

## Query Building for Beginning Users of SQL Server Management Studio (SSMS)

#### Presented by: Geoff Johnson

Database Administrator/Business Intelligence Developer



©2013 Shelby Systems, Inc. Other brand and product names are trademarks or registered trademarks of the respective holders.

#### Objective

This session will introduce you to the SQL Server Management Studio (SSMS) and show you how to use the querying tools inside of it.

#### What You Will Learn

- How to install SSMS on your workstation.
- How to connect to your ArenaDB or ShelbyDB database inside SSMS.
- How to use the Query Builder tool to create a query.

#### **Install SSMS on Your Workstation**

SQL Server Management Studio (SSMS) was probably installed by default on your SQL Server host computer during the regular SQL Server installation. However, you will often want to work at your desk and not at the server. To do that, you will need to install SSMS on your workstation. If you are using the Express edition of SQL Server, SSMS Express is available as a separate download from the Microsoft website. If you have the Standard edition, however, you will want to install the full version directly from the SQL Server install CD.

After you run the setup executable file on the SQL Server install CD, you will see the SQL Server Installation Center. Click on the choice for **Installation** options.



Next, click the option for a **New SQL Server stand-alone installation or add features to an existing installation**. This is not intuitive because it seems like you will be asking it to install a completely new instance of SQL Server; however, this is not the case.

	New SQL Server stand-alone installation or add features to an existing installation Launch a wizard to install SQL Server 2008 in a non-clustered environment or to add features to a existing SQL Server 2008 instance.
<b>F</b>	New SQL Server failover cluster installation Launch a wizard to install a single-node SQL Server 208 failover cluster.
şŶĨ	Launch a wizard to add a node to an existing SQL Server 2018 failover cluster.
1	Upgrade from SQL Server 2000 or SQL Server 2005 Launch a wizard to upgrade SQL Server 2000 or SQL Server 2005 to SQL Server 2008. Before you upgrade, you should run the Upgrade Advisor to detect potential problems.
4	Search for product updates Search Microsoft Update for SQL Server 2008 product updates.

On the Feature Selection screen, de-select all the Instance Features. Make sure that at least the **Business Intelligence Development Studio** (BIDS) and the **Management Tools (Basic and Complete)** options are checked. The BIDS option is for Reporting Services, should you want to use them, and the Management Tools include SSMS.



#### **Open SSMS and Connect to Your Server**

After installing SSMS, you will find it under a program group called Microsoft SQL Server 2008 R2.



After you start SSMS you will see a **Connect to Server** prompt. There are two options for how to make the connection: **Windows Authentication** and **SQL Server Authentication**.

Server type:	Database Engine		
Server name:	Geoff-PC\Shelby		
Authentication:			
User name:	Geoff-PC\Geoff		
Password:			

The Windows Authentication option uses the user account under which you are currently logged into Windows. To use this method, you must be logged into the domain with an account that has permission to access the database. Domain admins automatically have permission, but other accounts need to have that permission granted explicitly or need to be part of a group with permission. For most day-to-day access I recommend using the SQL Server Authentication method.

The SQL Server Authentication option allows you to provide a SQL Server specific user login name and password to get connected. These credentials must be recognized on the server. The standard user names to use are ShelbyRead, ShelbyGF, and ShelbyPR. These are all read-only accounts, so they can be used to protect the server from inadvertent changes. The sa account is also a valid login account, and it has full access to read data and to make changes to the database.

🚽 Connect to Server		×			
SQL SQL	<b>Server</b> "2008 R2				
Server type:	Database Engine	-			
Server name:	Geoff-PC\Shelby	Geoff-PC\Shelby			
Authentication:	SQL Server Authentication	-			
Login:	ShelbyGF	-			
Password:	******				
	Remember password				
Connec	ct Cancel Help Option	s >>			

	Supported Shelby v.5 Logins				
Login ID	Туре	Access to:	Default Password		
sa	Read/Write	Every table	ShelbySystems-2007		
ShelbyRead	Read Only	All data except CN, GF, and PR	SsI-01-Rd-XXXXX		
ShelbyCN	Read Only	All data except GF and PR	SsI-01-Cn-XXXXX		
ShelbyGF	Read Only	All data except CN and PR	SsI-01-Gf-XXXXX		
ShelbyPR	Read Only	All data except CN and GF	SsI-01-Pr-XXXXX		

#### Set ShelbyDB as the Default Database

To make querying easier later on, it is good to set your Shelby v.5 database as the default database for the SQL Server connection. To do that, click the **Options** button on the Connect to Server prompt.

Connect to Server	Server 2008 R2
Server type:	Database Engine 💌
Server name:	Geoff-PC\Shelby -
Authentication:	Windo Authentication 🔹
User name:	Geoff-PC ff
Password:	
Conne	Remember password  Remember password  Cancel Help Options >>

Change to the **Connection Properties** tab if you are not taken there automatically. Then change the **Connect to database** drop-down option to ShelbyDB.

igin Connection Properties	Connection Parameters		
Type or select the name of the dat Connect to database:	tabase for the connection. ShelbyDB		
Network			
Network protocol:	<default></default>		
Network packet size:	4096 🔶 bytes		
Connection			
Connection time-out:	15 🚔 seconds		
Execution time-out:	0 🚖 seconds		
Encrypt connection			
Use custom color:	Select		

#### **Explore the Tables and Columns**

In the right-hand section of the SSMS window is the Object Explorer pane. It shows all the items in the server, including the databases. Just click the + next to the Databases container to show a list of all the databases. And click the + next to the database name to see the kinds of things inside the database. Click the + next to the Tables container or the Views container to see the tables in your database.

🖃 间 ShelbyDB
🕀 🚞 Database Diagrams
🕀 🚞 Tables
🔪 🖃 🗀 Views 🚽
🕀 🛄 System Views
🗄 🔄 Shelby.VIEW_SHELBY_CUSTINFO_ALL
🗄 🔝 Shelby.VIEW_SHELBY_CUSTINFO_FORMAT
🗄 🔝 Shelby.VIEW_SHELBY_CUSTINFO_NUMBERFORM
🗄 🔄 Shelby.VIEW_SHELBY_FORMATTED_NAME_NO_I
🗄 🔝 Shelby.vw_APInvAPCCChg 🕺
😠 🔝 Shelby.vw_APInvDetailAPCCChgDetail 🕺
🗄 🔝 Shelby.vw_APInvoices 👌
🗄 🔝 Shelby.vw_APTransactions 🛛 💡
🖃 🔝 Shelby.vw_APVendors
🕞 📴 Columns 🔵 💦 👌
🔳 NameCounter (int, not null)
📃 FormalName (varchar(123), null)
📃 InformalName (varchar(102), null)
📃 FormalName_LastFirst (varchar(125), null)
📃 InformalName_LastFirst (varchar(103), nu§
📃 Salutation (varchar(40), null)
📃 CellPhone (varchar(20), not null)
🔳 Email (varchar(4000), null)
📃 MainPhone (varchar(20), not null)
📃 MainAddressLine1 (varchar(35), null) 🛛 👌
🔳 MainAddressLine2 (varchar(46), null) 🧃
📃 MainAddressLine3 (varchar(46), null)
City (varchar(28), not null)
📃 State (varchar(5), not null)
📃 ZIP (varchar(5), not null)
📃 Terms (varchar(25), null)
🔳 ProductLine (varchar(25), null)
Approver (varchar(25), null)

There are many other items listed in the object explorer besides the databases, tables, and columns you might use for a query, but those items are outside the scope of this session. Some of them are covered in the session on using SSMS to manage your database.

Some of the information about the tables, views, and columns is self-evident, but some of the naming conventions are not, so below are a couple of charts to help orient you to the tables and views in the ShelbyDB database.

AM	Expense Amortization	MB	Membership
AP	Accounts Payable	MG	Matching Gifts
AR	Accounts Receivable	MI	Profiles for Advanced Search
BR	Bank Reconciliation	NA	GlobaFILE
CA	Appointments	NU	Check-In
СК	Check Express	РО	Purchase Orders
СМ	Commemoratives	PR	Payroll
CN	Contributions	RC	Resource Calendar
DH	Deposits on Hand	RE	Receipting
FA	Fixed Assets	RG	Registrations
GF	Donors and Gifts	RM	Remittance
GL	General Ledger	SE	Servant
IN	Inventory	SG	Organizations and Small Groups
IR	Insurance and Retirement	SS	System Tables
LN	Loan Processing	ST	Statistics

#### ShelbyDB Table Prefix Abbreviations

#### **ShelbyDB Views**

vw_APInvoices	vw_GBNames	vw_MBLifeEvents
vw_APTransactions	vw_GBOtherRelations	vw_MBNames
vw_APVendors	vw_GBOtherRelationsLowerNames	vw_PRCheckDetails
vw_ARBillToNames	vw_GBOtherRelationsUpperNames	vw_PRChecks
vw_ARCustomerNames	vw_GBProfileInformation	vw_PREmployees
vw_ARInvoices	vw_GLAccounts ^	vw_SGAttendanceHistory ^
vw_ARTransactions	vw_GLActualsByPeriod	vw_SGAttendees
vw_CNGifts	vw_GLBudgetEntries	vw_SGEnrollmentCustomFields
vw_CNNames	vw_GLBudgetsAndActualsByPeriodAndProject	vw_SGEnrollments
vw_GFGifts	vw_GLBudgetsByPeriod	vw_SGGroupCustomFields
vw_GFNames	vw_GLClosingAccountActivity	vw_SGGroups
	vw_GLJournalEntries	vw_SGGroupStatistics
	vw_GLSpecialReports	

More information is available for these views. Just search the Shelby Community for the view name minus the opening vw\_, and you should find information about the contents of the view.

The CN (Contributions) and GF (Donors & Gifts) views have been improved since these were added to the database. The revised views can be downloaded from the Shelby Community, under the Downloads/ShelbyQUERY/Data Views path. I recommend using these views instead.

^ The GLAccounts and SGAttendanceHistory views can take a very long time to populate if you have a large number of records or a long history of related information in ShelbyDB. There is a new GLAccounts view available in the Shelby Community in the same Data Views path mentioned above, and the new GLAccounts view should not take so long to load. For the SGAttendanceHistory view, always filter it to only the necessary date range for your reports. That will help it run more quickly.

If you have trouble using any of these views, feel free to contact extreme.reporting@shelbyinc.com for assistance.

To explore the actual data inside any table or returned by any view, simply right-click on the name of the table or view and click the option to **Select Top 1000 Rows**.



The system will automatically open a new query window. In the top half of the query window you will see the query statement, with every column listed. And in the bottom half of the query window you will see the results, as shown on the next page.

The first line of the query (after a comment) is SELECT TOP 1000 plus the first column in the table or view. If you delete the words TOP 1000, you can get the full contents of the table or unfiltered results of the view.

To run the query and refresh the results, press the F5 key or click the Execute button the toolbar.



**Note:** Don't confuse the Execute button with the "play" button (>) to the right of it. That button is a Debug button, but it does not really work with SQL queries the way we will be working with them.

∕\$QI	Query13.sql - 1	GEOFF-V7\f (56))	SQLQuery11.sql - GEOFF	-V7\f (60))			4
	1 /****	** Script for Se	lectTopNRows com	mand from SSMS	*****/		
	2 SELEC	T TOP 1000 [Name	Counter]				j
	3	,[FormalName]					2
	4	,[InformalName]					
	5	,[FormalName_La	stFirst]				5
	6	,[InformalName_]	LastFirst]				
	7	,[Salutation]					
	8	,[CellPhone]					1
	10	,[Email]					1
	11	,[MainFhone]	nell				]
	12	,[MainAddressLi	nell				-
	13	.[MainAddressLi	ne3]				
	14	,[Citv]	,				
	15	,[State]					ļ
	16	,[ZIP]					-
	17	,[Terms]					
	18	,[ProductLine]					2
	19	,[Approver]					
	20	,[ContactName]					)
	21	,[Memo]					{
	22	,[LastInvoiceDa	te]				3
	23	,[LastInvoiceAm	ount]				ç
	24	,[LastInvoiceNu	mber]				-
	25: FRO.	m ISneidyusi.ISn	elbvi.ivw Apvend	orsi			
	iesults 🛅 Me:	ssages				1	
	NameCounter	FormalName	InformalName	FormalName_LastFirst	InformalName_LastFirst	Salutation	CellPhone
1	1	Acme Paper Supply	Acme Paper Supply	Acme Paper Supply	Acme Paper Supply	Bill Johnson	N/A
2	9	Mr. George Adams	George Adams	Adams, Mr. George	Adams, George	George	(901) 456-8654)
3	18	Herb Alpert	Herb Alpert	Alpert, Herb	Alpert, Herb	Herb	N/A }
4	28	Mrs. Grace Anders	Gracie Anders	Anders, Mrs. Grace	Anders, Gracie	Gracie	N/A j
5	40	American Express	American Express	American Express	American Express		N/A
6	71	Mr. Fred Astaire	Fred Astaire	Astaire, Mr. Fred	Astaire, Fred	Fred	N/A
7	198	Mr. George Barnes	George Barnes	Barnes, Mr. George	Barnes, George	George	N/A §
8	240	Christian Book Store	Christian Book Store	Christian Book Store	Christian Book Store		N/A S
9	275	Cordova Community Ch	Cordova Community Ch	Cordova Community Ch	Cordova Community Ch		N/A
10	339	EPS Technologies	EPS Technologies	EPS Technologies	EPS Technologies		N/A
11	567	IBM Corporation	IBM Corporation	IBM Corporation	IBM Corporation		N/A
12	632	Jones Lumber	Jones Lumber	Jones Lumber	Jones Lumber		N/A
13	637	K-Mart	K-Mart	K-Mart	K-Mart		N/A 💈
للحرب		~	~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	~L/~L/~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~		~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	

#### Start a New Blank Query

The easiest way to start a new query is to press **Ctrl-N** or click the **New Query** button on the toolbar, shown circled in the screenshot below.



Make sure the drop-down on the query toolbar shows the database you want to work with. If you set the default database for the connection, it should be correct. If you did not set the default database, it will be set to Master by default, and you will need to change it to ShelbyDB.



#### **Open the Graphical Query Editor**

Initially there is no toolbar icon for the graphical Query Editor, so the only ways to open are to select **Query > Design Query in Editor...** from the menu or press **Ctrl-Shift-Q**. If you want to use the query editor frequently, I recommend adding it to the query toolbar by using the Tools > Customize menu option. The icon is listed as part of the Query set of commands. Just drag the icon to the Query toolbar wherever you want it to appear.



#### Add One or More Tables, Views, or Functions

When you first open the query designer, the Add Table window will pop up in front of the main designer window. In the ShelbyDB database you can select from the Tables and Views tabs. In ArenaDB there are options under Tables, Views, and Functions. Neither database provides anything under the Synonyms tab.

**Tables** are the structures that store the underlying data captured by the software and stored in the database.

**Views** are saved queries that have done some of the work of putting the raw data into a form that can be turned into a usable report.

**Functions** are like views but accept parameters and return a very specific value or set of values. Most queries do not require functions.

A	dd Table			? <mark>- x</mark>
(	Tables	Views	Functions Synonyms	
	vw_GBO	therRelat	onsUpperNames (Shelby)	*
	vw_GBPi	rofileInfo	mation (Shelby)	
	vw_GFGi	iftDetails	(Shelby)	
	vw_GFGi	ifts (Shelb	<u>v)</u>	
	vw_GFN	ames (Sh	elby)	
	vw_GFPt	urposesA	ndPledges (Shelby)	
	vw_GLA	ccounts (	Shelby)	
	vw_GLA	cct_0 (Sh	lby)	
	vw_GLA	cct_1 (Sh	lby)	
	vw_GLA	cct_10 (Sł	elby)	
	vw_GLA	cct_11 (Sł	elby)	
	vw_GLA	cct_12 (Sł	elby)	-
l	GL A.	+ 10 /91	alle à	•
			Refresh Add	Close

To add any item from the list, double-click it. The examples to follow will use **vw\_GFGifts**, a view of all giving in Donors & Gifts.

After you select the view(s) or table(s) you need, click Close to work in the main Query Designer window.

#### **Select the Desired Column**

The top pane of the query designer shows the tables or views that were just added. Selecting the columns you want to see in the results is as easy as putting a checkmark next to the names of the columns you want. If you want all the columns, there is a shortcut: just click the asterisk at the very top of the list of columns.

When you put a checkmark next to a column name, that column also appears in the middle pane of the query designer. It also appears in the query text editor in the bottom pane.

Query	Designer						×
	vw_GFGifts (Shelby)  * (All Columns) GiftKey PurposeKey PledgeKey GiftDate GiftDate GiftDate PurposeCode PurposeCode GiftAmount GiftAmount SisNonCash Project BatchNumber IsAnonymous CheckNumber ReceiptNumber	30					
•				<del>-</del>			*
	Column /	Alias Lable	Output	Sort Type	Sort Order	Filter	Or
	Citronicer	vw_GFGITC	. 💌				
	GircDate	vw_GFGITC	. 🔽				
	PurposeCode	vw_GFGIT	. 💌				
	PurposeDescri	VW_GFGIFt	. 🔽				
		VW_GFGIRt	. 💌				
	IsiNonCash	VW_GFGIFt	. 🔽				
		VW_GFGIFt	. 🔽				
	MotivationDes	vw_GFGIFC	. 🔽				
		III					
LECT .OM	T NameCounter, GiftD Shelby.vw_GFGifts	ate, PurposeCode, Purpo	seDescriptio	on, GiftAmount,	IsNonCash, Motivati	onCode, Motivat	ionDescript
			III				
						ок С	Cancel

Click **OK** at the bottom of the query designer to return the text of the query to the main SSMS query text editor. Click the **Execute** button on the toolbar or press **F5** to execute the query.

#### Use the Query Designer to Modify an Existing Query

Most of the time when you open the Query Designer it will be empty and you will choose your tables and columns from scratch. However, if you already have a query you can load it into the Query Designer by selecting the text of the query first (Ctrl-A is a handy way to do that) before opening Query Designer.

This is one of the major advantages over the ShelbyQUERY tool in Shelby v.5 because ShelbyQUERY cannot load pre-existing query logic into its graphical Query Builder tool.

#### Add a Filter to One or More Column Values

The Filter column in the middle pane allows you to filter the rows returned by the query to the rows that match a certain value for the column. In the example below you can see a filter to return only the building program giving.

	Column	Alias	Table	Output	Sort Type	Sort Order	Filter	Qr 🗠		
	NameCounter		vw_GFGift	1						
	GiftDate		vw_GFGift	1						
•	PurposeCode		vw_GFGift	~			= 'BUILD'			
	PurposeDescri		vw_GFGift	~						
	GiftAmount		vw_GFGift	1						
•		1		Freedo						
SELECT	NameCounter, Gift	tDate, Purpo	seCode, Purpose	Descriptio	n, GiftAmount, IsN	onCash, Motivation	Code, MotivationDesc	criptio		
WHER	(E (PurposeCode = 'BUILD')									

Notice that the value of the PurposeCode column is enclosed in single quotation marks. All nonnumeric values (generally text and dates) need single quotation marks. Numeric values should not have single quotation marks around them.

#### **Common Comparison Operators**

Operator	Description				
=	Equal to				
<	Less than				
>	Greater than				
<=	Less than or equal to				
>=	Greater than or equal to				
<>	Not equal to				
LIKE	Matches a text pattern, with _ standing in for any one alphanumeric character and % standing in for any number of alphanumeric characters (including zero characters).				

**BETWEEN** *X* **AND** *Y* allows you to compare a column to a range of values, not just to a single value. In the illustration below you see a comparison that will include all donations in 2013. The system adds CONVERT() functions to make the date values you type in as strings into datetime values.

	Column	Alias	Table	Output	s	Sort Order	Filter	Or	Or	*	
	NameCounter		vw_GFGift	-							
•	GiftDate		vw_GFGift	-		<b>C</b>	BETWEEN '1/1/2013' AND '12/31/	/20		E	
	PurposeCode		vw_GFGift	<b>V</b>						_	
	PurposeDescri		vw_GFGift	<b>V</b>							
	GiftAmount		vw_GFGift	<b>V</b>							
	TANLE		CTCS	112						Ξ.	
				111					•		
SELECT FROM	SELECT         NameCounter, GiftDate, PurposeCode, PurposeDescription, GiftAmount, IsNonCash, MotivationCode, MotivationDescription           FROM         Shelby, vw. GEGifts           VMERGE         College Set Set Set Set Set Set Set Set Set Se										
WHEN			(DATETIME, 2013	01-01-00	.00.00	, 102) AND C	SWEEKI(DATETINE, 2013-12-310	0.00.00, 102)			

**NOT BETWEEN X AND Y** is also a valid comparison statement.

**IN** ( $X, Y, \ldots Z$ ) allows you to compare a column to a set of values. There is no limit to the number of values that can be inside the parentheses. **NOT IN ()** also works.

**IS NULL** allows you to compare a column to see if the value is a NULL value. NULL is a special placeholder for information that is unknown. This is differentiated from information that is empty. Consider this simple "table":

FullName	DogOwned	Breed		
John Smith	Y	Great Dane		
Mary Smith	Y	NULL		
Susan Smith	Ν	{empty}		

In one case there is a dog whose breed is known, in one case there is a dog but an unknown breed, and in one case there is no dog, and therefore no breed.

This is the theoretical difference between "blank" and "NULL," but in actual practice, there are often values that you would expect to be blank that are NULL and vice versa. The important thing to remember is that there are not just two possible statuses for a value (filled in or blank) but three (filled in, blank, or NULL).

If you want to include all rows that match certain criteria or have no information available for that column, put IS NULL in the **OR...** property to the right of the Criteria.

**IS NOT NULL** allows you to find the values that are known and are therefore not NULL. This can include values that are blank.

There are three **Or...** column properties to allow for more complex filtering options.

#### Add More Tables to the Design

If you need columns from a table not yet in the query design, you can add tables at any time. Just rightclick anywhere in the Query Deisgner and click the **Add Table...** option. This will re-open the Add Table requestor and give you the chance to add more tables.

Query D	Designer						×	3
	vw_GFGifts (Shell	by)						•
				Execute SQL				
				Add Group By				
				Change Type	•			
				Redo				
				Undo				
				Add Table				Ŧ
-				Add New Derived Table			P	_
	Column	Alias	Table	Pane	r ∎	Filter	10	â
•	NameCounter		VW_GEGIN	Clear Results				
	GiftDate		vw_GFGift	Clear (could		BETWEEN '1/		
_	PurposeCode		vw_GFGift	Properties				-
•	PurnoseDescri	1	III	- man -			Þ.	
SELECT FROM WHERE	NameCounter, Gif Shelby.vw_GFGif (GiftDate BETWEE	tDate, Purp its IN CONVER	ooseCode, Purpo: T(DATETIME, '20	seDescription, GiftAmount, IsNonCas 13-01-01 00:00:00', 102) AND CONV	h, Motivatic	mCode, MotivationDe IME, '2013-12-31 00:	scrip:	otio 10',
•				III				Þ.
					0	K Cance	: <b> </b>	]

In the following examples, the **vw\_GFNames** view will be added to the query.

#### **Create a Join Between Tables**

With rare exceptions, if you have more than one table in the query, you will need to join the tables together. A "join" defines the relationship between the tables so that information from one table can be aligned properly with the information in the other table. In the SSMS Query Designer, you define a join by dragging the column of one table to the related column of the other table. Because this is one area in which new query writers are prone to make mistakes, we'll take it step by step.

#### Step One: Determine the Related (Key) Columns in Each Table

In the example below, the **vw\_GFGifts** view is related to the **vw\_GFNames** view through the **NameCounter** column in each view. NameCounter is a column used frequently to tie information about people together.



For the underlying tables in Shelby v.5 (not the views) the table definitions, including key column relationships between the tables, are stored at

http://www.shelbyinc.com/Documents/pdfs/v5/HSQTBLs.pdf.

For the Shelby v.5 views, search the Shelby Community for the name of the view, and you should find an article that describes the key columns.

### Step Two: Drag the Column Name from One Table to the Related Column in the Other Table

This is a simple drag-and-drop process.



When you let go, the join relationship will be represented on the screen by a bar between the tables and a diamond at the center of the bar.



Another major advantage of SSMS Query Designer over the ShelbyQUERY designer is that you can drag-and-drop multiple columns, if necessary, to join two tables on more than one common value. In financial queries it is often necessary to join on BeginDate, CoNu, FundNu, DeptNu, and AcctNu. In ShelbyQUERY's Query Builder, this is impossible. In SSMS Query Designer, it is easy.

#### Step Three: If the Join Should Not Be an Inner Join, Change It

The default join type when you drag-and-drop the column is an inner join. This means that there must be a matching value on both sides of the relationship. But sometimes you will want all the rows from one of the tables regardless of whether or not there is a match in the second table. To change the join type, right-click on the diamond in the middle of the join bar and select the description that best describes the results you want. For example, if it were possible to have a gift with no NameCounter associated with it, we might select "Select All Rows from vw\_GFGifts (Shelby)" so that we would get all the gifts regardless of a missing NameCounter value.



The diamond shape will change to reflect the join type, with a rectangle appearing below the diamond on the side of the bar nearest the table from which all rows will be returned.

Module Header		Key Column	Detail Table	Key Column	View
Exp Amort.	AMMst	Counter	AMDest	AMMstCounter	
AP	APInv	Counter	APInvDetail	Counter vw_APTransactions	
AR ARInv		HeaderCounter	ARInvDetail	HeaderCounter	vw_ARTransactions
Contributions	Contributions CNHst		CNHstDet	HstCounter	CNGifts
Dnrs & Gfts GFHs		Counter	GFHstDet	HstCounter	GFGifts
Gnrl Ledger	GLEntri	HeaderCounter	GLDetail	HeaderCounter	vw_GLJournalEntries
Payroll	PRHist	HistoryNu	PRHistDetail	HistoryNu	vw_PRCheckDetails

#### **Commonly Joined Financial Tables and Related Views**

#### **Tweak the Output**

#### Add Aliases for Column Names

If you want a column to appear in the results with a friendlier name than the one in the table or view, you can change the name of the column by putting your preferred name into the Alias property of the selected column in the middle pane.

	Column	Alias	Table	Output	Sort Type
	NameCounter		vw_GFGift	1	-
•	FormalName	Donor	vw_GFNa	1	
	GiftDate		vw_GFGift	1	
	PurposeCode		vw_GFGift	1	
	PurposeDescri		vw_GFGift	1	
	GiftAmount		vw_GFGift	1	;
	IsNonCash		vw_GFGift		~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

#### **Reorder Columns**

Columns will appear in the results in the same order from right-to-left that they appear top-to-bottom in the selected columns pane. You can re-order columns by clicking in the box to the left of the column name and dragging a column reference up or down, then dropping it in the new position.

#### Remove a Column from the Output

If you need a column for filtering purposes but do not necessarily want it in the output, you can uncheck the Output box. The column will not appear in the results.

#### Sort the Results

You can use the Sort Type and Sort Order properties to dictate in what order the results will appear. If you do not set these values, the order of the output is considered random. Even if it appears to have a "natural" sequence, don't rely on it to be the same every time unless you specify the sort order.

In the screenshot below you can see the FormalName\_LastFirst has been added for sorting purposes only. The Output checkmark has been removed. The Sort Order for FormalName\_LastFirst has been set to 1, and the Sort Order for NameCounter has been set to 2, which will keep individual gifts separate even if two donors have the same name. Both sorting columns are Ascending for the Sort Order.

Column	Alias	Table	Output	Sort Type	Sort Order	Filter
NameCounter		vw_GFGift	<b>V</b>	Ascending	2	Ì
FormalName_L		vw_GFNa		Ascending	1	4
FormalName	Donor	vw_GFNa	1			+
GiftDate		vw_GFGift	1			BETWEEN '1/
PurposeCode		vw_GFGift	1			
PurposeDescri		vw_GFGift	1			-
 CiftAreasta		NUM COCIED				~ <u>~</u>

#### **Add Aggregate Functions**

Sometimes you are not interesting in individual values but want to see aggregates, such as total giving to a specific purpose code or totals from each donor in a date range. When that is the goal, you need to add aggregate functions. Doing that is pretty simple in SSMS Query Designer.

First you have to "activate" the aggregate function options by right-clicking anywhere in the window and selecting "Add Group By."

Query	Designer				×
	<ul> <li>ww_GFGifts (S</li> <li>* (All Columns)</li> <li>GiftKey</li> <li>PurposeKey</li> <li>PledgeKey</li> <li>NameCounter</li> <li>GiftDate</li> <li>PurposeCode</li> <li>PurposeDescrip</li> <li>GiftAmount</li> <li>CashAmount</li> <li>IsNonCash</li> <li>Project</li> <li>BatchNumber</li> </ul>	Shelby) ĝ↓ tion		ww_GFNames (Shelby) FormalName InformalName_LastFirst \$↓ InformalName_LastFirst Salutation Gender Execute SQL Add Group By Change Type Redo Undo Add Table	•
	Column	Alias	Table	Add New Derived Table er Filter o	Dr 🔺
	NameCounter		vw_GFGifl	Dana	
	FormalName_L		vw_GFNa	Class Desults	
	FormalName	Donor	vw_GFNa	Clear Results	
	GiftDate		vw_GFGifl	Properties BETWEEN '1/	Ξ
	PurposeCode		vw_GFGift		

Once you turn on the Group By options, you will get an additional column in the middle pane of selected columns called, appropriately, Group By, which is also the default value for the property on each selected column. You only need to change this property for the columns that should be aggregated or need to be ignored because they are only for filtering purposes.

When you change the Group By property to any aggregate function (such as SUM, AVG, MIN, or MAX), an Alias of EXPR1 will be added. You should change these default expressions to meaningful descriptions of the aggregated columns.

In the screenshot below you can see several features of the Query Designer used for aggregating values.

Query D	)esigner								×
	↓       ww_GFGifts (S         ↓       All Columns)         □       GiftKey         ↓       PurposeKey         ↓       PledgeKey         ✔       NameCounter         □       GiftDate         ✔       PurposeCode         ₽       Project         ₽       PurposeCode         ₽       PurposeCode         ₽	Shelby)		Formal Informal Informal Salutat Gender DateOf Age Marital CellPho Email Familyf	FNames (Shelb Name alName LastFirs alName_LastFi ion Birth Status one MainPhone MainAddressLir				*
			- 11						
-	Column	Alias	Table	Output	Sort Type	Sort Order	Group By	Filter	Or
	NameCounter		VW_GFGIFt		Ascending	2	Group By		
2 1	FormalName_L		VW_GENa		Ascending	1	Group By		
	FormalName	Donor	VW_GENa	v			Group By		
	GiftDate		vw_GFGift				Where	BETWEEN '1/	)
	PurposeCode		vw_GFGift	<b>v</b>			Group By		
	PurposeDescri		vw_GFGift	1			Group By		
	GiftAmount	TotalGift	vw_GFGift	1			Sum		)
P.	IsNonCash		vw_GFGift			-	Where	= 'No'	
	MotivationCode		vw_GFGift	1			Group By		
-	MotivationDes		vw_GFGift	<b>v</b>			Group By		
•			I	11					E.
SELECT	Shelby.vw_GFGift SUM(Shelby.vv Shelbv.vw GFGif	s.NameCounto v_GFGifts.Gift ts LEFT OUTE	er, Shelby.vw_GF Amount) AS Total R JOIN	Names.Fo IGift, Shelb	rmalName AS Do by.vw_GFGifts.M	onor, Shelby.vw_GF lotivationCode, She	Gifts.PurposeCode lby.vw_GFGifts.Mo	, Shelby.vw_GFGifts.F tivationDescription	PurposeD 🔺 +
							C	ОКС	Cancel

- In the selected tables pane, the columns used as GROUP BY values have a grouping icon next to them; whereas, the aggregated columns have a Greek sigma symbol next to them. (Note also the funnels marking the filtered columns and the A-Z icon for columns being used for sorting.)
- In the selected column pane, you can see the Group By property set to Where for GiftDate (a filter for the current year) and for IsNonCash (a filter for only cash gifts, not gifts in kind).
- The Group By property for GiftAmount is set to Sum, and the default EXPR1 alias has been replaced with the more meaningful TotalGift.

#### **Finishing Up**

#### Save the Query

After you click OK in the Query Designer and press F5 to get the results of the query, you should verify the results to make sure you are getting the results you expect. If you don't, the problem is probably with a WHERE clause or a JOIN condition. But once you are satisfied, press Ctrl-S to save your query. You can give it any name you want. SSMS puts an extension of .sql at the end of the file name, but it is just a text file that you can open in Notepad and share with others if you want.

#### Get the Data Out of SSMS

Unless you have many thousands of rows of data, the best way to get the data out of SSMS is to click the empty box in the upper left corner of the query results (above the row numbers and to the left of the column headers). This will select all the rows and columns. Then right-click on the grid of results and select **Copy with Headers**. Then open Excel and paste the clipboard into a blank worksheet. That is the only way to get the headers along with the data.

The other option, **Save Results As**, works fine to export the data to a CSV file, but it does not include the column headers too. If you do have many thousands of rows, your RAM memory contraints may make this your only option. But the headers will have to be re-added to the Excel file when you open it.

# HQ 2013

#### Geoff Johnson Database Adminisrator/ Business Intelligence Developer geoff.johnson@shelbyinc.com



Geoff Johnson has been working with Shelby Systems since 2003 and is currently the Database Administrator/Business Intelligence Developer. As a DBA his primary task is improving the effectiveness of the storage and delivery of the data inside each Shelby software solution. As a BI Developer his primary task is providing quality reports that facilitate the work of our customers and thereby expand the Kingdom of God.

Geoff has a passion for bringing the latest business intelligence innovations from Microsoft to the ministry of the church. He has helped many Shelby Systems customers implement Reporting Services solutions, PivotTable and PowerPivot analytical reports, and other custom report designs ranging in complexity from simple directories to complex GAAP-compliant financial statements.

He has worked closely with several denominational headquarters offices to improve and streamline their data collection and reporting processes.

