTINCT cs.customer_id) AS
per_customer_avg_Sales
FROM
    city AS ct
        JOIN
    customers cs ON ct.city_id = cs.city_id
        JOIN
    sales AS s ON cs.customer_id = s.customer_id
GROUP BY 2;


-- -- Q5
-- Top Selling Products by City
-- What are the top 3 selling products in each city?

with t as
(select p.product_name,ct.city_name,count(s.sale_id) as total_orrders,
dense_rank() over(partition by ct.city_name order by count(s.sale_id)
desc) as top_3
from products p
join sales s
on p.product_id = s.product_id
join customers cs
on s.customer_id = cs.customer_id
```

```
join city ct
on ct.city_id = cs.city_id
group by 1,2)
select * from t where t.top_3 < 4
;


select * from
(select p.product_name,ct.city_name,count(s.sale_id) as total_orrders,
dense_rank() over(partition by ct.city_name order by count(s.sale_id)
desc) as top_3
from products p
join sales s
on p.product_id = s.product_id
join customers cs
on s.customer_id = cs.customer_id
join city ct
on ct.city_id = cs.city_id
group by 1,2) as t
where t.top_3 < 4
;

-- Q.6
-- Customer Segmentation by City
-- How many unique customers are there in each city who have purchased
coffee products?
SELECT
    ct.city_name, COUNT(DISTINCT cs.customer_id)
FROM
    customers cs
        JOIN
    city ct ON cs.city_id = ct.city_id
        JOIN
    sales s ON cs.customer_id = s.customer_id
        JOIN
    products p ON s.product_id = p.product_id
WHERE
    s.product_id IN (1 , 2,
        3,
        4,
        5,
        6,
        7,
        8,
        9,
        10,
        11,
        12,
        13,
        14)
GROUP BY 1
;

-- -- Q.7
-- Average Sale
-- Find each city and their average sale per customer

SELECT
```

```
    ct.city_name,
    ROUND(SUM(total) / COUNT(DISTINCT s.customer_id),
            2) avg_sale_per_customer
FROM
    city ct
        JOIN
    customers cs ON ct.city_id = cs.city_id
        JOIN
    sales s ON s.customer_id = s.customer_id
GROUP BY 1
ORDER BY 2
;


-- Q.8
-- Monthly Sales Growth
-- Sales growth rate: Calculate the percentage growth (or decline) in
sales over different time periods (monthly)
-- by each city
with t as
(select month(s.sale_date), year(s.sale_date),ct.city_name,
sum(s.total),
lag(sum(s.total)) over(partition by city_name order by
month(s.sale_date),year(s.sale_date)) -
lead(sum(s.total)) over(partition by city_name order by
month(s.sale_date),year(s.sale_date)) as diffence_of_monthly_sales
from sales s
join customers cs
on s.customer_id = cs.customer_id
join city as ct
on cs.city_id = cs.city_id
group by 1,2,3)
select * from t where t.diffence_of_monthly_sales is not null;


-- Q.9
-- Market Potential Analysis
-- Identify top 3 city based on highest sales, return city name, total
sale, total rent, total customers.
with t as
(select ct.city_name,sum(total),sum(ct.estimated_rent),count(distinct
s.customer_id),
dense_rank() over(order by sum(total) desc) as top_3
 from sales s
join customers cs
on s.customer_id = cs.customer_id
join city ct
on ct.city_id = cs.city_id
group by 1)
select * from t where t.top_3 < 4
;
```