SQL QUERIES- group A

Main Business Problems

1. **To discover if the annual salaries of customers lead to a better credit ID.**

Query:

```
SELECT creditid, round(avg(annualincome))
FROM customer
GROUP BY creditid
order by creditid
```

Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

creditid	round(avg(annualincome))
1	29500
2	48684
3	58607
4	59438
5	63494
6	59458
7	NULL

2. **To discover if there is any correlation between credit rating and number of encounters**

Query:

```
SELECT CUSTOMER.creditid, count(purchase) as Total_Number_of_Encounters, COUNT(CASE WHEN PURCHASE ='TRUE' THEN 1 END) AS Number_of_TRUE_PURCHASE, COUNT(CASE WHEN purchase='FALSE' THEN 1 END) AS Number_of_FALSE_PURCHASE
FROM customer
JOIN encounter
ON CUSTOMER.customerid = encounter.customerid
GROUP BY creditid
order by creditid
```

Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

creditid	Total_Number_of_Encounters	Number_of_TRUE_PURCHASE	Number_of_FALSE_PURCHASE
1	8	3	5
2	23	6	17
3	33	8	25
4	42	8	34
5	48	14	34
6	54	18	36
7	8	2	6

3. **To discover any connection between experienced salesperson and their number of successful encounters sales.**

Query:

select sfirstname as salesperson , salesperson.salesid, shiredate, count(encounter.purchase) AS Number_of_Encounters
FROM salesperson
JOIN encounter
On salesperson.salesid = encounter.salesid
group by sfirstname
order by shiredate

Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

salesperson	salesid	shiredate	Number_of_Encounters
Ron	3	1989-05-02	39
Hermione	2	1989-05-02	45
Dudley	4	1996-04-27	38
Katie	5	1996-05-15	22
Ginny	6	2001-06-01	35
Neville	7	2004-11-09	36
Colin	8	2006-04-26	1

4. **Weekly report with respect to encounters.**

Query:

Select count(*) AS positiveSaleinWeek1 from encounter WHERE encounter.purchase = 'TRUE' AND encounter.encdate BETWEEN '2015-07-01' AND '2015-07-07'; Select count(*) AS positiveSaleinWeek2 from encounter WHERE encounter.purchase = 'TRUE' AND encounter.encdate

BETWEEN '2015-07-08' AND '2015-07-14'; Select count(*) AS positiveSaleinWeek3 from encounter WHERE encounter.purchase = 'TRUE' AND encounter.encdate BETWEEN '2015-07-15' AND '2015-07-22'; Select count(*) AS positiveSaleinWeek4 from encounter WHERE encounter.purchase = 'TRUE' AND encounter.encdate BETWEEN '2015-07-23' AND '2015-07-31';

Result:

Result Grid | Filter Ro

	positiveSaleinWeek1
▶	10

Result Grid | Filter Rows:

	positiveSaleinWeek2
▶	14

Result Grid | Filter Rows:

	positiveSaleinWeek3
▶	16

Result Grid | Filter Rows:

	positiveSaleinWeek4
▶	19

5. To identify the loyal customers for Gaskins

Query:

```
SELECT cfirstname, count(purchase = 'TRUE')
FROM customer
JOIN encounter
ON customer.CustomerID = encounter.CustomerID
where purchase = 'True'
```

GROUP BY CFirstName
order by count(purchase = 'TRUE') DESC
LIMIT 10

Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

cfirstname	count(purchase = 'TRUE')
Katrina	2
Owen	1
Danny	1
George	1
Stephen	1
Mitchell	1
Dan	1
Bob	1
Vicki	1
Paul	1

Potential Business Problems

1. How many customers paid cash without looking into financing options?

Query:

```
SELECT count(customer.customerid) AS Customers_paid_cash_without_looking_into_financing_options
FROM customer
INNER JOIN creditrating
ON customer.creditid = creditrating.creditid
WHERE  customer.creditid = '7'
```

Result:

```
SELECT count(customer.customerid) AS Customers_paid_cash_without_looking_into_financing_options
FROM customer
INNER JOIN creditrating
ON customer.creditid = creditrating.creditid
WHERE  customer.creditid = '7'
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Customers_paid_cash_without_looking_into_finan
6

2. Get the total number of customers on each date that purchased.

Query:

```
SELECT encdate AS Date,  count(customerid) AS Total_number_of_Customers
FROM encounter
WHERE purchase = 'TRUE'
GROUP BY (encdate)
```

Result:

```
SELECT encdate AS Date,  count(customerid) AS Total_number_of_Customers FROM encounter
WHERE purchase = 'TRUE'  GROUP BY (encdate)
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Date	Total_number_of_Customers
2015-07-02	2
2015-07-03	1
2015-07-04	2
2015-07-05	1
2015-07-06	1
2015-07-07	1
2015-07-08	1
2015-07-09	1
2015-07-10	1
2015-07-11	4
2015-07-12	2
2015-07-13	2
2015-07-14	3
2015-07-15	1
2015-07-17	4
2015-07-18	2
2015-07-19	2
2015-07-20	2
2015-07-21	4
2015-07-22	1
2015-07-23	3
2015-07-24	2
2015-07-26	1
2015-07-27	3
2015-07-28	2
2015-07-29	3

3. Which Salesperson have encountered most?

Query:

```
SELECT sfirstname AS Salesperson_encountered_most
FROM salesperson
INNER JOIN encounter
ON salesperson.salesid = encounter.salesid
GROUP BY sfirstname
order by count(purchase) DESC
limit 1
```

Result:

```
SELECT sfirstname AS Salesperson_encountered_most
FROM salesperson
INNER JOIN encounter
ON salesperson.salesid = encounter.salesid
GROUP BY sfirstname
order by count(purchase) DESC
limit 1
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

Salesperson_encountered_most
Hermione

4. Identify the salesperson who sold the most.

Query:

SELECT sfirstname AS Salesperson_sold_most
FROM salesperson
INNER JOIN encounter
ON salesperson.salesid = encounter.salesid
where purchase = 'True'
GROUP BY sfirstname
order by count(purchase = 'TRUE') DESC
LIMIT 1

Result:

```
SELECT sfirstname AS Salesperson_sold_most
FROM salesperson
INNER JOIN encounter
ON salesperson.salesid = encounter.salesid
where purchase = 'True'
GROUP BY sfirstname
order by count(purchase = 'TRUE') DESC
LIMIT 1
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

Salesperson_solda_most
Hermione

5. Get details of customers that purchased most.

Query:

SELECT *

FROM customer
INNER JOIN encounter
ON customer.CustomerID = encounter.CustomerID
where purchase = 'True'
GROUP BY CFirstName
order by count(purchase = 'TRUE') DESC
LIMIT 1

Result:

```
SELECT *
FROM customer
INNER JOIN encounter
ON customer.CustomerID = encounter.CustomerID
where purchase = 'True'
GROUP BY CFirstName
order by count(purchase = 'TRUE') DESC
LIMIT 1
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

customerid	cfirstname	clastname	cphone	annualincome	creditid	encid	salesid	customerid	encdate	purchase
65	Katrina	Newsom	(801) 763-7396	54500	5	91	6	65	2015-07-14	true

6. What is the percentage of poor and good credit ratings in customers?
Query:

SELECT COUNT(customerid) FROM project.'customer (1)';

SELECT COUNT(customerid) FROM project.'customer (1)'

WHERE credited>=4

WHERE credited<4

Result:

```
SELECT COUNT(customerid) FROM project.`customer (1)`
;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Conte

COUNT(customerid)
177

GOOD:

```
SELECT count(customerid) FROM project.`customer (1)`
WHERE creditid>=4
;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

count(customerid)
123

(123/177)*100=**69.5%**

BAD:

```
SELECT count(customerid) FROM project.`customer (1)`
WHERE creditid<4
;
```

Limit to 1000 rows

Result Grid | Filter Rows: | Export: | Wrap Cell Content

count(customerid)

(54/177)*100=**30.5%**

7. How many sales were there?

Query:
SELECT COUNT(encid) FROM project.encounter
WHERE purchase = 'true';

```
SELECT COUNT(encid) FROM project.encounter
WHERE purchase = 'true';
```

Result Grid | Filter Rows: | Export: | Wrap C

COUNT(encid)
59

8. Average Credit of Customers

Query:

SELECT avg(creditrating.creditid) FROM creditrating;

Result:

```
SELECT avg(creditrating.creditid) FROM creditrating;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

avg(creditrating.creditid)
4.0000

9. What number of customers paid with cash

Query:

Highest Annual Income of customers

Select MAx(annualincome) as HighestIncome from customer;

Result:

```
16 •  Select MAx(annualincome) as HighestIncome from customer;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

HighestIncome
124375

10. Lowest Annual Income of customers

Query:
Select Min(annualincome) as LowestIncome from customer;

Result:

```
16 •  Select Min(annualincome) as LowestIncome from customer;
```

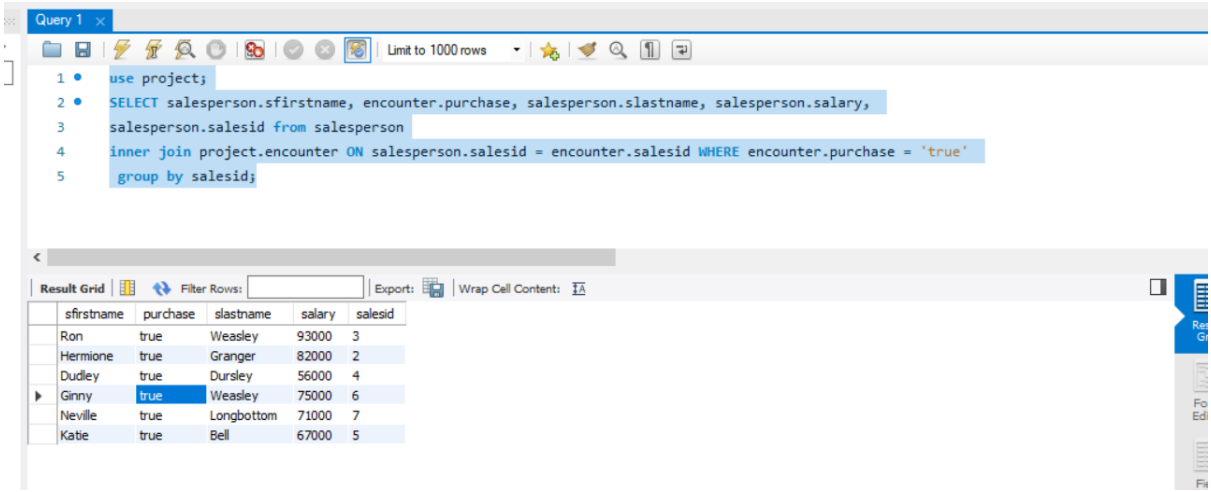
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

LowestIncome
20250

11. Employee having highest salary and their sales

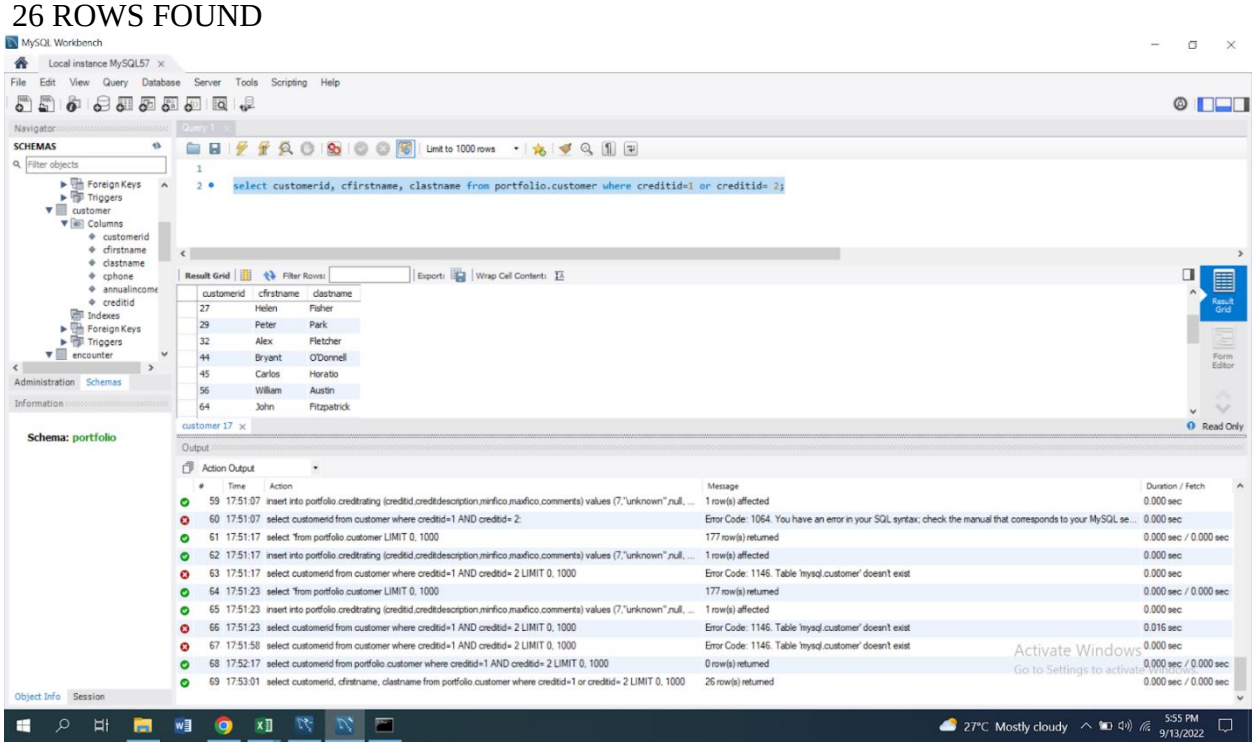
Query:
SELECT salesperson.sfirstname, encounter.purchase, salesperson.slastname, salesperson.salary, salesperson.salesid from salesperson
inner join project.encounter ON salesperson.salesid = encounter.salesid WHERE encounter.purchase = 'true'
group by salesid;

Result:



12. IDENTIFY THE RECORDS WHERETHE CUSTOMERS HAVE A BAD CREDIT (id 1,2)
Query:
select customerid, cfirstname, clastname from portfolio.customer where creditid=1 or creditid= 2;

Result:
26 ROWS FOUND



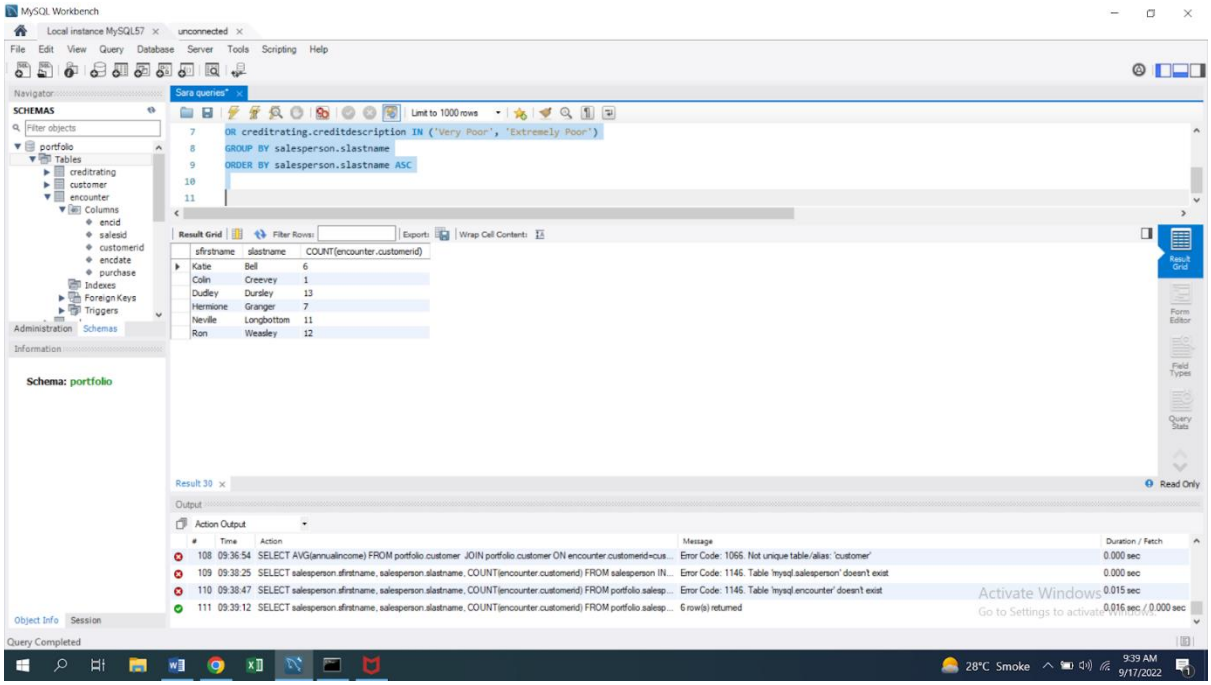
13. Create a list that shows each salesperson (first and last name) and the number of encounters each has had with customers that have an annual income of $25,000 or less or have a credit

description that is “Very Poor” or “Extremely Poor.” Order the results alphabetically by the salesperson’s last name

Query:

```
SELECT salesperson.sfirstname, salesperson.slastname, COUNT(encounter.customerid)
FROM portfolio.salesperson
INNER JOIN portfolio.encounter ON salesperson.salesid=encounter.salesid
INNER JOIN portfolio.customer ON encounter.customerid=customer.customerid
INNER JOIN portfolio.creditrating ON customer.creditid=creditrating.creditid
WHERE customer.annualincome <= 25000
OR creditrating.creditdescription IN ('Very Poor', 'Extremely Poor')
GROUP BY salesperson.slastname
ORDER BY salesperson.slastname ASC
```

Result:

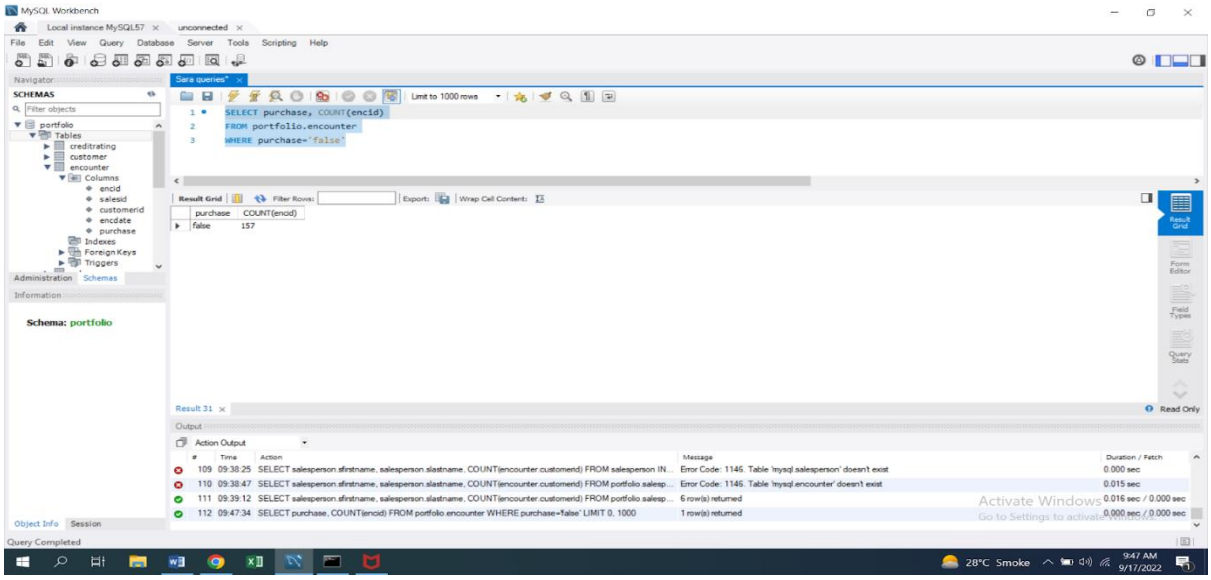


14. Count no. of false purchase

Query:

```
SELECT purchase, COUNT(encid)
FROM portfolio.encounter
WHERE purchase='false'
```

Result:



- **In which day do they get the maximum order quantity?**

Query:

SELECT encdate AS Date, count(customerid) AS Total_number_of_Customers FROM encounter WHERE purchase = 'TRUE' GROUP BY (encdate) order by count(purchase = 'TRUE') DESC

Result:

```
SELECT encdate AS Date,   count(customerid) AS Total_number_of_Customers FROM encounter WHERE purchase = 'TRUE'
GROUP BY (encdate) order by count(purchase = 'TRUE') DESC
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Date	Total_number_of_Customers
2015-07-11	4
2015-07-17	4
2015-07-21	4
2015-07-14	3
2015-07-23	3
2015-07-27	3
2015-07-29	3
2015-07-30	3
2015-07-01	2
2015-07-02	2
2015-07-04	2
2015-07-12	2
2015-07-13	2
2015-07-18	2
2015-07-19	2
2015-07-20	2
2015-07-24	2
2015-07-28	2
2015-07-31	2
2015-07-03	1
2015-07-05	1
2015-07-06	1
2015-07-07	1
2015-07-08	1
2015-07-09	1

15. Monthly/yearly salary of an employ

Query:

SELECT sfirstname, salary FROM salesperson

monthly/yearly salary of an employ SELECT sfirstname, salary FROM salesperson

Result:

```
SELECT sfirstname, salary
FROM salesperson
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

sfirstname	salary
Harry	140000
Hermione	82000
Ron	93000
Dudley	56000
Katie	67000
Ginny	75000
Neville	71000
Colin	62000

16. Which customer have more income?

Query:
SELECT annualincome, cfirstname FROM customer GROUP BY cfirstname order by max(annualincome) DESC LIMIT 1

Result:

```
SELECT annualincome, cfirstname
FROM customer
GROUP BY cfirstname
order by max(annualincome) DESC
LIMIT 1
```

Result Grid | Filter Rows: | Export: | W

annualincome	cfirstname
124375	Sue