

PV226/MSSQL

Microsoft SQL Server 2012

Kapitola 6: Tvorba pokročilých T-SQL dotazů

Bc. David Gešvindr
MCT | MSP | MCTS | MCITP | MCPD

Obsah

- ➞ 1. Načítání dat pomocí příkazu SELECT
- ➞ 2. Modifikace dat

1. Načítání dat pomocí příkazu SELECT

Logické zpracování dotazu

- ➞ Popisuje chování dotazu
- ➞ Jeho fáze, jejich vstupy a výstupy nutně nesouvisí s vlastním fyzickým zpracováním dotazu

Zpracování dotazu

```
7 SELECT 8 DISTINCT 9 TOP <select list>  
1 FROM <left table>  
    3 <join type> JOIN <right table>  
        2 ON <join condition>  
4 WHERE <where condition>  
5 GROUP BY <group by list>  
6 HAVING <having condition>  
10 ORDER BY <order by list>
```

1 FROM, 2 ON, 3 JOIN

Customers	
CustomerID	Name
1	David
2	Jirka
3	Martin

X
CROSS
JOIN

Orders	
OrderID	CustomerID
1	1
2	2
3	1
4	4

Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
1	David	2	2
1	David	3	1
...			
3	Martin	4	4

VT1

2 ON

Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
1	David	2	2
...			
3	Martin	4	4

VT1

ON CONDITION **Customers.CustomerID = Orders.CustomerID**

Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
1	David	3	1
2	Jirka	2	2

VT2

= INNER JOIN



3 LEFT OUTER JOIN

Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
1	David	3	1
2	Jirka	2	2

VT2



Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
1	David	3	1
2	Jirka	2	2
3	Martin	<i>NULL</i>	<i>NULL</i>

VT3

3 RIGHT OUTER JOIN

Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
1	David	3	1
2	Jirka	2	2

VT2



Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
1	David	3	1
2	Jirka	2	2
<i>NULL</i>	<i>NULL</i>	4	4

VT3

3 FULL OUTER JOIN

Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
1	David	3	1
2	Jirka	2	2

VT2



Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
1	David	3	1
2	Jirka	2	2
3	Martin	<i>NULL</i>	<i>NULL</i>
<i>NULL</i>	<i>NULL</i>	4	4

VT3

Zpracování dotazu

```
7 SELECT 8 DISTINCT 9 TOP <select list>  
1 FROM <left table>  
    3 <join type> JOIN <right table>  
        2 ON <join condition>  
4 WHERE <where condition>  
5 GROUP BY <group by list>  
6 HAVING <having condition>  
10 ORDER BY <order by list>
```

4 WHERE

Three-valued Logic

True Unknown False

- ⊕ V podmínice ON, WHERE a HAVING:
 - ⊕ *NULL* chápáno jako FALSE
- ⊕ V check-constraint:
 - ⊕ *NULL* chápáno jako TRUE
- ⊕ V GROUP BY, ORDER BY a UNIQUE omezeních jsou si *NULL* rovny



5 GROUP BY



Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
1	David	3	1
2	Jirka	2	2
3	Martin	NULL	NULL

VT4

GROUP BY Customers.CustomerID

GROUPS	RAW		
Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
	David	3	1
2	Jirka	2	2
3	Martin	NULL	NULL

VT5

6 HAVING

GROUPS	RAW		
Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
	David	3	1
2	Jirka	2	2
3	Martin	<i>NULL</i>	<i>NULL</i>

VT5

HAVING COUNT(Orders.OrderID) > 0

GROUPS	RAW		
Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	1	1
	David	3	1
2	Jirka	2	2

VT6



7 SELECT

- ➞ Vyhodnocení výrazů v části SELECT
- ➞ Vytvořené aliasy sloupců použitelné až v dalších krocích

VT6



VT7

„All-at-once“

```
SELECT c1 + 1 AS e1, e1 + 1 AS e2
```

```
UPDATE Customers  
SET FirstName = LastName,  
    LastName = FirstName
```

```
UPDATE dbo.T1 SET c1 = c1 + (SELECT MAX(c1)  
FROM dbo.T1)
```

8 DISTINCT

- ➔ Odstranění duplicitních řádků
- ➔ Redundantní s **GROUP BY**



9 TOP

- ➞ Vrání **n** prvních výsledků z dodaného tabulky
- ➞ Pokud je definováno **ORDER BY**, ovlivní pořadí řádků
- ➞ **WITH TIES**
- ➞ Deterministický x nedeterministický



9 ORDER BY

- ➞ Seřazení množiny výsledků
- ➞ Dle ANSI SQL:1999
 - ➞ při použití **DISTINCT** jsme omezeni na sloupce vracené příkazem **SELECT** (VT7), jinak použitelná i VT6
- ➞ Vrací **kurzor**
 - ➞ Nelze použít jako table-expression
 - ➞ View, Inline table-valued function, subquery, derived table nebo common table expression



Zpracování dotazu

```
7 SELECT 8 DISTINCT 9 TOP <select list>  
1 FROM <left table>  
    3 <join type> JOIN <right table>  
        2 ON <join condition>  
4 WHERE <where condition>  
5 GROUP BY <group by list>  
6 HAVING <having condition>  
10 ORDER BY <order by list>
```

Když norma nestačila

The background features a vibrant blue sky with wispy white clouds. A bright light source, likely the sun, is positioned on the left side, creating a lens flare effect. Overlaid on the sky are several flowing, wavy lines in various shades of blue and white, which create a sense of movement and depth. The overall aesthetic is clean, modern, and optimistic.

CROSS APPLY

Customers	
CustomerID	Name
1	David
2	Jirka
3	Martin

```
SELECT TOP 1 *  
WHERE Orders.CustomerID =  
Customers.CustomerID  
ORDER BY Orders.OrderID DESC
```

Orders	
OrderID	CustomerID
3	1



Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	3	1
2	Jirka	2	2

OUTER APPLY

Customers	
CustomerID	Name
1	David
2	Jirka
3	Martin

```
SELECT TOP 1 *  
WHERE Orders.CustomerID =  
Customers.CustomerID  
ORDER BY Orders.OrderID DESC
```

Orders	
OrderID	CustomerID
3	1



Customers. CustomerID	Customers. Name	Orders. OrderID	Orders. CustomerID
1	David	3	1
2	Jirka	2	2
3	Martin	<i>NULL</i>	<i>NULL</i>

PIVOT

CustomerID	City	Category
1	Brno	Small
2	Brno	Medium
3	Praha	<i>Large</i>
4	Praha	<i>Large</i>
5	Ostrava	<i>Medium</i>



City	Small	Medium	Large
Brno	1	1	0
Praha	0	0	2
Ostrava	0	1	0

UNPIVOT

City	Small	Medium	Large
Brno	1	1	0
Praha	0	0	2
Ostrava	0	1	0



City	Category	CustomersCount
Brno	Small	1
Brno	Medium	1
Praha	<i>Large</i>	2
Ostrava	<i>Medium</i>	1

2. Modifikace dat

