

Unit 1 Research Project

Eddie S. Jackson

Kaplan University

IT526: SQL Query Design

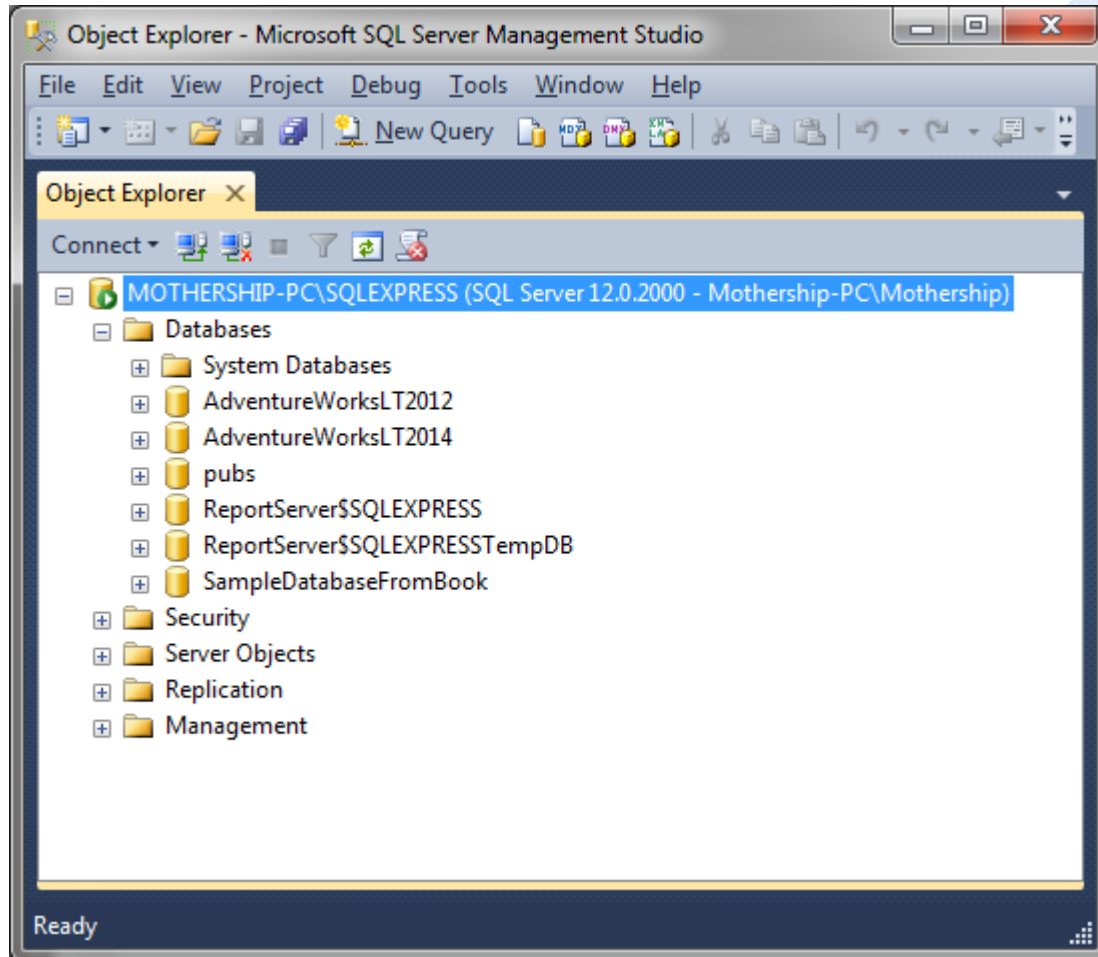
Jon Walter McKeeby, DSc, MBA

06/21/2014

Unit 1 Research Project

Part 1

Install SQL Express

**Part 2**

a. VARCHAR and CHAR are data types. The main difference between VARCHAR and CHAR is that CHAR has a fixed length, whereas VARCHAR has a variable length. The default range is 1-8000, where CHAR has a default of 10 characters and VARCHAR a default of 50 characters (Microsoft, n.d.). To use these data types in examples, “Last Name” can be represented as CHAR(9), note the space is included. In VARCHAR, a good example would be how an address is represented. For instance, 3415 George Washington St would have 25 characters but VARCHAR(50) could be used to contain up to 50 characters (a variable length). **What is stored**

sizewise is 9 in CHAR (as defined) and 25 in VARCHAR, from the amount of characters in the address itself.

b. The following query will generate an error due to the order of processing in the SQL query. When “AS” is used (which defines an alias), the respective variable will not be available in the WHERE clause. This is because the SELECT clause actually gets processed after the WHERE clause. The SQL query in question is below. Notice the names in red and the processing order. **It fails due to YearHired not being available. The correct usage in the WHERE should just be DATEPART(year, hiredate).**

```
Processed 3rd - SELECT empid, firstname, lastname,
                DATEPART(year, hiredate) AS YearHired
Processed 1st - FROM HR.Employees
Processed 2nd - WHERE YearHired = 2004;
```

c. In the contrasting WHERE clauses,

```
WHERE FileName LIKE '!_%' ESCAPE '!'
WHERE LEFT(FileName, 1) = '_'
```

the first WHERE is better than the second WHERE due to performance; the first WHERE will perform significantly faster. There could be an index associated to WHERE number one (thus the gain in performance), whereas the second WHERE is considered a function and will not use indexes (SQL Server Performance, 2004).

Part 3

a. Query the HR.Employees table. The result set column headers should be Employee ID, Last Name, and First Name.

Query

```
SELECT EmpID, LastName, FirstName
FROM HR.Employees;
```

Results

	EmpID	LastName	FirstName
1	1	Davis	Sara
2	2	Funk	Don
3	3	Lew	Judy
4	4	Peled	Yael
5	5	Buck	Sven
6	6	Suurs	Paul
7	7	King	Russell
8	8	Cameron	Maria
9	9	Dolgopyatova	Zoya

Query executed successfully.

b. Return the empid, firstname, lastname, and city for the employee with empid 9.

Query

```
SELECT EmpID,FirstName,LastName,City
FROM HR.Employees
WHERE EmpID = 9;
```

Results

	EmpID	FirstName	LastName	City
1	9	Zoya	Dolgopyatova	London

Query executed successfully.

c. Return a list of employee id's, last names, and first names for employees hired on or after January 1, 2004.

Query

```
SELECT EmpID,LastName,FirstName
FROM HR.Employees
WHERE Hiredate >= '20040101';
```

Results

	EmpID	LastName	FirstName
1	7	King	Russell
2	8	Cameron	Maria
3	9	Dolgopyatova	Zoya

Query executed successfully.

d. Retrieve the product name and unit price of all products that have unit price between \$10 and \$15. Use the BETWEEN keyword.

Query

```
SELECT ProductName,UnitPrice
FROM Production.Products
WHERE UnitPrice BETWEEN 10 AND 15;
```

Results

	ProductName	UnitPrice
1	Product IMEJH	10.00
2	Product VJZZH	10.00
3	Product LYLNI	14.00
4	Product XWOXC	12.50
5	Product SWNJY	14.00
6	Product RJVNM	14.00
7	Product CBRRL	12.00
8	Product MYNXN	12.75
9	Product ACRVI	13.25
10	Product XLXQF	14.00
11	Product TBTBL	12.50
12	Product TOONT	15.00
13	Product WEUJZ	15.00
14	Product BKAZJ	10.00
15	Product LUNZZ	13.00

Query executed successfully.

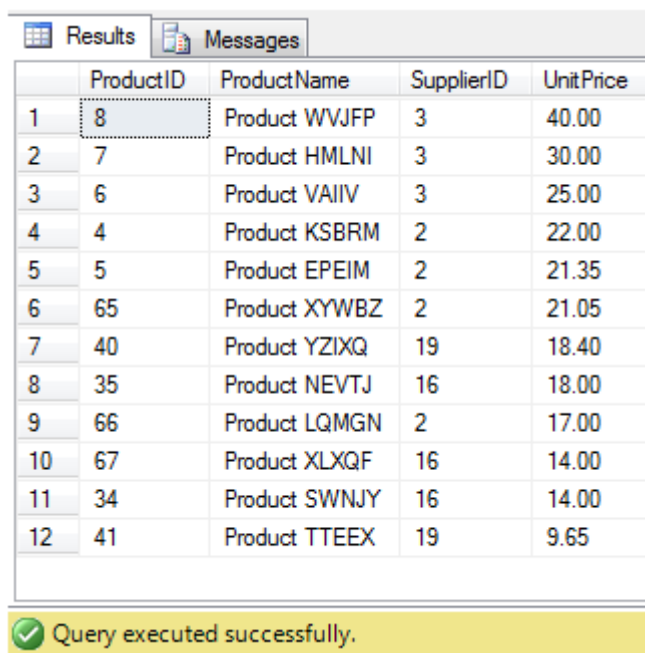
e. Retrieve the product id, product name, supplier id, and unit price of all products that are supplied by suppliers 2, 3, 16, or 19. Use the IN keyword. Return the results in order of largest to smallest unit price.

Query

```
SELECT ProductID,ProductName,SupplierID,UnitPrice
FROM Production.Products
WHERE SupplierID IN ('2','3','16','19')
ORDER BY UnitPrice DESC;
```

In the seminar example, IN was used with single quotes with numbers, but when I ran it with and without the quotes, I received the same results. Why is that?

Results



	ProductID	ProductName	SupplierID	UnitPrice
1	8	Product WVJFP	3	40.00
2	7	Product HMLNI	3	30.00
3	6	Product VAIIV	3	25.00
4	4	Product KSBRM	2	22.00
5	5	Product EPEIM	2	21.35
6	65	Product XYWBZ	2	21.05
7	40	Product YZIXQ	19	18.40
8	35	Product NEVTJ	16	18.00
9	66	Product LQMGN	2	17.00
10	67	Product XLXQF	16	14.00
11	34	Product SWNJY	16	14.00
12	41	Product TTEEX	19	9.65

Query executed successfully.

f. List the category name and description for all product categories that contain the character string 'ee' anywhere in the description. Use the LIKE keyword.

Query

```
SELECT CategoryName,Description
FROM Production.Categories
WHERE Description LIKE '%ee%';
```

Results

	CategoryName	Description
1	Beverages	Soft drinks, coffees, teas, beers, and ales
2	Condiments	Sweet and savory sauces, relishes, spreads, and ...
3	Confections	Desserts, candies, and sweet breads
4	Dairy Products	Cheeses
5	Seafood	Seaweed and fish

Query executed successfully.

g. Retrieve the order id, order date, employee id, and ship (to) city of all orders for which the shipped date is unknown.

Query

```
SELECT OrderID,OrderDate,EmpID,ShipCity
FROM Sales.Orders
WHERE ShippedDate IS NULL;
```

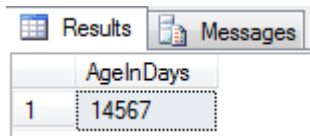
Results

	OrderID	OrderDate	EmpID	ShipCity
1	11008	2008-04-08 00:00:00.000	7	Graz
2	11019	2008-04-13 00:00:00.000	6	Buenos Aires
3	11039	2008-04-21 00:00:00.000	1	I. de Margarita
4	11040	2008-04-22 00:00:00.000	4	Eugene
5	11045	2008-04-23 00:00:00.000	6	Tsawassen
6	11051	2008-04-27 00:00:00.000	7	Toulouse
7	11054	2008-04-28 00:00:00.000	8	Buenos Aires
8	11058	2008-04-29 00:00:00.000	9	Mannheim
9	11059	2008-04-29 00:00:00.000	2	Rio de Janeiro
10	11061	2008-04-30 00:00:00.000	4	Eugene
11	11062	2008-04-30 00:00:00.000	4	Reggio Emilia
12	11065	2008-05-01 00:00:00.000	8	Barquisimeto
13	11068	2008-05-04 00:00:00.000	8	Sao Paulo
14	11070	2008-05-05 00:00:00.000	2	Frankfurt a.M.
15	11071	2008-05-05 00:00:00.000	1	Barquisimeto
16	11072	2008-05-05 00:00:00.000	4	Graz
17	11073	2008-05-05 00:00:00.000	2	México D.F.
18	11074	2008-05-06 00:00:00.000	7	Kobenhavn
19	11075	2008-05-06 00:00:00.000	8	Genève
20	11076	2008-05-06 00:00:00.000	4	Marseille
21	11077	2008-05-06 00:00:00.000	1	Albuquerque

h. [No particular database.] Write a query that returns the number of days old you are. Use the DATEDIFF function.

Query `SELECT DATEDIFF(DD,'1974-08-05',GETDATE()) AS [AgeInDays];`

Results



	AgeInDays
1	14567

KAPLAN UNIVERSITY

References

Ben-Gan, I., Sarka, D., & Talmage, R. (2013). *Training Kit (Exam 70-461): Querying*

Microsoft® SQL Server® 2012. Microsoft Press.

Microsoft. (n.d.). The valid default length range for SQL Server data types char, varchar, binary

and varbinary is 1-8000. <http://technet.microsoft.com/en-us/library/ms165682>

(v=sql.105).aspx

SQL Server Performance. (07/14/2004). Performance of LIKE vs LEFT/LEN. Retrieved from

<http://www.sql-server-performance.com/forum/threads/performance-of-like-vs-left->

len.16160/