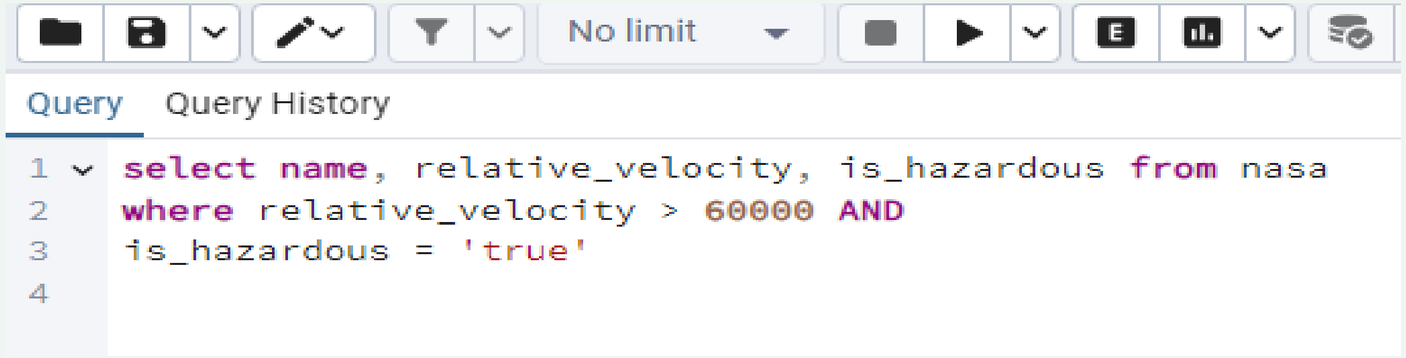
ASTEROID DATA ANALYSIS: ADVANCED SQL QUERIES AND INSIGHTS

By Chirag Sharma

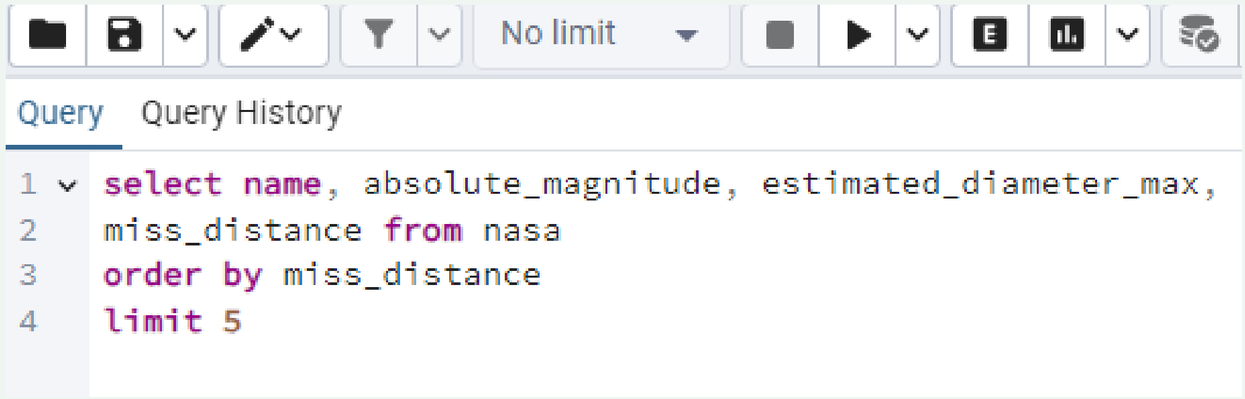
1.Write a query to find all hazardous asteroids that have a relative velocity higher than 60,000 km/h. Return their names, relative velocity, and miss distance.



Data Output | Messages | Notifications

	name character varying (255)	relative_velocity double precision	is_hazardous boolean
1	349507 (2008 QY)	109949.7571484926	true
2	494975 (2009 WO106)	68306.5927565258	true
3	140039 (2001 SO73)	106453.0869492776	true
4	(2016 CL136)	92214.2461809838	true
5	481532 (2007 LE)	74321.5865408294	true
6	(2013 WF108)	62591.0473432282	true
7	434188 (2003 AD23)	139874.0817159504	true
8	(2013 QU1)	69535.1179617595	true
9	(2022 BX1)	74338.0833320291	true
10	(2019 FJ)	98004.329790642	true
11	7822 (1991 CS)	75731.1381531594	true

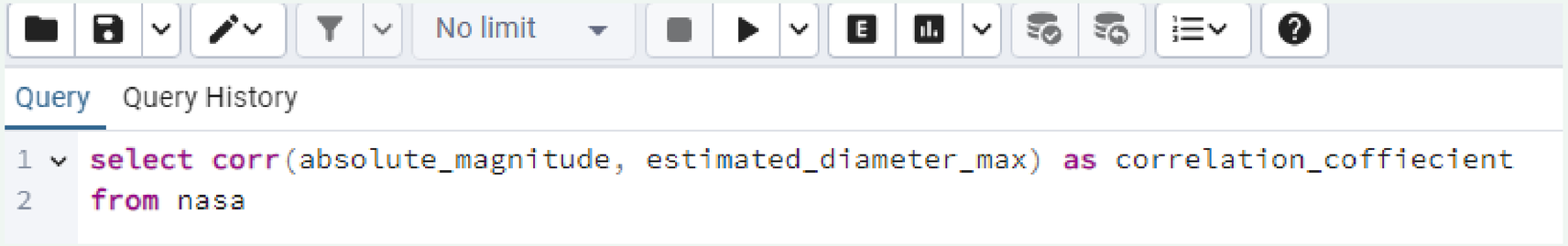
2. Write a query to identify the top 5 asteroids with the smallest miss distance. Include their names, absolute magnitude, estimated diameter max, and miss distance.

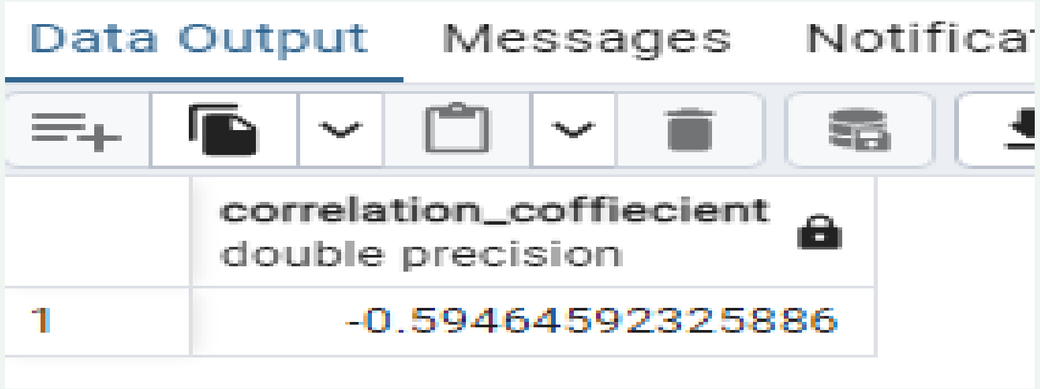


Data Output | Messages | Notifications

	name character varying (255)	absolute_magnitude double precision	estimated_diameter_max double precision	miss_distance double precision
1	(2020 VT4)	28.61	0.0112730122	6745.532515957
2	(2024 LH1)	30.811	0.0040910989	8098.256295645
3	(2020 QG)	29.9	0.0062235757	9316.925424026
4	(2021 UA1)	31.84	0.0025470647	9426.685381245
5	(2023 BU)	29.7	0.0068240151	9966.973048537

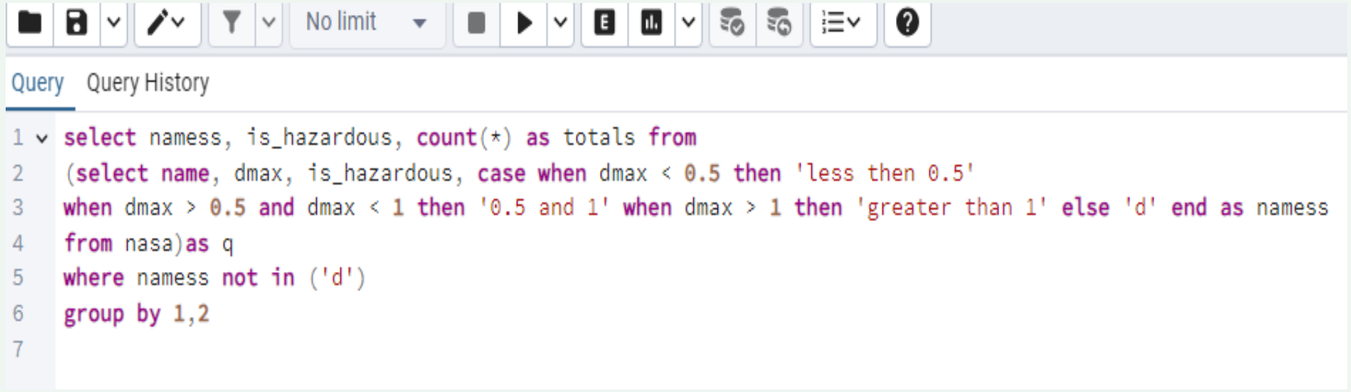
3. Write a query to find the correlation between the absolute magnitude and the estimated maximum diameter of asteroids.

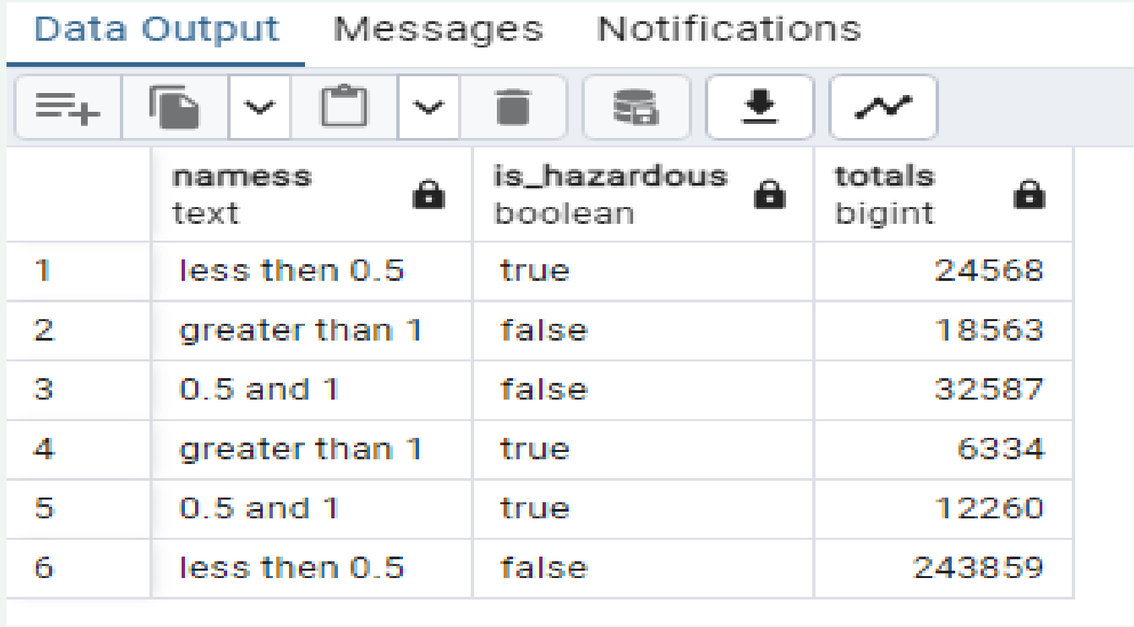




A **negative correlation** (like -0.5946) between absolute magnitude and estimated maximum diameter means that as the absolute magnitude **increases** (which actually means the asteroid is **dimmer**), the estimated maximum diameter tends to **decrease**.

4. Write a query to group asteroids by their estimated maximum diameter (e.g., less than 0.5 km, between 0.5 and 1 km, greater than 1 km) and count how many asteroids fall into each group. Include a column indicating if any of the asteroids in each group are hazardous.





	namess text	is_hazardous boolean	totals bigint
1	less then 0.5	true	24568
2	greater than 1	false	18563
3	0.5 and 1	false	32587
4	greater than 1	true	6334
5	0.5 and 1	true	12260
6	less then 0.5	false	243859

5. Write a query to find the top 3 non-hazardous asteroids with the largest estimated maximum diameter. Return their names, absolute magnitude, and estimated diameter max.

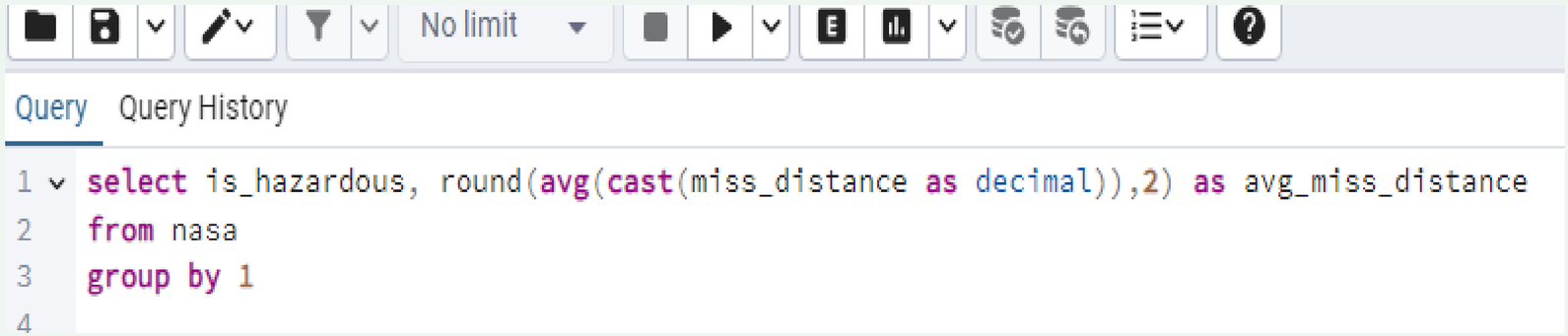
Query | Query History

```sql
select name, absolute_magnitude, dmax
from nasa
where is_hazardous = 'false'
and dmax is not null
order by dmax desc
limit 3
```

Data Output | Messages | Notifications

	name character varying (255)	absolute_magnitude double precision	dmax double precision
1	1036 Ganymed (A924 UB)	9.25	83.9537266171
2	1036 Ganymed (A924 UB)	9.25	83.9537266171
3	1036 Ganymed (A924 UB)	9.25	83.9537266171

6. Write a query to calculate the average miss distance for hazardous and non-hazardous asteroids separately. Return the average miss distance for both categories.



Data Output | Messages | Notifications

	is_hazardous boolean	avg_miss_distance numeric
1	false	41587314.11
2	true	41180152.82

7. Write a query to identify all asteroids that have an absolute magnitude less than 20 and a miss distance of more than 50 million kilometers. Return their names, estimated diameter max, and miss distance.

Query | Query History

```
1  select name, miss_distance, dmax, absolute_magnitude
2  from nasa
3  where absolute_magnitude < 20 and
4  miss_distance > 50000000
5
```

Data Output | Messages | Notifications

	name character varying (255)	miss_distance double precision	dmax double precision	absolute_magnitude double precision
1	357618 (2005 EM30)	53926958.82627829	0.912094798	19.07
2	(2021 PC7)	71134291.76365572	0.7871182442	19.39
3	396593 (2001 HC)	72474926.61426796	0.912094798	19.07
4	(2006 NL)	73285367.52491198	0.602615057	19.97
5	(2002 UZ30)	62883608.41805307	0.7482383761	19.5
6	608017 (2002 UZ30)	62883600.57912468	0.6166517648	19.92
7	(2019 GT2)	54753461.32719465	0.8995803882	19.1
8	679440 (2019 GT2)	54753462.5389374	0.8995803882	19.1
9	142464 (2002 TC9)	55701750.49665165	1.2590413069	18.37
10	54686 (2001 DU8)	66092092.50350917	2.8186423881	16.62
11	(2023 EL)	63776060.197492346	0.8670416872	19.18
12	162117 (1998 SD15)	53598472.496762986	0.8831611957	19.14

8. Write a query to find the asteroid with the highest relative velocity among those with an absolute magnitude greater than 22. Return its name, relative velocity, and miss distance.

Query | Query History

```
1  select name, relative_velocity, miss_distance, absolute_magnitude
2  from nasa
3  where absolute_magnitude > 22
4  order by relative_velocity desc
```

Data Output | Messages | Notifications

	name character varying (255)	relative_velocity double precision	miss_distance double precision	absolute_magnitude double precision
1	(2019 AM13)	184719.9375113735	70816111.14140572	22.05
2	(2020 XV6)	182428.2557519296	73068285.12824818	22.72
3	(2019 AM13)	182007.8297119332	66799767.27703499	22.05
4	(2018 YC2)	177726.9439764907	57280025.30370563	22.7
5	(2019 AM13)	176314.4795337412	62061310.96366454	22.05
6	(2019 AM13)	174651.7589022704	59186535.37040998	22.05
7	(2021 GF11)	166920.0530192586	65432378.223673664	23.38
8	(2020 XV6)	166732.7333501113	57921514.00024011	22.72
9	(2019 AM13)	166324.1731663432	50279519.83865699	22.05
10	(2020 XV6)	163882.3294451671	55084089.24021801	22.72
11	(2021 GF11)	163244.1850250724	71525149.54439595	23.38
12	(2020 CY1)	162793.6647074082	70837137.69050612	23.69

9. Write a query to identify the asteroid that has the smallest estimated diameter but the highest relative velocity. Return its name, estimated diameter min, and relative velocity.

Query | Query History

```sql
select name, estimated_diameter_min, relative_velocity
from nasa
order by estimated_diameter_min asc,
relative_velocity desc
limit 5
```

Data Output | Messages | Notifications

	name character varying (255)	estimated_diameter_min double precision	relative_velocity double precision
1	(2022 WJ1)	0.0005111578	74391.1024120236
2	(2022 WJ1)	0.0005111578	72770.4241991748
3	(2022 WJ1)	0.0005111578	71727.1103874359
4	(2022 WJ1)	0.0005111578	45204.2350372289
5	(2008 TS26)	0.0006089126	56751.0170897491

10. Write a query to find the asteroids that have orbited Earth more than once, where their minimum estimated diameter is greater than 0.5 km. Return their names, the number of times they orbited, and their average relative velocity.

Query | Query History

```
1  select * from
2  (select name, estimated_diameter_min, count(*) as total_orbition
3  from nasa
4  group by 1,2)as q
5  where estimated_diameter_min > 0.5 AND
6  total_orbition > 1
7  order by total_orbition
8
```

Data Output | Messages | Notifications

	name character varying (255)	estimated_diameter_min double precision	total_orbition bigint
1	189552 (2000 RL77)	1.0878148336	2
2	219527 (2001 QK142)	1.0581688593	2
3	(2017 VJ2)	0.7701139256	2
4	661492 (2004 TX37)	1.9794975866	2
5	31210 (1998 BX7)	1.3949382293	2
6	(2001 KY18)	0.6005580277	2
7	458745 (2011 QY37)	0.5451989071	2
8	363076 (2000 PH6)	0.6707411893	2
9	(2021 JQ24)	0.7187121332	2
10	162168 (1999 GT6)	0.9048049682	2
11	141761 (2002 MC)	0.8640820365	2

11. Write a query to identify the top 5 asteroids with the greatest difference between their minimum and maximum estimated diameters. Return their names, absolute magnitude, estimated diameter min, estimated diameter max, and the difference.

Query | Query History

```sql
select * from
(select name, absolute_magnitude, estimated_diameter_min, dmax,
round(cast(dmax - estimated_diameter_min as decimal),3) as difference
from nasa)as q
where difference is not null
order by difference desc
limit 5
```

Data Output | Messages | Notifications

	name character varying (255)	absolute_magnitude double precision	estimated_diameter_min double precision	dmax double precision	difference numeric
1	1036 Ganymed (A924 UB)	9.25	37.5452479361	83.9537266171	46.408
2	1036 Ganymed (A924 UB)	9.25	37.5452479361	83.9537266171	46.408
3	1036 Ganymed (A924 UB)	9.25	37.5452479361	83.9537266171	46.408
4	433 Eros (A898 PA)	10.41	22.0067027115	49.2084832235	27.202
5	433 Eros (A898 PA)	10.41	22.0067027115	49.2084832235	27.202

12. Write a query to rank the asteroids by their miss distance in ascending order. For each asteroid, include its name, absolute magnitude, estimated diameter max, and whether it is hazardous.

Query | Query History

```
1  select * from
2  (select name, absolute_magnitude, is_hazardous, miss_distance,
3  rank()over(partition by name order by miss_distance) as ranks
4  from nasa)as j
5
```

Data Output | Messages | Notifications

	name character varying (255)	absolute_magnitude double precision	is_hazardous boolean	miss_distance double precision	ranks bigint
1	(1979 XB)	18.6	true	5473840.412206171	1
2	(1982 YA)	18.01	false	24265966.868863046	1
3	(1982 YA)	18.01	false	27043782.62224835	2
4	(1982 YA)	18.01	false	28441767.53670059	3
5	(1982 YA)	18.01	false	30274978.46185063	4
6	(1982 YA)	18.01	false	31233783.1199063	5
7	(1982 YA)	18.01	false	32654935.717933107	6
8	(1982 YA)	18.01	false	34704870.2741351	7
9	(1982 YA)	18.01	false	35889719.36325159	8
10	(1982 YA)	18.01	false	40522852.732388094	9
11	(1982 YA)	18.01	false	43149711.47424509	10

13. Write a query to find the asteroid with the smallest absolute magnitude that is also classified as hazardous. Return its name, absolute magnitude, estimated diameter max, and miss distance

Query | Query History

```sql
select name, is_hazardous, absolute_magnitude
from nasa
where is_hazardous = 'true'
order by absolute_magnitude
limit 1
```

Data Output | Messages | Notifications

	name character varying (255)	is_hazardous boolean	absolute_magnitude double precision
1	3122 Florence (1981 ET3)	true	14.1

14. Write a query to determine the average estimated diameter (max) of all non-hazardous asteroids. Compare this value with the average estimated diameter (max) of hazardous asteroids. Return both averages.

Query | Query History

```sql
with non_haz as (
select name, is_hazardous, round(cast(avg(dmax) as decimal), 3) as avg_max_diameter,
row_number()over(order by is_hazardous) as numbers
from nasa
where is_hazardous = 'false'
group by 1,2),

haz as (
    select name, is_hazardous, round(cast(avg(dmax) as decimal),3) as avg_max_diameter,
    row_number()over(order by is_hazardous) as num
    from nasa
    where is_hazardous = 'true'
    group by 1,2
)

select a.name, a.avg_max_diameter, b.name, b.avg_max_diameter
from non_haz as a
join haz as b
ON a.numbers = b.num
```

Data Output | Messages | Notifications

	name character varying (255)	avg_max_diameter numeric	name character varying (255)	avg_max_diameter numeric
1	458452 (2011 BR15)	0.661	(2015 RH2)	0.236
2	(2024 GC2)	0.056	(1999 XK136)	0.459
3	(2021 GM26)	0.043	380636 (2004 XN14)	0.664
4	(2020 QQ)	0.022	(2007 RV9)	0.508
5	(2010 UP)	0.060	(2004 LB)	0.284
6	(2005 OW)	0.393	(2006 WJ3)	0.597
7	(2020 VT2)	0.014	(2013 LE16)	1.170
8	(2019 VW)	0.027	(2012 JS11)	0.518
9	(2017 XM2)	0.086	(2017 FQ64)	0.386
10	(2022 UG16)	0.052	(2023 FM)	0.255
11	(2023 HL12)	0.072	86819 (2000 GK137)	1.906
12	(2012 XM145)	0.494	(2017 TN6)	0.348
13	(2010 GJ65)	0.258	452707 (2010 XY72)	1.067

15. Write a query to identify the asteroid that has the closest miss distance but is not hazardous. Return its name, miss distance, absolute magnitude, and estimated diameter max.

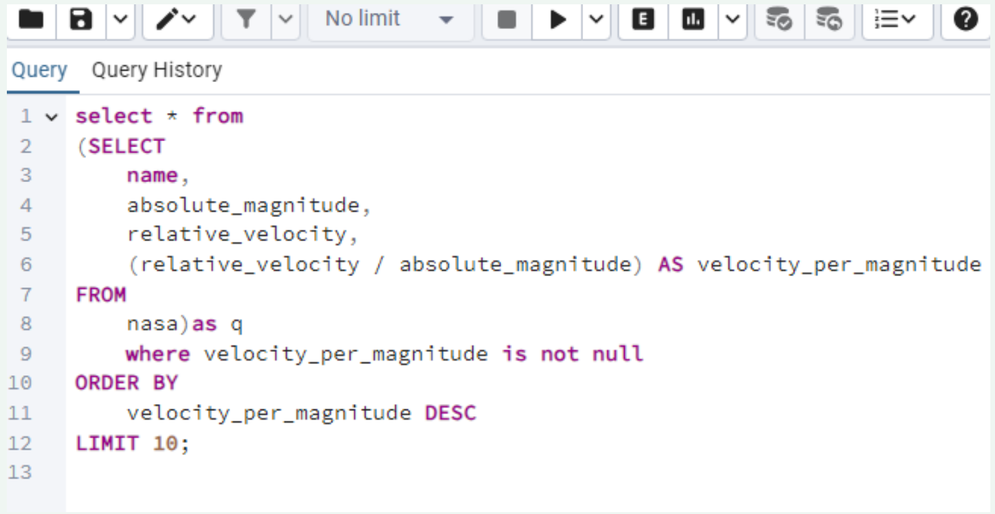
Query | Query History

```
1  select name, is_hazardous, round(cast(miss_distance as decimal),2) as miss_distance
2  from nasa
3  where is_hazardous = 'false'
4  order by miss_distance
5  limit 1
```

Data Output | Messages | Notifications

	name character varying (255)	is_hazardous boolean	miss_distance numeric
1	(2020 VT4)	false	6745.53

16. Write a query to find the top 10 asteroids with the highest relative velocity per unit of absolute magnitude. Return their names, relative velocity, absolute magnitude, and this calculated ratio.




```sql
select * from
(SELECT
    name,
    absolute_magnitude,
    relative_velocity,
    (relative_velocity / absolute_magnitude) AS velocity_per_magnitude
FROM
    nasa)as q
    where velocity_per_magnitude is not null
ORDER BY
    velocity_per_magnitude DESC
LIMIT 10;
```

Data Output | Messages | Notifications

	name character varying (255)	absolute_magnitude double precision	relative_velocity double precision	velocity_per_magnitude double precision
1	343158 Marsyas (2009 HC82)	16	291781.1066131202	18236.31916332001
2	343158 Marsyas (2009 HC82)	16	286612.8729117869	17913.30455698668
3	343158 Marsyas (2009 HC82)	16	280795.7229586679	17549.732684916744
4	343158 Marsyas (2009 HC82)	16	275303.1834955411	17206.44896847132
5	343158 Marsyas (2009 HC82)	16	269539.1865080766	16846.19915675479
6	343158 Marsyas (2009 HC82)	16	263450.7812432873	16465.673827705457
7	343158 Marsyas (2009 HC82)	16	257788.7925003576	16111.79953127235
8	343158 Marsyas (2009 HC82)	16	252111.0546057838	15756.940912861488
9	(2020 BZ12)	18.12	279142.2492064088	15405.201391082162
10	343158 Marsyas (2009 HC82)	16	246265.2504780808	15391.57815488005

Thank You