

Question 1 :

```
SELECT id_membre, sum(prix) from ventes v  
inner join carte c  
on c.id_menu = v.id_menu  
group by id_membre
```

The screenshot shows a PostgreSQL query editor interface. At the top, there is a toolbar with various icons for file operations, search, and execution. Below the toolbar, the connection name "new/postgres@PostgreSQL 14" is displayed. The "Query Editor" tab is active, showing the following SQL query:

```
1 SELECT id_membre, sum(prix) from ventes v  
2 inner join carte c  
3 on c.id_menu = v.id_menu  
4 group by id_membre
```

Below the query editor, the "Data Output" tab is active, displaying the results of the query in a table format. The table has two columns: "id_membre" (character varying (200)) and "sum" (double precision). The results are as follows:

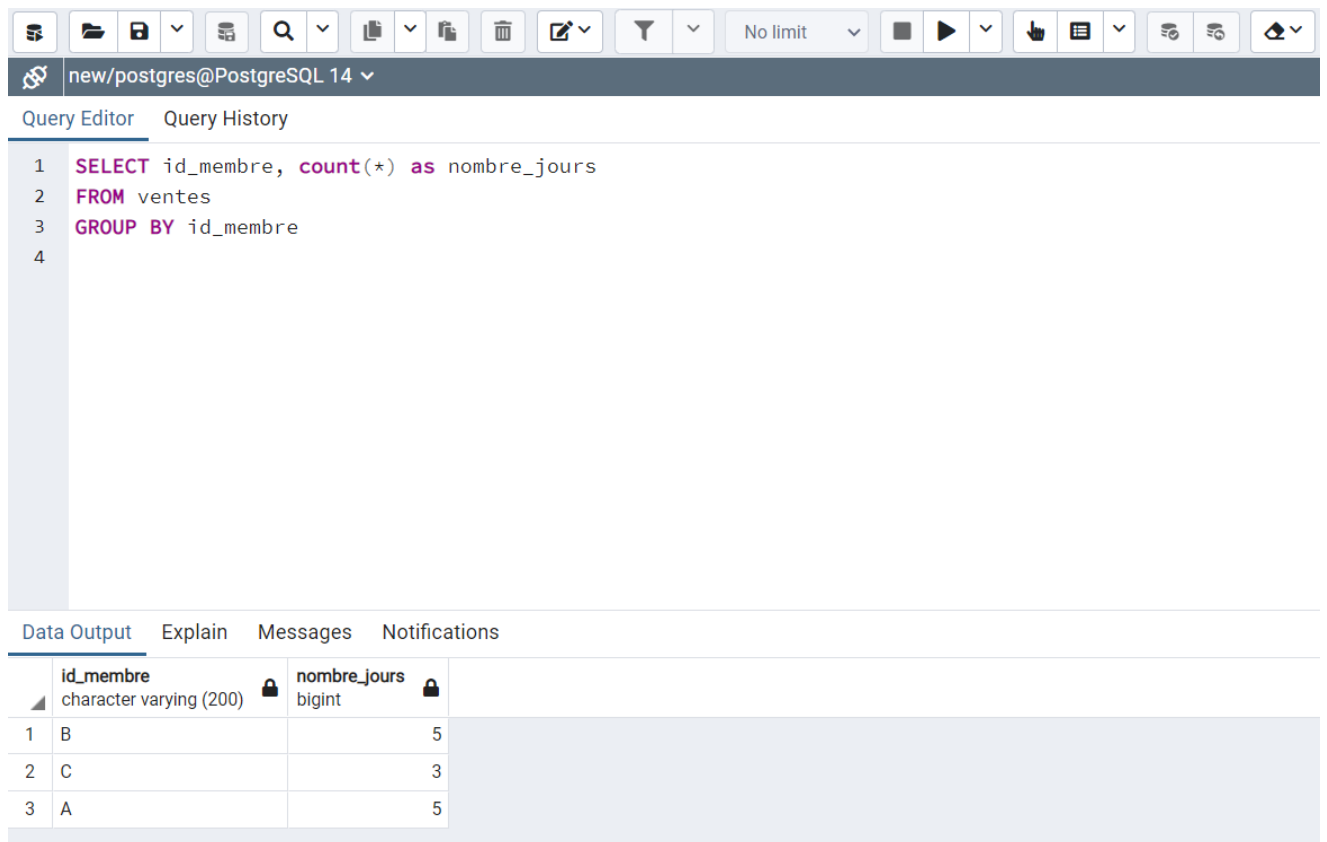
	id_membre character varying (200)	sum double precision
1	B	475
2	C	315
3	A	550

Question 2 :

SELECT id_membre, count(*) as nombre_jours

FROM ventes

GROUP BY id_membre



The screenshot shows a PostgreSQL query editor interface. At the top is a toolbar with various icons for file operations, search, and execution. Below the toolbar, the connection name 'new/postgres@PostgreSQL 14' is displayed. The 'Query Editor' tab is active, showing a SQL query with four lines. Below the editor, the 'Data Output' tab is active, displaying a table with three columns: 'id_membre', 'nombre_jours', and a third column that is not explicitly named but contains values 5, 3, and 5. The table has three rows of data.

```
1 SELECT id_membre, count(*) as nombre_jours
2 FROM ventes
3 GROUP BY id_membre
4
```

	id_membre character varying (200)	nombre_jours bigint
1	B	5
2	C	3
3	A	5

Question 3 :

```
SELECT v.id_membre, c.nom_menu  
FROM ventes v  
INNER JOIN (SELECT id_membre, MIN(date) as date1  
FROM ventes  
GROUP BY id_membre) v2  
ON v.id_membre = v2.id_membre AND v.date = v2.date1  
INNER JOIN carte c  
ON v.id_menu = c.id_menu
```

The screenshot shows a PostgreSQL query editor interface. At the top is a toolbar with various icons for file operations, search, and execution. Below the toolbar is a header bar indicating the connection to 'new/postgres@PostgreSQL 14'. The main area is divided into two tabs: 'Query Editor' and 'Query History'. The 'Query Editor' tab is active, displaying the following SQL query:

```
1 SELECT v.id_membre, c.nom_menu  
2 FROM ventes v  
3 INNER JOIN (SELECT id_membre, MIN(date) as date1  
4 FROM ventes  
5 GROUP BY id_membre) v2  
6 ON v.id_membre = v2.id_membre AND v.date = v2.date1  
7 INNER JOIN carte c  
8 ON v.id_menu = c.id_menu
```

Below the query editor are four tabs: 'Data Output', 'Explain', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing the results of the query in a table format. The table has two columns: 'id_membre' and 'nom_menu'. The results are as follows:

	id_membre character varying (200)	nom_menu character varying (200)
1	A	menu déjeuner
2	B	Menu Dégustation
3	C	Menu Dégustation

Question 4 :

```
WITH cte AS (SELECT
                id_menu,
                count(*) as amount
            from
                ventes
            GROUP BY
                id_menu)

select
    id_menu,amount
from
    cte
where amount in
    (select max(amount) from cte)
```

The screenshot shows a PostgreSQL query editor interface. At the top, there is a toolbar with various icons for file operations, search, and execution. Below the toolbar, the connection name 'new/postgres@PostgreSQL 14' is displayed. The 'Query Editor' tab is active, showing the following SQL query:

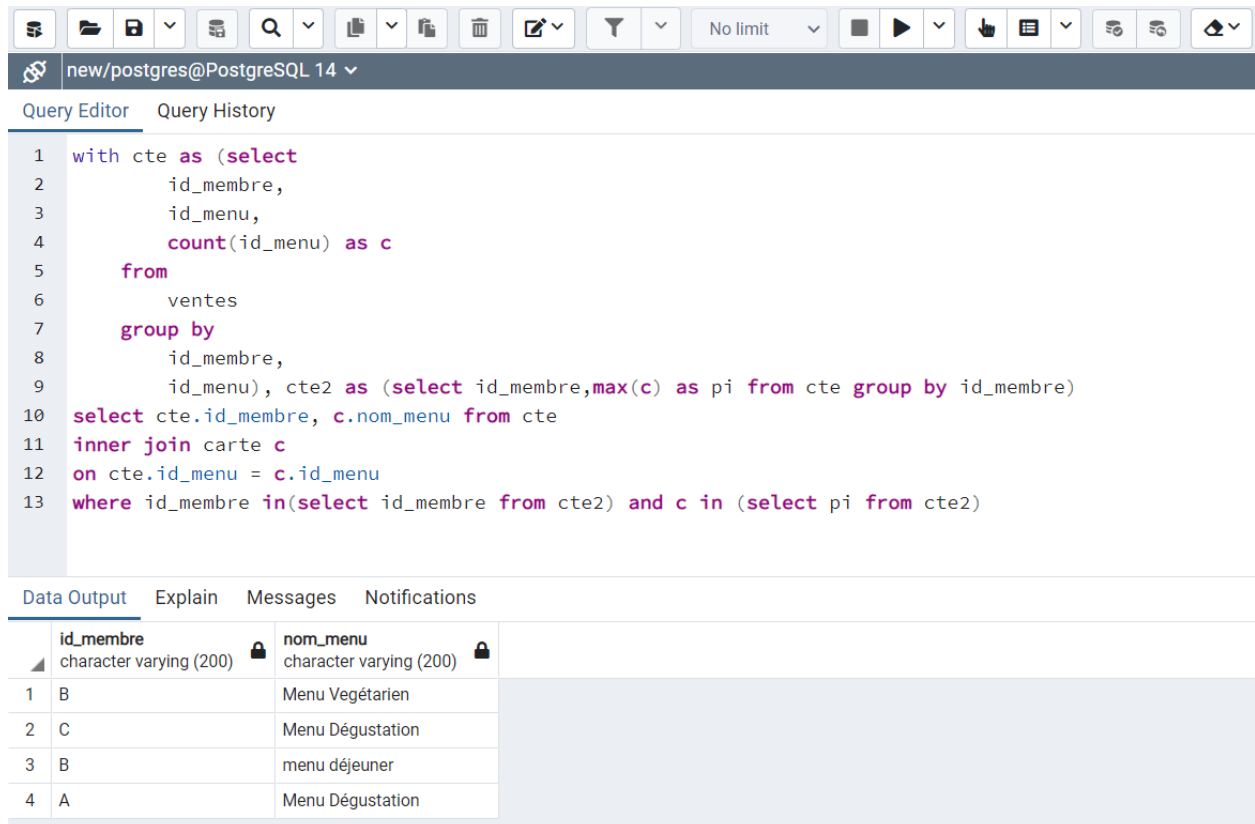
```
1 WITH cte AS (SELECT
2     id_menu,
3     count(*) as amount
4     from
5     ventes
6     GROUP BY
7     id_menu)
8 select
9     id_menu,amount
10 from
11     cte
12 where amount in
13     (select max(amount) from cte)
14
```

Below the query editor, the 'Data Output' tab is active, displaying the results of the query in a table format. The table has two columns: 'id_menu' (integer) and 'amount' (bigint). The results are as follows:

	id_menu integer	amount bigint
1	2	6

Question 5:

```
with cte as (select
    id_membre,
    id_menu,
    count(id_menu) as c
from
    ventes
group by
    id_membre,
    id_menu), cte2 as (select id_membre,max(c) as pi from cte group by id_membre)
select cte.id_membre, c.nom_menu from cte
inner join carte c
on cte.id_menu = c.id_menu
where id_membre in(select id_membre from cte2) and c in (select pi from cte2)
```



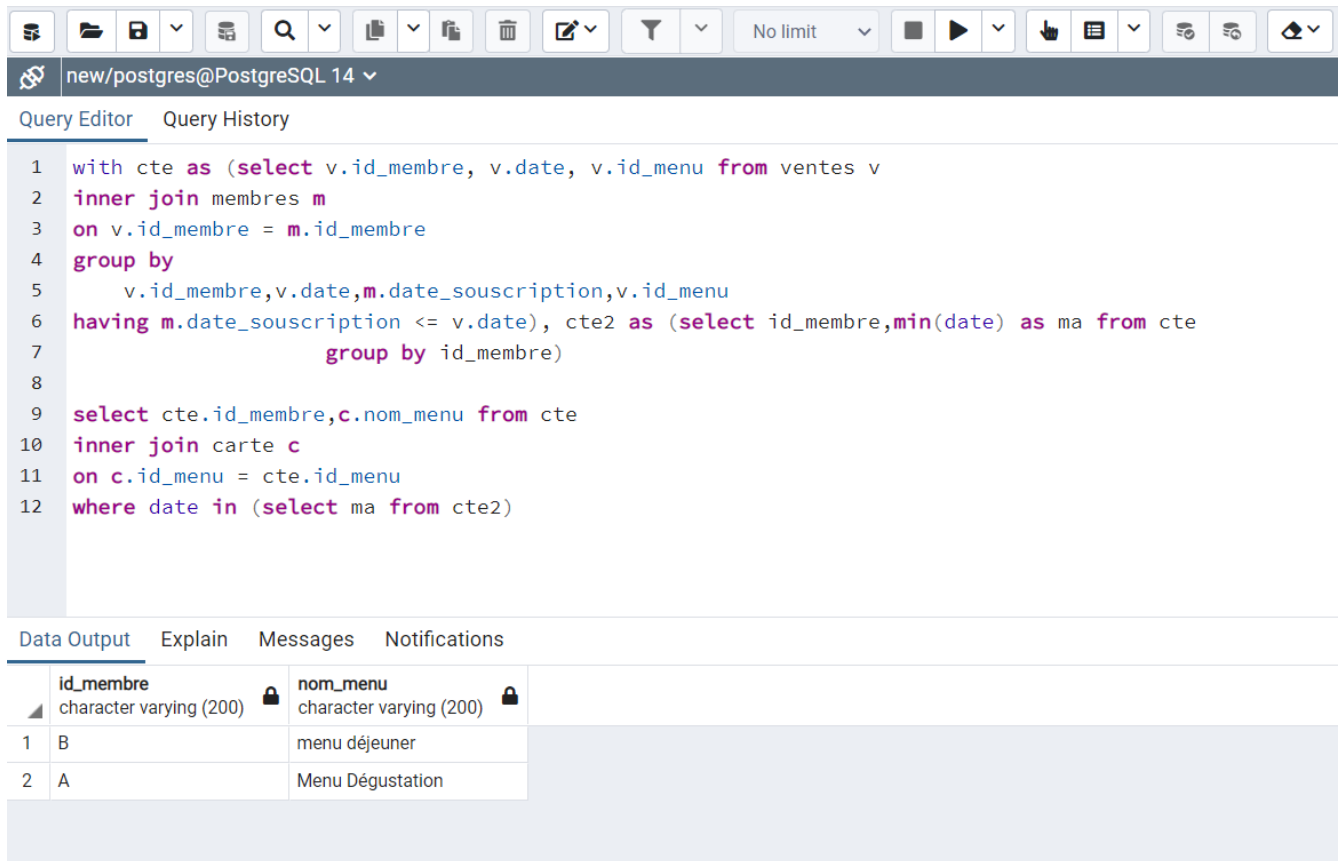
The screenshot shows a PostgreSQL query editor interface. The top toolbar includes icons for file operations, search, and execution. The main area displays the SQL query from the previous block. Below the query editor, the 'Data Output' tab is active, showing a table with two columns: 'id_membre' and 'nom_menu'. The table contains four rows of data.

	id_membre character varying (200)	nom_menu character varying (200)
1	B	Menu Végétarien
2	C	Menu Dégustation
3	B	menu déjeuner
4	A	Menu Dégustation

Question 6:

```
with cte as (select v.id_membre, v.date, v.id_menu from ventes v
inner join membres m
on v.id_membre = m.id_membre
group by
    v.id_membre,v.date,m.date_souscription,v.id_menu
having m.date_souscription <= v.date), cte2 as (select id_membre,min(date) as ma from cte
group by id_membre)

select cte.id_membre,c.nom_menu from cte
inner join carte c
on c.id_menu = cte.id_menu
where date in (select ma from cte2)
```



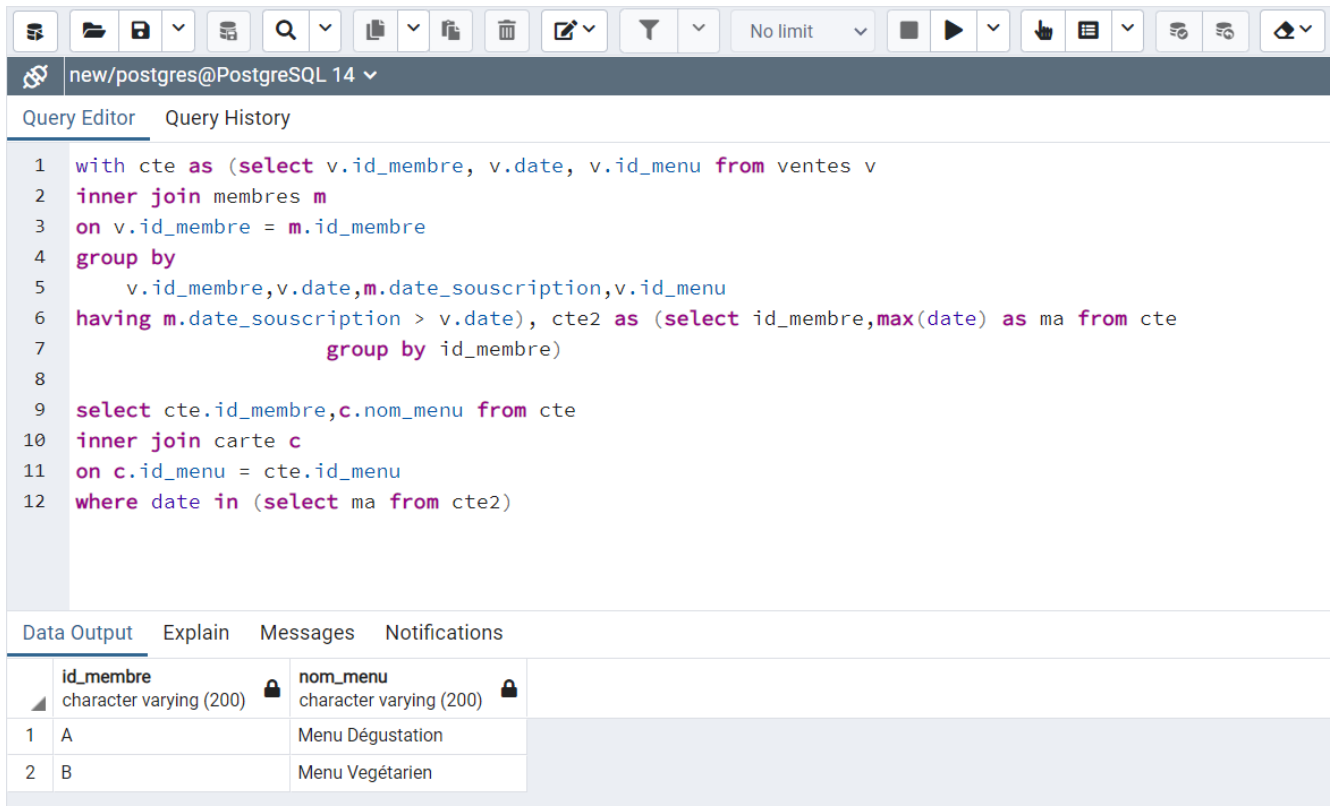
The screenshot shows a PostgreSQL query editor interface. The top toolbar contains various icons for file operations, search, and execution. The main area displays the SQL query from the previous block. Below the query editor, the 'Data Output' tab is active, showing the results of the query in a table format.

	id_membre character varying (200)	nom_menu character varying (200)
1	B	menu déjeuner
2	A	Menu Dégustation

Question 7:

```
with cte as (select v.id_membre, v.date, v.id_menu from ventes v
inner join membres m
on v.id_membre = m.id_membre
group by
    v.id_membre,v.date,m.date_souscription,v.id_menu
having m.date_souscription > v.date), cte2 as (select id_membre,max(date) as ma from cte
group by id_membre)

select cte.id_membre,c.nom_menu from cte
inner join carte c
on c.id_menu = cte.id_menu
where date in (select ma from cte2)
```



The screenshot shows a PostgreSQL query editor interface. The top toolbar contains various icons for file operations, search, and execution. The main area displays the SQL query from the previous block. Below the query editor, the 'Data Output' tab is active, showing the results of the query in a table format.

	id_membre character varying (200)	nom_menu character varying (200)
1	A	Menu Dégustation
2	B	Menu Végétarien

Question 8:

with cte as (select v.id_membre, v.date, v.id_menu from ventes v

inner join membres m

on v.id_membre = m.id_membre

group by

v.id_membre,v.date,m.date_souscription,v.id_menu

having m.date_souscription > v.date), cte2 as (select ct.id_membre, ct.id_menu,
(c.prix*count(ct.id_menu)) as total from cte ct

inner join

carte c

on c.id_menu

= ct.id_menu

group by ct.id_membre, ct.id_menu, c.prix)

select id_membre, count(id_menu), sum(total) as total from cte2

group by id_membre

The screenshot shows a PostgreSQL query editor interface. The query is as follows:

```
1 with cte as (select v.id_membre, v.date, v.id_menu from ventes v
2 inner join membres m
3 on v.id_membre = m.id_membre
4 group by
5     v.id_membre,v.date,m.date_souscription,v.id_menu
6 having m.date_souscription > v.date), cte2 as (select ct.id_membre, ct.id_menu,
7     (c.prix*count(ct.id_menu)) as total from cte ct
8     inner join carte c
9     on c.id_menu = ct.id_menu
10    group by ct.id_membre, ct.id_menu, c.prix)
11
12 select id_membre, count(id_menu), sum(total) as total from cte2
13 group by id_membre
```

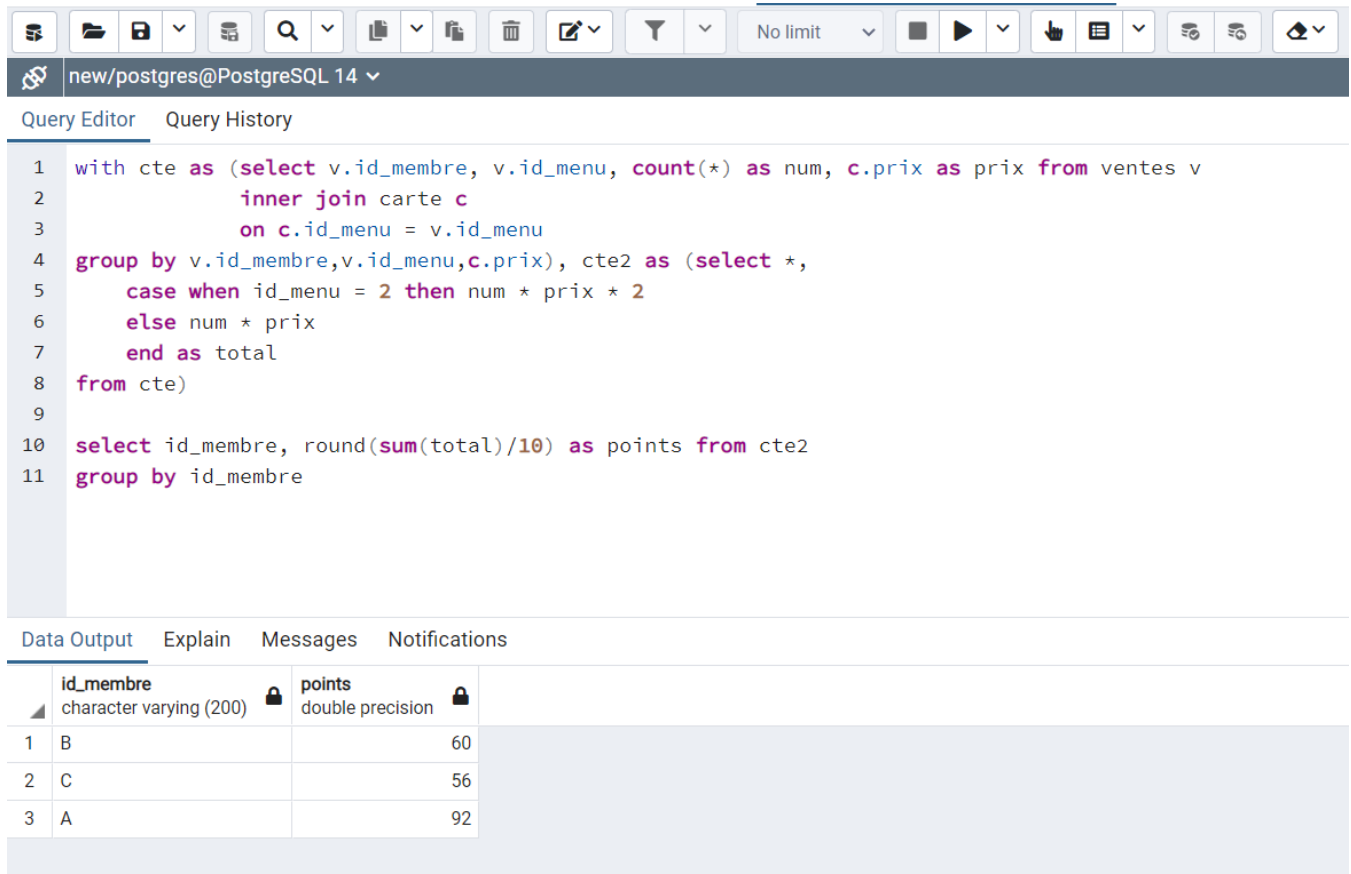
The results are displayed in a table with the following columns: id_membre (character varying (200)), count (bigint), and total (double precision).

	id_membre	count	total
1	B	3	410
2	A	2	190

Question 9:

```
with cte as (select v.id_membre, v.id_menu, count(*) as num, c.prix as prix from ventes v
              inner join carte c
              on c.id_menu = v.id_menu
group by v.id_membre,v.id_menu,c.prix), cte2 as (select *,
          case when id_menu = 2 then num * prix * 2
          else num * prix
          end as total
from cte)

select id_membre, round(sum(total)/10) as points from cte2
group by id_membre
```



The screenshot shows a PostgreSQL Query Editor interface. The top toolbar contains various icons for file operations, search, and execution. Below the toolbar, the connection name is 'new/postgres@PostgreSQL 14'. The 'Query Editor' tab is active, displaying a SQL query. The query defines two CTEs: 'cte' which joins 'ventes' and 'carte' tables and calculates a total value based on the menu ID, and 'cte2' which calculates the average total points for each member. The 'Data Output' tab is selected, showing a table with two columns: 'id_membre' (character varying (200)) and 'points' (double precision). The results show three rows: member B with 60 points, member C with 56 points, and member A with 92 points.

```
1 with cte as (select v.id_membre, v.id_menu, count(*) as num, c.prix as prix from ventes v
2               inner join carte c
3               on c.id_menu = v.id_menu
4 group by v.id_membre,v.id_menu,c.prix), cte2 as (select *,
5           case when id_menu = 2 then num * prix * 2
6           else num * prix
7           end as total
8 from cte)
9
10 select id_membre, round(sum(total)/10) as points from cte2
11 group by id_membre
```

	id_membre character varying (200)	points double precision
1	B	60
2	C	56
3	A	92