PURCHASING PATTERNS & LOYALTY ANALYTICS

by Chirag Sharma

1.What is the total purchase amount for each customer across all categories?

Query | Query History

```
1  select customername, sum(amount) as Total_amount
2  from customers
3  group by 1
4  order by 2 desc
5
```

Data Output | Messages | Notifications

	customername character varying (100)	total_amount numeric
1	Michael Smith	605469.00
2	Michael Johnson	595251.00
3	John Smith	545150.00
4	James Smith	540245.00
5	Michael Brown	522981.00
6	Jennifer Smith	479199.00
7	Michael Jones	452720.00
8	David Johnson	415818.00
9	Michael Williams	413358.00
10	David Smith	386086.00
11	Christopher Jones	383500.00

2. Retrieve the top 5 customers with the highest total purchase amount in the dataset.

Query | Query History

```sql
select customername, sum(amount) as Total_amounts
from customers
group by 1
order by 2 desc
limit 5
```

Data Output | Messages | Notifications

	customername character varying (100)	total_amounts numeric
1	Michael Smith	605469.00
2	Michael Johnson	595251.00
3	John Smith	545150.00
4	James Smith	540245.00
5	Michael Brown	522981.00

3. What is the average purchase amount for each product category across all customers?

Query | Query History

```
1  select productcategory,
2  round(avg(amount),1) as avg_amount
3  from customers
4  group by 1
```

Data Output | Messages | Notifications

	productcategory character varying (50)	avg_amount numeric
1	Books	2728.8
2	Clothing	2726.4
3	Electronics	2719.0
4	Home	2726.2

4. Find the customer who has returned the most items and calculate their total returns.

Query | Query History

```sql
select customername, sum(returns) as Total_returns
from customers
where returns is not null
group by 1
order by 2 desc
limit 1
```

Data Output | Messages | Notifications

	customername character varying (100)	total_returns bigint
1	Michael Johnson	91

5. How many purchases were made using each payment method, and what is the total amount for each method?

Query | Query History

```
1  select paymentmethod, count(*) as Total_Purchases,
2  sum(amount) as Total_amount
3  from customers
4  group by 1
5  order by 2 desc
```

Data Output | Messages | Notifications

	paymentmethod character varying (50)	total_purchases bigint	total_amount numeric
1	Credit Card	184033	502975311.00
2	PayPal	158278	431126143.00
3	Cash	132906	361129405.00
4	Crypto	24783	67458123.00

6. Identify the top 3 most frequently purchased product categories by each customer.

Query | Query History

```sql
select productcategory, customername, total_purchases from
(select customername,
productcategory,
count(*) as Total_purchases,
row_number()over(partition by customername order by count(*) desc) as ranks
from customers
group by 1,2)as d
where ranks <= 3
```

Data Output | Messages | Notifications

	productcategory character varying (50)	customername character varying (100)	total_purchases bigint
1	Books	Aaron Acosta	3
2	Home	Aaron Acosta	1
3	Electronics	Aaron Acosta	1
4	Clothing	Aaron Alexander	6
5	Books	Aaron Alexander	4
6	Home	Aaron Alexander	3
7	Electronics	Aaron Archer	2
8	Books	Aaron Archer	2
9	Clothing	Aaron Armstrong	3
10	Books	Aaron Armstrong	2
11	Home	Aaron Armstrong	1

7. Calculate the total number of purchases and total amount spent by customers above the age of 50.

Query | Query History

```
1  select count(*) as Total_Purchases,
2  sum(amount) as Total_amount
3  from customers
4  where customerage > 50
5
```

Data Output | Messages | Notifications

	total_purchases bigint	total_amount numeric
1	187342	526952647.00

8. Find all customers who have made purchases in more than two different product categories.

Query | Query History

```
1  select customername
2  from customers
3  group by customername
4  having count(distinct productcategory) > 2
```

Data Output | Messages | Noti

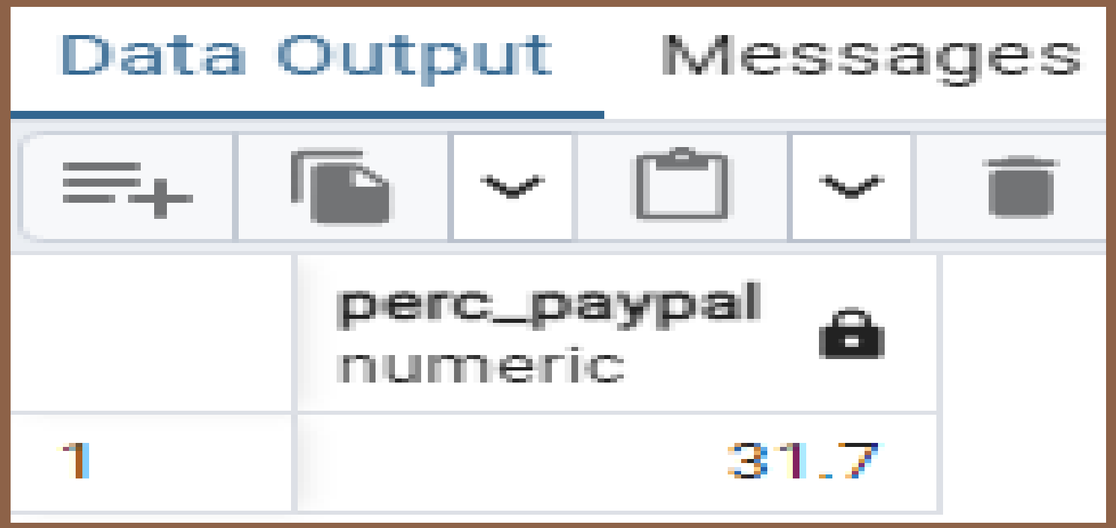
	customername character varying (100)
1	Aaron Acosta
2	Aaron Alexander
3	Aaron Armstrong
4	Aaron Austin
5	Aaron Avery
6	Aaron Avila
7	Aaron Ayala
8	Aaron Bailey
9	Aaron Baker
10	Aaron Barber
11	Aaron Bautista

9. Determine the percentage of purchases made via PayPal compared to other payment methods.

Query | Query History

```sql
SELECT
    round((SUM(CASE WHEN PaymentMethod = 'PayPal' THEN 1 ELSE 0 END) * 100.0) / COUNT(*),1) AS perc_paypal
FROM
    customers;

```



10. What is the average product price for each product category and the total purchase amount for each category?

Query | Query History

```
1  select productcategory, round(avg(amount),1) as Avg_price,
2  round(sum(amount),0) as Total_amount
3  from customers
4  group by 1
```

Data Output | Messages | Notifications

	productcategory character varying (50)	avg_price numeric	total_amount numeric
1	Books	2728.8	374284837
2	Clothing	2726.4	375248527
3	Electronics	2719.0	306745492
4	Home	2726.2	306410126

11. List the customers who have made purchases in every year present in the dataset

Query | Query History

```sql
with customeryears as (
    select customername, extract(year from purchasedate) as years
    from customers
),

second as (
    select count(distinct extract(years from purchasedate)) as total_years
    from customers
)

select customername
from customeryears
group by 1
having count(distinct years) = (select total_years from second)
```

	customername character varying (100)
1	Aaron Alexander
2	Aaron Avila
3	Aaron Ayala
4	Aaron Baker
5	Aaron Bautista
6	Aaron Boone
7	Aaron Boyer
8	Aaron Brady
9	Aaron Cantrell
10	Aaron Chapman
11	Aaron Clark
12	Aaron Colon
13	Aaron Cook
14	Aaron Curry
15	Aaron David

12. Find the customer with the highest number of purchases made in a single month, along with the month and year.

Query | Query History

```sql
select customername, to_char(date, 'month') as month, extract(year from date) as Years,
count(*) as Total_Transactions
from customers
where to_char(date, 'month') like 'september' and
extract(year from date) = 2022
group by 1,2,3
order by 4 desc
limit 3
```

Data Output | Messages | Notifications

	customername character varying (100)	month text	years numeric	total_transactions bigint
1	Michael Johnson	september	2022	11
2	Jennifer Smith	september	2022	7
3	Matthew Smith	september	2022	7

13. Calculate the churn rate for male and female customers.

Query | Query History

```sql
select gender,
round(((sum(churn))*100.0/(count(churn))),1) as Churn_Rate
from customers
group by 1
```

Data Output | Messages | Notifications

	gender character varying (10)	churn_rate numeric
1	Female	19.9
2	Male	20.1

14. Retrieve the total purchase amount, number of purchases, and average purchase amount for each customer, grouped by gender.

Query | Query History

```
1  select customername, gender, count(*) as Total_Purchases,
2  sum(amount) as Total_purchase_amount,
3  round(avg(amount),1) as Avg_purchase_Amount
4  from customers
5  group by 1,2
6  order by 3 desc
```

Data Output | Messages | Notifications

	customername character varying (100)	gender character varying (10)	total_purchases bigint	total_purchase_amount numeric	avg_purchase_amount numeric
1	John Smith	Male	128	355617.00	2778.3
2	Michael Smith	Female	127	341791.00	2691.3
3	James Smith	Female	123	363873.00	2958.3
4	Michael Johnson	Male	117	311219.00	2660.0
5	David Johnson	Male	108	295535.00	2736.4
6	Michael Johnson	Female	100	284032.00	2840.3
7	Michael Brown	Female	100	289752.00	2897.5
8	Michael Smith	Male	96	263678.00	2746.6
9	Jennifer Smith	Male	95	267505.00	2815.8

15. Identify the most purchased product category by each age group (e.g., 18-25, 26-35, etc.).

Query | Query History

```sql
select * from
(select *, rank()over(partition by age_group order by total_purchases desc) as ranks from
(select productcategory,
    case
    when age >= 18 and age <= 25 then '18-25'
    when age > 25 and age <= 35 then '25-35'
    when age > 35 and age <= 45 then '35-45'
    when age > 45 and age <= 55 then '45-55'
    when age > 55 and age <= 65 then '55-65'
    else '65+'
    end as Age_group,
    count(*) as Total_purchases
 from customers
group by 1,2
order by 3 desc)as d)as d
where ranks = 1
```

Data Output | Messages | Notifications

	productcategory character varying (50)	age_group text	total_purchases bigint	ranks bigint
1	Clothing	18-25	21317	1
2	Clothing	25-35	26166	1
3	Clothing	35-45	26030	1
4	Clothing	45-55	25626	1
5	Books	55-65	25841	1
6	Books	65+	13127	1

16. Find the top 3 product categories with the highest total returns and the amount returned for each.

Query | Query History

```sql
select productcategory, sum(returns) as Total_Returns,
sum(amount) as TotalReturn_amount
from customers
where returns = 1
group by 1
order by 3 desc
```

Data Output | Messages | Notifications

	productcategory character varying (50)	total_returns bigint	totalreturn_amount numeric
1	Books	55635	152112847.00
2	Clothing	55424	150965493.00
3	Electronics	45719	124473049.00
4	Home	45467	123868763.00

17. Determine the customer with the longest gap (in days) between two consecutive purchases.

Query | Query History

```sql
with gaps as (
    select customername,
    date,
    lag(date)over(partition by customername order by date) as previous_Purchase,
    date_part('day', date::timestamp - lag(date)over(partition by customername order by date)) as gaps_in_days
    from customers
)

select customername,
max(gaps_in_days) as Longest_gap
from gaps
where gaps_in_days is not null
group by 1
order by 2 desc
```

Data Output | Messages | Notifications

	customername character varying (100)	longest_gap double precision
1	Colleen Hanson	1326
2	Cynthia Marsh	1324
3	Julie Bennett	1319
4	Gina Rowland	1316
5	Richard Bishop Jr.	1305
6	Jack Medina	1294
7	Gary Boyd	1293
8	Jeffrey Hansen	1292
9	Catherine Russell	1289
10	Ashley Hodge	1283
11	Karen Duke	1282

18. Determine the customer with the longest gap (in Months) between two consecutive purchases.

Query | Query History

```sql
with gaps as (
    select customername,
    date,
    lag(date)over(partition by customername order by date) as previous_date,
    extract(month from AGE(date, lag(date)over(partition by customername order by date))) as gaps_in_months
    from customers
)

select customername,
max(gaps_in_months) as Longest_gap
from gaps
where gaps_in_months is not null
group by 1
order by 2 desc
```

Data Output | Messages | Notifications

	customername character varying (100)	longest_gap numeric
1	George Parker	11
2	Dakota Rodgers	11
3	George Mclaughlin	11
4	Dale Ball	11
5	Adam Carney	11
6	Amanda Buchanan	11
7	George Jenkins	11
8	George Howell	11
9	Dale Hernandez	11
10	Amanda Burns	11
11	George Gordon	11
12	Raymond Johnson	11

THANK YOU