



Credit Card Spending

SQL Case Study

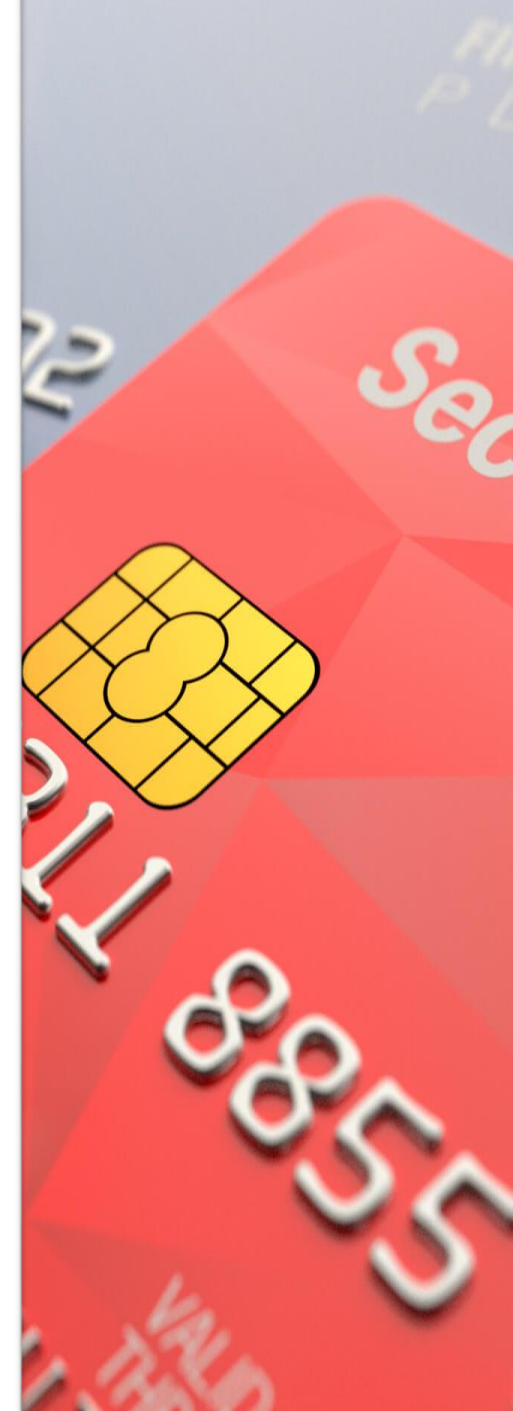
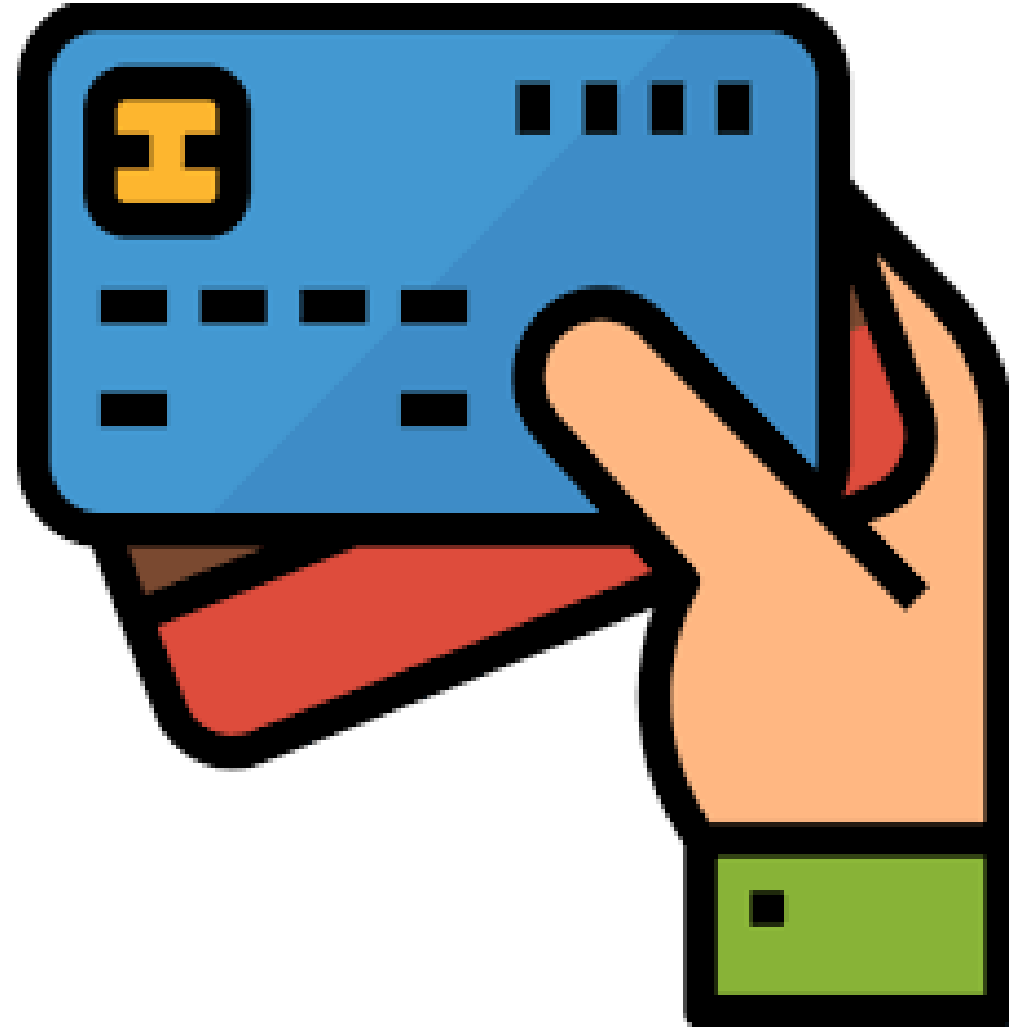
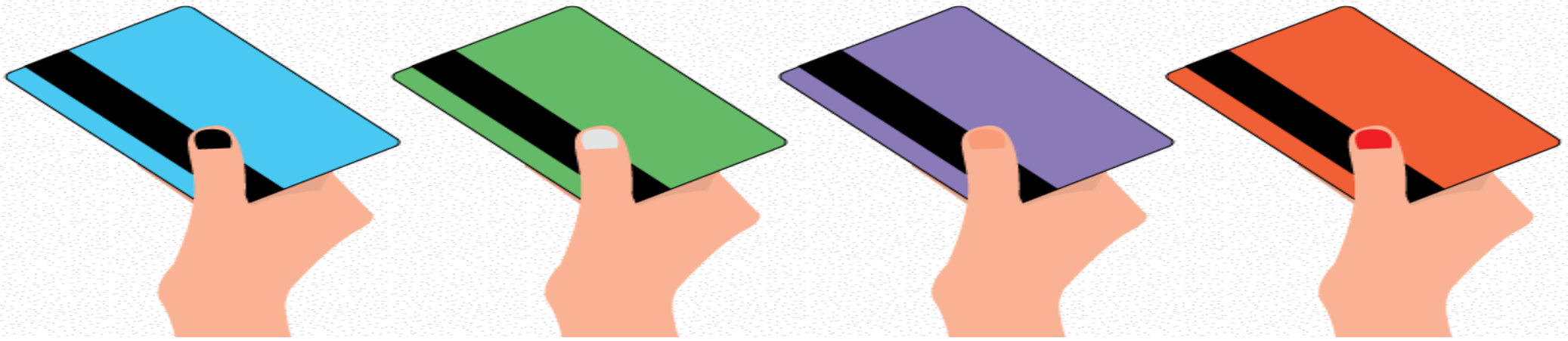


Table of Contents

- Introduction
- Problem Statement
- Datasets
- Insights
- Conclusion





Introduction

- In this personal project, we delve into the realm of data analytics using SQL queries to analyze credit card transaction data.
- Through a series of queries, we explore various aspects of spending patterns, customer behavior, and operational insights.
- By examining top spending cities, card types, growth trends, and operational metrics, this project aims to extract actionable insights to optimize business strategies and drive profitability.
- **RDBMS Used:** MYSQL



Problem Statement

- In this project, we aim to analyze credit card spending patterns to derive insights into customer behavior and preferences.
- The goal is to understand factors such as visit frequency, spending amounts, and preferred categories of expenditure. By leveraging credit card transaction data, we seek to identify trends and patterns that can inform strategic decisions related to customer engagement and loyalty programs.
- Additionally, we aim to create accessible datasets for easy examination by stakeholders, facilitating data-driven decision-making without the need for advanced SQL expertise



Datasets

We have 1 key dataset for this case study:

- Credit Card Sales

	transaction_id	city	transaction_date	card_type	exp_type	gender	amount
►	1	Delhi	29-10-2014	Gold	Bills	F	82475
	2	Greater Mumbai	22-08-2014	Platinum	Bills	F	32555
	3	Bengaluru	27-08-2014	Silver	Bills	F	101738
	4	Greater Mumbai	12-04-2014	Signature	Bills	F	123424
	5	Bengaluru	05-05-2015	Gold	Bills	F	171574
	6	Delhi	08-09-2014	Silver	Bills	F	100036
	7	Delhi	24-02-2015	Gold	Bills	F	143250
	8	Greater Mumbai	26-06-2014	Platinum	Bills	F	150980
	9	Delhi	28-03-2014	Silver	Bills	F	192247
	10	Delhi	01-09-2014	Platinum	Bills	F	67932

Discover the city with the lowest percentage spent for gold card holders.

```
with fetch_data as (  
  select  
    city,  
    card_type,  
    sum(case when card_type = 'Gold' then amount end) as gold_amount,  
    sum(amount) as total_amount  
  from  
    credit_card_trans  
  group by 1, 2  
  order by 1, 2  
)  
  
select  
  city,  
  ( sum(gold_amount) / sum(total_amount) ) * 100 as percentage  
from  
  fetch_data  
group by 1  
having sum(gold_amount) is not null  
order by 2  
limit 1
```

	city	percentage
►	Dhamtari	0.3330



Discover the city with the lowest percentage spent for gold card holders.

```
with fetch_data as (  
    select  
    city,  
    card_type,  
    sum(case when card_type = 'Gold' then amount end) as gold_amount,  
    sum(amount) as total_amount  
    from  
    credit_card_trans  
    group by 1, 2  
    order by 1, 2  
)  
  
select  
city,  
( sum(gold_amount) / sum(total_amount) ) * 100 as percentage  
from  
fetch_data  
group by 1  
having sum(gold_amount) is not null  
order by 2  
limit 1
```

	city	percentage
►	Dhamtari	0.3330



```
# top 5 cities that lead in credit card spending, and what percentage do they contribute to the total?
```

```
with fetch_data as (  
  select  
    city,  
    sum(amount) as total_amount_spent,  
    (select sum(amount) from credit_card_trans) as whole_amount  
  from  
    credit_card_trans  
  group by city  
  order by 2 desc  
)  
  
select  
  city,  
  total_amount_spent,  
  round((total_amount_spent / whole_amount) * 100,2) as contribution_percentage  
from  
  fetch_data  
limit 5;
```

	city	total_amount_spent	contribution_percentage
►	Greater Mumbai	576751476	14.15
	Bengaluru	572326739	14.05
	Ahmedabad	567794310	13.93
	Delhi	556929212	13.67
	Kolkata	115466943	2.83


```
# Generate a dynamic report showcasing city, highest_expense_type, and lowest_expense_type.
```

```
with fetch_data as (  
  select  
    city,  
    exp_type,  
    sum(amount) as total_amount  
  from  
    credit_card_trans  
  group by 1,2  
  order by 1  
)  
  
min_and_max_expense as (  
  select  
    city,  
    min(total_amount) as min_spent,  
    max(total_amount) as max_spent  
  from  
    fetch_data  
  group by 1  
)  
  
select  
  m1.city,  
  max(case when m1.max_spent = total_amount then exp_type end) as highest_expense_type,  
  min(case when m1.min_spent = total_amount then exp_type end) as lowest_expense_type  
  from  
    min_and_max_expense as m1  
  join  
    fetch_data as f  
  on m1.city = f.city  
  group by 1
```

	city	highest_expense_type	lowest_expense_type
▶	Achalpur	Grocery	Entertainment
	Adilabad	Bills	Food
	Adityapur	Food	Grocery
	Adoni	Bills	Entertainment
	Adoor	Fuel	Bills
	Afzalpur	Fuel	Food
	Agartala	Grocery	Food
	Agra	Bills	Grocery
	Ahmedabad	Bills	Grocery
	Ahmednagar	Fuel	Grocery

```
# Identify the city with the highest spending-to-transaction ratio on weekends.
```

```
with fetch_data as (  
    select  
        transaction_id,  
        city,  
        STR_TO_DATE(transaction_date, '%d-%m-%Y') as transaction_date,  
        card_type,  
        exp_type,  
        gender,  
        amount  
    from  
        credit_card_trans  
)  
  
filter_data as (  
    select  
        *,  
        dayofweek(transaction_date) as week_day  
    from  
        fetch_data  
    where dayofweek(transaction_date) in (1,7)  
)  
  
select  
    city,  
    sum(amount) / count(transaction_id) as spending_to_transaction_ratio  
from  
    filter_data  
group by city  
order by 2 desc  
limit 1
```

	city	spending_to_transaction_ratio
►	Sonepur	299905.0000

Discover the city with the lowest percentage spent for gold card holders.

```
with fetch_data as (  
  select  
    city,  
    card_type,  
    sum(case when card_type = 'Gold' then amount end) as gold_amount,  
    sum(amount) as total_amount  
  from  
    credit_card_trans  
  group by 1, 2  
  order by 1, 2  
)  
  
select  
  city,  
  ( sum(gold_amount) / sum(total_amount) ) * 100 as percentage  
from  
  fetch_data  
group by 1  
having sum(gold_amount) is not null  
order by 2  
limit 1
```

	city	percentage
►	Dhamtari	0.3330

```

# Retrieve transaction details for each card type upon reaching a cumulative spend of 1,000,000.

with fetch_data as (
    select
        transaction_id,
        city,
        card_type,
        exp_type,
        gender,
        sum(amount) over(partition by card_type order by transaction_date,transaction_id) as cumulative_sum
    from
        credit_card_trans
),

filter_data as (
    select
        *,
        dense_rank() over(partition by card_type order by cumulative_sum) as ranking_order
    from
        fetch_data
    where cumulative_sum >= 1000000
)

select
    transaction_id,
    city,
    card_type,
    exp_type,
    gender
from
    filter_data
where ranking_order = 1

```

transaction_id	city	transaction_date	card_type	exp_type	gender	amount	cumulative_sum
23644	Chennai	01-01-2014	Gold	Grocery	M	129379	1050417
16494	Ozar	01-01-2014	Platinum	Bills	F	199575	1051910
7790	Bengaluru	01-01-2014	Signature	Bills	F	292221	1268866
741	Ahmedabad	01-01-2015	Silver	Food	F	209232	1197835

```
# Uncover the percentage contribution of female spending across different expense type

# the reason behind not using where condition over here is we need female contribution over "different
expense type" that means from the sum of amounts of different expense types plays a pivot role and
among them we need to get the share of female contributors.

# so the group by takes care of expense type and returns the overall sum of each of the expense type

# now using a case statment on that will return female contribution and contribution percentage

# using a where condition here will skew the results and will constantly result in same overall sum
value irrespective of expense type so that has to be avoided

select
exp_type,
(sum(case when gender = 'F' then amount else 0 end ) / sum(amount)) * 100 as percentage_contribution
from
credit_card_trans
group by exp_type
```

	exp_type	percentage_contribution
►	Bills	63.9459
	Food	54.9053
	Entertainment	49.3729
	Grocery	50.9110
	Fuel	49.7104
	Travel	51.1329

```
# Find the card and expense type combination with the highest month over month growth in January 2014
```

```
with fetch_data as (  
    select  
        transaction_id,  
        city,  
        STR_TO_DATE(transaction_date, '%d-%m-%Y') as transaction_date,  
        card_type,  
        exp_type,  
        gender,  
        amount  
    from  
        credit_card_trans  
)  
  
filter_data as (  
    select  
        card_type,  
        exp_type,  
        extract(year from transaction_date) as year,  
        extract(month from transaction_date) as month,  
        sum(amount) as total_amount,  
        lag(sum(amount)) over(partition by card_type, exp_type order by extract(year from transaction_date),  
        extract(month from transaction_date)) as previous_month  
    from  
        fetch_data  
    group by card_type, exp_type, extract(year from transaction_date), extract(month from  
transaction_date)  
)  
  
select  
*,  
((total_amount - previous_month) / previous_month) * 100 as growth  
from  
filter_data  
where previous_month is not null and year = 2014 and month = 1  
order by growth desc  
limit 1
```

	card_type	exp_type	year	month	total_amount	previous_month	growth
►	Gold	Travel	2014	1	2092554	1113534	87.9201

How can we determine the highest spending month and amounts for each card type?

```
with fetch_data as (  
    select  
        transaction_id,  
        city,  
        STR_TO_DATE(transaction_date, '%d-%m-%Y') as transaction_date,  
        card_type,  
        exp_type,  
        gender,  
        amount  
    from  
        credit_card_trans  
),  
  
filter_data as (  
    select  
        card_type,  
        month(transaction_date) as month,  
        year(transaction_date) as year,  
        sum(amount) as total_spent,  
        dense_rank() over(partition by card_type order by sum(amount) desc) as ranking_order  
    from  
        fetch_data  
    group by 1, 2, 3  
)  
  
select  
    card_type,  
    month as highest_spending_month,  
    year,  
    total_spent  
from  
    filter_data  
where ranking_order = 1
```

	card_type	highest_spending_month	year	total_spent
►	Gold	1	2015	55455064
	Platinum	8	2014	57936507
	Signature	12	2013	58799522
	Silver	3	2015	59723549

```
# Explore the city that achieved its 500th transaction in the shortest time after the first.
```

```
with fetch_data as (  
    select  
        transaction_id,  
        city,  
        STR_TO_DATE(transaction_date, '%d-%m-%Y') as transaction_date,  
        card_type,  
        exp_type,  
        gender,  
        amount,  
        dense_rank() over(partition by city order by STR_TO_DATE(transaction_date, '%d-%m-%Y'), transaction_id) as ranking_order  
    from  
        credit_card_trans  
)
```

```
days_took as (  
    select  
        city,  
        min(transaction_date) as min_date,  
        max(transaction_date) as max_date,  
        datediff(max(transaction_date), min(transaction_date)) as days_took  
    from  
        fetch_data  
    where ranking_order = 1 or ranking_order = 500  
    group by city  
    having count(ranking_order) = 2  
)
```

```
select  
    *  
from  
    days_took  
order by days_took  
limit 1
```

	city	min_date	max_date	days_took
►	Bengaluru	2013-10-04	2013-12-24	81

Conclusion

- **Customer A** has made the highest total purchase at the restaurant.
- **Customer B** was the most frequent visitor to the restaurant.
- The customers' first orders were **Sushi**, **Curry** and **Ramen** for Customer A, Customer B, and Customer C respectively.
- **Ramen** was the most purchased item on the menu.
- **Curry** and **Sushi** was purchased first by the Customer A and B respectively after they became a member.
- Before becoming a member, **Customer A** spent \$25 on 2 items whereas **Customer B** spent \$40 on 3 items.



A wide-angle photograph of a modern restaurant interior. The ceiling is a prominent feature, composed of a grid of wooden panels with recessed lighting. A long, dark wooden bar runs across the middle of the frame, with several dark wooden chairs tucked under it. In the background, there is a large, open kitchen area with a glass partition. To the right, there are more tables and chairs, some with green cushions. The floor is polished and reflects the overhead lights. The overall atmosphere is warm and contemporary.

Thank you !