SQL AGGREGATE FUNCTIONS

What is Aggregate function in SQL?

1. Aggregate functions helps to summarize large volumes of data.
2. These functions help us to produce a single value for an entire group or table.
3. These functions operate on multiple rows and generate one value as a result.
4. With aggregate functions two more keywords comes into existence i.e. GROUP BY and HAVING.
5. Whenever, you are writing a aggregate function query and you want to restrict the data you need to use HAVING Clause instead of WHERE.

E.g. Let’s suppose I want to find out the avg of the salary column for every department and then want to see only those rows where the average is more than 10000.

```
Select department, avg(salary) from employees
group by department
having avg(salary) > 10000;
```

As we need to calculate the avg salary for every department I grouped the data based on department and as I am giving condition on an aggregated column I used having clause.

AGGREGATE FUNCTIONS IN SQL

1. COUNT

1. SUM

1. AVERAGE

1. MIN

1. MAX

COUNT()

1. COUNT() function returns the number of rows in a table.
2. It can also help you to give the count of rows matching a specific condition.
3. Let’s consider the below example.

I want to count the number of rows in employees table so I can write the below query

```
Select count(*) from employees;
```

I can also pass a column name in the count() function however, it will not make any difference as it will never happen in a table that one column has 100 rows and another column has only 10 rows.

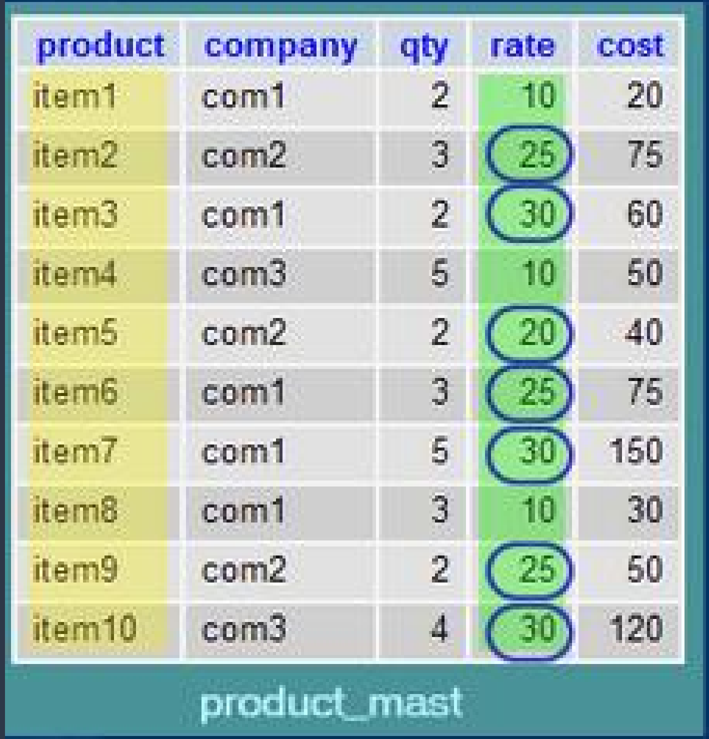
This is only applicable when you are counting the number of rows in a column where there are null values as the count function will not count the NULLs.

So, the below query is exactly the same as the above query.

```
Select count(salary) from employees;
```

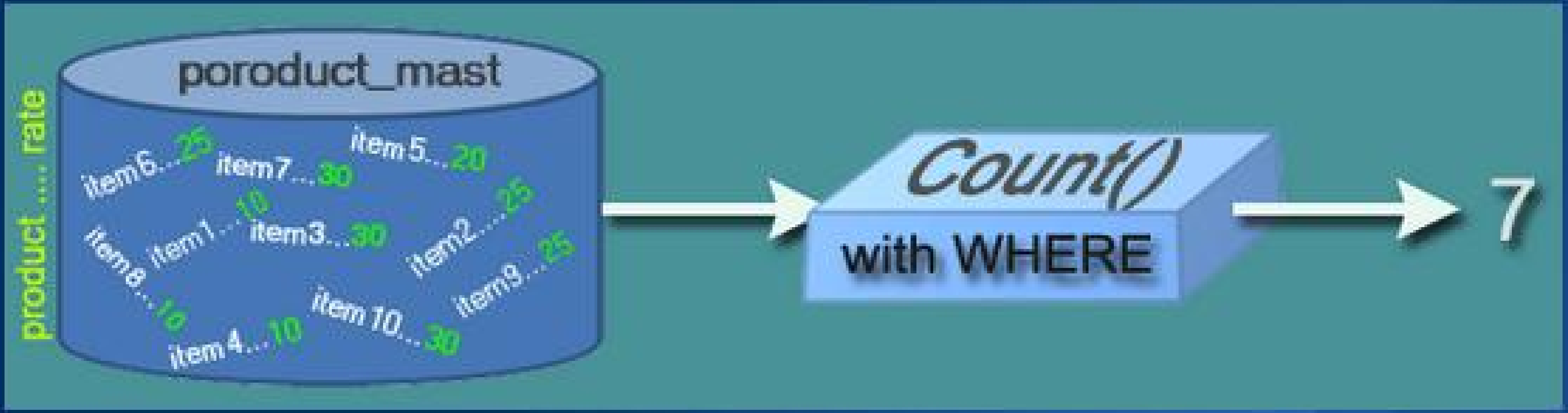
Example : COUNT() with WHERE

Example : SELECT COUNT(*)
FROM product_mast
WHERE rate>=20;



product	company	qty	rate	cost
item1	com1	2	10	20
item2	com2	3	25	75
item3	com1	2	30	60
item4	com3	5	10	50
item5	com2	2	20	40
item6	com1	3	25	75
item7	com1	5	30	150
item8	com1	3	10	30
item9	com2	2	25	50
item10	com3	4	30	120

product_mast



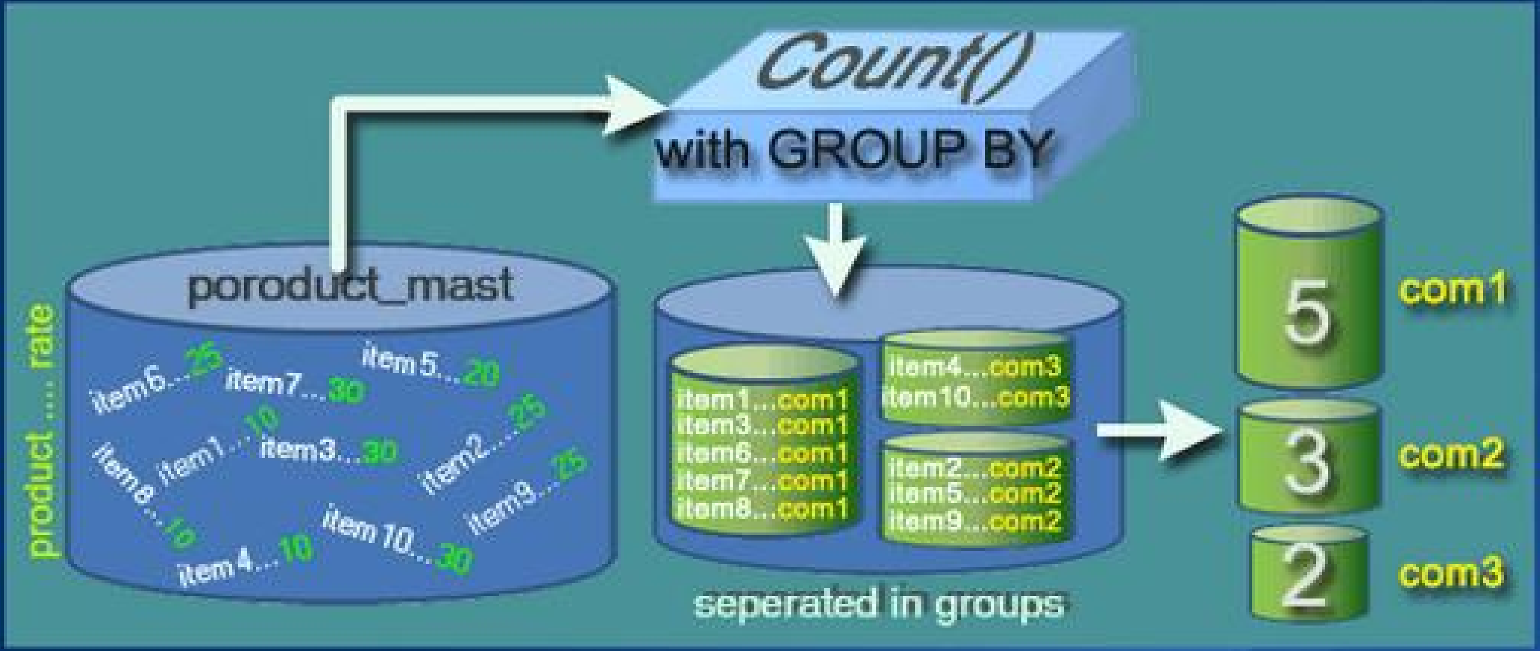
Output : 7

Example : COUNT() with GROUP BY

product	company	qty	rate	cost
item1	com1	2	10	20
item2	com2	3	25	75
item3	com1	2	30	60
item4	com3	5	10	50
item5	com2	2	20	40
item6	com1	3	25	75
item7	com1	5	30	150
item8	com1	3	10	30
item9	com2	2	25	50
item10	com3	4	30	120

product_mast

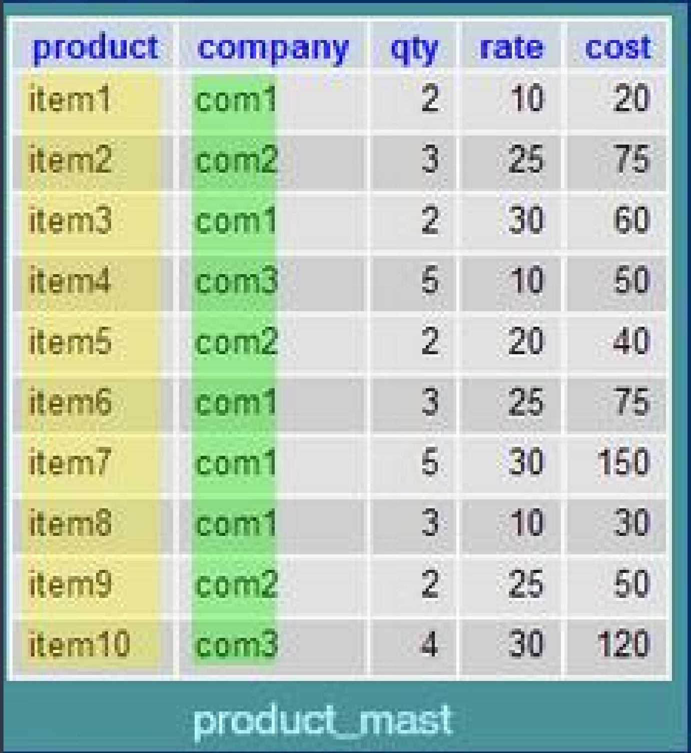
Example : SELECT company, COUNT(*)
FROM product_mast GROUP BY company;



Output :

```
com1   5
com2   3
com3   2
```

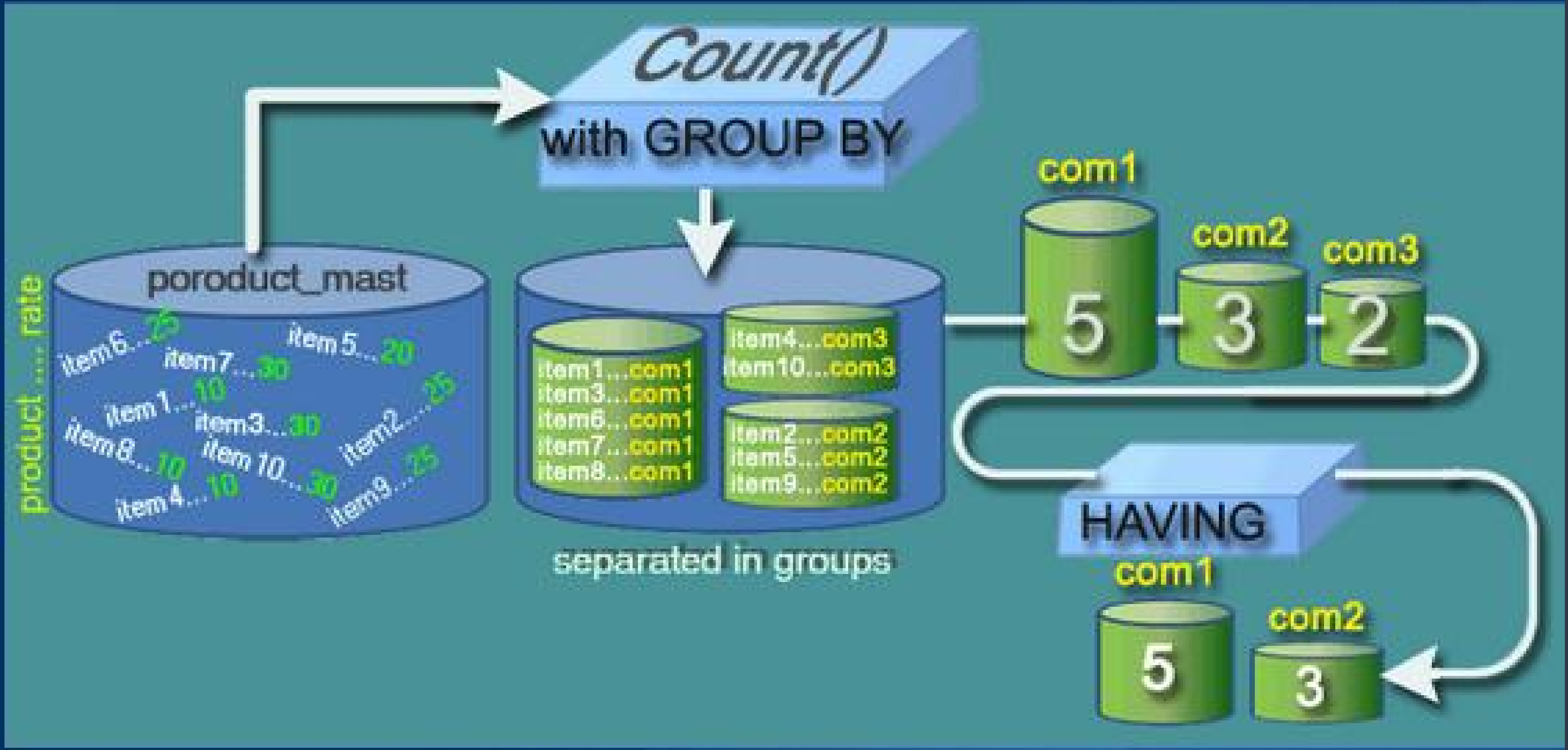
Example : COUNT() with HAVING



product	company	qty	rate	cost
item1	com1	2	10	20
item2	com2	3	25	75
item3	com1	2	30	60
item4	com3	5	10	50
item5	com2	2	20	40
item6	com1	3	25	75
item7	com1	5	30	150
item8	com1	3	10	30
item9	com2	2	25	50
item10	com3	4	30	120

product_mast

Example : SELECT company, COUNT(*) FROM product_mast GROUP BY company HAVING COUNT(*)>2;



Output :

com1 5

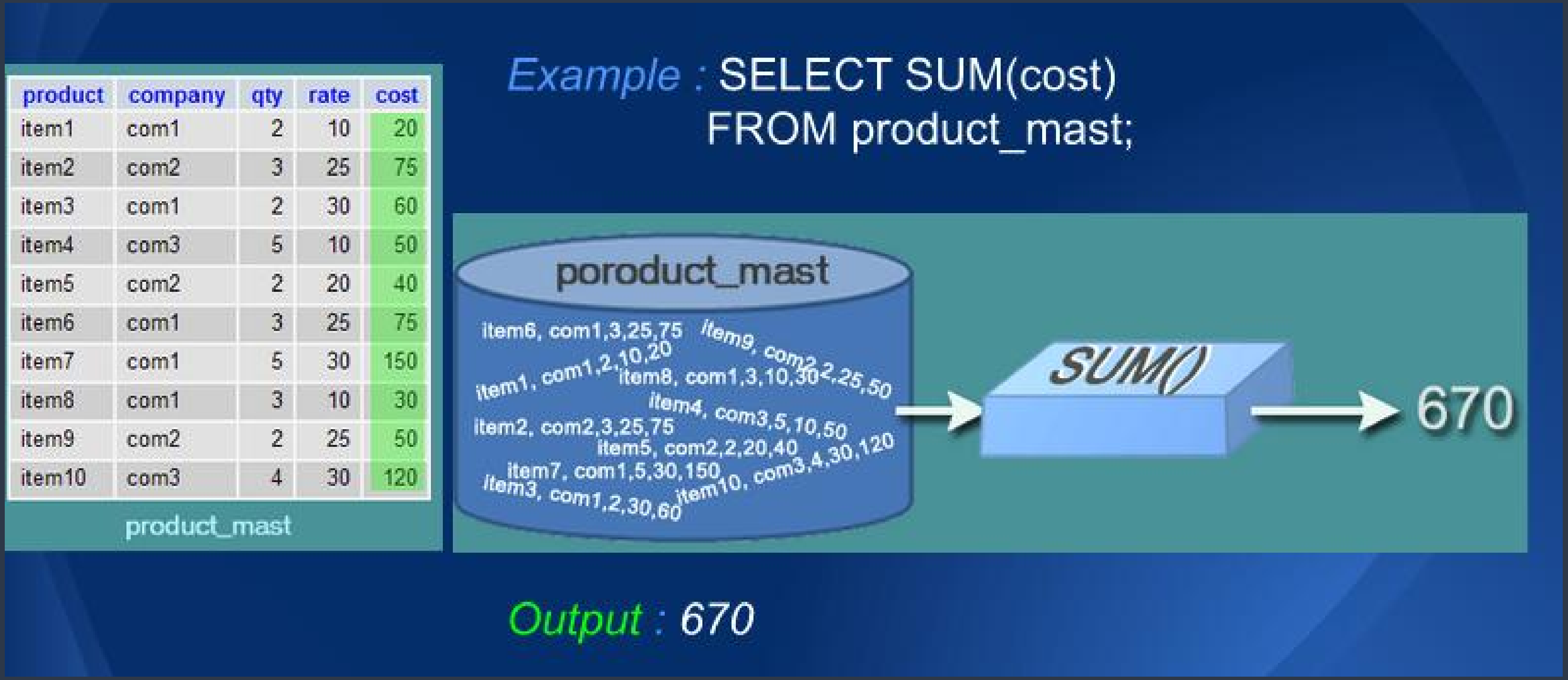
com2 3

SUM()

SUM() inside this function you need to pass a numerical column and it will add all the rows in that column and will give you one value as an output.

SYNTAX:

```
select sum(column_name) from table_name
```



product	company	qty	rate	cost
item1	com1	2	10	20
item2	com2	3	25	75
item3	com1	2	30	60
item4	com3	5	10	50
item5	com2	2	20	40
item6	com1	3	25	75
item7	com1	5	30	150
item8	com1	3	10	30
item9	com2	2	25	50
item10	com3	4	30	120

product_mast

Example : SELECT SUM(cost)
FROM product_mast;

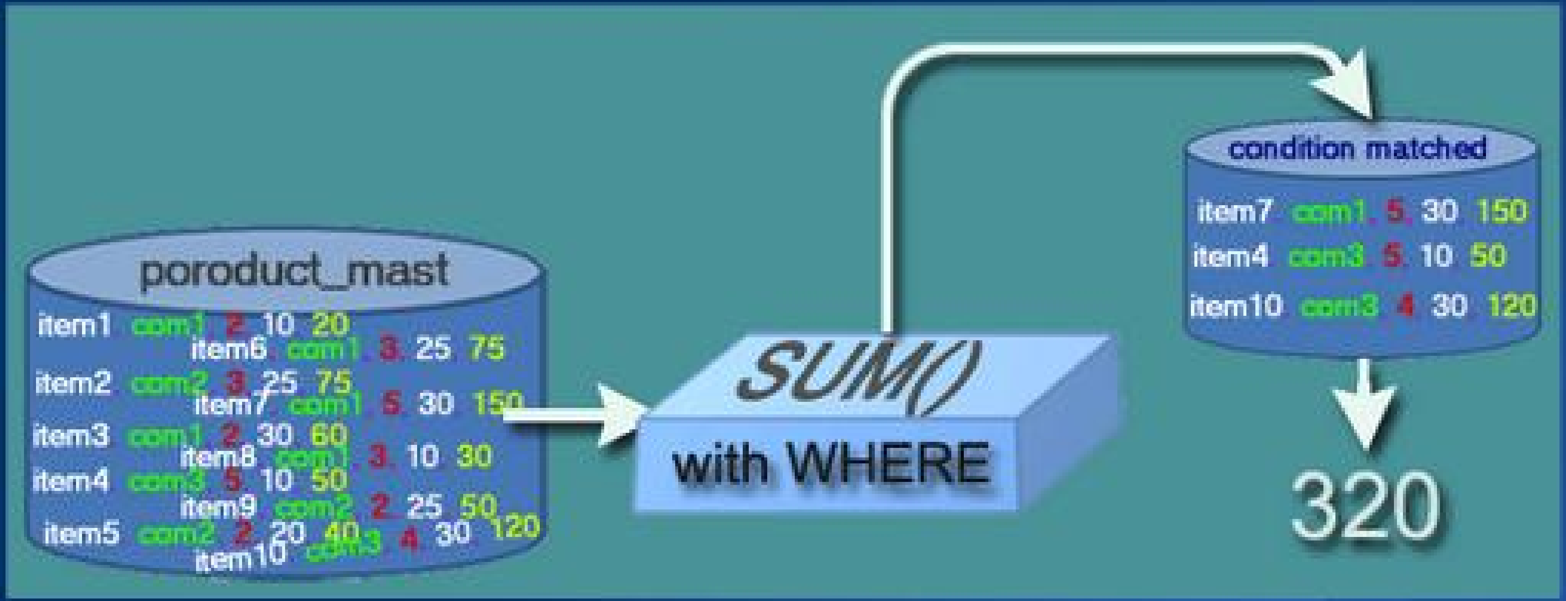
Output : 670

Example : SUM() with WHERE

Example :
```
SELECT SUM(cost)
FROM product_mast
WHERE qty>3;
```

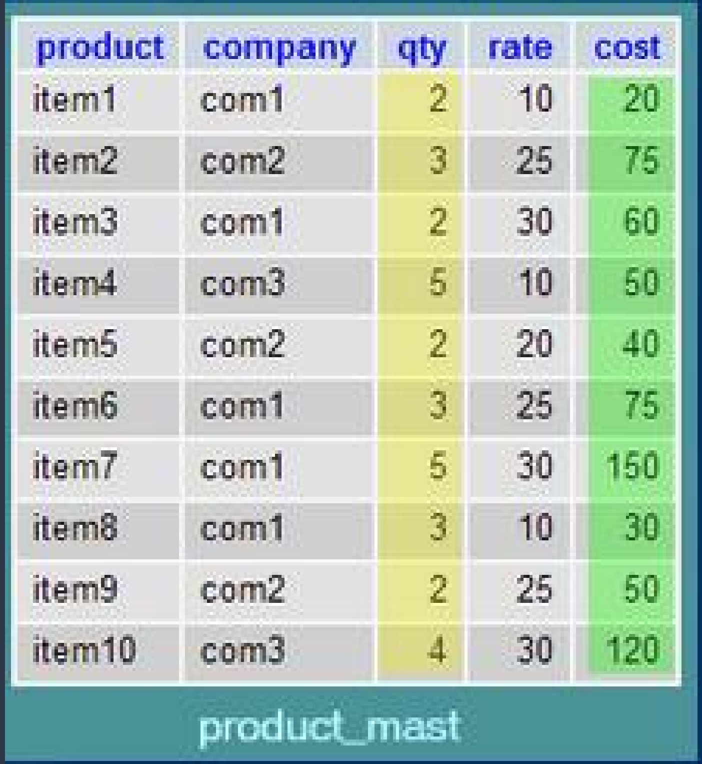
product	company	qty	rate	cost
item1	com1	2	10	20
item2	com2	3	25	75
item3	com1	2	30	60
item4	com3	5	10	50
item5	com2	2	20	40
item6	com1	3	25	75
item7	com1	5	30	150
item8	com1	3	10	30
item9	com2	2	25	50
item10	com3	4	30	120

product_mast



Output : *320*

Example : SUM() with GROUP BY



product	company	qty	rate	cost
item1	com1	2	10	20
item2	com2	3	25	75
item3	com1	2	30	60
item4	com3	5	10	50
item5	com2	2	20	40
item6	com1	3	25	75
item7	com1	5	30	150
item8	com1	3	10	30
item9	com2	2	25	50
item10	com3	4	30	120

product_mast

Example :

```
SELECT SUM(cost)
FROM product_mast
WHERE qty>3
        GROUP BY
company;
```

Output :

```
com1 150
com3 170
```

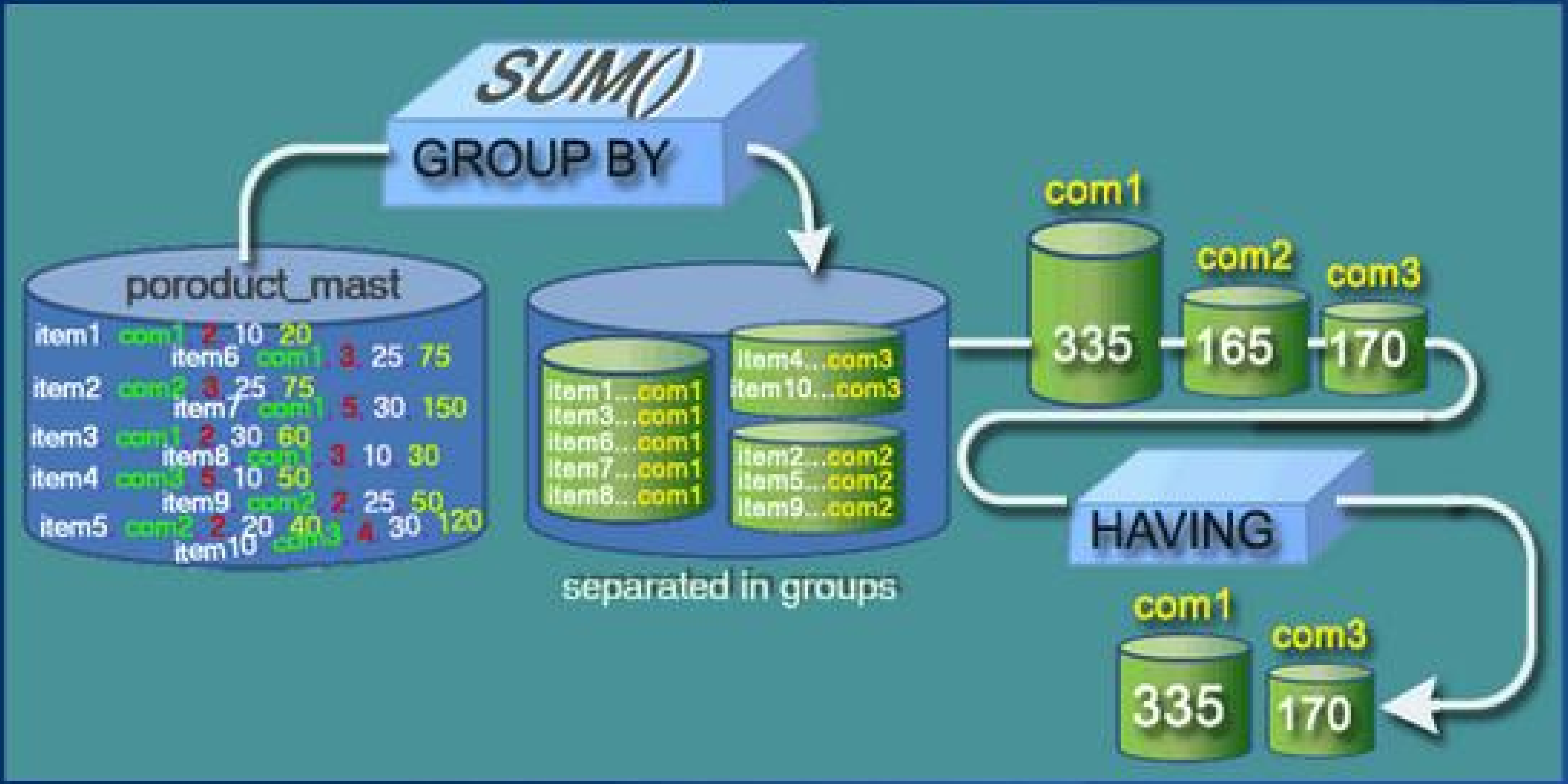
Example : SUM() with HAVING

product	company	qty	rate	cost
item1	com1	2	10	20
item2	com2	3	25	75
item3	com1	2	30	60
item4	com3	5	10	50
item5	com2	2	20	40
item6	com1	3	25	75
item7	com1	5	30	150
item8	com1	3	10	30
item9	com2	2	25	50
item10	com3	4	30	120

product_mast

Example :

```
SELECT company, SUM(cost)
FROM product_mast
GROUP BY company
HAVING SUM(cost)>=170;
```



Output :

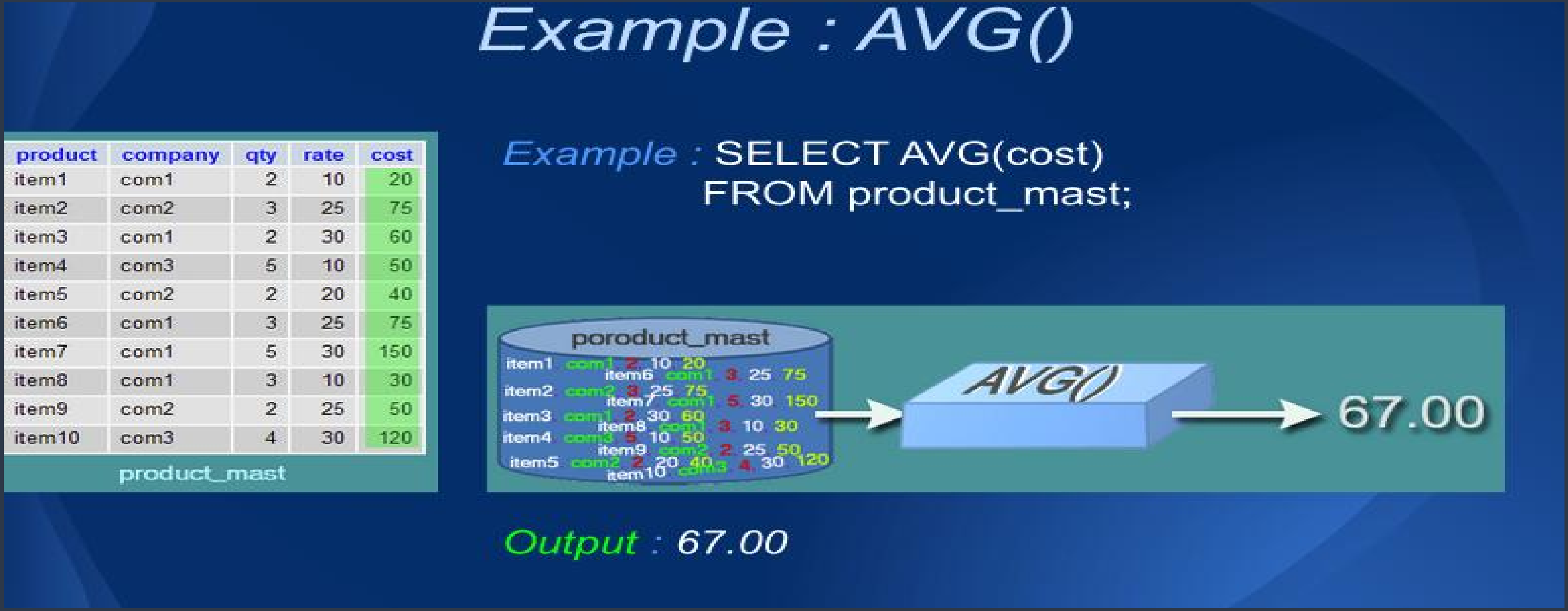
com1 335
com3 170

AVG()

AVG() function stands for Average. You need to pass a column inside this function and it will add all the values in that column and divide it with the total number of rows and will give you one value as an output.

SYNTAX:

```
select avg(column_name) from table_name;
```



Example : AVG()

product	company	qty	rate	cost
item1	com1	2	10	20
item2	com2	3	25	75
item3	com1	2	30	60
item4	com3	5	10	50
item5	com2	2	20	40
item6	com1	3	25	75
item7	com1	5	30	150
item8	com1	3	10	30
item9	com2	2	25	50
item10	com3	4	30	120

product_mast

Example : SELECT AVG(cost)
FROM product_mast;

Output : 67.00

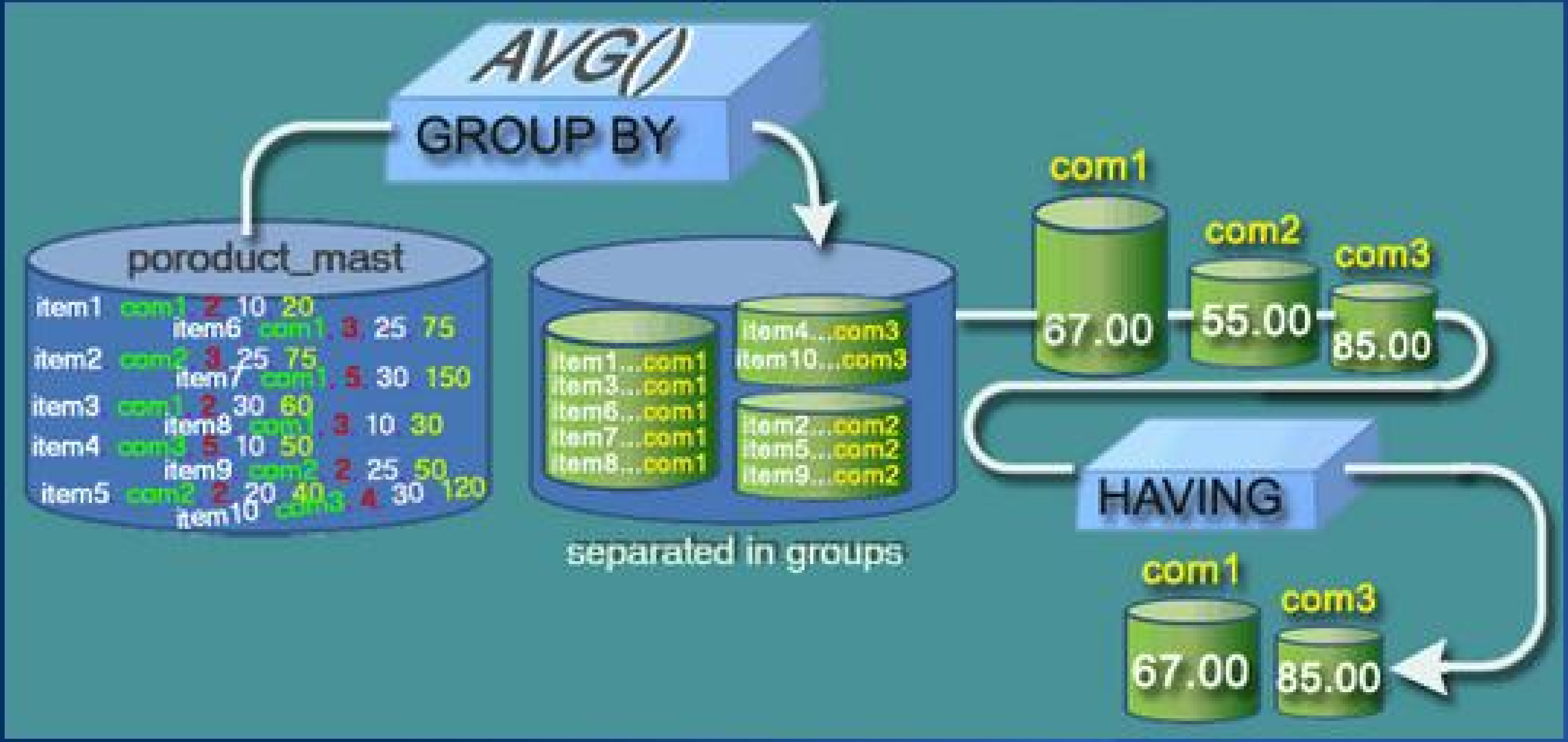
Example : AVG() with HAVING

Example :

```
SELECT company, AVG(cost)
FROM product_mast
GROUP BY company
HAVING AVG(cost)>=65;
```

product	company	qty	rate	cost
item1	com1	2	10	20
item2	com2	3	25	75
item3	com1	2	30	60
item4	com3	5	10	50
item5	com2	2	20	40
item6	com1	3	25	75
item7	com1	5	30	150
item8	com1	3	10	30
item9	com2	2	25	50
item10	com3	4	30	120

product_mast



Output :

com1 67.00
com3 85.00

MIN() & MAX()

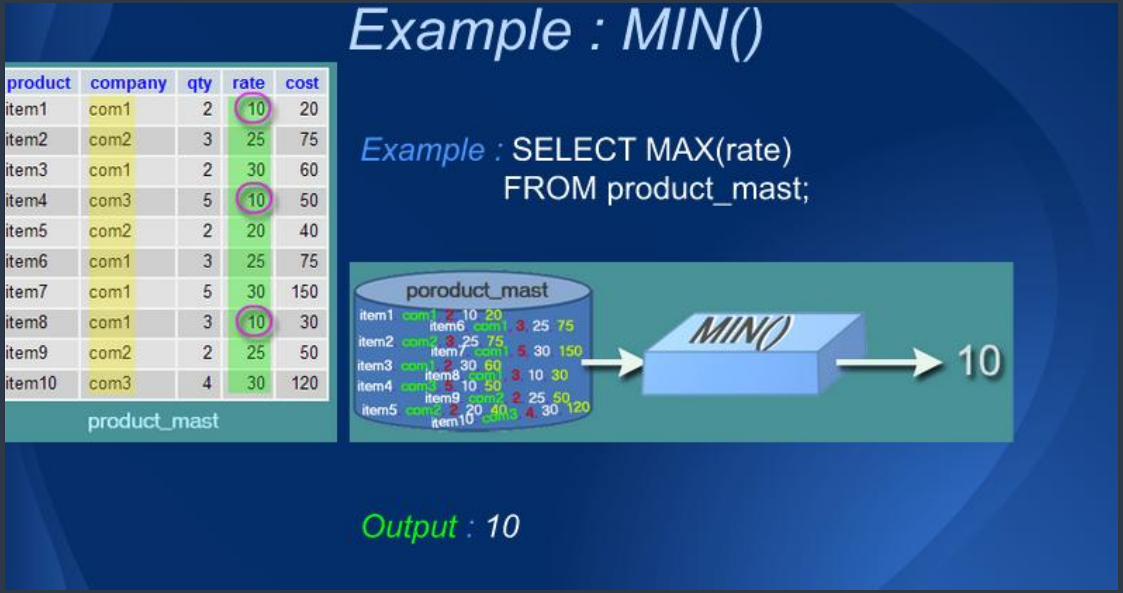
Whenever, you would like to find out the minimum value in a column you can make use of MIN() function.

SYNTAX:

```
select min(column_name) from table_name;
```

Whenever, you would like to find out the maximum value in a column you can make use of MAX() function

```
select max(column_name) from table_name;
```



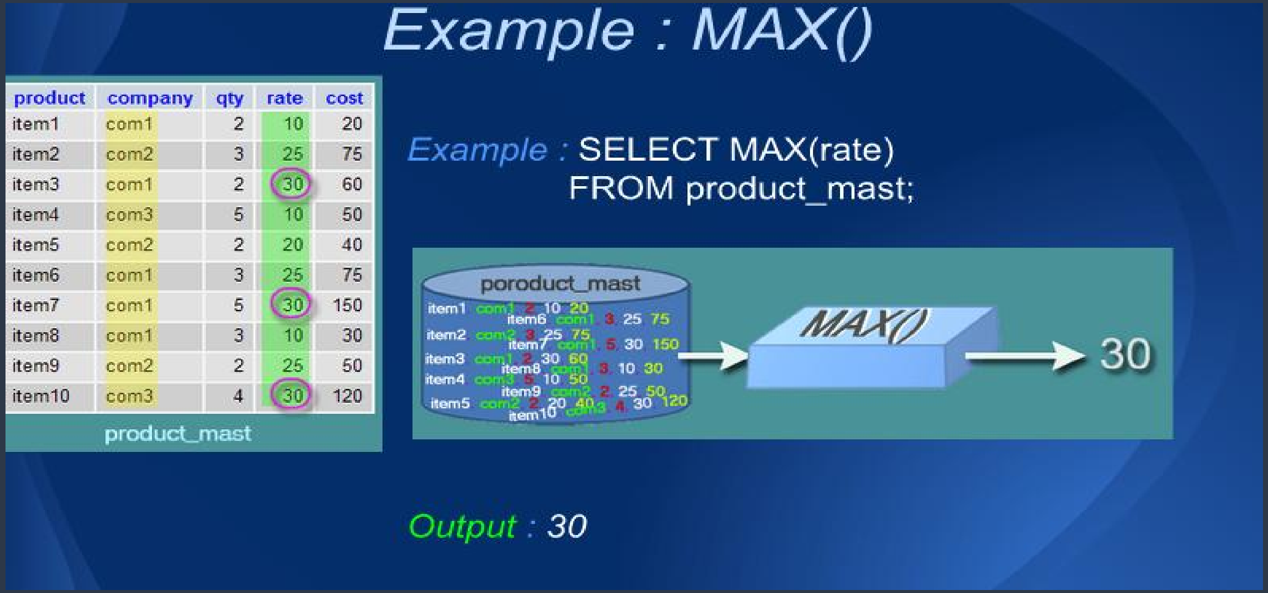
Example : MIN()

product	company	qty	rate	cost
item1	com1	2	10	20
item2	com2	3	25	75
item3	com1	2	30	60
item4	com3	5	10	50
item5	com2	2	20	40
item6	com1	3	25	75
item7	com1	5	30	150
item8	com1	3	10	30
item9	com2	2	25	50
item10	com3	4	30	120

product_mast

Example : SELECT MAX(rate)
FROM product_mast;

Output : 10



Example : MAX()

product	company	qty	rate	cost
item1	com1	2	10	20
item2	com2	3	25	75
item3	com1	2	30	60
item4	com3	5	10	50
item5	com2	2	20	40
item6	com1	3	25	75
item7	com1	5	30	150
item8	com1	3	10	30
item9	com2	2	25	50
item10	com3	4	30	120

product_mast

Example : SELECT MAX(rate)
FROM product_mast;

Output : 30