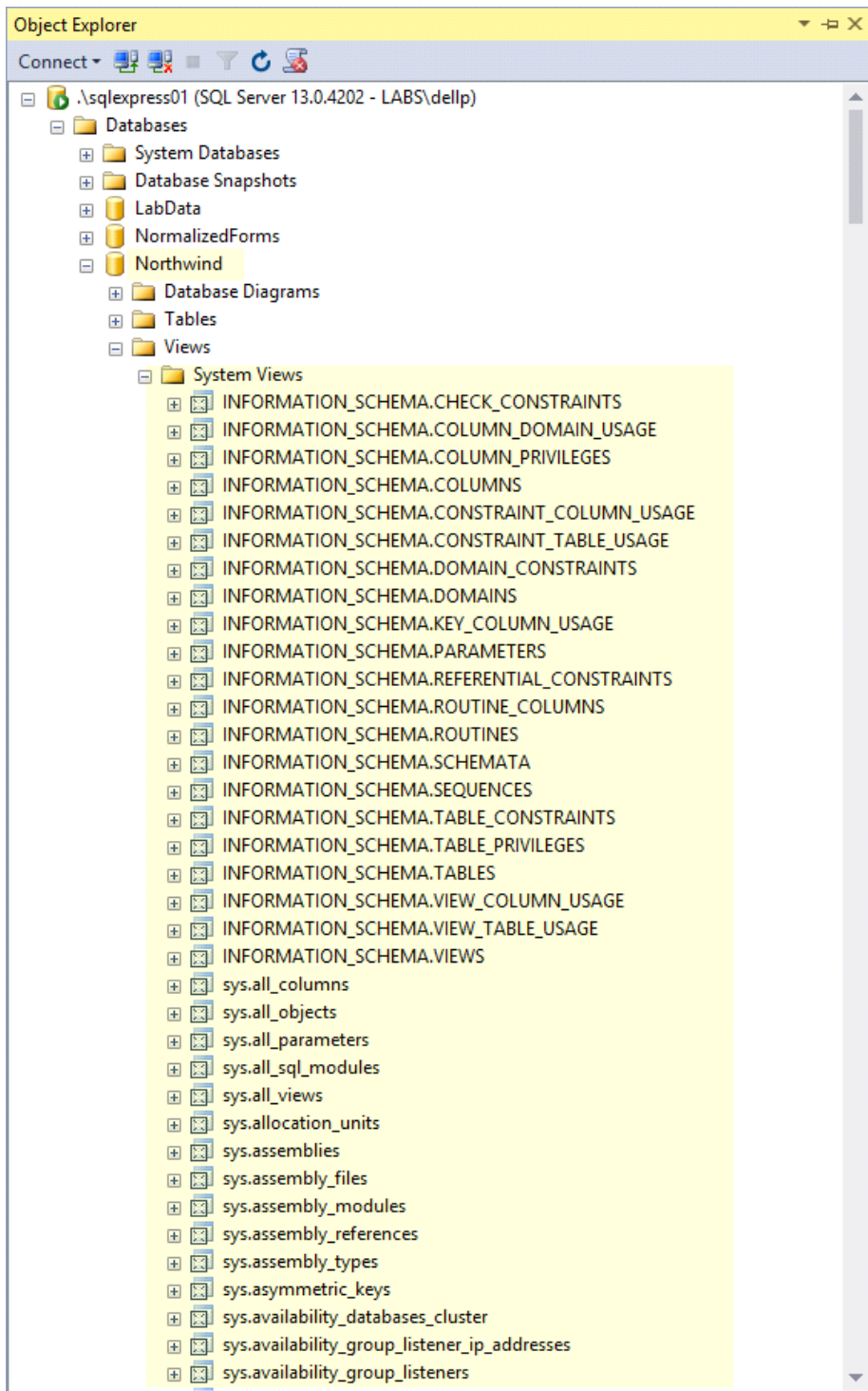


System View List

Friday, May 26, 2017 12:57 PM

☐ SelectSystemTablesViews.sql



```
Select * From Sys.Tables;
Select * From Sys.Views;
```

127 %

Results

Messages

	name	object_id	principal_id	schema_id	parent_object_id	type	type_desc	create_date	modify_date	is_m
1	sysdiagrams	98099390	NULL	1	0	U	USER_TABLE	2017-05-23 11:39:12.373	2017-05-23 11:39:12.373	0
2	CustomerDemographics	565577053	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:01.640	2017-04-11 17:52:18.860	0
3	Region	597577167	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:01.663	2017-04-11 17:52:19.010	0
4	Employees	629577281	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:01.707	2017-04-11 17:52:18.970	0
5	Categories	661577395	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:01.827	2017-04-11 17:52:18.990	0
6	Customers	693577509	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:01.913	2017-04-11 17:52:18.957	0
7	Shippers	725577623	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:02.523	2017-04-11 17:52:18.980	0
8	Suppliers	757577737	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:02.550	2017-04-11 17:52:19.000	0
9	Orders	789577851	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:02.757	2017-04-11 17:52:18.980	0
10	Products	821577965	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:07.400	2017-04-11 17:52:19.000	0
11	Order Details	853578079	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:07.803	2017-04-11 17:52:18.940	0
12	CustomerCustomerDemo	885578193	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:17.987	2017-04-11 17:52:18.873	0
13	Territories	917578307	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:17.997	2017-04-11 17:52:19.010	0
14	EmployeeTerritories	949578421	NULL	1	0	U	USER_TABLE	2017-04-11 17:52:18.270	2017-04-11 17:52:18.910	0

	name	object_id	principal_id	schema_id	parent_object_id	type	type_desc	create_date	modify_date	is
1	Orders Qty	981578535	NULL	1	0	V	VIEW	2017-04-11 17:52:18.513	2017-04-11 17:52:18.513	0
2	Quarterly Orders	997578592	NULL	1	0	V	VIEW	2017-04-11 17:52:18.523	2017-04-11 17:52:18.523	0
3	Invoices	1013578649	NULL	1	0	V	VIEW	2017-04-11 17:52:18.530	2017-04-11 17:52:18.530	0
4	Product Sales for 1997	1029578706	NULL	1	0	V	VIEW	2017-04-11 17:52:18.540	2017-04-11 17:52:18.540	0
5	Current Product List	1125579048	NULL	1	0	V	VIEW	2017-04-11 17:52:18.587	2017-04-11 17:52:18.587	0
6	Order Details Extended	1141579105	NULL	1	0	V	VIEW	2017-04-11 17:52:18.593	2017-04-11 17:52:18.593	0
7	Products Above Average Price	1157579162	NULL	1	0	V	VIEW	2017-04-11 17:52:18.600	2017-04-11 17:52:18.600	0
8	Products by Category	1173579219	NULL	1	0	V	VIEW	2017-04-11 17:52:18.607	2017-04-11 17:52:18.607	0
9	Alphabetical list of products	1189579276	NULL	1	0	V	VIEW	2017-04-11 17:52:18.613	2017-04-11 17:52:18.613	0
10	Order Subtotals	1205579333	NULL	1	0	V	VIEW	2017-04-11 17:52:18.620	2017-04-11 17:52:18.620	0
11	Customer and Suppliers by City	1221579390	NULL	1	0	V	VIEW	2017-04-11 17:52:18.633	2017-04-11 17:52:18.633	0
12	Sales Totals by Amount	1237579447	NULL	1	0	V	VIEW	2017-04-11 17:52:18.640	2017-04-11 17:52:18.640	0
13	Sales by Category	1253579504	NULL	1	0	V	VIEW	2017-04-11 17:52:18.650	2017-04-11 17:52:18.650	0
14	Summary of Sales by Quarter	1301579675	NULL	1	0	V	VIEW	2017-04-11 17:52:18.673	2017-04-11 17:52:18.673	0
15	Summary of Sales by Year	1317579732	NULL	1	0	V	VIEW	2017-04-11 17:52:18.680	2017-04-11 17:52:18.680	0
16	Category Sales for 1997	1333579789	NULL	1	0	V	VIEW	2017-04-11 17:52:18.687	2017-04-11 17:52:18.687	0

DMV's

Friday, May 26, 2017 12:15 PM

DynamicManagementViews.sql

The screenshot shows a SQL Server Enterprise Manager window with two tabs: 'CreateSystemViewsT...s (LABS\delip (53))' and 'DynamicManagemen... (LABS\delip (54))'. The active tab contains the following T-SQL code:

```
USE master;
GO

--Get All Cached Plans
-----
SELECT *
FROM sys.dm_exec_cached_plans cp
CROSS APPLY sys.dm_exec_query_plan(cp.plan_handle);
GO

--Get plans and average CPU time for the top five queries.
-----
SELECT TOP 5 total_worker_time/execution_count AS [Avg CPU Time],
Plan_handle, query_plan
FROM sys.dm_exec_query_stats AS qs
CROSS APPLY sys.dm_exec_query_plan(qs.plan_handle)
ORDER BY total_worker_time/execution_count DESC;
GO
```

The 'Results' tab displays a table with 10 rows and 9 columns. The first row is highlighted. The columns are: bucketid, refcounts, usecounts, size_in_bytes, memory_object_address, cacheobjtype, objtype, and plan_handle.

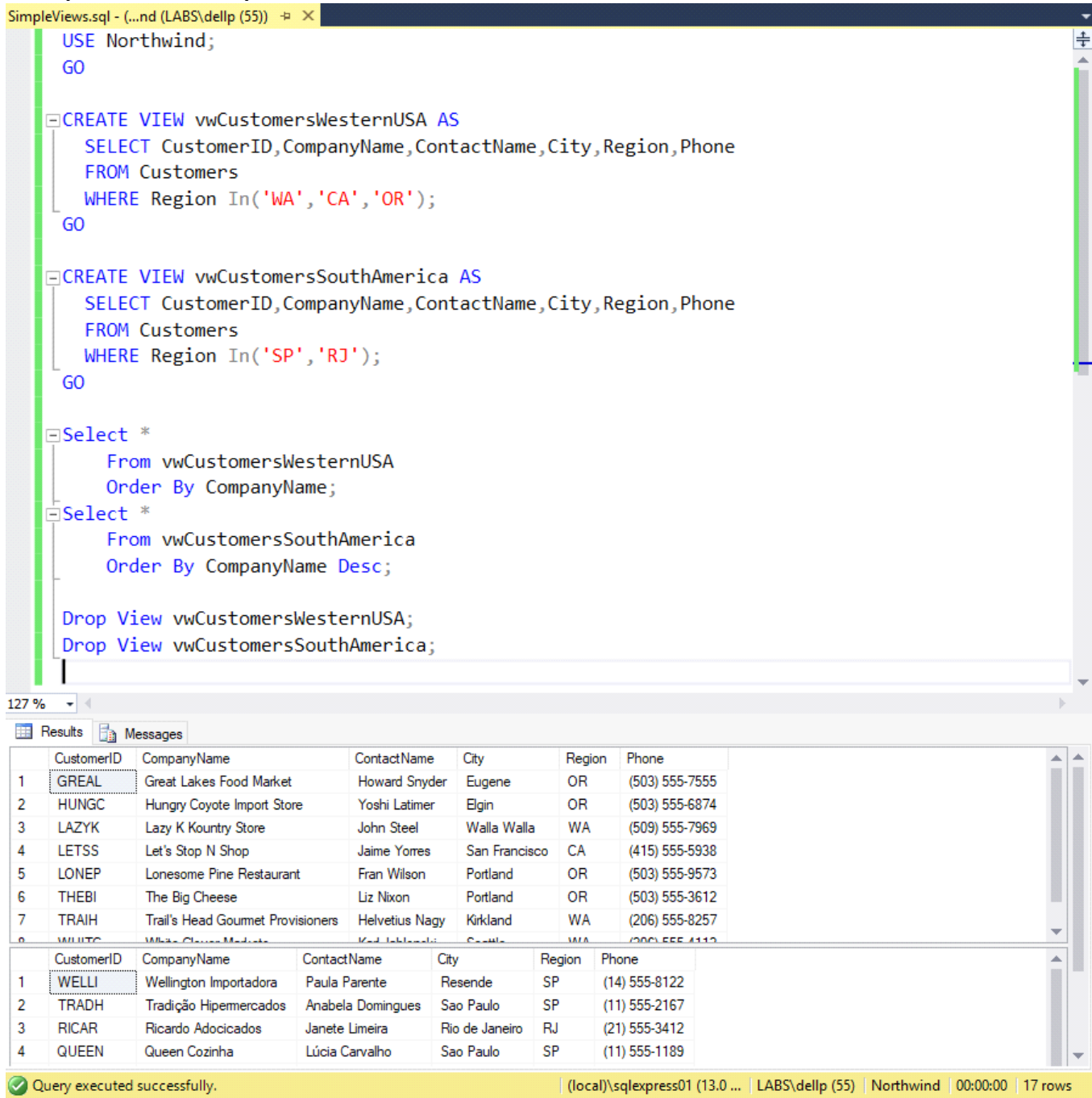
bucketid	refcounts	usecounts	size_in_bytes	memory_object_address	cacheobjtype	objtype	plan_handle
9640	2	1	65536	0x000001FCA3802060	Compiled Plan	Proc	0x0500060035F9F31A700D7AA9FC01000001000000000000...
27283	2	1	40960	0x000001FCA87D4060	Compiled Plan	Proc	0x0500060029B1F43310067AA9FC01000001000000000000...
17020	2	1	548864	0x000001FCA9782060	Compiled Plan	Proc	0x050006004413B31B307E2AA5FC01000001000000000000...
7133	2	1	57344	0x000001FCA90D2060	Compiled Plan	Proc	0x05000600A741DC1CD0762AA5FC01000001000000000000...
8344	2	1	1622016	0x000001FCA0848060	Compiled Plan	Proc	0x05000600CC264B66706F2AA5FC01000001000000000000...
15865	2	1	147456	0x000001FCA6A0A060	Compiled Plan	Proc	0x05000600764D926010682AA5FC01000001000000000000...
7039	2	1	73728	0x000001FCA7868060	Compiled Plan	Proc	0x05000600958E8176407F3CB4FC01000001000000000000...
37123	2	1	475136	0x000001FCA6092060	Compiled Plan	Proc	0x050006003D299E5FE0783CB4FC01000001000000000000...
2166	2	1	278528	0x000001FCA7FFA060	Compiled Plan	Proc	0x05000600142CDA7380713CB4FC01000001000000000000...
22474	2	1	81820	0x000001FCA50AA060	Compiled Plan	Proc	0x05000600B07E6770627CB4FC01000001000000000000...

The 'Messages' tab shows a status bar at the bottom: 'Query executed successfully. (local)\sqlexpress01 (13.0 ... | LABS\delip (54) | master | 00:00:00 | 5 rows

Simple Views

Thursday, May 25, 2017 3:17 PM

SimpleViews.sql



The screenshot displays a SQL script named 'SimpleViews.sql' in a text editor window. The script contains the following SQL commands:

```
USE Northwind;
GO

CREATE VIEW vwCustomersWesternUSA AS
SELECT CustomerID, CompanyName, ContactName, City, Region, Phone
FROM Customers
WHERE Region In('WA', 'CA', 'OR');
GO

CREATE VIEW vwCustomersSouthAmerica AS
SELECT CustomerID, CompanyName, ContactName, City, Region, Phone
FROM Customers
WHERE Region In('SP', 'RJ');
GO

Select *
From vwCustomersWesternUSA
Order By CompanyName;

Select *
From vwCustomersSouthAmerica
Order By CompanyName Desc;

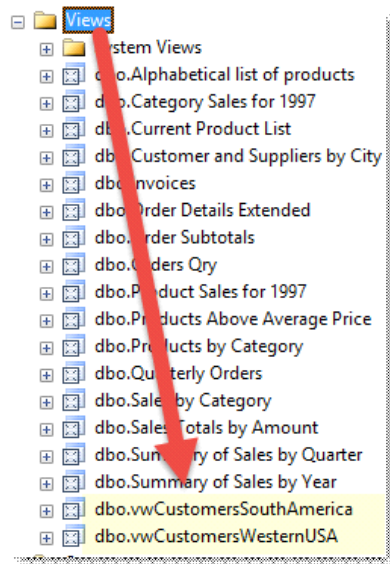
Drop View vwCustomersWesternUSA;
Drop View vwCustomersSouthAmerica;
```

Below the script, the 'Results' pane shows the output of the queries. It contains two tables. The first table, 'vwCustomersWesternUSA', lists customers from the Western US (WA, CA, OR) ordered by CompanyName. The second table, 'vwCustomersSouthAmerica', lists customers from South America (SP, RJ) ordered by CompanyName Desc.

CustomerID	CompanyName	ContactName	City	Region	Phone	
1	GREAL	Great Lakes Food Market	Howard Snyder	Eugene	OR	(503) 555-7555
2	HUNGC	Hungry Coyote Import Store	Yoshi Latimer	Elgin	OR	(503) 555-6874
3	LAZYK	Lazy K Kountry Store	John Steel	Walla Walla	WA	(509) 555-7969
4	LETSS	Let's Stop N Shop	Jaime Yorres	San Francisco	CA	(415) 555-5938
5	LONEP	Lonesome Pine Restaurant	Fran Wilson	Portland	OR	(503) 555-9573
6	THEBI	The Big Cheese	Liz Nixon	Portland	OR	(503) 555-3612
7	TRAIH	Trail's Head Gourmet Provisioners	Helvetius Nagy	Kirkland	WA	(206) 555-8257
8	WLUIC	White Clover Markets	Ked Colburn	Seattle	WA	(206) 555-4112

CustomerID	CompanyName	ContactName	City	Region	Phone	
1	WELLI	Wellington Importadora	Paula Parente	Resende	SP	(14) 555-8122
2	TRADH	Tradição Hipermarcados	Anabela Domingues	Sao Paulo	SP	(11) 555-2167
3	RICAR	Ricardo Adocicados	Janete Limeira	Rio de Janeiro	RJ	(21) 555-3412
4	QUEEN	Queen Cozinha	Lúcia Carvalho	Sao Paulo	SP	(11) 555-1189

At the bottom of the screenshot, a status bar indicates that the query was executed successfully, showing the server name '(local)\sqlexpress01 (13.0 ...)', the database 'LABS\dellp (55)', the context 'Northwind', and the execution time '00:00:00' with '17 rows' returned.



Alter View

Friday, May 26, 2017

1:35 PM



ALterViews.sql

AlterView.sql - (loc...nd (LABS\dellp (51))

```

USE Northwind;
GO

CREATE VIEW vwCustomersWesternUSA AS
    SELECT CustomerID,CompanyName,ContactName,City,Region,Phone
    FROM Customers
    WHERE Region In('WA','CA','OR');
GO

Select *
    From vwCustomersWesternUSA
    Order By CompanyName;
GO

ALTER VIEW vwCustomersWesternUSA
AS
    SELECT CompanyName,ContactName,City,Region,Phone
    FROM Customers
    WHERE Region In('WA','CA','OR');
GO

Select *
    From vwCustomersWesternUSA
    Order By CompanyName;

DROP VIEW vwCustomersWesternUSA;

```

127 %

Results Messages

	CustomerID	CompanyName	ContactName	City	Region	Phone
1	GREAL	Great Lakes Food Market	Howard Snyder	Eugene	OR	(503) 555-7555
2	HUNGC	Hungry Coyote Import Store	Yoshi Latimer	Elgin	OR	(503) 555-6874
3	LAZYK	Lazy K Kountry Store	John Steel	Walla Walla	WA	(509) 555-7969
4	LETSS	Let's Stop N Shop	Jaime Yorres	San Francisco	CA	(415) 555-5938
5	LONEP	Lonesome Pine Restaurant	Fran Wilson	Portland	OR	(503) 555-9573
6	THEBI	The Big Cheese	Liz Nixon	Portland	OR	(503) 555-3612

	CompanyName	ContactName	City	Region	Phone
1	Great Lakes Food Market	Howard Snyder	Eugene	OR	(503) 555-7555
2	Hungry Coyote Import Store	Yoshi Latimer	Elgin	OR	(503) 555-6874
3	Lazy K Kountry Store	John Steel	Walla Walla	WA	(509) 555-7969
4	Let's Stop N Shop	Jaime Yorres	San Francisco	CA	(415) 555-5938
5	Lonesome Pine Restaurant	Fran Wilson	Portland	OR	(503) 555-9573

Query executed successfully. (local)\sqlexpress01 (13.0 ... LABS\dellp (51) Northwind 00:00:00 16 rows

Updatable Views

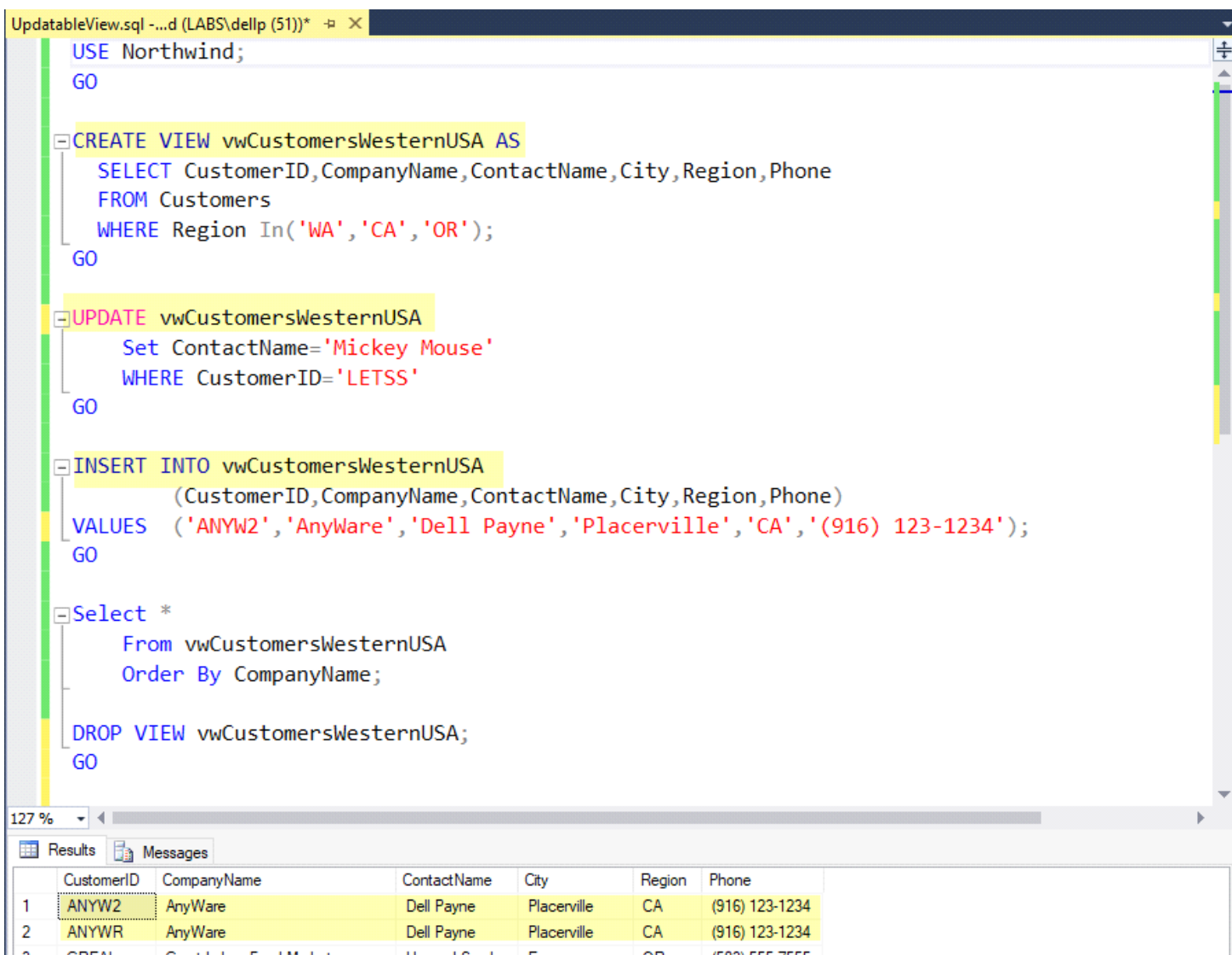
Friday, May 26, 2017 2:09 PM

When can a view be updated?

1. The view is defined based on one and only **one table**.
2. The view must include the **PRIMARY KEY** of the table based upon which the view has been created.
3. The view should **not have** any field made out of **aggregate functions**.
4. The view must **not have any DISTINCT** clause in its definition.
5. The view must **not have any GROUP BY or HAVING** clause in its definition.
6. The view must **not have any SUBQUERIES** in its definitions.
7. If the view you want to update is based upon another view, the later should be updatable.
8. Any of the selected output fields (of the view) must not use constants, strings or value expressions.

Syntax:

```
UPDATE < view_name > SET<column1>=<value1>,<column2>=<value2>,...  
WHERE <condition>;
```



```
USE Northwind;  
GO  
  
CREATE VIEW vwCustomersWesternUSA AS  
SELECT CustomerID,CompanyName,ContactName,City,Region,Phone  
FROM Customers  
WHERE Region In('WA','CA','OR');  
GO  
  
UPDATE vwCustomersWesternUSA  
Set ContactName='Mickey Mouse'  
WHERE CustomerID='LETSS'  
GO  
  
INSERT INTO vwCustomersWesternUSA  
(CustomerID,CompanyName,ContactName,City,Region,Phone)  
VALUES ('ANYW2','AnyWare','Dell Payne','Placerville','CA','(916) 123-1234');  
GO  
  
Select *  
From vwCustomersWesternUSA  
Order By CompanyName;  
  
DROP VIEW vwCustomersWesternUSA;  
GO
```

	CustomerID	CompanyName	ContactName	City	Region	Phone
1	ANYW2	AnyWare	Dell Payne	Placerville	CA	(916) 123-1234
2	ANYWR	AnyWare	Dell Payne	Placerville	CA	(916) 123-1234
3	GRAF1	Great Lakes Food Market	Howard Snyder	Eugene	OR	(503) 555-7555

	CustomerID	CompanyName	ContactName	City	Region	Phone
1	ANYW2	AnyWare	Dell Payne	Placerville	CA	(916) 123-1234
2	ANYWR	AnyWare	Dell Payne	Placerville	CA	(916) 123-1234
3	GREAL	Great Lakes Food Market	Howard Snyder	Eugene	OR	(503) 555-7555
4	HUNGC	Hungry Coyote Import Store	Yoshi Latimer	Elgin	OR	(503) 555-6874
5	LAZYK	Lazy K Kountry Store	John Steel	Walla Walla	WA	(509) 555-7969
6	LETSS	Let's Stop N Shop	Mickey Mouse	San Francisco	CA	(415) 555-5938
7	LONEP	Lonesome Pine Restaurant	Fran Wilson	Portland	OR	(503) 555-9573
8	THEBI	The Big Cheese	Liz Nixon	Portland	OR	(503) 555-3612
9	TRAIH	Trail's Head Gourmet Provisioners	Helvetius Nagy	Kirkland	WA	(206) 555-8257
10	WHITC	White Clover Markets	Karl Jablonski	Seattle	WA	(206) 555-4112

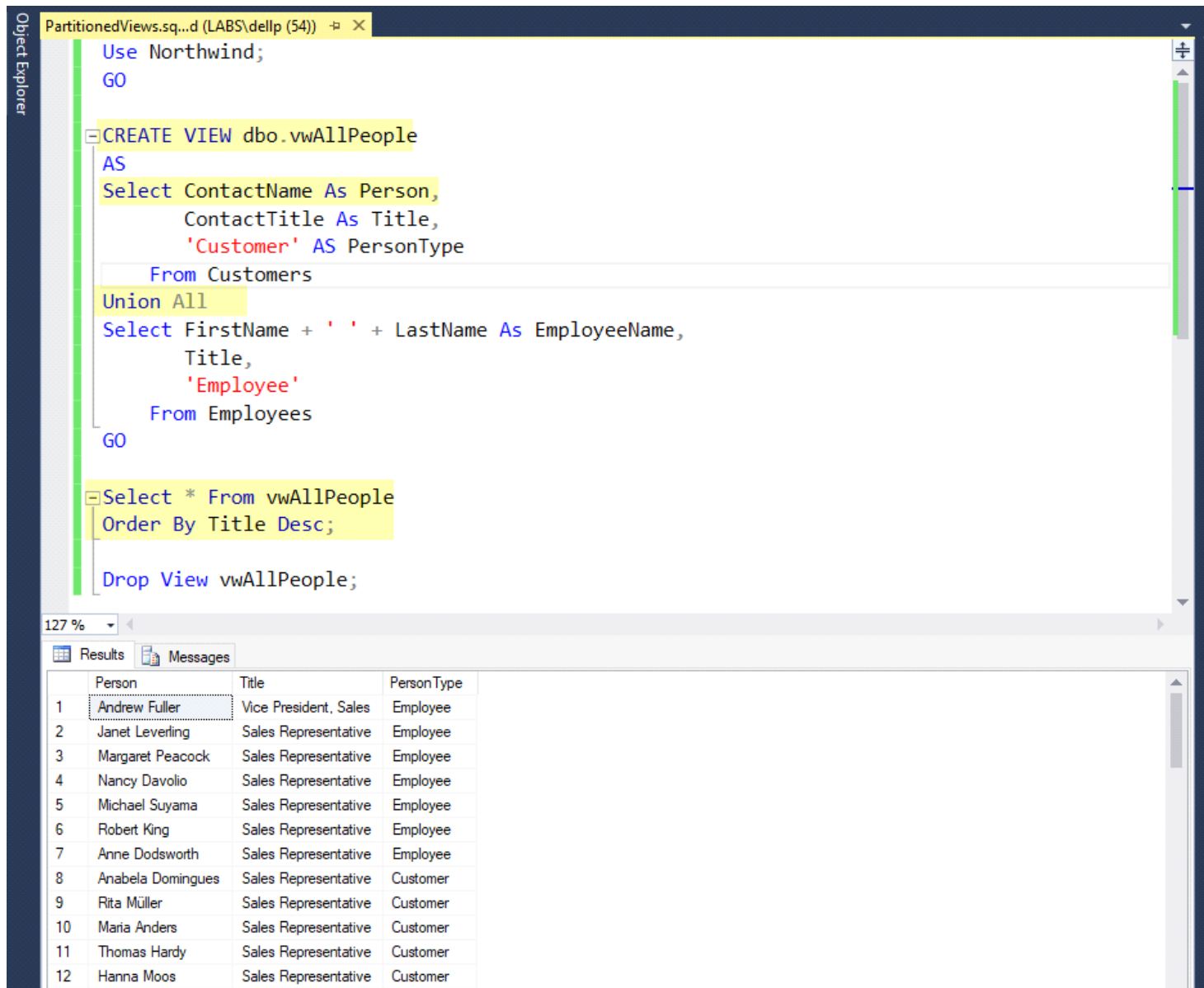


Query executed successfully.

(local)\sqlexpress01 (13.0 ... LABS\dellp (51) Northwind 00:00:00 10 rows

Partitioned View

Thursday, May 25, 2017 3:20 PM



The screenshot shows the SQL Server Enterprise Manager interface. The top pane displays a SQL script in the 'Script' tab, and the bottom pane shows the 'Results' tab with a table of data.

SQL Script:

```
Use Northwind;
GO

CREATE VIEW dbo.vwAllPeople
AS
Select ContactName As Person,
       ContactTitle As Title,
       'Customer' AS PersonType
  From Customers
Union All
Select FirstName + ' ' + LastName As EmployeeName,
       Title,
       'Employee'
  From Employees
GO

Select * From vwAllPeople
Order By Title Desc;

Drop View vwAllPeople;
```

Results Table:

	Person	Title	PersonType
1	Andrew Fuller	Vice President, Sales	Employee
2	Janet Leverling	Sales Representative	Employee
3	Margaret Peacock	Sales Representative	Employee
4	Nancy Davolio	Sales Representative	Employee
5	Michael Suyama	Sales Representative	Employee
6	Robert King	Sales Representative	Employee
7	Anne Dodsworth	Sales Representative	Employee
8	Anabela Domingues	Sales Representative	Customer
9	Rita Müller	Sales Representative	Customer
10	Maria Anders	Sales Representative	Customer
11	Thomas Hardy	Sales Representative	Customer
12	Hanna Moos	Sales Representative	Customer

IF There is TIME...

```
1 Use Northwind;
2 Go
3
4 --Returns Rows that are NOT Common between each other
5 Select Country, City From Customers
6 Except
7 Select Country, City From Suppliers
8 Order By Country, City;
9 Go
10
11 --Returns Rows that ARE Common between each other
12 Select Country, City From Customers
13 Intersect
14 Select Country, City From Suppliers
15 Order By Country, City;
16 Go
```

Indexed Views

Thursday, May 25, 2017 3:20 PM

IF THERE IS TIME

IndexedViews

The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows a SQL script in the 'Script' tab. The script creates a view named 'vwCustomerOrders' with schema binding, selects data from 'Customers' and 'Orders', and then creates a unique clustered index on the view. The bottom pane shows the 'Results' tab with a table of 14 rows. A red arrow points to the 'Execution plan' tab, which is currently inactive. The status bar at the bottom indicates the query was executed successfully, returning 830 rows.

```
USE Northwind
GO

CREATE VIEW vwCustomerOrders
WITH SCHEMABINDING
AS
SELECT Customers.CustomerID, Customers.CompanyName, Orders.OrderID, Orders.OrderDate
FROM   dbo.Customers INNER JOIN
       dbo.Orders ON dbo.Customers.CustomerID = dbo.Orders.CustomerID
GO

--Materialize the Make Virtual Table persist to Disk
-- Must have Standard/Developer/Enterprise edition
-- TURN ON ActualExecution Plan for comparison when running with/wo index.
CREATE UNIQUE CLUSTERED INDEX CustomerOrders_Index
ON vwCustomerOrders(CustomerID, OrderID);
GO

Select * From vwCustomerOrders;
GO

DROP VIEW dbo.vwCustomerOrders;
GO
```

	CustomerID	CompanyName	OrderID	OrderDate
1	VINET	Vins et alcools Chevalier	10248	1996-07-04 00:00:00.000
2	TOMSP	Toms Spezialitäten	10249	1996-07-05 00:00:00.000
3	HANAR	Hanari Cames	10250	1996-07-08 00:00:00.000
4	VICTE	Victuailles en stock	10251	1996-07-08 00:00:00.000
5	SUPRD	Suprêmes délices	10252	1996-07-09 00:00:00.000
6	HANAR	Hanari Cames	10253	1996-07-10 00:00:00.000
7	CHOPS	Chop-suey Chinese	10254	1996-07-11 00:00:00.000
8	RICSU	Richter Supermarkt	10255	1996-07-12 00:00:00.000
9	WELLI	Wellington Importadora	10256	1996-07-15 00:00:00.000
10	HILAA	HILARION-Abastos	10257	1996-07-16 00:00:00.000
11	ERNSH	Ernst Handel	10258	1996-07-17 00:00:00.000
12	CENTC	Centro comercial Moctezuma	10259	1996-07-18 00:00:00.000
13	OTTIK	Ottlies Käseladen	10260	1996-07-19 00:00:00.000
14	QUEDE	Que Delicia	10261	1996-07-19 00:00:00.000

Query executed successfully. (local)\sqlexpress01 (13.0 ... LABS\delip (56) Northwind 00:00:00 830 rows