



# Day 8

# MySQL ROLLUP





# WITH ROLLUP

- The **WITH ROLLUP** modifier adds extra rows to the resultset that represent super-aggregate summaries.
- The super-aggregated column is represented by a **NULL** value.
- Multiple aggregates over different columns will be added if there are multiple GROUP BY columns.



# General Syntax :

```
SELECT
    c1, c2, aggregate_function(c3)
FROM
    table_name
GROUP BY
    c1, c2 WITH ROLL UP;
```



## Key points :

- If you have more than one column specified in the GROUP BY clause, the **ROLLUP** clause assumes a hierarchy among the input columns.

```
GROUP BY c1, c2, c3 WITH ROLL UP;
```

- The **ROLLUP** assumes that there is the following hierarchy:

**c1 > c2 > c3**



# Use Case Example :

```
SELECT
    productLine,
    orderYear,
    SUM(orderValue) AS totalOrderValue
FROM
    sales
GROUP BY
    productLine,
    orderYear
WITH ROLLUP;
```

# Output :

Subtotal  
Aggregate

	productLine	orderYear	totalOrderValue
▶	Classic Cars	2003	1374832.22
	Classic Cars	2004	1763136.73
	Classic Cars	2005	715953.54
	Classic Cars	NULL	3853922.49
	Motorcycles	2003	348909.24
	Motorcycles	2004	527243.84
	Motorcycles	2005	245273.04
	Motorcycles	NULL	1121426.12
	Planes	2003	309784.20
	Planes	2004	471971.46
	Planes	2005	172881.88
	Planes	NULL	954637.54
	Ships	2003	222182.08
	Ships	2004	337326.10
	Ships	2005	104490.16
	Ships	NULL	663998.34
	Trains	2003	65822.05
	Trains	2004	96285.53
	Trains	2005	26425.34
	Trains	NULL	188532.92
	Trucks and...	2003	376657.12
	Trucks and...	2004	465390.00
	Trucks and...	2005	182066.45
	Trucks and...	NULL	1024113.57
	Vintage Cars	2003	619161.48
	Vintage Cars	2004	854551.85
	Vintage Cars	2005	323846.30
	Vintage Cars	NULL	1797559.63
	NULL	NULL	9604190.61

Grand total  
Aggregate