SQL

Overview

This query using dataset ‘thelook_ecommerce’ who contains about customers, products, orders, logistics, web events, and digital marketing campaign. This dataset contains 7 table, there are distribution_centers, events, inventory_items, order_items, orders, products, and users

Table used

There are 4 tables used in this assignment, they are:

- products table
- orders table
- order_items table
- users table

Get number of unique users, number of orders, and total sale price per status and month

Scheme
(number of unique users and number of orders)

Field name	Type	Mode
total_unique_user	INTEGER	NULLABLE
total_order	INTEGER	NULLABLE

Scheme
(total sale price per status and month)

Field name	Type	Mode
month	INTEGER	NULLABLE
total_sale_price	FLOAT	NULLABLE
status	STRING	NULLABLE

Number of unique users and number of orders

Query

```
SELECT COUNT (DISTINCT table1.id) AS total_unique_user, COUNT (table2.total_order) as total_order
FROM
    (
      SELECT id, COUNT (DISTINCT id) AS total_unique_user
      FROM `bigquery-public-data.thelook_ecommerce.users`
      GROUP BY id
    ) AS table1
INNER JOIN
    (
      SELECT user_id, COUNT (order_id) AS total_order
      FROM `bigquery-public-data.thelook_ecommerce.orders`
      GROUP BY user_id
    ) AS table2
ON (table1.id = table2.user_id)
;
```

Result

total_unique_user	total_order
80107	80107

Total sale price per status and month

Query

```
SELECT EXTRACT(MONTH from date(created_at)) AS month
  ,(sale_price) AS total_sale_price
  ,status
FROM `bigquery-public-data.thelook_ecommerce.order_items`
WHERE 1=1
AND created_at BETWEEN '2022-01-01' AND '2022-08-31'
ORDER BY 1 ASC
```

Result

month	total_sale_price	status
1	3.5	Shipped
1	3.5	Complete
1	4.5	Shipped
1	5	Shipped
1	5	Shipped
1	5	Returned
1	5.5	Shipped
1	6	Shipped
1	6	Complete
1	6	Processing
1	6.25	Shipped
1	6.25	Complete
1	6.5	Shipped
1	6.5	Complete
1	6.5	Processing
1	6.5	Processing
1	6.5	Processing
1	7	Complete
1	7	Complete
1	7	Returned
1	7	Processing

Get frequencies, average order value and total number of unique users where status is complete grouped by month

Scheme

Field name	Type	Mode
month_year	DATE	NULLABLE
frequency	FLOAT	NULLABLE
aov	FLOAT	NULLABLE
number_of_unique_users	INTEGER	NULLABLE

Query

```
SELECT DATE_TRUNC(DATE(shipped_at), MONTH) AS month_year
      ,ROUND(COUNT(DISTINCT order_id)/COUNT(DISTINCT user_id),2) AS frequency
      ,ROUND(SUM(sale_price)/COUNT(DISTINCT order_id),2) AS AOV
      ,COUNT(DISTINCT(user_id)) AS number_of_unique_users
FROM `bigquery-public-data.thelook_ecommerce.order_items`
WHERE shipped_at BETWEEN '2019-01-01' AND '2022-09-01'
    AND status = 'Complete'
GROUP BY 1
ORDER BY 1
```

Result

month_year	frequency	AOV	number_of_unique_users
2019-01-01	1	173	8
2019-02-01	1	65.27	18
2019-03-01	1	76.82	36
2019-04-01	1.03	87.89	38
2019-05-01	1	84.36	69
2019-06-01	1	78.26	80
2019-07-01	1	91.61	111
2019-08-01	1	86.17	119
2019-09-01	1.01	87.25	143
2019-10-01	1	96.94	167
2019-11-01	1	92.02	184
2019-12-01	1	95.48	215
2020-01-01	1	87.25	243
2020-02-01	1	81.61	252
2020-03-01	1	87.31	261
2020-04-01	1.01	95.94	301
2020-05-01	1	88.64	295
2020-06-01	1	80.74	288
2020-07-01	1.01	82.94	398
2020-08-01	1	95.93	364

Find user id, email, first and last name of users whose status is refunded on Aug 22

Scheme

Field name	Type	Mode
id	INTEGER	NULLABLE
first_name	STRING	NULLABLE
last_name	STRING	NULLABLE
email	STRING	NULLABLE

Query

```sql
SELECT users.id
  ,first_name
  ,last_name
  ,email
FROM `bigquery-public-data.thelook_ecommerce.users` as users
INNER JOIN `bigquery-public-data.thelook_ecommerce.order_items` as order_items
ON users.id=order_items.user_id
WHERE users.created_at BETWEEN '2022-08-01' and '2022-08-31'
AND status = 'Returned'
```

Result

id	first_name	last_name	email
3916	David	Stevenson	davidstevenson@example.org
66612	Sherry	Miller	sherrymiller@example.net
70955	Dana	Steele	danasteele@example.net
43948	Christy	Petersen	christypetersen@example.com
42230	Emily	Robinson	emilyrobinson@example.org
98594	Casey	Petersen	caseypetersen@example.net
82452	Hannah	Garcia	hannahgarcia@example.com
30590	Matthew	Cruz	matthewcruz@example.org
25119	Joseph	Decker	josephdecker@example.org
69518	Yolanda	Bautista	yolandabautista@example.net
22685	Tammie	Jensen	tammiejensen@example.com
42687	Daniel	Hernandez	danielhernandez@example.com
73505	Kelly	Ruiz	kellyruiz@example.net
10794	William	Williams	williamwilliams@example.com
58771	Jacob	Archer	jacobarcher@example.com
69324	Mark	Mccall	markmccall@example.com
91853	Taylor	Harrison	taylorharrison@example.com
3156	Francisco	Larsen	franciscolarsen@example.com
3217	Jessica	Arnold	jessicaarnold@example.net
8036	Albert	Rodriguez	albertrodriguez@example.org
23863	Kirk	Smith	kirksmith@example.com

Get the top 5 least and most profitable product over all time

Scheme

Field name	Type	Mode
order_date	DATE	NULLABLE
product_category	STRING	NULLABLE
MTD	FLOAT	NULLABLE

Query

1.
```
WITH product_sales AS
(SELECT orders.product_id
      ,products.name
      ,products.retail_price
      ,products.cost
      ,orders.sale_price - products.cost AS profit
  FROM `bigquery-public-data.thelook_ecommerce.order_items` orders
  JOIN `bigquery-public-data.thelook_ecommerce.products` products
  ON orders.product_id = products.id
  WHERE status = 'Complete'),

product_profit AS
(SELECT product_id
      ,name
      ,retail_price
      ,cost
      ,SUM(profit) AS total_profit
  FROM product_sales
  WHERE name IS NOT NULL
  GROUP BY 1,2,3,4
  ORDER BY 2)
```

2.
```
(SELECT product_id
      ,name
      ,retail_price
      ,cost
      ,product_profit.total_profit
  FROM product_profit
  ORDER BY 5 DESC
  LIMIT 5)
  UNION ALL

(SELECT product_id
      ,name
      ,retail_price
      ,cost
      ,product_profit.total_profit
  FROM product_profit
  ORDER BY 5 ASC
  LIMIT 5)
```

Result

product_id	name	retail_price	cost	total_profit
14235	Indestructable Aluminum Aluma Wallet - RED	0.01999999955	0.008299999778	0.03509999932
28700	Wayfarer Style Sunglasses Dark Lens Black Frame	1.5	0.605999998	0.894000002
15332	Blank Long Cuff Beanie Cap (Choose Many Colors Available)	1.820000052	0.8918000261	0.9282000263
14159	Set of 2 - Replacement Insert For Checkbook Wallets Card Or Picture Insert	0.4900000095	0.1773800033	0.9378600186
9043	Pink Ribbon Breast Cancer Awareness Knee High Socks Great for Sports Teams Fundraising Relay for Life Walk Survivor (Style 26)	1.950000048	0.7683000158	1.181700032
23546	Alpha Industries Rip Stop Short	999	482.5169991	2582.415005
23646	Diesel Men's Lophophora Leather Jacket	898	408.5900009	2447.049995
7804	MiH Jeans Women's Aztec Jacket	495	169.7850011	1951.289993
24071	Men's Andrew Marc Nucky Lambskin Leather Bomber Jacket with Sheepskin Collar	479	205.9700007	1911.209995
18340	Canada Goose Men's The Chateau Jacket	815	337.4100014	1910.359994

Create a query to get month to date of total profit in each product categories of past 3 months (current date 15 august 2022), breakdown by month and categories

Scheme

Field name	Type	Mode
order_date	DATE	NULLABLE
product_category	STRING	NULLABLE
MTD	FLOAT	NULLABLE

Query

```sql
WITH profit AS
(SELECT DATE(order_items.shipped_at) AS order_date,
        product.category AS product_category,
        ROUND(SUM(order_items.sale_price - product.cost),2) AS category_profit
  FROM `bigquery-public-data.thelook_ecommerce.order_items` AS order_items
  INNER JOIN `bigquery-public-data.thelook_ecommerce.products` AS product
  ON order_items.product_id = product.id
  WHERE order_items.created_at BETWEEN "2022-01-01" AND "2022-08-16"
  AND status = "Complete"
  GROUP BY 1,2
  ORDER BY 2,1),

mtd_table AS
(SELECT order_date,
        product_category,
        category_profit,
        SUM(category_profit) OVER (PARTITION BY product_category
        ,EXTRACT(MONTH FROM order_date) ORDER BY product_category,order_date) AS MTD
  FROM profit
  ORDER BY 2,1)



mtd_table AS
(SELECT order_date,
        product_category,
        category_profit,
        SUM(category_profit) OVER (PARTITION BY product_category
        ,EXTRACT(MONTH FROM order_date) ORDER BY product_category,order_date) AS MTD
  FROM profit
  ORDER BY 2,1)

SELECT order_date,product_category, MTD
FROM mtd_table
WHERE order_date BETWEEN "2022-06-01" AND "2022-08-16"
  AND EXTRACT(DAY FROM order_date) = 15
```

Result

order_date	product_category	MTD
2022-06-15	Accessories	1483.51
2022-07-15	Accessories	1486.52
2022-08-15	Accessories	2607.88
2022-06-15	Active	1664.28
2022-07-15	Active	1603.6
2022-08-15	Active	1935.08
2022-06-15	Blazers & Jackets	1317.01
2022-08-15	Blazers & Jackets	906.62
2022-06-15	Dresses	1087.28
2022-07-15	Dresses	2280.85
2022-08-15	Dresses	1819.09
2022-06-15	Fashion Hoodies & Sweatshirts	1971.43
2022-07-15	Fashion Hoodies & Sweatshirts	1702.23
2022-08-15	Fashion Hoodies & Sweatshirts	2759.79
2022-06-15	Intimates	1075.17
2022-07-15	Intimates	1673.74
2022-08-15	Intimates	1553.58
2022-06-15	Jeans	3195.2
2022-07-15	Jeans	4520.91
2022-08-15	Jeans	4580.5
2022-06-15	Jumpsuits & Rompers	364.26