



CREDIT CARD DATA ANALYSIS

Presented By:
Niroj Lawati





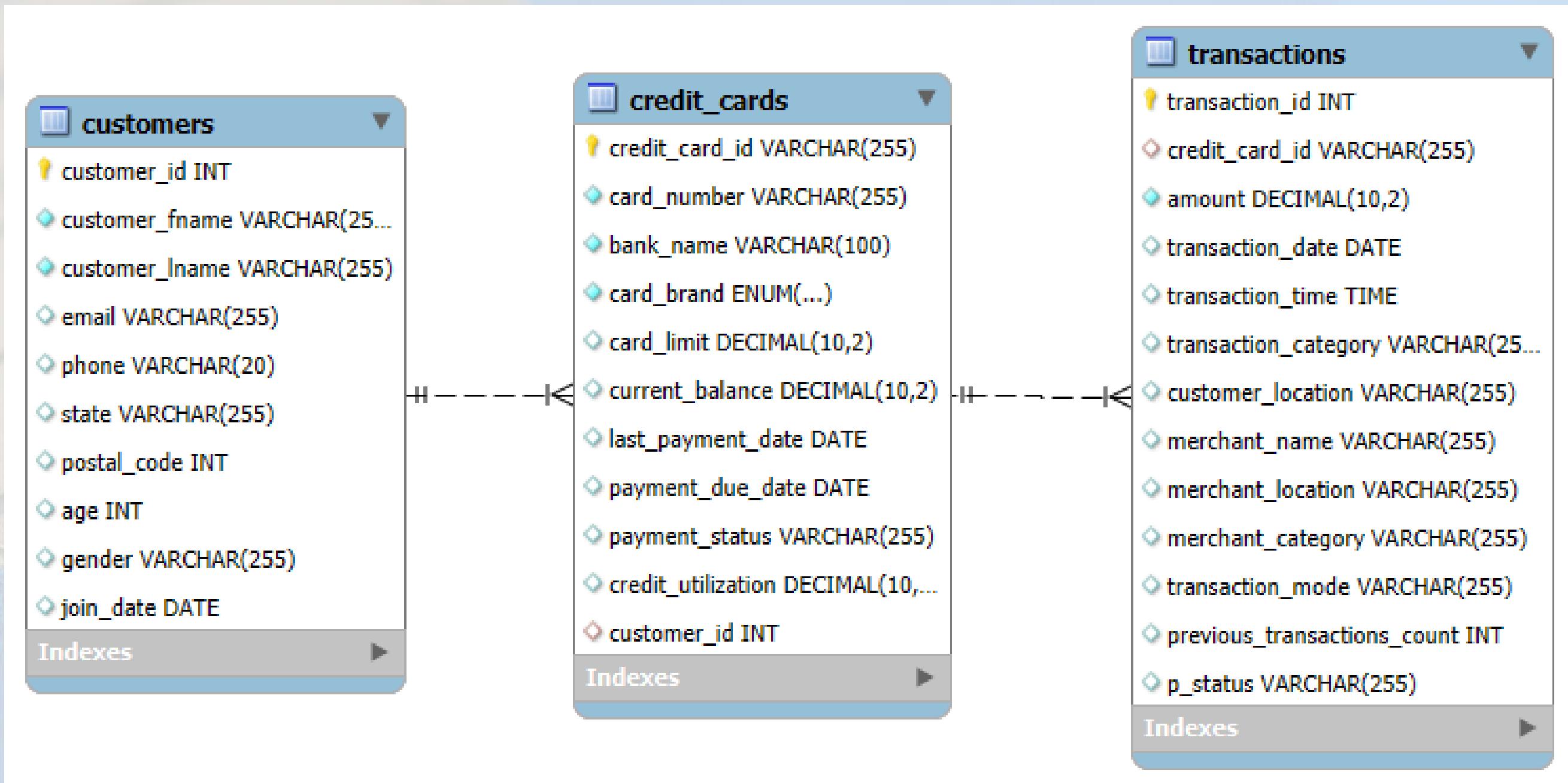
ABOUT PROJECT

This project focuses on analyzing credit card data to uncover key insights into customer behavior, spending patterns, and business performance.

The analysis utilizes structured data stored in a relational database, with tables for customers, credit cards, and transactions. The project aims to deliver valuable insights for decision-making, such as identifying high-value customers, understanding spending trends, and improving risk management.



ER DIAGRAM



1. FIND TRANSACTIONS WHERE CREDIT_CARD_ID IS MISSING.

```
select *  
from creditcard.transactions t1  
where t1.credit_card_id is null;
```



	transaction_id	credit_card_id	amount	transaction_date
*	NULL	NULL	NULL	NULL
...
...
...

• • • • • • •

2. IDENTIFY CREDIT CARDS THAT REFERENCE A NON-EXISTING CUSTOMER.

```
select *  
from creditcard.credit_cards c  
left join creditcard.customers c2  
on c2.customer_id = c.customer_id  
where c.customer_id is null;
```

credit_card_id	card_number	bank_name	card_brand	card_limit

.....

3. DELETE TRANSACTIONS WITH A MISSING OR INCORRECT CREDIT_CARD_ID.

```
set sql_safe_updates = 0;
delete from creditcard.transactions
where credit_card_id is null
or credit_card_id not in (
    select credit_card_id from creditcard.credit_cards
);
```



Output

Action Output

#	Time	Action	Message
1	10:22:11	set sql_safe_updates = 0	0 row(s) affected
2	10:22:16	delete from creditcard.transactions where credit_card_id is null or credit_card_id not in (select credit_card_id from creditcard.credit_cards)	0 row(s) affected

4. REMOVE CREDIT CARDS LINKED TO CUSTOMERS WHO DON'T EXIST IN THE CUSTOMER TABLE.

```
set sql_safe_updates = 0;
delete from creditcard.credit_cards c
where c.customer_id not in (
    select c2.customer_id from creditcard.customers c2
)
```

Action Output			
#	Time	Action	Message
1	10:24:34	set sql_safe_updates = 0	0 row(s) affected
2	10:24:43	delete from creditcard.credit_cards c where c.customer_id not in (select c2.customer_id from creditcard.customers c2)	0 row(s) affected

5. TOTAL NUMBER OF TRANSACTION

```
select count(*) as total_transaction  
from creditcard.transactions t1;
```

Result Grid	
	total_transaction
▶	12327



.....

6. WHAT IS THE TOTAL AMOUNT SPENT?

```
select sum(t1.amount) as total_amount  
from creditcard.transactions t1;
```

Result Grid | Filter Rows:

	total_amount
▶	33978710.62



.....

7. HOW DOES SPENDING CHANGE MONTH BY MONTH?

```
select  
month(t1.transaction_date) as month,  
dense_rank() over(order by week(t1.transaction_date)) as week,  
sum(amount) as total_spent  
from creditcard.transactions t1  
group by month(t1.transaction_date), week(t1.transaction_date)  
order by week;
```

VALID DATES 00/00

	month	week	total_spent
▶	10	1	4016904.20
	10	2	7537706.84
	10	3	6876256.21
	10	4	10744617.81
	10	5	4803225.56

.....

8. THE TOP 5 CUSTOMERS BY TOTAL SPENDING

```
select dense_rank() over (order by SUM(t1.amount) desc) as Top_spender,
concat(c1.customer_fname, ' ', c1.customer_lname) as full_name,
sum(t1.amount) as total_spending
from creditcard.customers c1
inner join creditcard.credit_cards cc1
on cc1.customer_id = c1.customer_id
inner join creditcard.transactions t1
on t1.credit_card_id = cc1.credit_card_id
group by c1.customer_id, c1.customer_fname, c1.customer_lname
order by total_spending desc
limit 5;
```

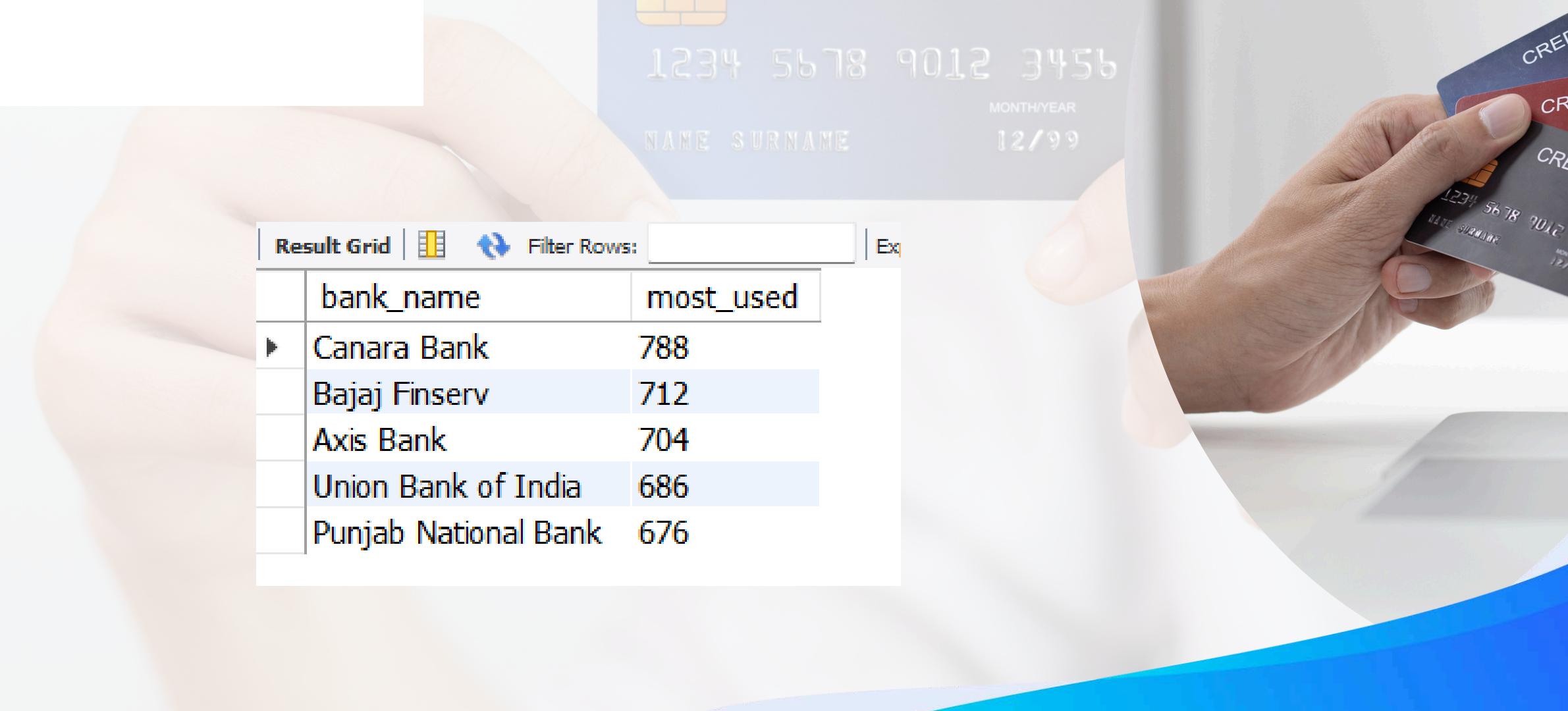
Result Grid | Filter Rows: _____ | Export: Wrap

	Top_spender	full_name	total_spending
1	Eshana Wagle	90086.47	
2	Jagat Doshi	78410.31	
3	Vamakshi Handa	77699.32	
4	Vihaan Setty	75598.91	
5	Abhimanyu Raju	74345.51	



9. WHICH BANK'S CREDIT CARDS ARE USED THE MOST?

```
select cc1.bank_name, count(transaction_id) as most_used  
from creditcard.credit_cards cc1  
inner join creditcard.transactions t1  
on t1.credit_card_id = cc1.credit_card_id  
group by cc1.bank_name  
order by most_used desc  
limit 5;
```



A hand holds a light blue credit card with the text "CREDIT CARD" at the top. The card features a yellow chip, the number "1234 5678 9012 3456", and the expiration date "12/99". Below the card is a software interface showing a "Result Grid" with the following data:

	bank_name	most_used
▶	Canara Bank	788
	Bajaj Finserv	712
	Axis Bank	704
	Union Bank of India	686
	Punjab National Bank	676

• • • • • • • •

10. BIG SPENDERS: - CUSTOMERS WITH THE HIGHEST SPENDING

```
select c1.customer_id, c1.customer_fname, c1.customer_lname, sum(t1.amount) as highest_spending
from creditcard.customers c1
inner join creditcard.credit_cards cc1
on cc1.customer_id = c1.customer_id
inner join creditcard.transactions t1
on t1.credit_card_id = cc1.credit_card_id
group by c1.customer_id, c1.customer_fname, c1.customer_lname
order by highest_spending desc
limit 10;
```

customer_id	customer_fname	customer_lname	highest_spending
1463	Eshana	Wagle	90086.47
244	Jagat	Doshi	78410.31
1513	Vamakshi	Handa	77699.32
788	Vihaan	Setty	75598.91
1257	Abhimanyu	Raju	74345.51
365	Hardik	Saini	73955.21
1266	Indali	Deshpande	72583.13
750	Yashasvi	Mani	72068.88
1051	Vritti	Kala	72058.88
294	Gagan	Singh	70826.09
⋮	⋮	⋮	⋮



11. BANKS WITH THE HIGHEST TRANSACTIONS CAN ATTRACT MORE CUSTOMERS.

```
select cc1.bank_name, sum(t1.amount) as highest_transaction  
from creditcard.credit_cards cc1  
inner join creditcard.transactions t1  
on t1.credit_card_id = cc1.credit_card_id  
group by cc1.bank_name  
order by highest_transaction desc  
limit 5;
```

bank_name	highest_transaction
Canara Bank	2171064.99
Bajaj Finserv	1953746.38
Axis Bank	1911830.39
Union Bank of India	1880907.76
Bank of Maharashtra	1876609.57



• • • • • • • •

12. IDENTIFYING PEAK SPENDING TIMES CAN HELP IN MARKETING AND PROMOTIONS.

```
select dayname(t1.transaction_date) as days,  
sum(t1.amount) as total_spending  
from creditcard.transactions t1  
group by dayname(t1.transaction_date)  
order by total_spending desc;
```

Result Grid | Filter Rows:

	days	total_spending
▶	Thursday	6921250.05
	Friday	6159600.92
	Tuesday	6049373.44
	Wednesday	4838041.69
	Sunday	3957574.80
	Saturday	3693317.68
	Monday	2359552.04





THANK YOU

