Bike Rental Shop: Case Study

Overview

Emily, the owner of a bike rental shop, is seeking your help as an SQL specialist to gather data that will support her business decisions. She wants to get insights into various aspects of the business, including rental revenue, membership data, and bike categories. To help Emily make informed decisions, you will be working with the shop's database, which contains five tables.

Database Structure

The database consists of the following five tables:

1. **Customer Table**: Contains details about customers of the bike rental shop.
   id, name, email

2. **Bike Table**: Stores information about the bikes the shop owns.
   id, model, category, price_per_hour, price_per_day, status

3. **Rental Table**: Links customers with bikes they've rented.
   id, customer_id, bike_id, start_timestamp, duration, total_paid

4. **Membership Type Table**: Information about the different membership types available for purchase.
   id, name, description, price

5. **Membership Table**: Stores details about individual memberships purchased by customers.
   id, membership_type_id, customer_id, start_date, end_date, total_paid

Emily's Business Questions

1. Bike Count by Category: How many bikes does the shop own in each category? Show only categories with more than 2 bikes.

2. Membership Purchases by Customer: What is the total number of memberships purchased by each customer?

3. Rental Price Discount for Winter Months: Calculate the discounted rental prices for different bike categories (e.g., electric, mountain, etc.).

4. Available vs. Rented Bikes: How many bikes are available and rented by category?

5.Revenue from Rentals: What is the total revenue from rentals by month, year, and all-time?

6.Revenue from Memberships: What is the total revenue from memberships by year, month, and membership type?

7. Revenue from Memberships in 2023: What is the total revenue from memberships purchased in 2023, broken down by month and membership type?

8. Customer Segmentation by Rentals: How many customers fall into each category based on the number of rentals (more than 10, between 5 and 10, fewer than 5)?

DATA PREPARATION

```
# creating schema for bike rental
create schema bike_rental_case_study;

# selecting bike rental schema
use bike_rental_case_study;

select * from customer;
select * from bike;
select * from rental;
select * from membership_type;
select * from membership;
```

Q1.
How many bikes does the shop own in each category? Show only categories with 2 or more bikes.

```sql
SELECT
    category , COUNT(*) as number_of_bike
FROM bike
GROUP BY category;
```

category	number_of_bike
mountain bike	3
road bike	3
hybrid	2
electric	2

Q2.
What is the total number of memberships purchased by each customer?

```sql
SELECT
    c.name,
    count(m.membership_type_id) as membership_count
FROM customer c
LEFT JOIN membership m
    ON c.id=m.customer_id
GROUP BY c.name
ORDER BY membership_count DESC;
```

name	membership_count
Alice Smith	3
Bob Johnson	3
John Doe	2
Eva Brown	2
Michael Lee	2
Sarah White	0
David Wilson	0
Emily Davis	0
Daniel Miller	0
Olivia Taylor	0

Q3.
Calculate the discounted rental prices for different bike categories (e.g., electric, mountain, etc.).

```sql
SELECT
    id , category ,
    price_per_hour AS old_price_per_hour ,
    CASE WHEN category = 'electric' THEN ROUND(price_per_hour* 0.9 , 2)
        WHEN category = 'mountain bike' THEN ROUND(price_per_hour*0.8,2)
        ELSE price_per_hour*0.5  END AS new_price_per_hour,
    price_per_day AS old_price_per_day,
    CASE WHEN category = 'electric' THEN ROUND(price_per_hour* 0.8 ,2)
        WHEN category = 'mountain bike' THEN ROUND(price_per_hour*0.5,2)
        ELSE price_per_hour*0.5 END AS new_price_per_day
FROM bike ;
```

id	category	old_price_per_hour	new_price_per_hour	old_price_per_day	new_price_per_day
1	mountain bike	10	8.0	50	5.0
2	road bike	12	6.0	60	6.0
3	hybrid	8	4.0	40	4.0
4	electric	15	13.5	75	12.0
5	mountain bike	10	8.0	50	5.0
6	road bike	12	6.0	60	6.0
7	hybrid	8	4.0	40	4.0
8	electric	15	13.5	75	12.0
9	mountain bike	10	8.0	50	5.0
10	road bike	12	6.0	60	6.0

```sql
SELECT
    category ,
    CASE WHEN status='available' THEN COUNT(status) ELSE 0 END AS available_bike_count,
    CASE WHEN status = 'rented' THEN COUNT(status) ELSE 0 END AS rented_bike_count
FROM bike
GROUP BY category,status;
```

category	available_bike_count	rented_bike_count
mountain bike	1	0
road bike	3	0
hybrid	0	1
electric	2	0
mountain bike	0	0
hybrid	0	0
mountain bike	0	1

```sql
SELECT
    YEAR(start_timestamp) AS year,
    MONTH(start_timestamp) AS month,
    SUM(total_paid) AS revneue
FROM rental
GROUP BY year, month WITH ROLLUP;
```

year	month	revneue
2022	11	200
2022	12	150
2022	NULL	350
2023	1	110
2023	2	40
2023	3	110
2023	4	90
2023	5	120
2023	6	115
2023	7	150
2023	8	125
2023	9	175
2023	10	335
2023	NULL	1370

```sql
SELECT
    YEAR(m.start_date) AS year,MONTH(m.start_date) AS month ,mt.name AS membership_type_name,
    SUM(total_paid) AS total_revenue
FROM membership m
JOIN membership_type mt
    ON m.membership_type_id=mt.id
GROUP BY year , month,membership_type_name;
```

year	month	membership_type_name	total_revenue
2023	8	Basic Annual	500
2023	8	Basic Monthly	100
2023	8	Premium Monthly	200
2023	9	Basic Monthly	100
2023	9	Basic Annual	500
2023	9	Premium Monthly	200
2023	10	Basic Monthly	100
2023	10	Basic Annual	500
2023	10	Premium Monthly	200
2023	11	Premium Monthly	200
2023	11	Basic Annual	500
2023	11	Basic Monthly	100

Q7
What is the total revenue from memberships purchased in 2023, broken down by month and membership type?

```sql
SELECT
    MONTH(m.start_date) AS month ,
    mt.name AS membership_type_name ,
    SUM(total_paid) AS total_revenue
FROM membership m
JOIN membership_type mt
    ON m.membership_type_id=mt.id
WHERE year(m.start_date) =2023
GROUP BY  membership_type_name,month WITH ROLLUP;
```

month	membership_type_name	total_revenue
8	Basic Annual	500
9	Basic Annual	500
10	Basic Annual	500
11	Basic Annual	500
NULL	Basic Annual	2000
8	Basic Monthly	100
9	Basic Monthly	100
10	Basic Monthly	100
11	Basic Monthly	100
NULL	Basic Monthly	400
8	Premium Monthly	200
9	Premium Monthly	200
10	Premium Monthly	200
11	Premium Monthly	200

Q8.
How many customers fall into each category based on the number of rentals (more than 10, between 5 and 10, fewer than 5)?

```sql
SELECT customer_id,name ,COUNT(*) AS customer_count,
    CASE WHEN count(*) >10 THEN 'more than 10'
        WHEN COUNT(*) >=5 THEN 'between 5 and 10'
        WHEN COUNT(*) <5 THEN 'fewer than 5'
        END AS rental_count_category
FROM rental r
JOIN customer c
    ON c.id=r.customer_id
GROUP BY customer_id ,name;
```

customer_id	name	customer_count	rental_count_category
1	John Doe	13	more than 10
2	Alice Smith	4	fewer than 5
3	Bob Johnson	4	fewer than 5
4	Eva Brown	1	fewer than 5
5	Michael Lee	1	fewer than 5
6	Sarah White	1	fewer than 5
7	David Wilson	1	fewer than 5
8	Emily Davis	1	fewer than 5
9	Daniel Miller	1	fewer than 5
10	Olivia Taylor	5	between 5 and 10

What Did We Learn?

- **Bike Inventory Analysis**: By categorizing and counting bikes, we helped Emily understand the distribution of bikes across categories and identify underrepresented categories.

- **Customer Engagement**: By analyzing the number of memberships purchased by each customer, Emily can identify loyal customers and create personalized marketing strategies.

- **Discounted Pricing**: By applying discount logic based on bike categories, we helped prepare special pricing for the winter months, making the business more attractive during the off-season.

- **Rental Revenue Tracking**: By calculating rental revenue by month and year, we provided insights into the shop's financial performance and seasonality trends.

- **Membership Revenue Analysis**: We helped Emily track membership revenue by type and time, enabling better planning for membership promotions and offers.

- **Customer Segmentation**: By categorizing customers based on rental frequency, Emily can focus on high-value customers while nurturing others to rent more often.