

capstone project/postgres@PostgreSQL 15

No limit

Query Query History

```
1 select * from donor_data;
2 select * from donation_data;
3 |
4 -- (1) How much is the total donation?
5 select sum(donation)Total_Donation
6 from donation_data;
```

Data Output Messages Notifications

| | total_donation bigint |
|---|--------------------------|
| 1 | 249085 |

Q2:

capstone project/postgres@PostgreSQL 15

Query Query History

```

12
13 -- (3) Show the total donation and number of donations by gender.
14 select gender, sum(donation) total_donation, count(donation) no_of_donations
15 from donation_data
16 group by gender;
17
18
19
20
21
22
23
24
25
26

```

Data Output Messages Notifications

| | gender character varying (50) 🔒 | total_donation bigint 🔒 | no_of_donations bigint 🔒 |
|---|------------------------------------|----------------------------|-----------------------------|
| 1 | Female | 121457 | 508 |
| 2 | Male | 127628 | 492 |

Q4.

capstone project/postgres@PostgreSQL 15

Query Query History

```

22 -- (5) Total donation and number of donation by job field.
23 select job_field, sum(donation)total_donation, count(donation)number_of_donations
24 from donation_data
25 group by job_field;
26

```

Data Output Messages Notifications

| | job_field character varying (50) | total_donation bigint | number_of_donations bigint |
|----|-------------------------------------|--------------------------|-------------------------------|
| 1 | Marketing | 18255 | 74 |
| 2 | Training | 21721 | 84 |
| 3 | Product Management | 22798 | 90 |
| 4 | Research and Development | 22862 | 84 |
| 5 | Business Development | 22266 | 94 |
| 6 | Sales | 19009 | 83 |
| 7 | Support | 19475 | 79 |
| 8 | Legal | 17309 | 66 |
| 9 | Accounting | 20504 | 80 |
| 10 | Services | 19858 | 80 |
| 11 | Human Resources | 23060 | 93 |
| 12 | Engineering | 21968 | 93 |

Total rows: 12 of 12 Query complete 00:00:00.249

Q6.

capstone project/postgres@PostgreSQL 15

No limit

Query Query History

26

27

28 -- (6) Total donation and number of donations above \$200.

29 select sum(donation)total_donation, count(donation)number_of_donations

30 from donation_data

31 where donation > 200;|

32

33

34

35

36

37

38

39

Data Output Messages Notifications

| | total_donation bigint | number_of_donations bigint |
|---|--------------------------|-------------------------------|
| 1 | 205892 | 586 |

Q7

capstone project/postgres@PostgreSQL 15

No limit

QueryQuery History

32

33

34 -- (7) Total donation and number of donations below \$200.

35 select sum(donation)total_donation, count(donation)number_of_donations

36 from donation_data

37 where donation < 200;

38

39

40

41

42

43

44

45

Data OutputMessagesNotifications

| | total_donation bigint | number_of_donations bigint |
|---|--------------------------|-------------------------------|
| 1 | 42593 | 411 |

Q8.

capstone project/postgres@PostgreSQL 15

Query Query History

```

53 --(8) Which top 10 states contributes the highest donations?
54
55 select state top_10_states, sum(donation) total_daonation
56 from donation_data
57 group by state
58 order by 2 desc
59 limit 10;
60

```

Data Output Messages Notifications

| | top_10_states character varying (50) | total_daonation bigint |
|----|---|---------------------------|
| 1 | California | 30264 |
| 2 | Texas | 24097 |
| 3 | Florida | 20562 |
| 4 | New York | 14759 |
| 5 | Virginia | 10750 |
| 6 | Illinois | 8674 |
| 7 | District of Columbia | 8376 |
| 8 | Tennessee | 8316 |
| 9 | Georgia | 8046 |
| 10 | Ohio | 6876 |

Total rows: 10 of 10 Query complete 00:00:00.254

Q9.

capstone project/postgres@PostgreSQL 15

Query Query History

```

60
61 --(9) Which top 10 states contributes the least donations?
62
63 select state top_10_states, sum(donation) total_daonation
64 from donation_data
65 group by state
66 order by 2 asc
67 limit 10;

```

Data Output Messages Notifications

| | top_10_states character varying (50) | total_daonation bigint |
|----|---|---------------------------|
| 1 | Wyoming | 232 |
| 2 | Maine | 258 |
| 3 | South Dakota | 401 |
| 4 | North Dakota | 651 |
| 5 | Alaska | 734 |
| 6 | West Virginia | 793 |
| 7 | South Carolina | 819 |
| 8 | New Hampshire | 841 |
| 9 | Hawaii | 875 |
| 10 | Montana | 1009 |

Q10.

capstone project/postgres@PostgreSQL 15

Query
Query History

```

69 --(10) What are the top 10 cars driven by the highest donors?
70
71 select concat (first_name, ' ', last_name)fullname, sum(donation)amount_donated,
72 dn.car
73 from donation_data dt
74 join donor_data dn on dt.id = dn.id
75 group by concat (first_name, ' ', last_name), dn.car
76 order by 2 desc
77 limit 10;
78

```

Data Output
Messages
Notifications

| | fullname text | amount_donated bigint | car character varying (100) |
|---|--------------------|--------------------------|--------------------------------|
| 1 | Wallie Leather | 500 | Lexus |
| 2 | Beverlie Andriesse | 500 | Ford |
| 3 | Clevie Camilletti | 499 | Buick |
| 4 | Peder Rilton | 499 | Mazda |
| 5 | Worthy Le feaver | 498 | MINI |
| 6 | Amalea Knill | 497 | Hyundai |
| 7 | Nathaniel McGenn | 494 | GMC |
| 8 | Tonnie Stockney | 494 | Chevrolet |

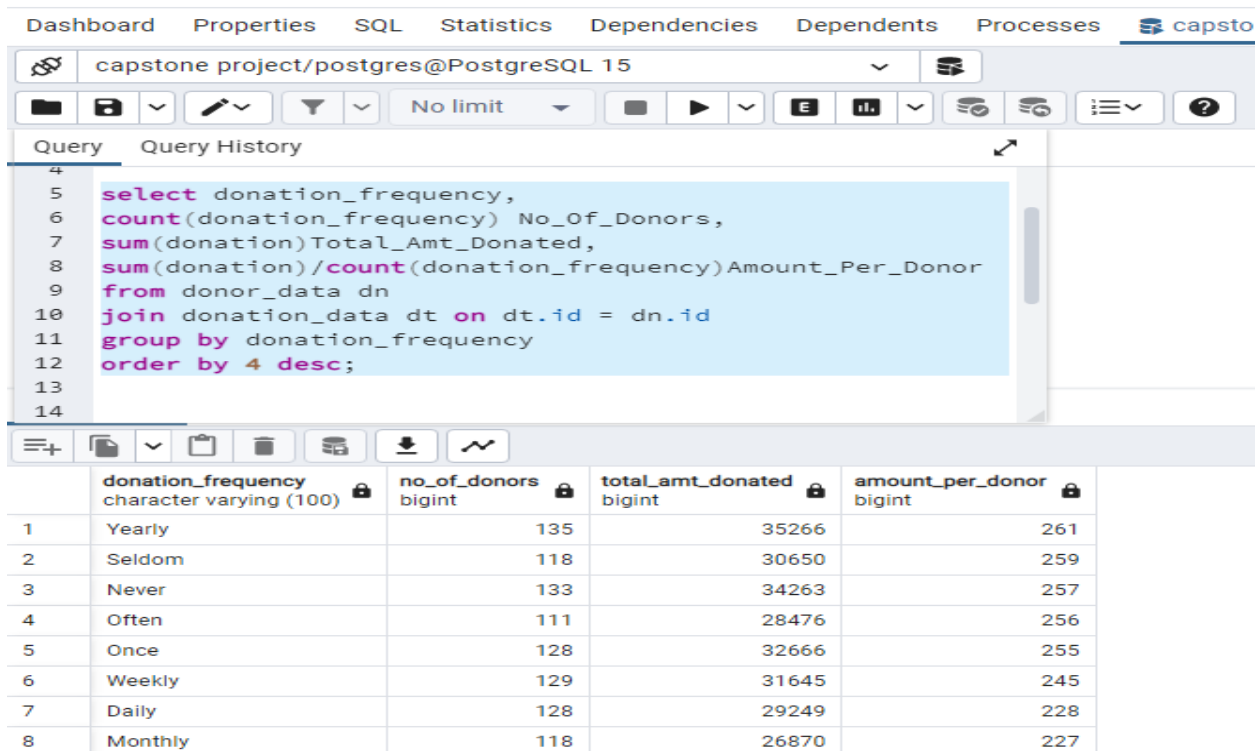
ANLAYSIS AND RECOMMENDATIONS

Some information needed to be clarified like "Never" as a frequency of donation variable. The data shows this group actually contribute more individually than other members in the same column. Therefore, the term "never" should not be taken literally.

Quick Analysis 1 (amount donated by frequency of donation)

Based on the screenshot attached, here are my insights:

- Those who donate yearly are the highest contributors to the course, with individual average donation of \$261.
- Those who donate monthly are the lowest contributors to the course, with individual average donation of \$227.



The screenshot displays a PostgreSQL query editor with the following SQL query:

```
select donation_frequency,
count(donation_frequency) No_Of_Donors,
sum(donation) Total_Amt_Donated,
sum(donation)/count(donation_frequency) Amount_Per_Donor
from donor_data dn
join donation_data dt on dt.id = dn.id
group by donation_frequency
order by 4 desc;
```

The results table below shows the data sorted by the average amount per donor in descending order:

| | donation_frequency character varying (100) | no_of_donors bigint | total_amt_donated bigint | amount_per_donor bigint |
|---|---|------------------------|-----------------------------|----------------------------|
| 1 | Yearly | 135 | 35266 | 261 |
| 2 | Seldom | 118 | 30650 | 259 |
| 3 | Never | 133 | 34263 | 257 |
| 4 | Often | 111 | 28476 | 256 |
| 5 | Once | 128 | 32666 | 255 |
| 6 | Weekly | 129 | 31645 | 245 |
| 7 | Daily | 128 | 29249 | 228 |
| 8 | Monthly | 118 | 26870 | 227 |

Quick Analysis 2 (no. of donors)

When the data is sorted by number of donors in relation to donation frequency (See screenshot), the yearly donors have 24 members more than donors categorized as "often" and 17 more members to both "seldom" and "monthly".

Dashboard Properties SQL Statistics Dependencies Dependents Processes

capstone project/postgres@PostgreSQL 15

No limit

Query Query History

```

14
15 select state, count(state)No_Of_Donors,
16 sum(donation)Total_Amt_Donated,
17 sum(donation)/count(donation_frequency)Amount_Per_Donor
18 from donor_data dn
19 join donation_data dt on dt.id = dn.id
20 group by state
21 order by 2 asc
22 limit 10;

```

| | state character varying (50) | no_of_donors bigint | total_amt_donated bigint | amount_per_donor bigint |
|----|---------------------------------|------------------------|-----------------------------|----------------------------|
| 1 | Maine | 1 | 258 | 258 |
| 2 | South Dakota | 1 | 401 | 401 |
| 3 | Wyoming | 1 | 232 | 232 |
| 4 | North Dakota | 2 | 651 | 325 |
| 5 | New Hampshire | 3 | 841 | 280 |
| 6 | Alaska | 3 | 734 | 244 |
| 7 | Montana | 4 | 1009 | 252 |
| 8 | Hawaii | 4 | 875 | 218 |
| 9 | Mississippi | 5 | 1391 | 278 |
| 10 | West Virginia | 6 | 793 | 132 |

Dashboard Properties SQL Statistics Dependencies Dependents Processes

capstone project/postgres@PostgreSQL 15

The session is idle and there is no current transaction.

Query Query History

```

14
15 select state, count(state)No_Of_Donors,
16 sum(donation)Total_Amt_Donated,
17 sum(donation)/count(donation_frequency)Amount_Per_Donor
18 from donor_data dn
19 join donation_data dt on dt.id = dn.id
20 group by state
21 order by 2 desc
22 limit 10;

```

| | state character varying (50) | no_of_donors bigint | total_amt_donated bigint | amount_per_donor bigint |
|----|---------------------------------|------------------------|-----------------------------|----------------------------|
| 1 | California | 113 | 30264 | 267 |
| 2 | Texas | 95 | 24097 | 253 |
| 3 | Florida | 90 | 20562 | 228 |
| 4 | New York | 58 | 14759 | 254 |
| 5 | Virginia | 39 | 10750 | 275 |
| 6 | Illinois | 34 | 8674 | 255 |
| 7 | Georgia | 33 | 8046 | 243 |
| 8 | North Carolina | 33 | 6328 | 191 |
| 9 | Ohio | 32 | 6876 | 214 |
| 10 | District of Columbia | 30 | 8376 | 279 |

Quick Analysis 4 (amount donated by job type)

It can be seen in the screenshot, donors from R&D, legal, training, accounting and product management are our biggest contributors per individual at \$272, \$262, \$258, \$256 and \$253 respectively. Sales is pulling last at \$229 per individual contributor, followed by donors from engineering and business development.

The screenshot shows a PostgreSQL query editor interface. The top navigation bar includes tabs for Dashboard, Properties, SQL, Statistics, Dependencies, Dependents, Processes, and a user profile for 'capstone'. The main window displays a query for the 'capstone project/postgres@PostgreSQL 15' database. The query is as follows:

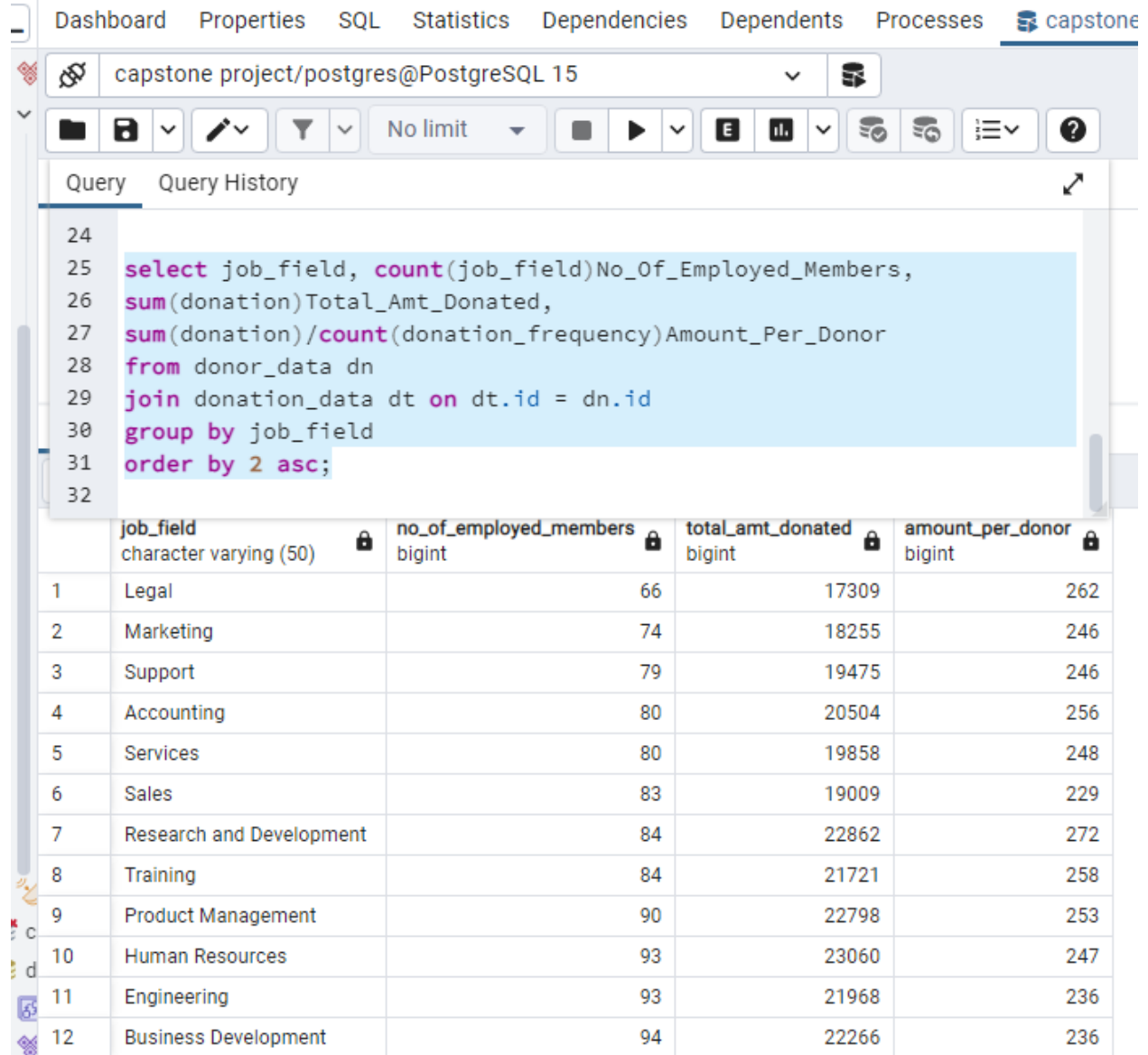
```
25 select job_field, count(job_field)No_Of_Employed_Members,  
26 sum(donation)Total_Amt_Donated,  
27 sum(donation)/count(donation_frequency)Amount_Per_Donor  
28 from donor_data dn  
29 join donation_data dt on dt.id = dn.id  
30 group by job_field  
31 order by 4 desc;  
32
```

Below the query editor, the results are displayed in a table with the following columns: job_field, no_of_employed_members, total_amt_donated, and amount_per_donor. The results are ordered by the amount per donor in descending order.

| | job_field | no_of_employed_members | total_amt_donated | amount_per_donor |
|----|--------------------------|------------------------|-------------------|------------------|
| 1 | Research and Development | 84 | 22862 | 272 |
| 2 | Legal | 66 | 17309 | 262 |
| 3 | Training | 84 | 21721 | 258 |
| 4 | Accounting | 80 | 20504 | 256 |
| 5 | Product Management | 90 | 22798 | 253 |
| 6 | Services | 80 | 19858 | 248 |
| 7 | Human Resources | 93 | 23060 | 247 |
| 8 | Marketing | 74 | 18255 | 246 |
| 9 | Support | 79 | 19475 | 246 |
| 10 | Business Development | 94 | 22266 | 236 |
| 11 | Engineering | 93 | 21968 | 236 |
| 12 | Sales | 83 | 19009 | 229 |

Quick Analysis 5 (job type donors)

According to the screenshot below, we have lower numbers of donors from legal, marketing, support, accounting and services compared to those from business development, engineering, HR and product management.



The screenshot shows a PostgreSQL query editor interface. The top navigation bar includes links for Dashboard, Properties, SQL, Statistics, Dependencies, Dependents, Processes, and a capstone logo. The main toolbar contains icons for file operations, query execution, and other database functions. The query editor displays a SQL query that selects job fields and calculates donor statistics. Below the query, the results are shown in a table format.

```
24
25 select job_field, count(job_field)No_Of_Employed_Members,
26 sum(donation)Total_Amt_Donated,
27 sum(donation)/count(donation_frequency)Amount_Per_Donor
28 from donor_data dn
29 join donation_data dt on dt.id = dn.id
30 group by job_field
31 order by 2 asc;
32
```

| | job_field character varying (50) | no_of_employed_members bigint | total_amt_donated bigint | amount_per_donor bigint |
|----|-------------------------------------|----------------------------------|-----------------------------|----------------------------|
| 1 | Legal | 66 | 17309 | 262 |
| 2 | Marketing | 74 | 18255 | 246 |
| 3 | Support | 79 | 19475 | 246 |
| 4 | Accounting | 80 | 20504 | 256 |
| 5 | Services | 80 | 19858 | 248 |
| 6 | Sales | 83 | 19009 | 229 |
| 7 | Research and Development | 84 | 22862 | 272 |
| 8 | Training | 84 | 21721 | 258 |
| 9 | Product Management | 90 | 22798 | 253 |
| 10 | Human Resources | 93 | 23060 | 247 |
| 11 | Engineering | 93 | 21968 | 236 |
| 12 | Business Development | 94 | 22266 | 236 |

Recommendations

- We have to encourage those who donate monthly to donate more. For instance, their individual amount is only \$34 short of the highest yearly donors. They only need to add \$2.83 to their monthly donations to meet match their yearly counterparts. Those in-between can also be encouraged to do this.
- We have to get more donors who will donate often, monthly or seldomly, as we are short of members in those areas.
- We need to reach out to the small cities like Maine, South Dakota, Wyoming, North Dakota etc, to encourage more people to contribute to the course. There should be an effort to have outreach programs in these towns and cities.
- We need to be more pro-active in encouraging our donors from sales, engineering and business development to add a little more to their previous contributions, so that they can match their counterparts from R&D and legal.
- Our donors under the legal job field need to be encouraged to bring awareness to their colleagues, since they are our second highest contributors in terms of average amount donated, but the least participating members. We also really need to encourage our donors from marketing and support to involve their colleagues as well.