



A simple rule to remember when using aggregates in a selection is if a column isn't aggregated it **must** be included in the group by clause.

The previous example would work perfectly once a group by is added for the stock number ...

The screenshot shows the SQL Developer interface. The top pane, titled 'Query Builder', contains the following SQL query:

```
select sum(od_quantity)
,od_stock_number
from order_details
group by od_stock_number
```

The bottom pane, titled 'Query Result', shows the results of the query. It includes a toolbar with icons for refresh, print, and SQL, and a status bar indicating 'All Rows Fetched: 13 in 0 seconds'. The results are displayed in a table with the following columns: ID, SUM(OD_QUANTITY), and OD_STOCK_NUMBER.

ID	SUM(OD_QUANTITY)	OD_STOCK_NUMBER
1	5	25
2	3	1
3	3	13
4	1	14
5	10	26
6	1	21
7	3	2
8	1	24
9	5	4
10	10	23
11	1	17
12	1	3
13	4	15



The User should be aware that the Where clause in a Group By will limit records before the records are grouped because a where clause works on the individual rows, if the User needs to limit records after they have been 'grouped' then the Having clause needs to be used.

The next examples will show the usage of the where clause and the order by, note that order by can be used on any column including the aggregated column ...

Aggregating Values – Using Where with Group By



Worksheet | Query Builder

```
select sum(od_quantity)
,od_stock_number
from order_details
where od_stock_number in (1,2,3,4,15,17)
group by od_stock_number
order by od_stock_number
```

Query Result x

SQL | All Rows Fetched: 6 in 0.016 seconds

	SUM(OD_QUANTITY)	OD_STOCK_NUMBER
1	3	1
2	3	2
3	1	3
4	5	4
5	4	15
6	1	17



The Having clause is used when the User wishes to limit records after they have been grouped. Therefore Having is the group by equivalent of a where clause.

Having can only be used when the Group By clause has been used and must follow the group by in the SQL syntax.

The Having clause must work on the aggregated column(s)

The following example has both Having and Order By ...

Aggregating Values – Using Having with Group By



Worksheet | Query Builder

```
select sum(od_quantity)
,od_stock_number
from order_details
where od_stock_number in (1,2,3,4,15,17)
group by od_stock_number
having sum(od_quantity) > 3
order by 1
```

Query Result x

SQL | All Rows Fetched: 2 in 0.015 seconds

	SUM(OD_QUANTITY)	OD_STOCK_NUMBER
1	4	15
2	5	4