

Chương 2: Truy vấn nâng cao và View

- Các hàm gộp nhóm.
- Thao tác với View

Các hàm gộp nhóm

- Giới thiệu về một số hàm gộp nhóm.
- Mệnh đề GROUP BY.

Giới thiệu một số hàm gộp nhóm

- Một số hàm thường dùng: AVG, COUNT, MAX, MIN, SUM,...

- Cách sử dụng:

```
SELECT {AVG | COUNT | MAX | MIN | SUM } (expression | *) }  
[,...n]
```

```
FROM table_list
```

```
[ WHERE search_conditions ]
```

Mệnh đề GROUP BY

- Cú pháp.
- Một số chú ý.
- Sử dụng Having với Group by.
- Group by với ROLLUP và CUBE.
- Mệnh đề COMPUTE và COMPUTE BY

Cú pháp

```
SELECT select_list FROM table_source  
[ WHERE search_condition ]  
[ GROUP BY [ALL] group_by_expression [...n]]  
[HAVING search_condition ]
```

Ví dụ

```
USE northwind
SELECT productid, orderid, quantity
FROM orderhist
```

<i>productid</i>	<i>orderid</i>	<i>quantity</i>
1	1	5
1	1	10
2	1	10
2	2	25
3	1	15
3	2	30

Only rows that
satisfy the WHERE
clause are grouped

```
USE northwind
SELECT productid, SUM (quantity)
FROM orderhist
GROUP BY productid
```

<i>productid</i>	<i>total_quantity</i>
1	15
2	35
3	45

<i>productid</i>	<i>total_quantity</i>
2	35

```
USE northwind
SELECT productid, SUM (quantity)
FROM orderhist
WHERE productid = 2
GROUP BY productid
```

Một số chú ý

- Các cột có trong SELECT phải được chỉ ra trong GROUP BY.
- Chỉ lấy giá trị thỏa mãn điều kiện ở mệnh đề WHERE
- Giá trị NULL được coi như 1 nhóm.
- Sử dụng ALL sẽ hiện thị cả những nhóm **không** thỏa mãn mệnh đề Where.

Sử dụng Having trong Group by(1)

- Thiết đặt điều kiện trên Group by

```
USE northwind
SELECT productid, orderid, quantity
FROM orderhist
```

<i>productid</i>	<i>orderid</i>	<i>quantity</i>
1	1	5
1	1	10
2	1	10
2	2	25
3	1	15
3	2	30

```
USE northwind
SELECT productid, SUM (quantity)
AS total_quantity
FROM orderhist
GROUP BY productid
HAVING SUM (quantity) >= 30
```

<i>productid</i>	<i>total_quantity</i>
2	35
3	45

Sử dụng Having trong Group by(2)

- Chú ý:
 - Chỉ được sử dụng kèm với Group by.

- VD:

```
USE Northwind
```

```
SELECT ProductID, SUM(Quantity) AS Total_Quantity
```

```
FROM OrderHist
```

```
GROUP BY ProductID
```

```
HAVING SUM(Quantity) >= 30
```

Group by với ROLLUP và CUBE

- Cú pháp
- Toán tử ROLLUP
- Toán tử CUBE
- Sự khác nhau giữa ROLLUP và CUBE

Cú pháp

SELECT *select_list* FROM *table_source*

[WHERE *search_condition*]

[[GROUP BY [ALL] *group_by_expression* [, ...*n*]] [WITH { CUBE |
ROLLUP }]]

[HAVING *search_condition*]

Toán tử ROLLUP

```
USE northwind
SELECT productid, orderhist, SUM (quantity) AS total_quantity
FROM orderhist
GROUP BY productid, orderid
WITH ROLLUP
ORDER BY productid, orderid
```

<i>productid</i>	<i>orderid</i>	<i>total_quantity</i>
NULL	NULL	95
1	NULL	15
1	1	5
1	2	10
2	NULL	35
2	1	10
2	2	25
3	NULL	45
3	1	15
3	2	30

Description
Grand total
Summarizes only rows for productid 1
Detail value for product 1, orderid 1
Detail value for product 1, orderid 2
Summarizes only rows for productid 2
Detail value for product 2, orderid 1
Detail value for product 2, orderid 2
Summarizes only rows for productid 3
Detail value for product 3, orderid 1
Detail value for product 3, orderid 2

Toán tử CUBE

```
USE northwind
SELECT productid, orderhist, SUM (quantity) AS total_quantity
FROM orderhist
GROUP BY productid, orderid
WITH CUBE
ORDER BY productid, orderid
```

The CUBE operator produces two more summary values than the ROLLUP operator

<i>productid</i>	<i>orderid</i>	<i>total quantity</i>
NULL	NULL	95
NULL	1	30
NULL	2	65
1	NULL	15
1	1	5
1	2	10
2	NULL	35
2	1	10
2	2	25
3	NULL	45
3	1	15
3	2	30

Description
Grand total
Summarizes only rows for orderid 1
Summarizes only rows for orderid 2
Summarizes only rows for productid 1
Detail value for product 1, orderid 1
Detail value for product 1, orderid 2
Summarizes only rows for productid 2
Detail value for product 2, orderid 1
Detail value for product 2, orderid 2
Summarizes only rows for productid 3
Detail value for product 3, orderid 1
Detail value for product 3, orderid 2

Sự khác nhau

- CUBE đưa ra kết quả là sự kết hợp tất cả các giá trị các cột trên Select.
- ROLLUP đưa ra kết quả là sự phân cấp trên các cột trong Select.

Mệnh đề Compute và Compute by (1)

- Cú pháp:

SELECT *select_list*

FROM *table_source*

[WHERE *search_condition*]

[COMPUTE { { AVG | COUNT | MAX | MIN | SUM | STDEV |
STDEVP | VAR | VARP } (*expression*) } [,...*n*]

[BY *expression* [,...*n*]]]

Mệnh đề Compute và Compute by (2)

COMPUTE

```
USE northwind
SELECT productid, orderid, quantity
FROM orderhist
ORDER BY productid, orderid
COMPUTE SUM (quantity)
```

<i>productid</i>	<i>orderid</i>	<i>quantity</i>
1	1	2
1	1	10
2	1	10
2	2	25
3	1	15
3	2	30
sum		95

COMPUTE BY

```
USE northwind
SELECT productid, orderid, quantity
FROM orderhist
ORDER BY productid, orderid
COMPUTE SUM (quantity) BY productid
```

<i>productid</i>	<i>orderid</i>	<i>quantity</i>
1	1	2
1	1	10
sum		15
2	1	10
2	2	25
sum		35
3	1	15
3	2	30
sum		45
sum		95

Thao tác với View

- Giới thiệu về view.
- Tạo, sửa, xóa view.
- Một số hạn chế của view.
- Truy cập dữ liệu với view.
- Một số ví dụ.

Giới thiệu về View

- Là cách tạo trước các câu SELECT.
- Một số dạng View thường gặp:
 - Tập con của các dòng, cột của bảng.
 - Hợp nhất các bảng.
 - Liên kết các bảng.
 - Thống kê tóm tắt bảng.
 - Tập con của View, kết hợp view và bảng,.

Tạo, sửa, xóa View (1)

- **Tạo view:**

CREATE VIEW [*schema*.]view_name

[(*column*[, ...*n*])]

[WITH ENCRYPTION]

AS

select_statement

[WITH CHECK OPTION]

- Trong đó:

- With Encryption: mã hóa nội dung View
- With Check Option: kiểm tra giá trị nhập. When a row is modified through a view, the WITH CHECK OPTION makes sure the data remains visible through the view after the modification is committed. CHECK OPTION cannot be specified if TOP is used anywhere in *select_statement*.

Tạo, sửa, xóa View (2)

- **Sửa view:**

```
ALTER VIEW [ < database_name > . ] [ < schema > . ] view_name [ ( column [ ,...n ] ) ]  
[ WITH ENCRYPTION ]  
AS  
    select_statement  
[ WITH CHECK OPTION ]
```

- **Xóa view:**

```
DROP VIEW { view } [ ,...n ]
```

Một số hạn chế của View

- Trong view không có mệnh đề:
 - COMPUTE, COMPUTE BY, INTO.
 - ORDER BY, unless there is also a TOP clause in the select list of the SELECT statement
 - Không thể tham chiếu tới bảng tạm.

Ví dụ 1

```
USE Northwind
```

```
GO
```

```
CREATE VIEW CustomerView
```

```
AS
```

```
    SELECT CustomerID, CompanyName FROM  
Customers  where CompanyName Like 'B%'
```

```
GO
```

```
SELECT * FROM CustomerView
```

Ví dụ 2

GO

INSERT CustomerView VALUES ('TEST1', 'Test Company')

GO

UPDATE CustomerView SET CustomerID = 'TEST2' WHERE CustomerID = 'TEST1'

GO

DELETE CustomerView WHERE CustomerID = 'TEST2'

Create view ViewRollup

as

Select salesOrderID, ProductID, sum(OrderQty) as SUMQTY
from sales.salesorderdetail group by
salesOrderID, ProductID with rollup