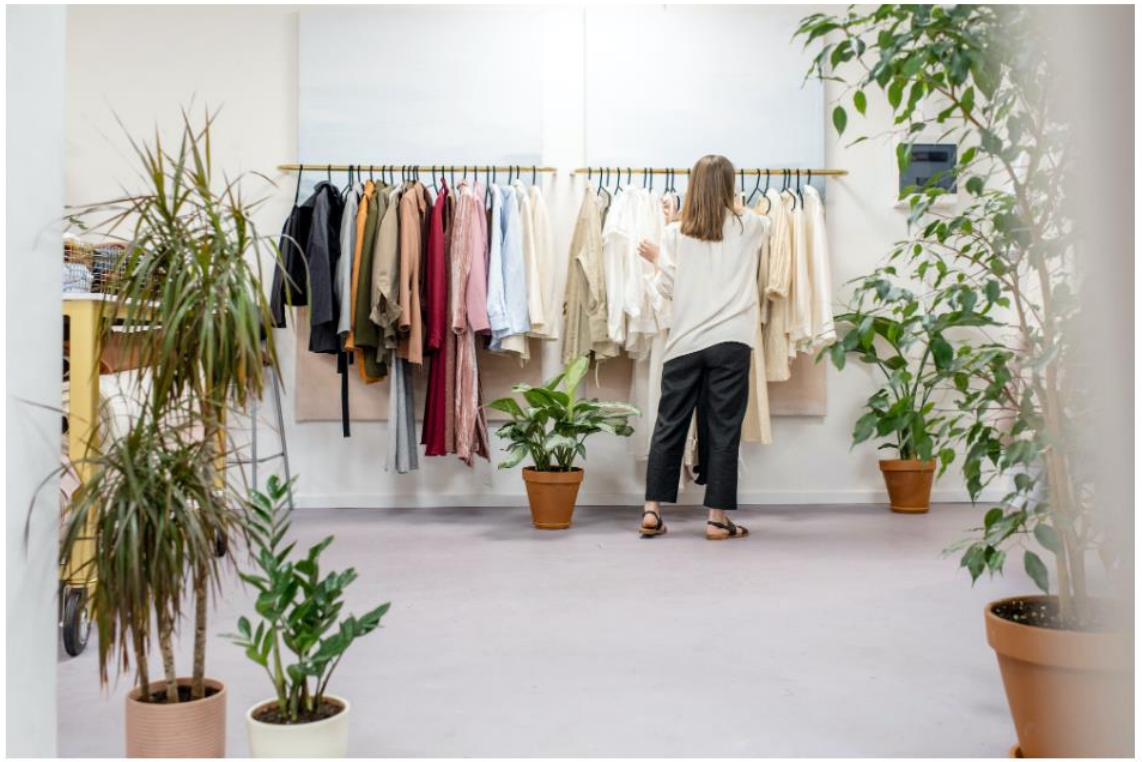
Sustainable Clothing Co.

Marketing Analysis

By Vikram Aiyer

Marketing Analysis



Intro

You are a Marketing Analyst

The 'Sustainable Clothing Co.' have been running several marketing campaigns and have asked you to provide your insight into whether they have been successful or not. Analyse the following data and answer the questions to form your answer.

Tables

Tables

Here are the tables you will be using

sustainable_clothing

Product ID	Product Name	Category	Size	Price
1	Organic Cotton T-Shirt	Tops	S	$29.99
2	Recycled Denim Jeans	Bottoms	M	$79.99
3	Hemp Crop Top	Tops	L	$24.99
4	Bamboo Lounge Pants	Bottoms	XS	$49.99
5	Eco-Friendly Hoodie	Outerwear	XL	$59.99
6	Linen Button-Down Shirt	Tops	M	$39.99
7	Organic Cotton Dress	Dresses	S	$69.99
8	Sustainable Swim Shorts	Swimwear	L	$34.99
9	Recycled Polyester Jacket	Outerwear	XL	$89.99
10	Bamboo Yoga Leggings	Activewear	XS	$54.99
11	Hemp Overalls	Bottoms	M	$74.99
12	Organic Cotton Sweater	Tops	L	$49.99
13	Cork Sandals	Footwear	S	$39.99
14	Recycled Nylon Backpack	Accessories	One Size	$59.99
15	Organic Cotton Skirt	Bottoms	XS	$34.99
16	Hemp Baseball Cap	Accessories	One Size	$24.99
17	Upcycled Denim Jacket	Outerwear	M	$79.99
18	Linen Jumpsuit	Dresses	L	$69.99
19	Organic Cotton Socks	Accessories	M	$9.99
20	Bamboo Bathrobe	Loungewear	XL	$69.99

transactions (first 10 shown)

transaction_id	product_id	quantity	purcahse_date
1	2	2	2023-06-02
1	14	1	2023-06-02
2	5	2	2023-06-05
3	2	1	2023-06-07
4	19	2	2023-06-10
5	2	1	2023-06-13
5	16	1	2023-06-13
6	10	2	2023-06-15
7	2	1	2023-06-18
8	4	1	2023-06-22
9	18	2	2023-06-26
10	2	1	2023-06-30
10	13	1	2023-06-30

marketing_campaigns

campaign_id	campaign_name	product_id	start_date	end_date
1	Summer Sale	2	2023-06-01	2023-06-30
2	New Collection Launch	10	2023-07-15	2023-08-15
3	Super Save	7	2023-08-20	2023-09-15

Questions

- 1. How many transactions were completed during each marketing campaign?
- 2. Which product had the highest sales quantity?
- 3. What is the total revenue generated from each marketing campaign?
- 4. What is the top-selling product category based on the total revenue generated?
- 5. Which products had a higher quantity sold compared to the average quantity sold?
- 6. What is the average revenue generated per day during the marketing campaigns?
- 7. What is the percentage contribution of each product to the total revenue?
- 8. Compare the average quantity sold during marketing campaigns to outside the marketing campaigns
- 9. Compare the revenue generated by products inside the marketing campaigns to outside the campaigns
- 10. Rank the products by their average daily quantity sold

1. How many transactions were completed during each marketing campaign?

Schema SQL

```
100 (43, 1, 1, '2023-10-01'),
101 (44, 7, 1, '2023-10-02'),
102 (45, 18, 2, '2023-10-03'),
103 (46, 12, 1, '2023-10-03'),
104 (47, 13, 1, '2023-10-04'),
105 (48, 4, 1, '2023-10-05'),
106 (49, 12, 2, '2023-10-05'),
107 (50, 7, 1, '2023-10-06'),
108 (51, 4, 2, '2023-10-08'),
109 (52, 8, 2, '2023-10-08'),
110 (53, 16, 1, '2023-10-09'),
111 (54, 19, 1, '2023-10-09'),
112 (55, 1, 1, '2023-10-10'),
113 (56, 18, 2, '2023-10-10'),
114 (57, 2, 1, '2023-10-10'),
115 (58, 15, 2, '2023-10-11'),
116 (59, 17, 2, '2023-10-13'),
117 (60, 13, 1, '2023-10-13'),
118 (61, 10, 2, '2023-10-13'),
119 (62, 9, 1, '2023-10-13'),
120 (63, 19, 2, '2023-10-13'),
121 (64, 20, 1, '2023-10-14')
```

Text to DDL

Query SQL

```
1  # 1. How many transactions were completed during each marketing campaign?
2
3  SELECT
4  campaign_name,
5  COUNT
6  (marketing_campaigns.product_id) AS total_transactions
7  FROM
8  marketing_campaigns
9
10 INNER JOIN transactions ON
11 transactions.product_id = marketing_campaigns.product_id -- we are inner joining transactions to
   marketing_campaigns using product_id key
12
13 GROUP BY campaign_name
14 ORDER BY total_transactions DESC;
```

Results

Copy as Markdown

Query #1 Execution time: 0ms

campaign_name	total_transactions
Summer Sale	7
New Collection Launch	6
Super Save	3

2. Which product had the highest sales quantity?

Schema SQL

```
product_id INT,
start_date DATE,
end_date DATE,
FOREIGN KEY (product_id) REFERENCES sustainable_clothing (product_id)
);
-- Insert data into the table
INSERT INTO marketing_campaigns (campaign_id, campaign_name, product_id, start_date, end_date)
VALUES
(1, 'Summer Sale', 2, '2023-06-01', '2023-06-30'),
(2, 'New Collection Launch', 10, '2023-07-15', '2023-08-15'),
(3, 'Super Save', 7, '2023-08-20', '2023-09-15');
-- Create the table
CREATE TABLE transactions (
transaction_id INT PRIMARY KEY,
product_id INT,
quantity INT,
purchase_date DATE,
FOREIGN KEY (product_id) REFERENCES sustainable_clothing (product_id)
);
-- Insert data into the table
INSERT INTO transactions (transaction_id, product_id, quantity, purchase_date)
VALUES
(1, 2, 2, '2023-06-02'),
```

Text to DDL

Query SQL

```
# 2. Which product had the highest sales quantity?

SELECT
(sustainable_clothing.product_name) AS name_of_product,-- to get the product name at initial level

SUM(transactions.quantity) AS total_qty_sold

FROM
transactions

LEFT JOIN sustainable_clothing ON
sustainable_clothing.product_id = transactions.product_id -- we are left joining sustainable_clothing to transactions using product_id key

GROUP BY product_name

ORDER BY total_qty_sold DESC

LIMIT 5;-- I have limited the rows to 5, there are more records
```

Results

Copy as Markdown

Query #1 Execution time: 0ms

name_of_product	total_qty_sold
Organic Cotton Sweater	9
Recycled Denim Jeans	8
Bamboo Yoga Leggings	8
Linen Jumpsuit	7
Organic Cotton Socks	7

3. What is the total revenue generated from each marketing campaign?

Schema SQL

```
VALUES
(1, 'Summer Sale', 2, '2023-06-01', '2023-06-30'),
(2, 'New Collection Launch', 10, '2023-07-15', '2023-08-15'),
(3, 'Super Save', 7, '2023-08-20', '2023-09-15');
-- Create the table
CREATE TABLE transactions (
transaction_id INT PRIMARY KEY,
product_id INT,
quantity INT,
purchase_date DATE,
FOREIGN KEY (product_id) REFERENCES sustainable_clothing (product_id)
);
-- Insert data into the table
INSERT INTO transactions (transaction_id, product_id, quantity, purchase_date)
VALUES
(1, 2, 2, '2023-06-02'),
(2, 14, 1, '2023-06-02'),
(3, 5, 2, '2023-06-05'),
(4, 2, 1, '2023-06-07'),
(5, 19, 2, '2023-06-10'),
(6, 2, 1, '2023-06-13'),
(7, 16, 1, '2023-06-13'),
```

Text to DDL

Query SQL

```
# 3. What is the total revenue generated from each marketing campaign?

SELECT

campaign_name,

ROUND(SUM(sustainable_clothing.price) * SUM(transactions.quantity),2) AS total_revenue -- We need to aggregate
price & -- quantity and multiply it.

FROM marketing_campaigns

INNER JOIN sustainable_clothing ON sustainable_clothing.product_id = marketing_campaigns.product_id -- As we are
finding revenue through campaign from product we are INNER JOINING through product_id key

INNER JOIN transactions ON marketing_campaigns.product_id = transactions.product_id -- in the 7th line we are
aggregating metrics from  2 different tables it is essential to join transactions to marketing_campaigns using
product_id key

GROUP BY 1 -- Groups the statement as campaign_name

ORDER BY 2 DESC; -- orders it by the highest return on ads

```

Results

Copy as Markdown

Query #1 Execution time: 0ms

campaign_name	total_revenue
Summer Sale	4479.44
New Collection Launch	2639.52
Super Save	629.91

4. What is the top-selling product category based on the total revenue generated?

Schema SQL

```sql
-- Create the table
CREATE TABLE sustainable_clothing (
product_id INT PRIMARY KEY,
product_name VARCHAR(100),
category VARCHAR(50),
size VARCHAR(10),
price FLOAT
);
-- Insert data into the table
INSERT INTO sustainable_clothing (product_id, product_name, category, size, price)
VALUES
(1, 'Organic Cotton T-Shirt', 'Tops', 'S', 29.99),
(2, 'Recycled Denim Jeans', 'Bottoms', 'M', 79.99),
(3, 'Hemp Crop Top', 'Tops', 'L', 24.99),
(4, 'Bamboo Lounge Pants', 'Bottoms', 'XS', 49.99),
(5, 'Eco-Friendly Hoodie', 'Outerwear', 'XL', 59.99),
(6, 'Linen Button-Down Shirt', 'Tops', 'M', 39.99),
(7, 'Organic Cotton Dress', 'Dresses', 'S', 69.99),
(8, 'Sustainable Swim Shorts', 'Swimwear', 'L', 34.99),
(9, 'Recycled Polyester Jacket', 'Outerwear', 'XL', 89.99),
(10, 'Bamboo Yoga Leggings', 'Activewear', 'XS', 54.99),
(11, 'Hemp Overalls', 'Bottoms', 'M', 74.99),
```

Text to DDL

Query SQL

```sql
# 4. What is the top-selling product category based on the total revenue generated?

SELECT

category,

ROUND(SUM(transactions.quantity) * SUM(sustainable_clothing.price), 2) AS total_revenue -- aggregating quanti
*price for total revenue

FROM sustainable_clothing

INNER JOIN transactions ON

sustainable_clothing.product_id = transactions.product_id -- We are inner joining sustainable_clothing to
transactions using product_id as key

GROUP BY category

ORDER BY total_revenue DESC;
```

Results

Copy as Markdown

category	total_revenue
Bottoms	21311.64
Outerwear	7319.04
Tops	6223.5
Dresses	4899.3
Accessories	3638.6
Activewear	2639.52
Footwear	639.84

5. Which products had a higher quantity sold compared to the average quantity sold?

Schema SQL

```sql
-- Create the table
CREATE TABLE sustainable_clothing (
product_id INT PRIMARY KEY,
product_name VARCHAR(100),
category VARCHAR(50),
size VARCHAR(10),
price FLOAT
);
-- Insert data into the table
INSERT INTO sustainable_clothing (product_id, product_name, category, size, price)
VALUES
(1, 'Organic Cotton T-Shirt', 'Tops', 'S', 29.99),
(2, 'Recycled Denim Jeans', 'Bottoms', 'M', 79.99),
(3, 'Hemp Crop Top', 'Tops', 'L', 24.99),
(4, 'Bamboo Lounge Pants', 'Bottoms', 'XS', 49.99),
(5, 'Eco-Friendly Hoodie', 'Outerwear', 'XL', 59.99),
(6, 'Linen Button-Down Shirt', 'Tops', 'M', 39.99),
(7, 'Organic Cotton Dress', 'Dresses', 'S', 69.99),
(8, 'Sustainable Swim Shorts', 'Swimwear', 'L', 34.99),
(9, 'Recycled Polyester Jacket', 'Outerwear', 'XL', 89.99),
(10, 'Bamboo Yoga Leggings', 'Activewear', 'XS', 54.99),
(11, 'Hemp Overalls', 'Bottoms', 'M', 74.99),
```

Text to DDL

Query SQL

```sql
# 5. Which products had a higher quantity sold compared to the average quantity sold?

SELECT

product_name,

SUM(transactions.quantity) AS total_qty,

ROUND(AVG(transactions.quantity), 2) AS avg_qty

FROM sustainable_clothing

INNER JOIN transactions ON

sustainable_clothing.product_id = transactions.product_id -- We are inner joining sustainable_clothing to transactions using product_id as key

GROUP BY 1

ORDER BY 2 DESC

LIMIT 7;
```

Results

Copy as Markdown

product_name	total_qty	avg_qty
Organic Cotton Sweater	9	1.80
Bamboo Yoga Leggings	8	1.33
Recycled Denim Jeans	8	1.14
Organic Cotton Socks	7	1.75
Linen Jumpsuit	7	1.75
Organic Cotton Skirt	5	1.67
Eco-Friendly Hoodie	5	1.67

6. What is the average revenue generated per day during the marketing campaigns?

Schema SQL

```sql
-- Create the table
CREATE TABLE sustainable_clothing (
product_id INT PRIMARY KEY,
product_name VARCHAR(100),
category VARCHAR(50),
size VARCHAR(10),
price FLOAT
);
-- Insert data into the table
INSERT INTO sustainable_clothing (product_id, product_name, category, size, price)
VALUES
(1, 'Organic Cotton T-Shirt', 'Tops', 'S', 29.99),
(2, 'Recycled Denim Jeans', 'Bottoms', 'M', 79.99),
(3, 'Hemp Crop Top', 'Tops', 'L', 24.99),
(4, 'Bamboo Lounge Pants', 'Bottoms', 'XS', 49.99),
(5, 'Eco-Friendly Hoodie', 'Outerwear', 'XL', 59.99),
(6, 'Linen Button-Down Shirt', 'Tops', 'M', 39.99),
(7, 'Organic Cotton Dress', 'Dresses', 'S', 69.99),
(8, 'Sustainable Swim Shorts', 'Swimwear', 'L', 34.99),
(9, 'Recycled Polyester Jacket', 'Outerwear', 'XL', 89.99),
(10, 'Bamboo Yoga Leggings', 'Activewear', 'XS', 54.99),
(11, 'Hemp Overalls', 'Bottoms', 'M', 74.99),
```

Text to DDL

Query SQL

```sql
# 6. What is the average revenue generated per day during the marketing campaigns?
SELECT
    mc.campaign_name,
    -- MIN(DAY(mc.start_date)) AS starting_day,
    -- MAX(DAY(mc.end_date)) AS ending_day,
    DATEDIFF(MAX(mc.end_date), MIN(mc.start_date)) AS campaign_period_days,
    ROUND(AVG(tr.avg_revenue), 2) AS avg_revenue_per_day
FROM marketing_campaigns mc
 INNER JOIN (
    SELECT
        DAY(t.purchase_date) AS transaction_day,
        t.product_id,
        ROUND(AVG(sc.price * t.quantity),2) avg_revenue
      FROM transactions t
      INNER JOIN sustainable_clothing sc ON t.product_id = sc.product_id
      GROUP BY 1, t.product_id
   ) AS tr ON mc.product_id = tr.product_id

 GROUP BY mc.campaign_name

   ORDER BY avg_revenue_per_day;

```

Results

Copy as Markdown

Query #1 Execution time: 1ms

campaign_name	campaign_period_days	avg_revenue_per_day
Super Save	26	69.99
New Collection Launch	31	73.32
Summer Sale	29	91.42

7. What is the percentage contribution of each product to the total revenue?

Schema SQL

```sql
-- Create the table
CREATE TABLE sustainable_clothing (
product_id INT PRIMARY KEY,
product_name VARCHAR(100),
category VARCHAR(50),
size VARCHAR(10),
price FLOAT
);
-- Insert data into the table
INSERT INTO sustainable_clothing (product_id, product_name, category, size, price)
VALUES
(1, 'Organic Cotton T-Shirt', 'Tops', 'S', 29.99),
(2, 'Recycled Denim Jeans', 'Bottoms', 'M', 79.99),
(3, 'Hemp Crop Top', 'Tops', 'L', 24.99),
(4, 'Bamboo Lounge Pants', 'Bottoms', 'XS', 49.99),
(5, 'Eco-Friendly Hoodie', 'Outerwear', 'XL', 59.99),
(6, 'Linen Button-Down Shirt', 'Tops', 'M', 39.99),
(7, 'Organic Cotton Dress', 'Dresses', 'S', 69.99),
(8, 'Sustainable Swim Shorts', 'Swimwear', 'L', 34.99),
(9, 'Recycled Polyester Jacket', 'Outerwear', 'XL', 89.99),
(10, 'Bamboo Yoga Leggings', 'Activewear', 'XS', 54.99),
(11, 'Hemp Overalls', 'Bottoms', 'M', 74.99),
```

Text to DDL

Query SQL

```sql
#7. What is the percentage contribution of each product to the total revenue?

SELECT
    sc.product_name,
    ROUND(SUM(tr.quantity * sc.price), 2) AS total_revenue,
    ROUND((SUM(tr.quantity * sc.price) / (SELECT SUM(tr.quantity * sc.price) FROM sustainable_clothing sc INNER JOIN transactions tr ON sc.product_id = tr.product_id)) * 100, 2) AS revenue_share
FROM
    sustainable_clothing sc
INNER JOIN
    transactions tr ON sc.product_id = tr.product_id
GROUP BY
    sc.product_name
ORDER BY
    total_revenue DESC
 LIMIT 7;
```

Results

Copy as Markdown

product_name	total_revenue	revenue_share
Recycled Denim Jeans	639.92	13.71
Linen Jumpsuit	489.93	10.49
Organic Cotton Sweater	449.91	9.64
Bamboo Yoga Leggings	439.92	9.42
Recycled Polyester Jacket	359.96	7.71
Eco-Friendly Hoodie	299.95	6.42
Bamboo Lounge Pants	249.95	5.35

8. Compare the average quantity sold during marketing campaigns to outside the marketing campaigns

Schema SQL

```
(20, 'Bamboo Bathrobe', 'Loungewear', 'XL', 69.99);
-- Create the table
CREATE TABLE marketing_campaigns (
campaign_id INT PRIMARY KEY,
campaign_name VARCHAR(100),
product_id INT,
start_date DATE,
end_date DATE,
FOREIGN KEY (product_id) REFERENCES sustainable_clothing (product_id)
);
-- Insert data into the table
INSERT INTO marketing_campaigns (campaign_id, campaign_name, product_id, start_date, end_date)
VALUES
(1, 'Summer Sale', 2, '2023-06-01', '2023-06-30'),
(2, 'New Collection Launch', 10, '2023-07-15', '2023-08-15'),
(3, 'Super Save', 7, '2023-08-20', '2023-09-15');
-- Create the table
CREATE TABLE transactions (
transaction_id INT PRIMARY KEY,
product_id INT,
quantity INT,
purchase_date DATE,
FOREIGN KEY (product_id) REFERENCES sustainable_clothing (product_id)
```

Text to DDL

Query SQL

```
#8. Compare the average quantity sold during marketing campaigns to outside the marketing campaigns

SELECT

    CASE WHEN mc.campaign_name IS NOT NULL THEN 'In Campaign' ELSE 'Outside Campaign' END AS campaign_category,
    ROUND(AVG(tr.quantity), 2) AS avg_qty
FROM
    transactions tr
LEFT JOIN
    marketing_campaigns mc ON mc.product_id = tr.product_id
GROUP BY
    campaign_category
ORDER BY
    campaign_category;
```

Results

Copy as Markdown

Query #1 Execution time: 1ms

campaign_category	avg_qty
In Campaign	1.19
Outside Campaign	1.44

9. Compare the revenue generated by products inside the marketing campaigns to outside the campaigns

Schema SQL

```sql
CREATE TABLE sustainable_clothing (
product_id INT PRIMARY KEY,
product_name VARCHAR(100),
category VARCHAR(50),
size VARCHAR(10),
price FLOAT
);
-- Insert data into the table
INSERT INTO sustainable_clothing (product_id, product_name, category, size, price)
VALUES
(1, 'Organic Cotton T-Shirt', 'Tops', 'S', 29.99),
(2, 'Recycled Denim Jeans', 'Bottoms', 'M', 79.99),
(3, 'Hemp Crop Top', 'Tops', 'L', 24.99),
(4, 'Bamboo Lounge Pants', 'Bottoms', 'XS', 49.99),
(5, 'Eco-Friendly Hoodie', 'Outerwear', 'XL', 59.99),
(6, 'Linen Button-Down Shirt', 'Tops', 'M', 39.99),
(7, 'Organic Cotton Dress', 'Dresses', 'S', 69.99),
(8, 'Sustainable Swim Shorts', 'Swimwear', 'L', 34.99),
(9, 'Recycled Polyester Jacket', 'Outerwear', 'XL', 89.99),
(10, 'Bamboo Yoga Leggings', 'Activewear', 'XS', 54.99),
(11, 'Hemp Overalls', 'Bottoms', 'M', 74.99),
(12, 'Organic Cotton Sweater', 'Tops', 'L', 49.99),
(13, 'Cork Sandals', 'Footwear', 'S', 39.99),
```

Text to DDL

Query SQL

```sql
#9. Compare the revenue generated by products inside the marketing campaigns to outside the campaigns

SELECT

    CASE WHEN mc.campaign_name IS NOT NULL THEN 'In Campaign' ELSE 'Outside Campaign' END AS campaign_category,
    ROUND(SUM(tr.quantity) * SUM(sc.price), 2) AS total_revenue
FROM
    transactions tr
LEFT JOIN
    marketing_campaigns mc ON mc.product_id = tr.product_id
LEFT JOIN
    sustainable_clothing sc ON sc.product_id = tr.product_id
GROUP BY
    campaign_category
ORDER BY
    campaign_category;
```

Results

Copy as Markdown

Query #1 Execution time: 1ms

campaign_category	total_revenue
In Campaign	20896.96
Outside Campaign	162461.88

10. Rank the products by their average daily quantity sold

Schema SQL

```
100 (43, 1, 1, '2023-10-01'),
101 (44, 7, 1, '2023-10-02'),
102 (45, 18, 2, '2023-10-03'),
103 (46, 12, 1, '2023-10-03'),
104 (47, 13, 1, '2023-10-04'),
105 (48, 4, 1, '2023-10-05'),
106 (49, 12, 2, '2023-10-05'),
107 (50, 7, 1, '2023-10-06'),
108 (51, 4, 2, '2023-10-08'),
109 (52, 8, 2, '2023-10-08'),
110 (53, 16, 1, '2023-10-09'),
111 (54, 19, 1, '2023-10-09'),
112 (55, 1, 1, '2023-10-10'),
113 (56, 18, 2, '2023-10-10'),
114 (57, 2, 1, '2023-10-10'),
115 (58, 15, 2, '2023-10-11'),
116 (59, 17, 2, '2023-10-13'),
117 (60, 13, 1, '2023-10-13'),
118 (61, 10, 2, '2023-10-13'),
119 (62, 9, 1, '2023-10-13'),
120 (63, 19, 2, '2023-10-13'),
121 (64, 20, 1, '2023-10-14')
```

Text to DDL

Query SQL

```
1  #10. Rank the products by their average daily quantity sold
2
3  SELECT
4  product_name,
5  DAY(tr.purchase_date) AS transaction_day,
6  ROUND(AVG(tr.quantity),0) AS avg_qty
7
8  FROM
9
10 sustainable_clothing sc
11
12 INNER JOIN
13
14 transactions tr ON tr.product_id = sc.product_id
15
16 GROUP BY product_name, DAY(tr.purchase_date)
17
18 ORDER BY transaction_day, avg_qty DESC;
19
20
21
22
```

Results

Copy as Markdown

Query #1 Execution time: 1ms

product_name	transaction_day	avg_qty
Organic Cotton Sweater	1	2
Organic Cotton T-Shirt	1	1
Hemp Baseball Cap	1	1
Recycled Denim Jeans	2	2
Organic Cotton Dress	2	1
Recycled Nylon Backpack	2	1
Linen Jumpsuit	3	2