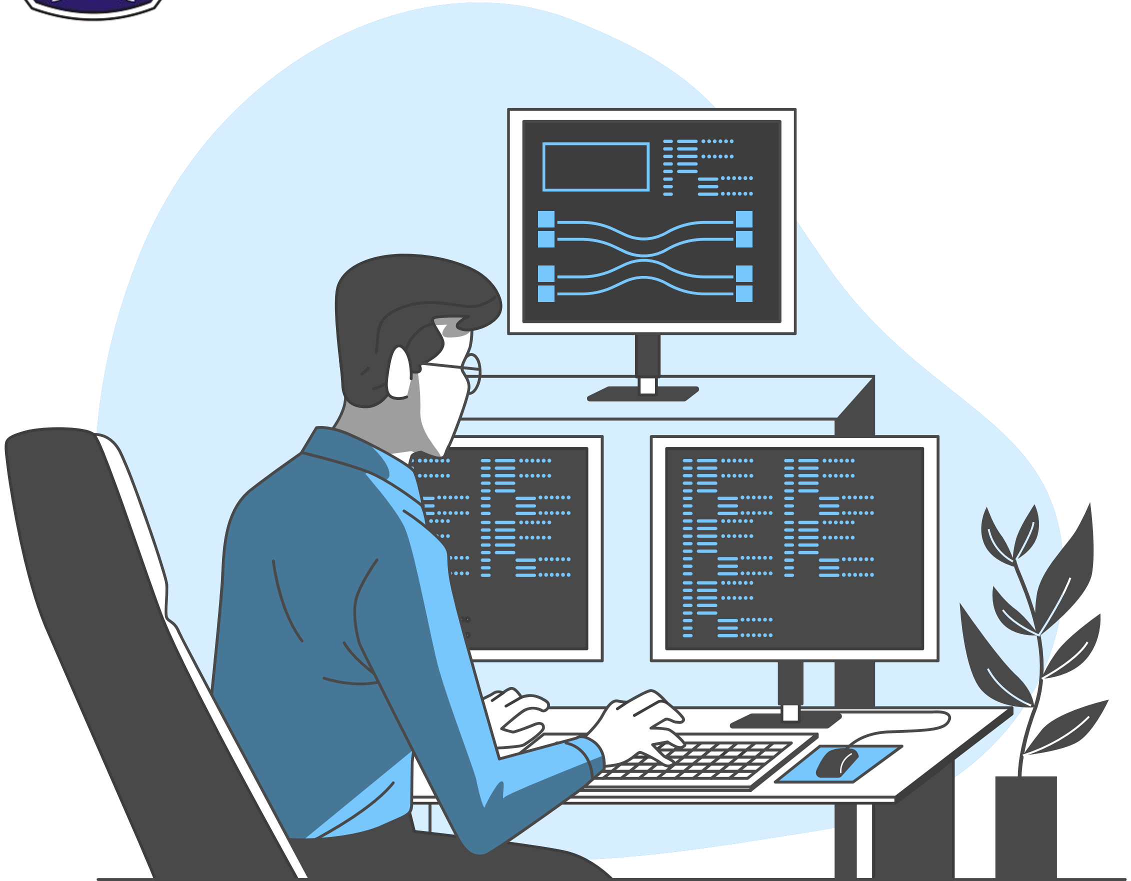


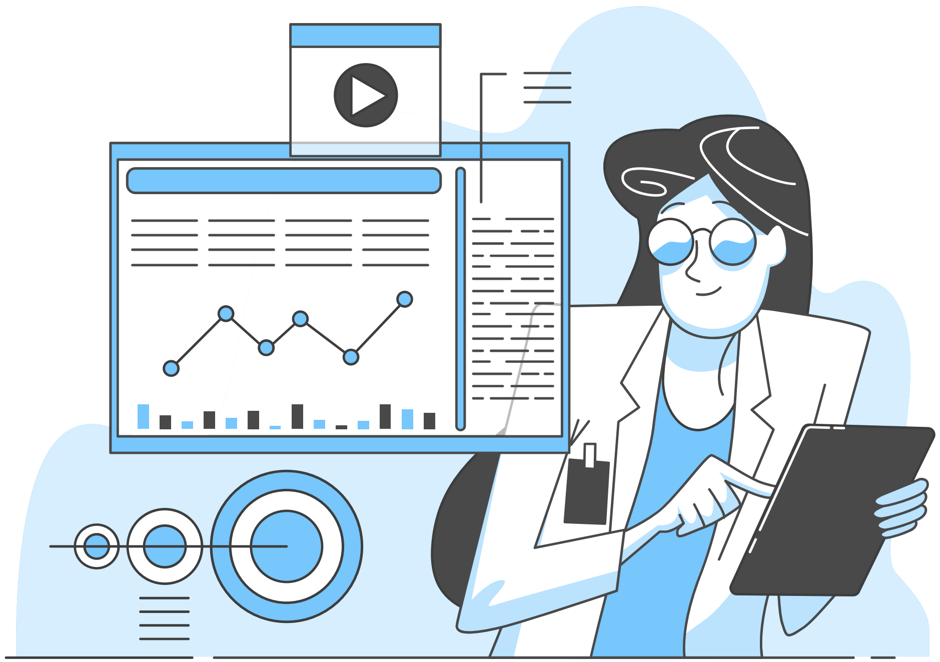
BASIS DATA LANJUT

Pertemuan 8

Pivoting dan Grouping Set

Team Teaching Basis Data Lanjut
JTI - Polinema 2024

Table of Contents



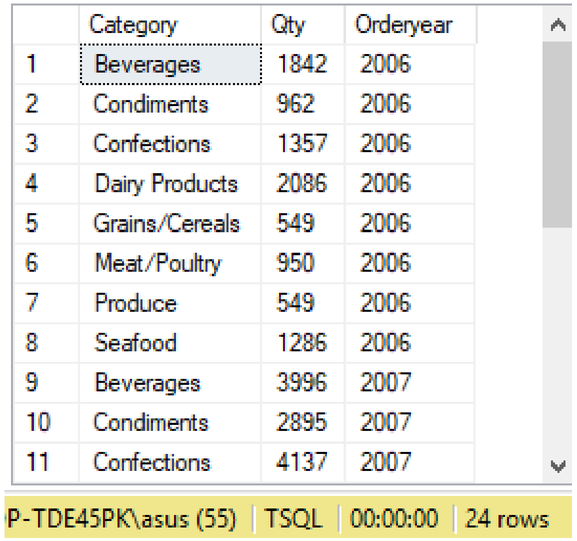
PIVOT & unpivot

KONSEP PIVOT

- Secara umum, output SQL menghasilkan data secara vertical
- Dalam banyak kasus, menampilkan data secara horizontal akan lebih mudah dimengerti
- Operator PIVOT digunakan untuk memutar (rotate) data dari semula berorientasi row-based (vertical) menjadi berorientasi columns-based (horizontal)
- Nilai dari suatu kolom yang berbeda akan digunakan sebagai judul untuk kolom lainnya

KONSEP PIVOT (1)

- Data secara vertikal

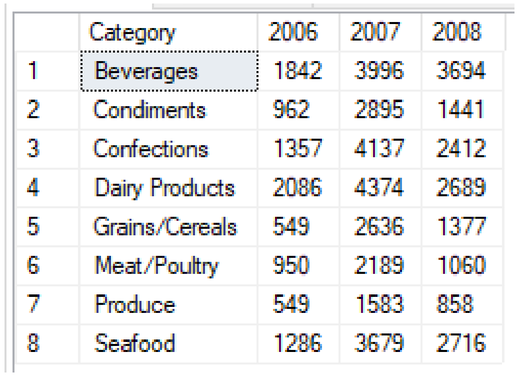


	Category	Qty	Orderyear
1	Beverages	1842	2006
2	Condiments	962	2006
3	Confections	1357	2006
4	Dairy Products	2086	2006
5	Grains/Cereals	549	2006
6	Meat/Poultry	950	2006
7	Produce	549	2006
8	Seafood	1286	2006
9	Beverages	3996	2007
10	Condiments	2895	2007
11	Confections	4137	2007

```
P-TDE45PK\asus (55) | TSQL | 00:00:00 | 24 rows
```

PIVOT

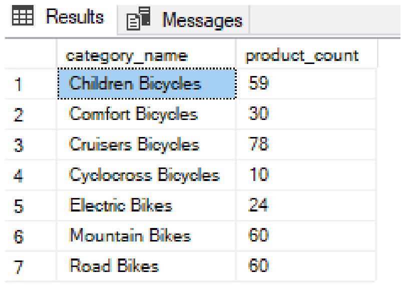
- Data secara horizontal



	Category	2006	2007	2008
1	Beverages	1842	3996	3694
2	Condiments	962	2895	1441
3	Confections	1357	4137	2412
4	Dairy Products	2086	4374	2689
5	Grains/Cereals	549	2636	1377
6	Meat/Poultry	950	2189	1060
7	Produce	549	1583	858
8	Seafood	1286	3679	2716

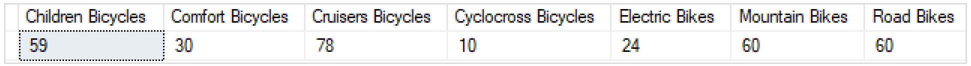
CONTOH DBASE TOKOSEPEDA

```
-- seleksi berdasarkan nama kategori dan
jumlah produk per kategori
SELECT
    category_name,
    COUNT(product_id) product_count
FROM
    production.products p
    INNER JOIN production.categories c
        ON c.category_id = p.category_id
GROUP BY
    category_name;
```

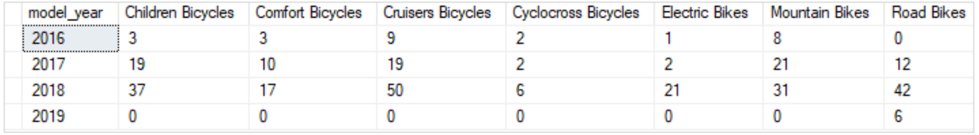


Results | Messages

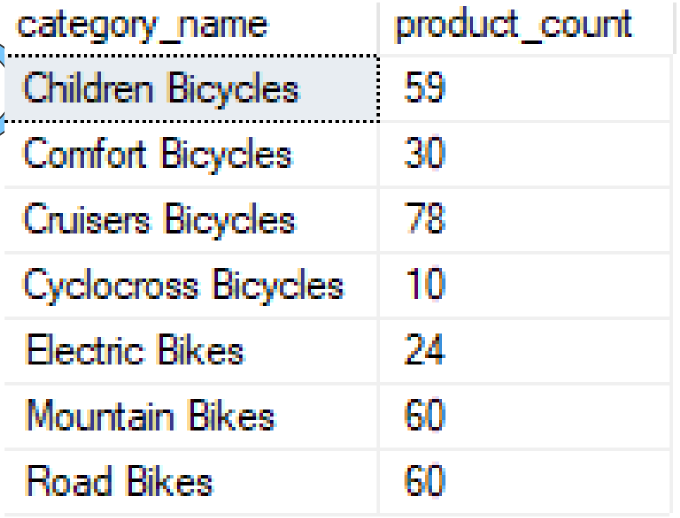
	category_name	product_count
1	Children Bicycles	59
2	Comfort Bicycles	30
3	Cruisers Bicycles	78
4	Cyclocross Bicycles	10
5	Electric Bikes	24
6	Mountain Bikes	60
7	Road Bikes	60



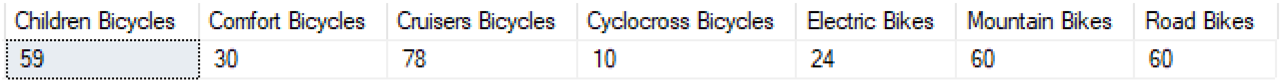
Children Bicycles	Comfort Bicycles	Cruisers Bicycles	Cyclocross Bicycles	Electric Bikes	Mountain Bikes	Road Bikes
59	30	78	10	24	60	60



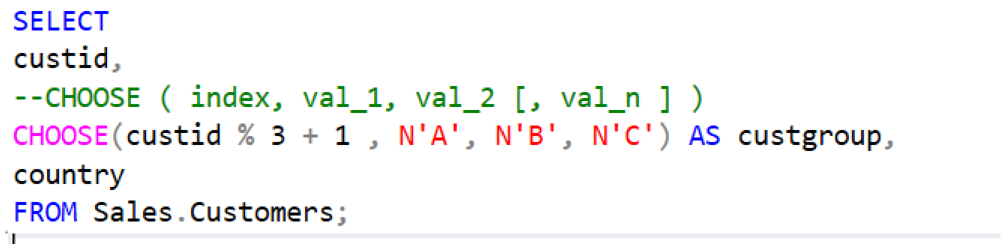
model_year	Children Bicycles	Comfort Bicycles	Cruisers Bicycles	Cyclocross Bicycles	Electric Bikes	Mountain Bikes	Road Bikes
2016	3	3	9	2	1	8	0
2017	19	10	19	2	2	21	12
2018	37	17	50	6	21	31	42
2019	0	0	0	0	0	0	6



category_name	product_count
Children Bicycles	59
Comfort Bicycles	30
Cruisers Bicycles	78
Cyclocross Bicycles	10
Electric Bikes	24
Mountain Bikes	60
Road Bikes	60



Children Bicycles	Comfort Bicycles	Cruisers Bicycles	Cyclocross Bicycles	Electric Bikes	Mountain Bikes	Road Bikes
59	30	78	10	24	60	60



```
SELECT
custid,
--CHOOSE ( index, val_1, val_2 [, val_n ] )
CHOOSE(custid % 3 + 1 , N'A', N'B', N'C') AS custgroup,
country
FROM Sales.Customers;
```

Cara membuat pivot query

- First, select a base dataset for pivoting.
- Second, create a temporary result by using a derived table or common table expression (CTE)
- Third, **apply the PIVOT operator.**

Syntax PIVOT

```
SELECT <non-pivoted column>,
    [first pivoted column] AS <column name>,
    [second pivoted column] AS <column name>,
    ...
    [last pivoted column] AS <column name>
FROM
    (<SELECT query that produces the data>)
    AS <alias for the source query>
PIVOT
(
    <aggregation function>(<column being aggregated>)
FOR
[<column that contains the values that will become column headers>]
    IN ( [first pivoted column], [second pivoted column],
    ... [last pivoted column])
) AS <alias for the pivot table>
<optional ORDER BY clause>;
```

ELEMEN PIVOT

- **GROUPING**

  menentukan kolom mana yang akan digunakan untuk pengelompokan data

- **SPREADING**

  menentukan list nilai yang akan dijadikan judul kolom untuk hasil pivot

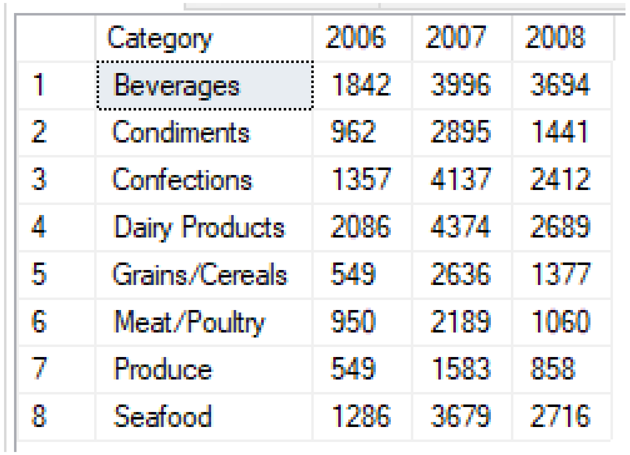
- **AGGREGATION**

  menentukan fungsi agregasi (SUM dkk) yang akan digunakan pada baris data yang dikelompokkan

Query PIVOT

```
SELECT Category, [2006],[2007],[2008]
FROM(
    SELECT Category, Qty, Orderyear FROM Sales.CategoryQtyYear) AS D
PIVOT(
      SUM(qty)
FOR orderyear IN ([2006],[2007],[2008])) AS pvt;
```

GROUPING : Category
SPREADING : ORDERYEAR IN (2006,2007,2008)
AGGREGATION : SUM(QTY)



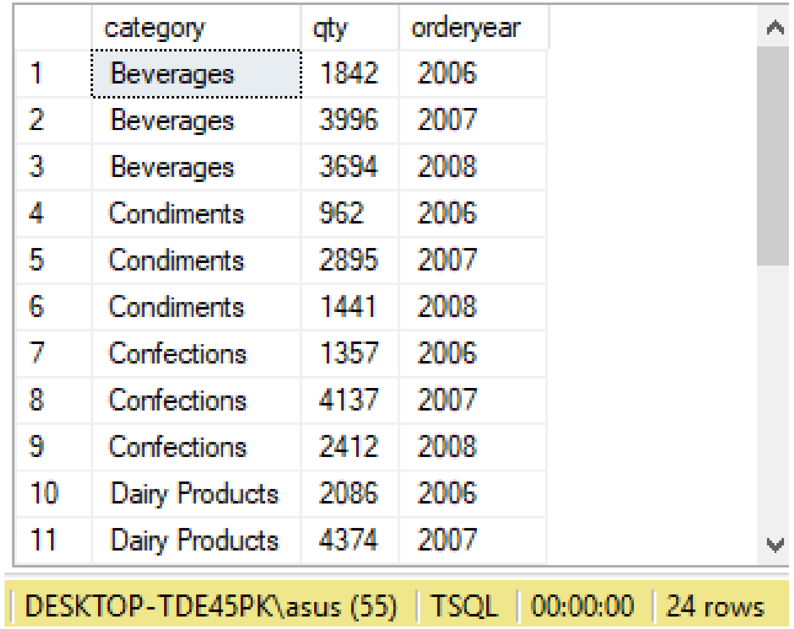
	Category	2006	2007	2008
1	Beverages	1842	3996	3694
2	Condiments	962	2895	1441
3	Confections	1357	4137	2412
4	Dairy Products	2086	4374	2689
5	Grains/Cereals	549	2636	1377
6	Meat/Poultry	950	2189	1060
7	Produce	549	1583	858
8	Seafood	1286	3679	2716

UNPIVOT

- Kebalikan dari logika PIVOT
- Mengembalikan data column-based menjadi row-based
- Untuk menggunakan operator UNPIVOT diperlukan:
  - ❑ kolom yang akan dilakukan UNPIVOT
  - ❑ nama untuk kolom baru yang akan menampilkan nilai UNPIVOT

Query unpivot

```sql
SELECT category, qty, orderyear
FROM Sales.PivotedCategorySales
UNPIVOT(qty FOR orderyear IN([2006],[2007],[2008])) AS unpvt;
```



	category	qty	orderyear
1	Beverages	1842	2006
2	Beverages	3996	2007
3	Beverages	3694	2008
4	Condiments	962	2006
5	Condiments	2895	2007
6	Condiments	1441	2008
7	Confections	1357	2006
8	Confections	4137	2007
9	Confections	2412	2008
10	Dairy Products	2086	2006
11	Dairy Products	4374	2007

DESKTOP-TDE45PK\asus (55) | TSQL | 00:00:00 | 24 rows

DEMO pivot

- Create View Sales.CategoryQtyYear

```
CREATE VIEW Sales.CategoryQtyYear
AS
SELECT  c.categoryname AS Category,
        od.qty AS Qty,
        YEAR(o.orderdate) AS Orderyear
FROM    Production.Categories AS c
        INNER JOIN Production.Products AS p ON c.categoryid=p.categoryid
        INNER JOIN Sales.OrderDetails AS od ON p.productid=od.productid
        INNER JOIN Sales.Orders AS o ON od.orderid=o.orderid;
GO
```

DEMO pivot (1)

- PIVOT category berdasarkan orderyear

```
SELECT  Category, [2006],[2007],[2008]
FROM (
SELECT  Category, Qty, Orderyear FROM Sales.CategoryQtyYear) AS D
    PIVOT(SUM(QTY) FOR orderyear IN ([2006],[2007],[2008])) AS pvt
ORDER BY Category;
```

DEMO UNPIVOT

- Create table yang menyimpan data yang telah dilakukan operasi PIVOT

```
CREATE TABLE [Sales].[PivotedCategorySales](
[Category] [nvarchar](15) NOT NULL,
[2006] [int] NULL,
[2007] [int] NULL,
[2008] [int] NULL);
GO
-- Populate it by pivoting from view
INSERT INTO Sales.PivotedCategorySales (Category, [2006],[2007],[2008])
SELECT Category, [2006],[2007],[2008]
FROM (SELECT  Category, Qty, Orderyear FROM Sales.CategoryQtyYear) AS D
    PIVOT(SUM(QTY) FOR orderyear IN ([2006],[2007],[2008]))AS p
```

DEMO UNPIVOT (1)

- Menerapkan operator UNPIVOT

```
SELECT category, qty, orderyear
FROM Sales.PivotedCategorySales
UNPIVOT(qty FOR orderyear IN([2006],[2007],[2008])) AS unpvt;
```

PIVOT dengan CTE

- Menampilkan data shipperid, shipcity, freight

```
SELECT shipperid, shipcity, freight
FROM Sales.Orders
WHERE shipcountry = N'Spain'
ORDER BY shipperid, shipcity;
```

- Menerapkan operator PIVOT

```
WITH PivotInput AS
 (
  SELECT shipperid, shipcity, freight
  FROM Sales.Orders
  WHERE shipcountry = N'Spain'
)
SELECT *
FROM PivotInput
  PIVOT( SUM(freight)
    FOR shipcity IN (Barcelona, Madrid, Sevilla) ) AS PivotOutput;
```

GROUPING SETS, CUBE, ROLLUP

Konsep grouping set

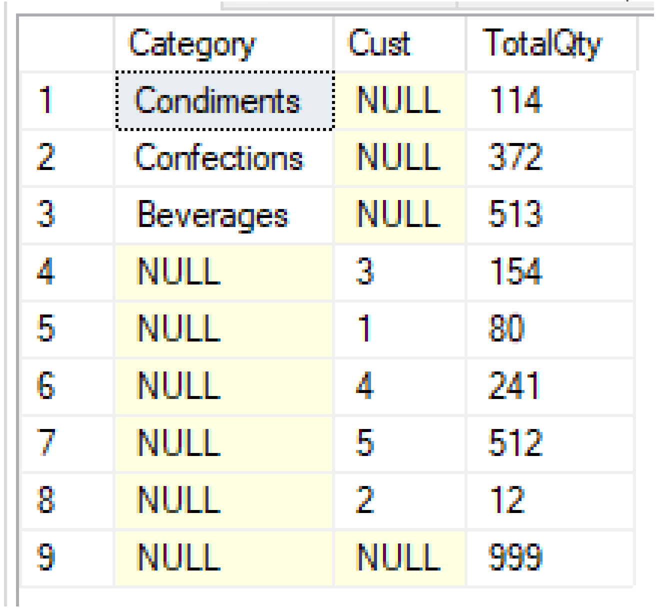
- Operator UNION digunakan untuk menggabungkan beberapa kueri
- GROUPING SET merupakan sub-clausa dari GROUP BY
- GROUPING SET memungkinkan banyak pengelompokkan untuk didefinisikan di dalam satu kueri

SIntaks

```
SELECT <column list with aggregate(s)>
FROM <source>
GROUP BY
GROUPING SETS
(
            <column_name>,--one or more columns
            <column_name>,--one or more columns
            () -- empty parentheses if aggregating all rows
);
```

Contoh Tanpa GROUPING SET

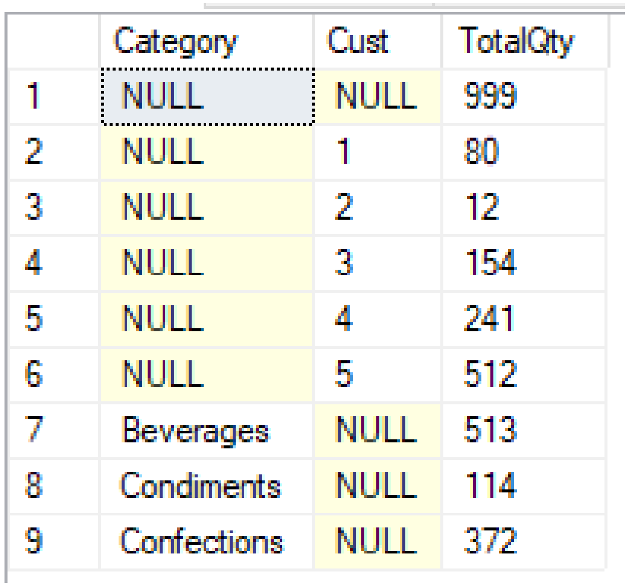
```
SELECT Category, NULL AS Cust, SUM(Qty) AS TotalQty
FROM Sales.CategorySales
GROUP BY category
UNION ALL
SELECT  NULL, Cust, SUM(Qty) AS TotalQty
FROM Sales.CategorySales
GROUP BY cust
UNION ALL
SELECT NULL, NULL, SUM(Qty) AS TotalQty
FROM Sales.CategorySales;
```



	Category	Cust	TotalQty
1	Condiments	NULL	114
2	Confections	NULL	372
3	Beverages	NULL	513
4	NULL	3	154
5	NULL	1	80
6	NULL	4	241
7	NULL	5	512
8	NULL	2	12
9	NULL	NULL	999

Contoh menggunakan grouping sets

```
SELECT Category, Cust, SUM(Qty) AS TotalQty
FROM Sales.CategorySales
GROUP BY
GROUPING SETS((Category),(Cust),())
ORDER BY Category, Cust;
```



	Category	Cust	TotalQty
1	NULL	NULL	999
2	NULL	1	80
3	NULL	2	12
4	NULL	3	154
5	NULL	4	241
6	NULL	5	512
7	Beverages	NULL	513
8	Condiments	NULL	114
9	Confections	NULL	372

ROLLUP VS CUBE

- ROLLUP dan CUBE merupakan sub-clausa dari GROUP BY
- Sama seperti GROUPING SETS, ROLLUP dan CUBE mendukung multiple grouping
- ROLLUP, akan menampilkan kombinasi dari set pengelompokkan dengan membentuk suatu hierarki
- CUBE, akan menampilkan semua kombinasi yang mungkin dari set pengelompokkan

CUBE

```
(d1, d2, d3)
(d1, d2)
(d2, d3)
(d1, d3)
(d1)
(d2)
(d3)
()
```

ROLL UP

```
(d1, d2, d3)
(d1, d2)
(d1)
()
```

ROLLUP, akan menampilkan kombinasi dari set pengelompokkan dengan membentuk suatu hierarki

CUBE, akan menampilkan semua kombinasi yang mungkin dari set pengelompokkan

cube

```
(d1, d2, d3)
(d1, d2)
(d2, d3)
(d1, d3)
(d1)
(d2)
(d3)
()
```

- k1, k2, k3
- K1,k2
- K1,k3
- K2,k3
- K1
- K2
- K3
- ()

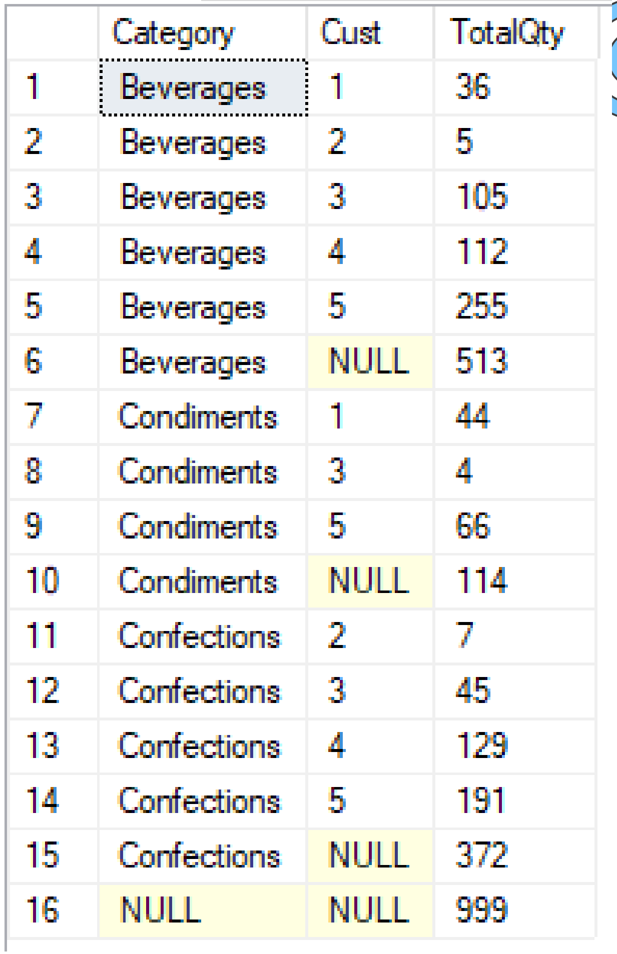
- K1,k2
- K1
- K2
- ()

- K1
- ()

ROLLUP

```
SELECT Category, Cust, SUM(Qty) AS TotalQty
FROM Sales.CategorySales
GROUP BY ROLLUP(Category,Cust);
```

- Menampilkan total qty untuk kombinasi pasangan category – cust
- Baris 6, 10, 15 adalah total qty untuk setiap category
- Baris 16 adalah total qty keseluruhan

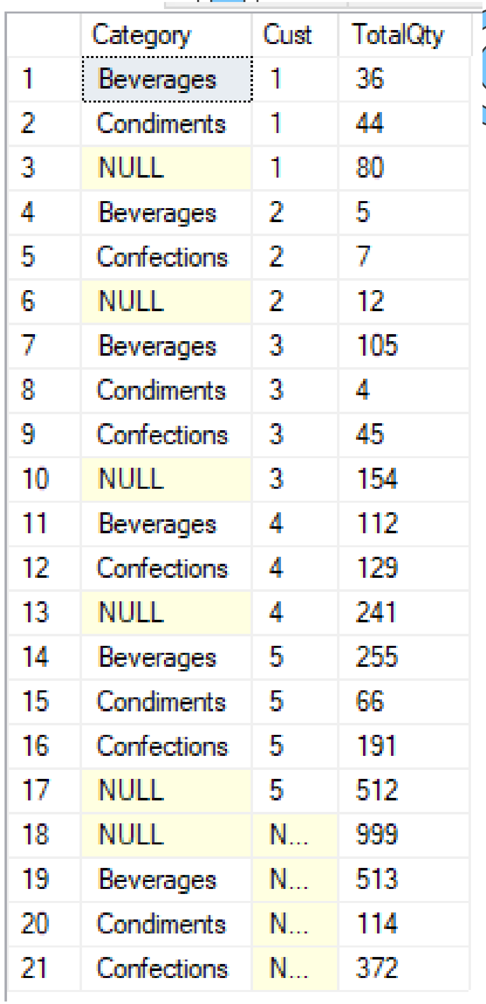


	Category	Cust	TotalQty
1	Beverages	1	36
2	Beverages	2	5
3	Beverages	3	105
4	Beverages	4	112
5	Beverages	5	255
6	Beverages	NULL	513
7	Condiments	1	44
8	Condiments	3	4
9	Condiments	5	66
10	Condiments	NULL	114
11	Confections	2	7
12	Confections	3	45
13	Confections	4	129
14	Confections	5	191
15	Confections	NULL	372
16	NULL	NULL	999

CUBE

```
SELECT Category, Cust, SUM(Qty) AS TotalQty
FROM Sales.CategorySales
GROUP BY CUBE(Category,Cust);
```

- Menampilkan total qty untuk seluruh kombinasi yang mungkin dari pasangan category – cust
- Baris 3, 6, 10,13,17 adalah total qty untuk setiap cust
- Baris 18 adalah total qty keseluruhan
- Baris 19-21 adalah total qty untuk setiap category



	Category	Cust	TotalQty
1	Beverages	1	36
2	Condiments	1	44
3	NULL	1	80
4	Beverages	2	5
5	Confections	2	7
6	NULL	2	12
7	Beverages	3	105
8	Condiments	3	4
9	Confections	3	45
10	NULL	3	154
11	Beverages	4	112
12	Confections	4	129
13	NULL	4	241
14	Beverages	5	255
15	Condiments	5	66
16	Confections	5	191
17	NULL	5	512
18	NULL	N...	999
19	Beverages	N...	513
20	Condiments	N...	114
21	Confections	N...	372

Thanks!

Do you have any questions?

Team Teaching Matakuliah Basis Data Lanjut
JTI POLINEMA