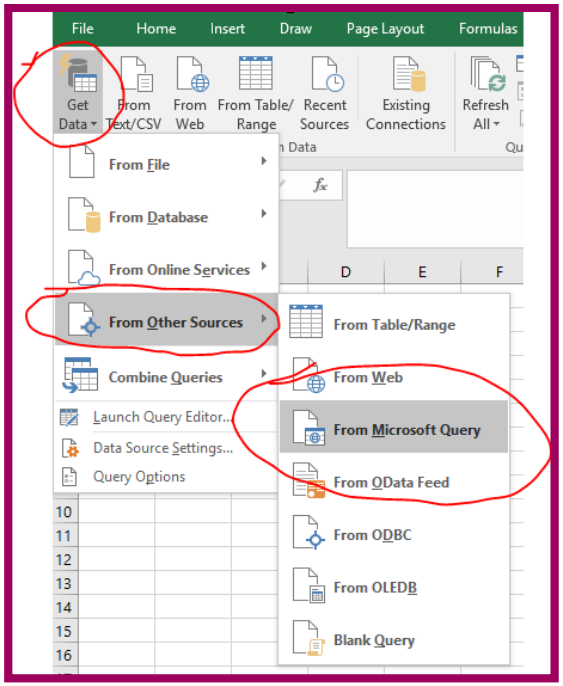
EXCEL & SAGE 50

A Beginners Guide to Linking Excel Spreadsheets to the Sage 50 Data

EgoPetram
Shaping Up Businesses

INITIATING A QUERY

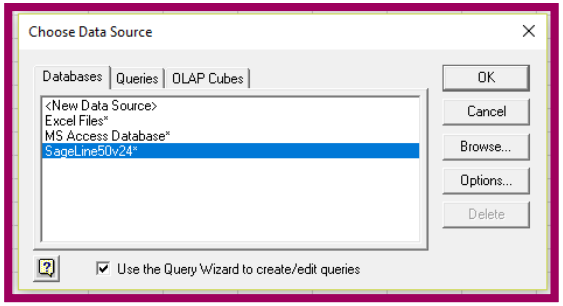


To start a query that extracts the data from Sage 50, select the DATA menu bar in Excel.

On the left of the DATA menu bar you will find a 'Get Data' option from which dropdown menus appear. Select 'From Microsoft Query' from the menu that extends the 'From Other Sources' option in the first dropdown box.

Note:
As you are using an ODBC connection to access the Sage data, you may be tempted to try the 'From ODBC' option.

Don't! For some reason it doesn't work properly. Sage don't seem to know why.



A pop-up box appears offering a choice of Data Sources.

Select the 'SageLine50v..' source.

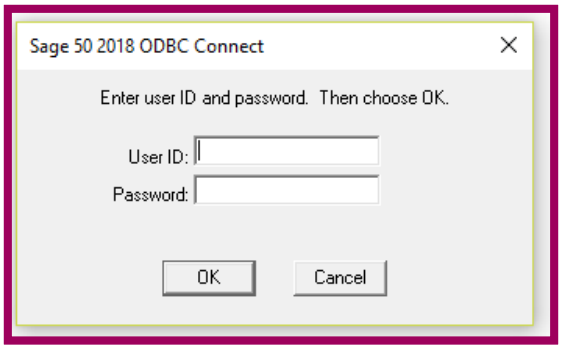
The version number should correspond with the version of Sage 50 that you are currently running. In this case v24.

Click OK

EgoPetram
Shaping Up Businesses

QUERY WIZARD

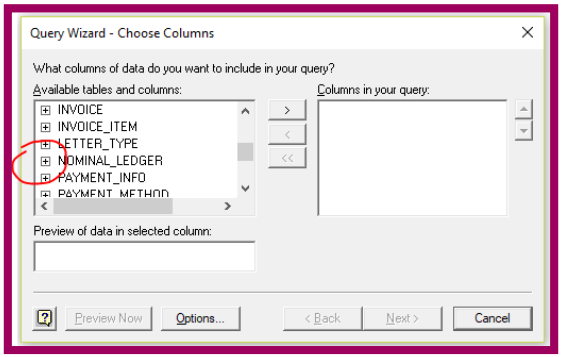
EgoPetram
Shaping Up Businesses



You are next asked for a User ID and Password.

These are any valid Sage 50 login and the associated password.

Note:
Once you are familiar with defining queries it is possible to save the login details in the definition of the connection. This has a disadvantage in that the password is visible!

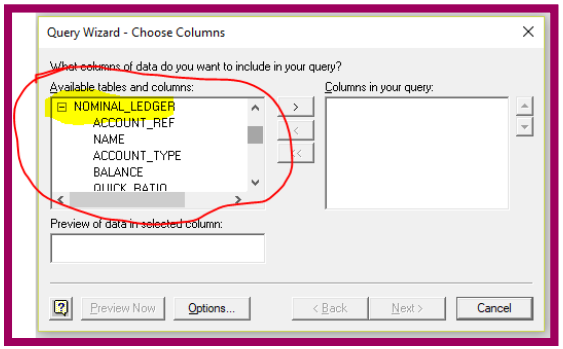


The 'Query Wizard – Choose Columns' pop-up is where you start deciding the data you wish to see.

The initial screen only shows the table names. This is fine if you wish to download a whole table but, often, tables contain fields that aren't useful so it makes sense to only select the fields you want to see.

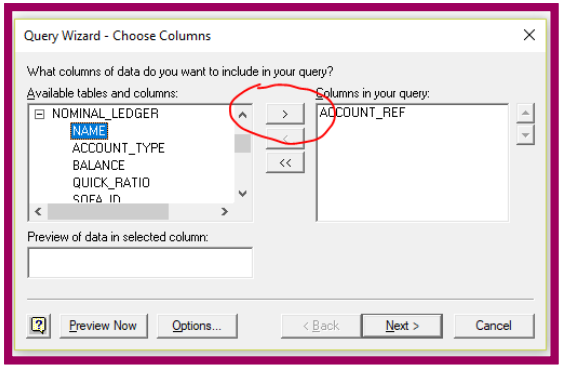
To see the fields in a given table expand the list by clicking the + to the right of the table name.

In the illustration the NOMINAL_LEDGER table.



In the lower illustration, you can see how the list expands to show the field names available.

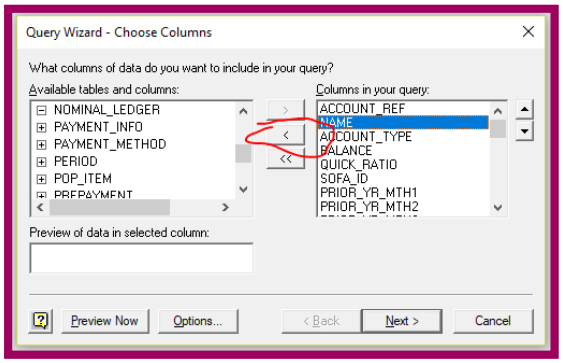
EgoPetram
Shaping Up Businesses



To select an item for your query by clicking the > button.

The field you have chosen then moves into the 'Columns in your query' box.

Clicking on the > button when the table name is highlighted results in all the fields in that table appearing in the 'Columns in your query' box.



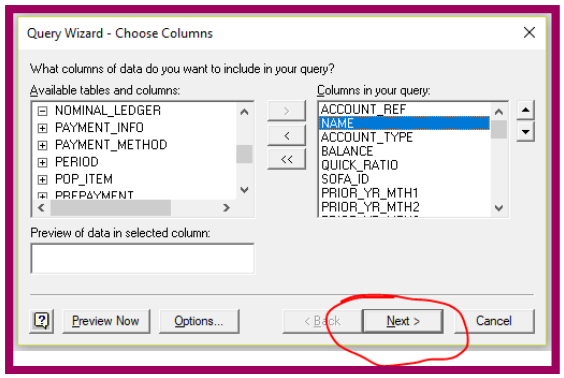
To remove a column from your query select the item in the 'Columns in your query' box that you wish to remove and the < button ceases to be 'greyed out'.

Click the button and the field returns to the 'Available tables and columns' box.

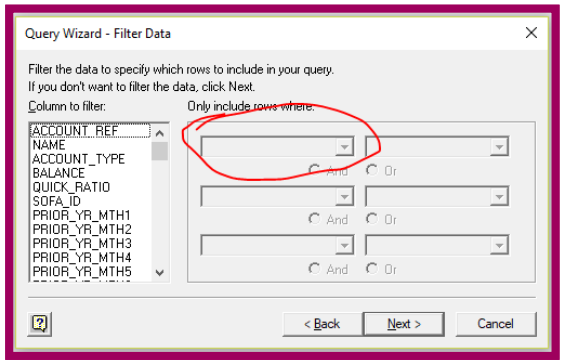
Using the << button removes all the fields chosen.

Note:
Until you are familiar with linking Excel to Sage 50 and with the way the data is held, it is best to limit the query to selecting from a single data table.

EgoPetram
Shaping Up Businesses

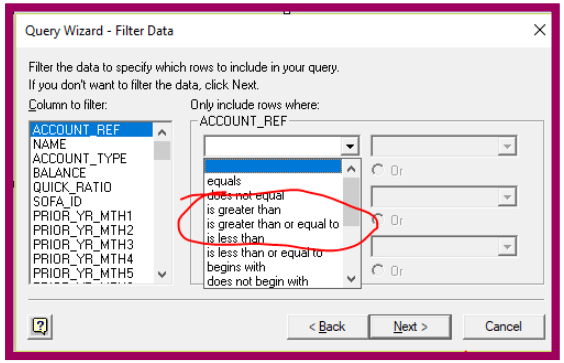


Once you are happy with the choice of columns for your query select the Next> button.



In this section of the Query Wizard you are presented with the opportunity to filter the data being brought into Excel.

Select a field that you wish to filter, in this case ACCOUNT_REF.

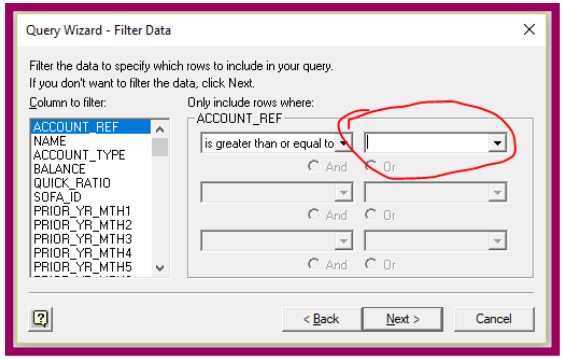


Once you have selected a field to be filtered the 1st 'greyed out' drop-down box becomes available.

Use the drop-down to select the type of filter you wish to apply.

In this example we will choose 'is greater than or equal to'

EgoPetram
Shaping Up Businesses

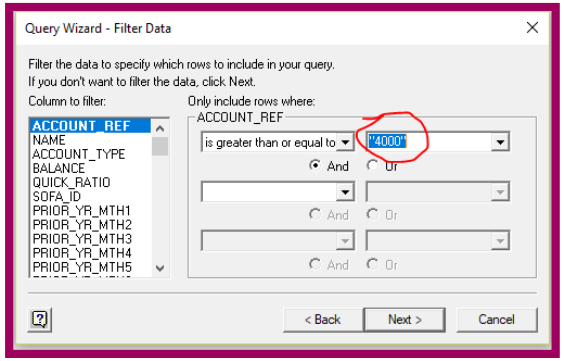


Having chosen the selection method the right hand box ceases to be 'greyed out'.

In this box you enter the value of the criteria you wish to set.

In this case "4000".

Note:
The ACCOUNT_REF is stored as text so the value of the filter must be entered within inverted commas.



Once you have entered the value, the selection method box on the second row ceases to be 'greyed out'

This is so that you can enter another criteria for the SAME field.

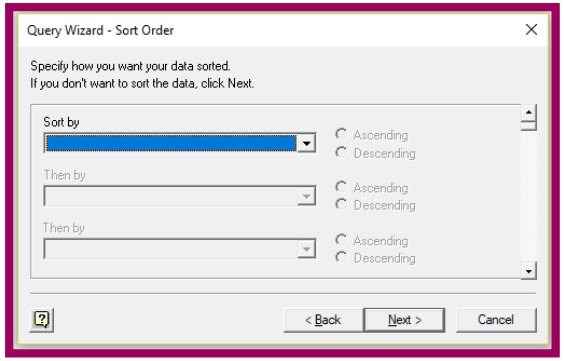
You might for example add that you ant the ACCOUNT_REF to be less than or equal to "5000" as well as greater than or equal to "4000"

Once you have finished adding criteria click the Next > button

Note:
You can choose to filter on more than one field. Having added criteria to one field, select another column to filter and repeat the process of adding criteria.

*Each column for which a criteria has been set will appear in **bold** in the 'Column to Filter' area.*

EgoPetram
Shaping Up Businesses

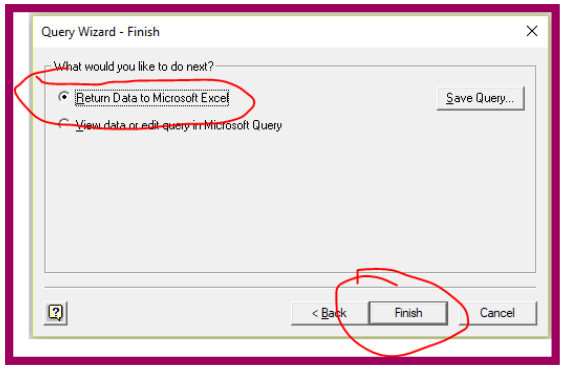


Once you have added any criteria that you require, the Query Wizard offers the opportunity to sort the returned records.

Use the drop-down to choose the field to be sorted and the 'Ascending' and 'Descending' buttons to identify the order of the sort.

In this example, the fields are being left in the order they are reported by Sage.

Select Next > when you have set the sorts you require.

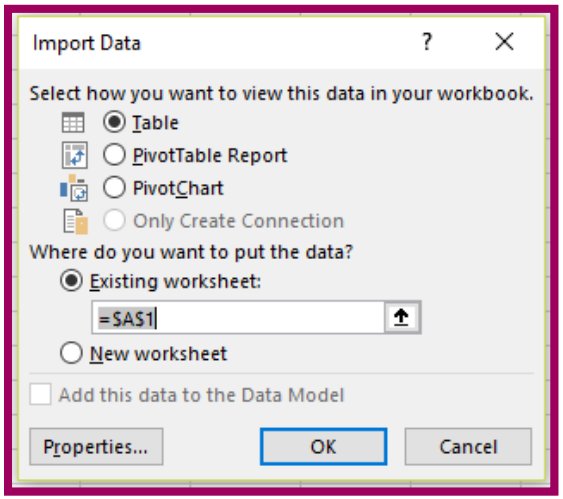


You have now created a Query.

Until you are familiar with the basics of linking data to Sage using the Query Wizard it is best to leave the 'Return Data to Microsoft Excel' button selected.

Select Finish

EgoPetram
Shaping Up Businesses



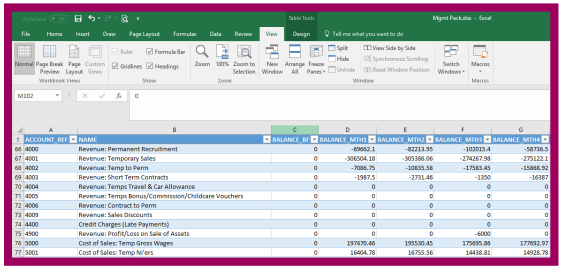
Having finished the Query design, Excel asks what you would like to do with the data.

The options allow you to select:

- Table – a list of the records
- PivotTable – allowing you to summarise the data
- PivotChart – allowing you to present the summary data as a graph.

The 'Table' option is the simplest and is the option to choose while you are still learning how Sage holds its data.

Select 'OK' to proceed

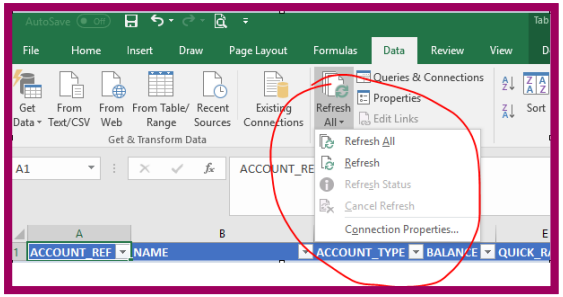


The data is then returned to Excel in the form and at the location you selected.

EgoPetram
Shaping Up Businesses

REFRESHING DATA

Refreshing Data



Data queries do not automatically update with the latest data.

To refresh the data go to the Data menu bar in Excel and select the Refresh option.

By default the option shows ‘Refresh All’. Selecting this option refreshes every data connection and every Pivot Table in the open workbook.

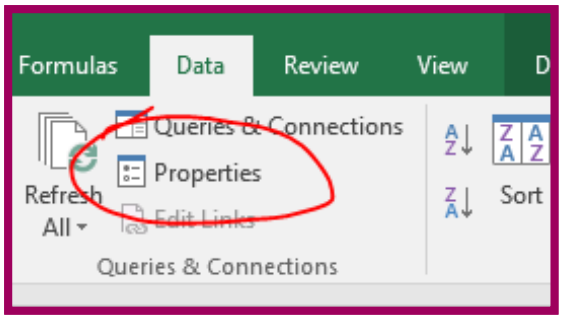
Using the drop-down menu, you are offered the option to just ‘Refresh’. This option only refreshes the currently selected connection.

EgoPetram
Shaping Up Businesses

CONNECTION PROPERTIES

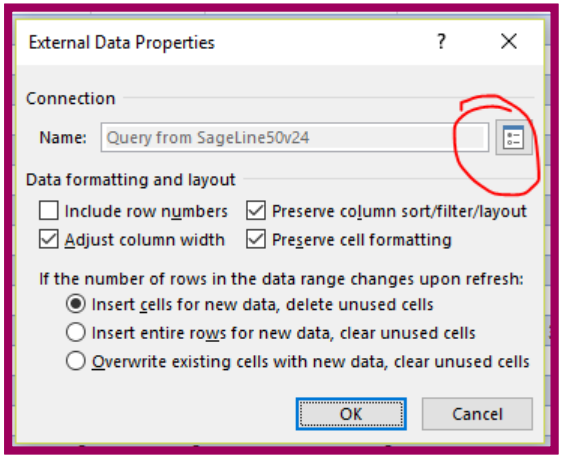
EgoPetram
Shaping Up Businesses

Connection Properties - 1



Once a query has been created, it's properties can be viewed using the 'Properites' option in the 'Queries and Connections' section of the Data menu bar.

This is also the route to take if you wish to edit/amend your query.

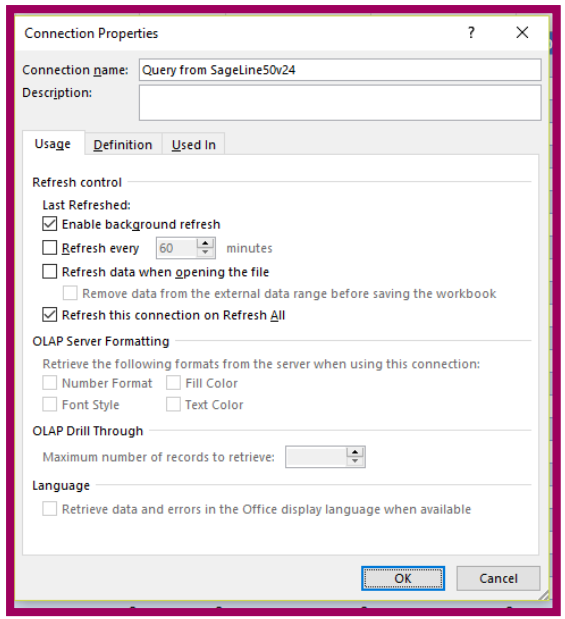


The 'External Data Properites' message box that appears when you select 'Properties' provides some options regarding the formatting and presentation of the data but is not particularly useful.

The interest is in the menu button on the top right of the message box alongside the connection name.

Clicking this gives access to much more detailed information about the Query and offers options to edit/amend the query itself.

EgoPetram
Shaping Up Businesses



The 'Connection Properties' message box has three tabs:

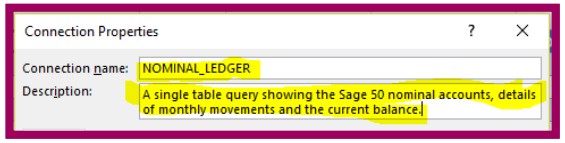
- Usage
- Definition
- Used In

The Usage Tab allows you to decide when the query is refreshed and how often.

The Definition Tab is the most useful.

The Used In tab gives details of all the parts of the spreadsheet in which this connection is used.

Note:
The OLAP (Online Analytical Processing) options are not available with ODBC queries



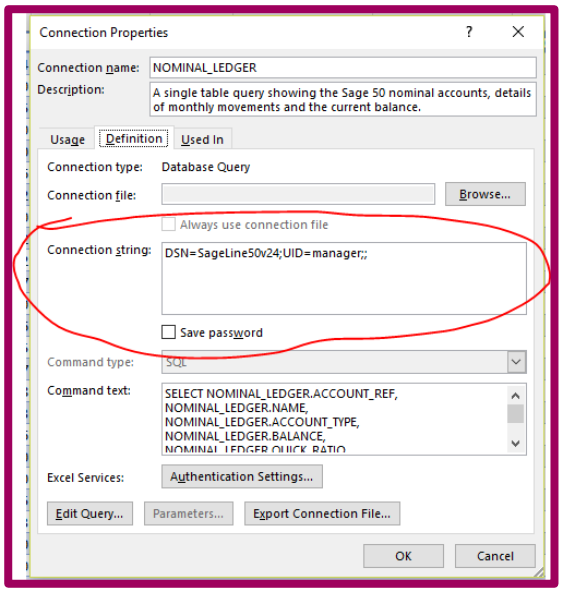
The first thing to note is that the connection name can be changed.

The Query Wizard default is to name a query as 'Query from [●]' where [●] is the datasource name.

It is sensible to change the name to something meaningful. In this case to 'NOMINAL_LEDGER'.

Adding a description can also help.

EgoPetram
Shaping Up Businesses



The Definition tab has two main areas of interest:

- The Connection String; and
- The Command Text

Let's focus first on the Connection String.

Query Wizard defaults to populating the connection string with just enough information to make the query work.

DSN is short for Datasource Name and names the ODBC link being used to connect to sage. In this case SageLine50v24.

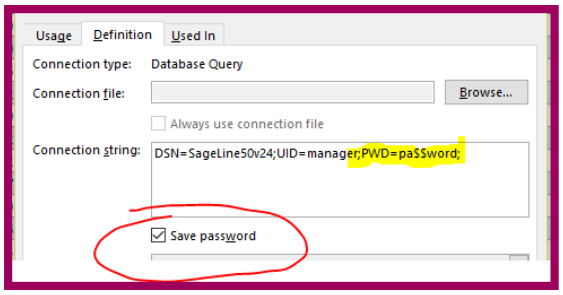
The UID is the userid used when the query was built to access Sage.

Each element of the connection string is separated by a semi-colon. In this example you may notice that the string ends with two semi-colons. This is because the password part of the string has been omitted.

Note:
If Sage 50 is upgraded the DSN name for the ODBC link will change and needs to be changed in all connections.

An alternative is to create a copy ODBC link with a meaningful name (eg Sage50) and then use that link. When Sage is updated you simply need to create a new Sage50 ODBC link as a copy of the latest ODBC driver and all your Excel queries will find the correct driver when the use the DSN=Sage50 connection string.

EgoPetram
Shaping Up Businesses



Not saving the password as part of the query means that each time it is run you are required to login to Sage using a pop-up message box.

This can be irritating, particularly if you are regularly refreshing your data.

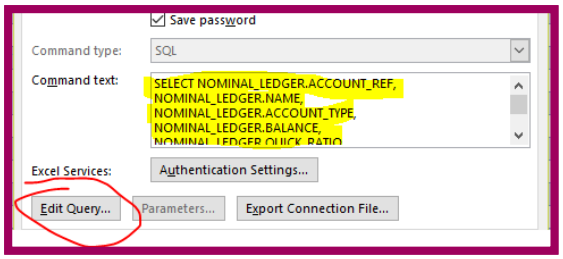
By checking the 'Save Password' box you can save the password with the query.

In this example checking the box has added 'PWD=pa$$word' to the connection string.

Note:
Saving the password as part of the query is not secure as the password becomes visible to anyone who has access to the spreadsheet.

If you check the 'Save Password' box Excel will pop-up a warning about the security. You need to confirm you are OK to proceed for the box to be ticked.

EgoPetram
Shaping Up Businesses



The bottom half of the 'Definition' tab is where the command text is displayed.

The command text is the instruction to Sage50 regarding which fields are to be returned, from which table and matching which criteria.

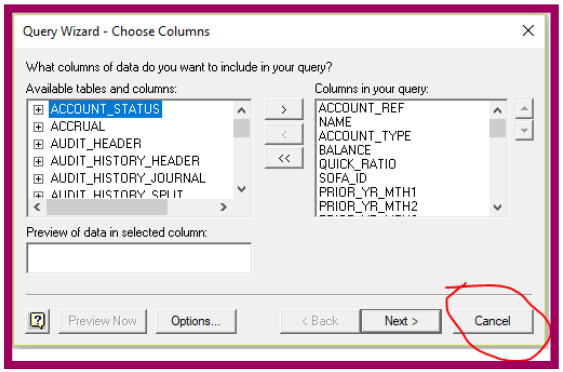
The query uses SQL (Structured Query Language) for the command text.

The format of a simple query is: -

SELECT *tablename.field1, tablename.field2, etc*
FROM *tablename*
[WHERE *tablename.field? = criteria_value*]

The WHERE section is only necessary if the records are being filtered based on one or more criteria.

Fortunately, you do not need to learn SQL to create your queries! Select 'Edit Query' and you will open the Query Wizard tool

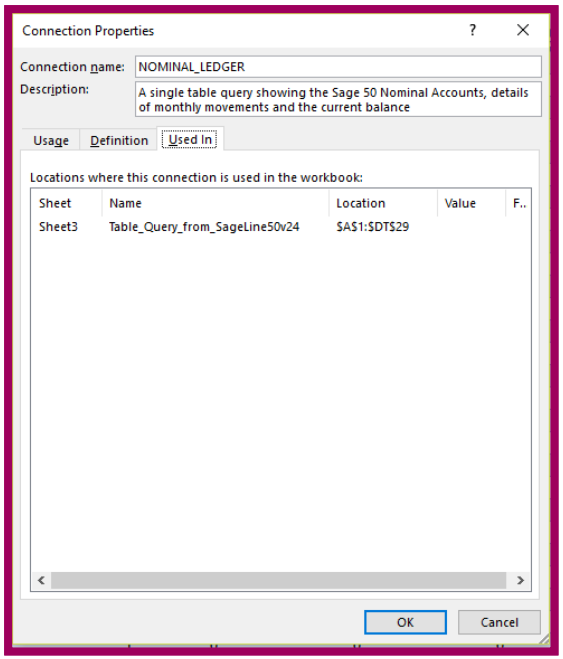


If you select 'Cancel' at this point you will be offered the opportunity to continue editing your query in 'Microsoft Query'

The 'Mircosoft Query' tool is very useful when building more complex queries such as those linking more than one table or with added calculations.

A basic guide to 'Microsoft Query' appears later in this document.

EgoPetram
Shaping Up Businesses



A single connection to Sage 50 can be used in multiple parts of an Excel Workbook.

The places it is used are listed on the 'Used In' tab of the 'Connection Properties' message box.

For example, a single connection might be used to display data:

- In one, or more, tables;
- In one, or more, pivot tables; and
- In one, or more, pivot charts; or
- In any combination of the above.

EgoPetram
Shaping Up Businesses

THE DATABASE

EgoPetram
Shaping Up Businesses

The Database - Overview

Name	Name	Name
COMPANY	GRN_ITEM	CIS_RETURN
PERIOD	GDN_ITEM	CIS_RETURNED_TRANSACTION
SALES_LEDGER	PROJECT	CIS_SETTINGS
PURCHASE_LEDGER	PROJECT_TRAN	CIS_SUBCONTRACTOR
NOMINAL_LEDGER	PROJECT_ONLY_TRAN	CIS_SUBCONTRACTOR_LEGACY
AUDIT_HEADER	PROJECT_STATUS	CIS_SUBCONTRACTOR_TAX_HISTORY
AUDIT_SPLIT	PROJECT_RESOURCE	ESUBMISSION_SETTINGS
AUDIT_HISTORY_HEADER	PROJECT_RESOURCE-TYPE	PROTX_PAYENT
AUDIT_HISTORY_SPLIT	PROJECT_COST_CODE	ECSALES_HEADER
AUDIT_HISTORY_JOURNAL	PROJECT_REVENUE_CODE	ECSALES_SPLIT
AUDIT_HISTORY_USAGE	PROJECCT_COST_TYPE	ECSALES_FILL
AUDIT_JOURNAL	PROJECT_RELATIONSHIP	VAT_DETAILS
AUDIT_VAT	PROJECT_BUDGET	VAT_ADJUSTMENT
AUDIT_USAGE	CLEAR_AUDIT_TRAIL_EVENT	VAT_RETURN_RECEIPT
DEPARTMENT	CLEARED_TRAN_RANGE	VAT_REV_CHARGE_HEADER
BANK	INTRASTAT	VAT_REV_CHARGE_RETURN
STOCK	COUNTRY_CODE	VAT_REV_CHARGE_DETAIL
STOCK_TRAN	DISPUTE_REASON	CALENDAR_EVENT
STOCK_CAT	BANK_STATEMENT	CALENDAR_EVENT_LABEL
STOCK_COMP	BANK_STATEMENT_LINE	CALENDAR_EVENT_LOCATION
INVOICE	FUND	CALENDAR_RECURRENCE_PATTERN
SALES_ORDER	FUND_TYPE	CONTACT_HISTORY
PURCHASE_ORDER	SOFA_CATEGORY	LETTER_TYPE
RECURRING	STOCK_ALLOCATION	COMMUNICATION_ADDRESS
FIXED_ASSET	CREDIT_BUREAU	EC_VAT_DESCRIPTION
FIXED_ASSET_CAT	CHART_LIST	CONTACT_HISTORY_CONTACT_MAJOR_TYPE
PREPAYMENT	CAT_TITLE	CONTACT_HISTORY_CONTACT_TYPE
ACCRUAL	TAX_CODE	CONTACT_HISTORY_CONTACT_OUTCOME
INVOICE_ITEM	UPDATE_LEDGER	DELETED_RECORDS
SOP_ITEM	CURRENCY	PAYMENT_METHOD
POP_ITE	VAT_SUMMARY	SUPPLIER_PAYMENT_INFO
SALES_DEL_ADDR	VAT_RETURN	PAYMENT_INFO
PURCHASE_DEL_ADDR	RTD_SUMMARY	ACCOUNT_STATUS
COMPANY_DEL_ADDR	SAGE_PAY_BY_LINK	DIRECT_DEBIT_PAYMENT_REQUESTS
PRICE	FINANCIAL_BUDGET	REMITTANCE
PRICE_LIST		

The 108 tables that make up the Sage 50 database are listed here.

Fortunately, the table names are self-explanatory.

Even more fortunately, there are only a small number of tables that contain the core financial data and associated information.

These are:

NOMINAL_LEDGER
FINANCIAL_BUDGET
DEPARTMENT
AUDIT_JOURNAL and
AUDIT_HISTORY_JOURNAL

EgoPetram
Shaping Up Businesses

THE DATABASE - NOMINAL_LEDGER

NOMINAL_LEDGER	NOMINAL_LEDGER	NOMINAL_LEDGER	NOMINAL_LEDGER
ACCOUNT_REF	PRIOR_YR3_MTH1	BUDGET_MTH1	DEBIT_BF
NAME	PRIOR_YR3_MTH2	BUDGET_MTH2	DEBIT_MTH1
ACCOUNT_TYPE	PRIOR_YR3_MTH3	BUDGET_MTH3	DEBIT_MTH2
BALANCE	PRIOR_YR3_MTH4	BUDGET_MTH4	DEBIT_MTH3
QUICK_RATIO	PRIOR_YR3_MTH5	BUDGET_MTH5	DEBIT_MTH4
SOFA_ID	PRIOR_YR3_MTH6	BUDGET_MTH6	DEBIT_MTH5
PRIOR_YR_MTH1	PRIOR_YR3_MTH7	BUDGET_MTH7	DEBIT_MTH6
PRIOR_YR_MTH2	PRIOR_YR3_MTH8	BUDGET_MTH8	DEBIT_MTH7
PRIOR_YR_MTH3	PRIOR_YR3_MTH9	BUDGET_MTH9	DEBIT_MTH8
PRIOR_YR_MTH4	PRIOR_YR3_MTH10	BUDGET_MTH10	DEBIT_MTH9
PRIOR_YR_MTH5	PRIOR_YR3_MTH11	BUDGET_MTH11	DEBIT_MTH10
PRIOR_YR_MTH6	PRIOR_YR3_MTH12	BUDGET_MTH12	DEBIT_MTH11
PRIOR_YR_MTH7	PRIOR_YR4_MTH1	BALANCE_BF	DEBIT_MTH12
PRIOR_YR_MTH8	PRIOR_YR4_MTH2	BALANCE_MTH1	DEBIT_FUTURE
PRIOR_YR_MTH9	PRIOR_YR4_MTH3	BALANCE_MTH2	CREDIT_BF
PRIOR_YR_MTH10	PRIOR_YR4_MTH4	BALANCE_MTH3	CREDIT_MTH1
PRIOR_YR_MTH11	PRIOR_YR4_MTH5	BALANCE_MTH4	CREDIT_MTH2
PRIOR_YR_MTH12	PRIOR_YR4_MTH6	BALANCE_MTH5	CREDIT_MTH3
PRIOR_YR2_MTH1	PRIOR_YR4_MTH7	BALANCE_MTH6	CREDIT_MTH4
PRIOR_YR2_MTH2	PRIOR_YR4_MTH8	BALANCE_MTH7	CREDIT_MTH5
PRIOR_YR2_MTH3	PRIOR_YR4_MTH9	BALANCE_MTH8	CREDIT_MTH6
PRIOR_YR2_MTH4	PRIOR_YR4_MTH10	BALANCE_MTH9	CREDIT_MTH7
PRIOR_YR2_MTH5	PRIOR_YR4_MTH11	BALANCE_MTH10	CREDIT_MTH8
PRIOR_YR2_MTH6	PRIOR_YR4_MTH12	BALANCE_MTH11	CREDIT_MTH9
PRIOR_YR2_MTH7	PRIOR_YR5_MTH1	BALANCE_MTH12	CREDIT_MTH10
PRIOR_YR2_MTH8	PRIOR_YR5_MTH2	BALANCE_FUTURE	CREDIT_MTH11
PRIOR_YR2_MTH9	PRIOR_YR5_MTH3		CREDIT_MTH12
PRIOR_YR2_MTH10	PRIOR_YR5_MTH4		CREDIT_FUTURE
PRIOR_YR2_MTH11	PRIOR_YR5_MTH5		INACTIVE_FLAG
PRIOR_YR2_MTH12	PRIOR_YR5_MTH6		RECORD_CREATE_DATE
	PRIOR_YR5_MTH7		RECORD_MODIFY_DATE
	PRIOR_YR5_MTH8		RECORD_DELETED
	PRIOR_YR5_MTH9		
	PRIOR_YR5_MTH10		
	PRIOR_YR5_MTH11		
	PRIOR_YR5_MTH12		

The NOMINAL_LEDGER table lists all the nominal ledger codes in the database together with summary information regarding the **movement** in each balance for:

1. Each month of the Current Year (BALANCE_MTHx)
2. Each month of each of the previous 5 years; and
3. The Budget

It also gives the current year figures split between DEBIT and CREDIT movements.

It is important to note that the figures for each period are the movement in the balance during that period and not the balance at the end of the period.

The control total for each column is £0 (Zero).

To arrive at the period end balance you must add the BALANCE_BF field and each monthly movement up to the period you require. For example the closing balance for Period three is BALANCE_BF + BALANCE_MTH1 + BALANCE_MTH2 + BALANCE_MTH3.

There is only one line of data per Nominal Code.

Note:
Sage retains details of deleted accounts. It is sometimes necessary to filter a query on the RECORD_DELETED entry to avoid including such accounts.

EgoPetram
Shaping Up Businesses

The Database - Financial_Budget & Department

FINANCIAL_BUDGET
ACCOUNT_REF
ANALYSIS_ID
YEAR
BUDGET
PERIOD
ACTUAL
CLEARED_ACTUAL
CATEGORY
RECORD_CREATE_DATE
RECORD_MODIFY_DATE
RECORD_DELETED

The FINANCIAL_BUDGET table stores the actual and budget movements for each department for each period.

The total of the ACTUAL field for a given ACCOUNT_REF (nominal code) in a given period in a given year will equal the corresponding period movement for the matching NOMINAL_CODE in the NOMINAL_LEDGER table

Similarly the total of the BUDGET field for a given ACCOUNT_REF in the current year will total the corresponding budget movement for the matching NOMINAL_CODE in the NOMINAL_LEDGER table.

DEPARTMENT
RECORD_CREATE_DATE
RECORD_MODIFY_DATE
RECORD_DELETED
NUMBER
NAME
REFERENCE

The DEPARTMENT table Lists all the department names.

The NUMBER field can be cross referenced to the ANALYSIS_ID field in the FINANCIAL_BUDGET table and the DEPT_NUMBER field in the AUDIT_JOURNAL and AUDIT_JOURNAL_HISTORY tables.

The Database - Audit_Journal

AUDIT_JOURNAL	AUDIT_JOURNAL_HISTORY
TRAN_NUMBER	TRAN_NUMBER
TYPE	TYPE
DATE	DATE
ACCOUNT_REF	USER_NAME
BANK_CODE	DETAILS
INV_REF	DISPUTED
USER_NAME	PAID_FLAG
DETAILS	PAID_STATUS
DISPUTED	DELETED_FLAG
PAID_FLAG	HEADER_NUMBER
PAID_STATUS	DATE_ENTERED
DELETED_FLAG	RECORD_CREATE_DATE
DATE_FLAG	RECORD_MODIFY_DATE
BANK_FLAG	RECORD_DELETED
HEADER_NUMBER	NOMINAL_CODE
DATE_ENTERED	EXTRA_REF
RECORD_CREATE_DATE	VAT_FLAG
RECORD_MODIFY_DATE	DEPT_NUMBER
RECORD_DELETED	DEPT_NAME
NOMINAL_CODE	TAX_CODE
EXTRA_REF	AMOUNT
STMT_TEXT	FOREIGN_AMOUNT
VAT_FLAG	SPLIT_NUMBER
DEPT_NUMBER	VAT_FLAG_CODE
DEPT_NAME	VAT_RECONCILED_DATE
TAX_CODE	
AMOUNT	
FOREIGN_AMOUNT	
RTD_FLAG	
SPLIT_NUMBER	
VAT_FLAG_CODE	
VAT_RECONCILED_DATE	
DISPUTE_CODE	
FUND_ID	
VAT_LEDGER_RETURN_ID	

The AUDIT_JOURNAL and AUDIT_JOURNAL_HISTORY tables both contain the transaction data posted to Sage.

The AUDIT_JOURNAL table contains current data and the AUDIT_JOURNAL_HISTORY table contains details of all entries that have been 'Archived' when the Sage 50 Archive program has been run. If your data hasn't been archived the AUDIT_JOURNAL_HISTORY table will be empty.

For a given NOMINAL_CODE in a given period the total of the transactions will equal the movement recorded in the NOMINAL_LEDGER table for that month.

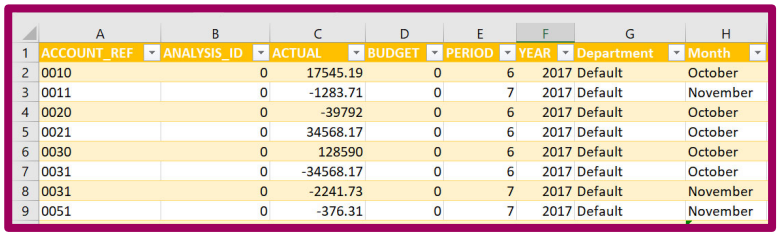
Pivot tables created from these tables can provide a useful 'drill-down' into a balance from Excel.

Given the volume of transactions that might exist in such a query, it can be better to rely on the summarising already done by Sage 50 and use the NOMINAL_LEDGER and FINANCIAL_BUDGET tables

Note:
Sage retains details of deleted accounts. It is sometimes necessary to filter a query on the RECORD_DELETED entry to avoid including such accounts.

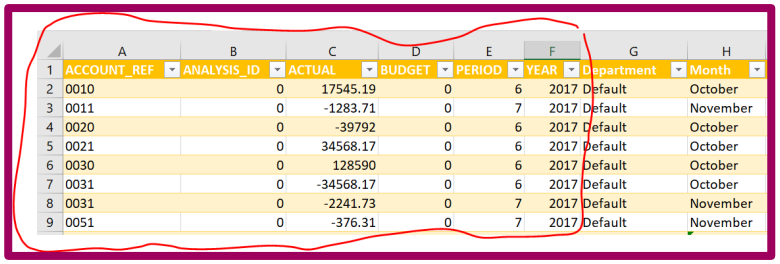
EgoPetram
Shaping Up Businesses

USING THE DATA



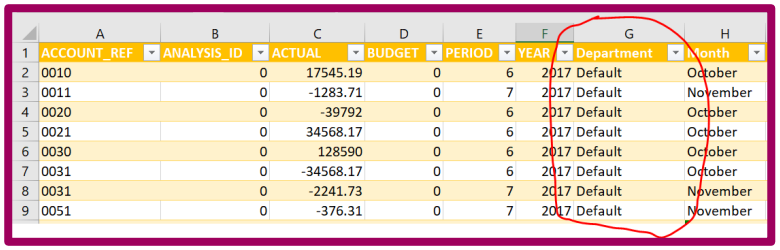
	A	B	C	D	E	F	G	H
1	ACCOUNT_REF	ANALYSIS_ID	ACTUAL	BUDGET	PERIOD	YEAR	Department	Month
2	0010		0	17545.19	0	6	2017 Default	October
3	0011		0	-1283.71	0	7	2017 Default	November
4	0020		0	-39792	0	6	2017 Default	October
5	0021		0	34568.17	0	6	2017 Default	October
6	0030		0	128590	0	6	2017 Default	October
7	0031		0	-34568.17	0	6	2017 Default	October
8	0031		0	-2241.73	0	7	2017 Default	November
9	0051		0	-376.31	0	7	2017 Default	November

Excel will automatically extend your data table if you add formulae in columns adjacent to the output of your Sage download.



	A	B	C	D	E	F	G	H
1	ACCOUNT_REF	ANALYSIS_ID	ACTUAL	BUDGET	PERIOD	YEAR	Department	Month
2	0010		0	17545.19	0	6	2017 Default	October
3	0011		0	-1283.71	0	7	2017 Default	November
4	0020		0	-39792	0	6	2017 Default	October
5	0021		0	34568.17	0	6	2017 Default	October
6	0030		0	128590	0	6	2017 Default	October
7	0031		0	-34568.17	0	6	2017 Default	October
8	0031		0	-2241.73	0	7	2017 Default	November
9	0051		0	-376.31	0	7	2017 Default	November

In this example, Columns A to F contain data from a connection to the Sage FINANCIAL_BUDGET table.



	A	B	C	D	E	F	G	H
1	ACCOUNT_REF	ANALYSIS_ID	ACTUAL	BUDGET	PERIOD	YEAR	Department	Month
2	0010		0	17545.19	0	6	2017 Default	October
3	0011		0	-1283.71	0	7	2017 Default	November
4	0020		0	-39792	0	6	2017 Default	October
5	0021		0	34568.17	0	6	2017 Default	October
6	0030		0	128590	0	6	2017 Default	October
7	0031		0	-34568.17	0	6	2017 Default	October
8	0031		0	-2241.73	0	7	2017 Default	November
9	0051		0	-376.31	0	7	2017 Default	November

Column G contains the formula:

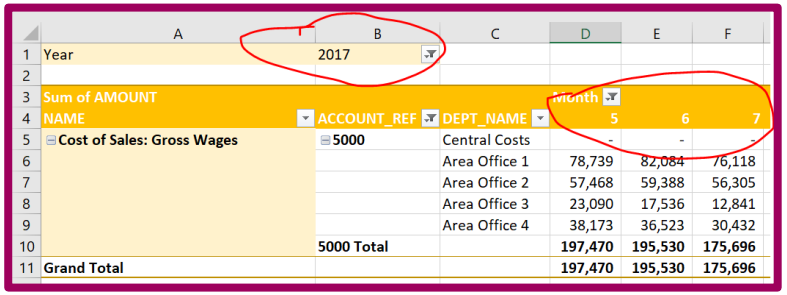
=VLOOKUP([@[ANALYSIS_ID]],Dept_List,2,0)

It looks up the row value of the ANALYSIS_ID field (column B) in a separate table in the workbook called 'Dept_List' and reports back the value in column 3 (ie 2 columns over) of that table.

Column H contains a similar formula

Where tables have been extended in this way, Excel will automatically extend the formulae to apply to all records in the dataset download from Sage. When 'Refresh Data' is used to refresh the connection the formulae in Columns G and H (in this case) will be copied to new rows automatically.

Note:
In the above example the table 'Dept_List' is actually a table arising from a connection to Sage where the Connection name has been changed from 'Query from SageLine50v23' to 'Dept_List'. This allows the lookup to be more meaningful.



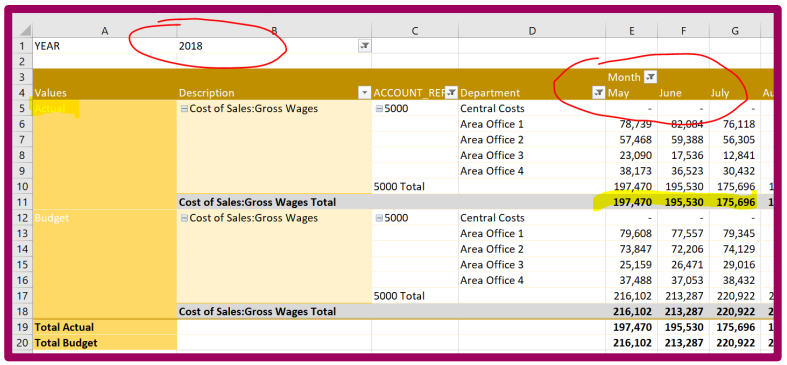
	A	B	C	D	E	F
1	Year	2017				
2						
3	Sum of AMOUNT			Month		
4	NAME	ACCOUNT_REF	DEPT_NAME	5	6	7
5	Cost of Sales: Gross Wages	5000	Central Costs	-	-	-
6			Area Office 1	78,739	82,084	76,118
7			Area Office 2	57,468	59,388	56,305
8			Area Office 3	23,090	17,536	12,841
9			Area Office 4	38,173	36,523	30,432
10		5000 Total		197,470	195,530	175,696
11	Grand Total			197,470	195,530	175,696

Summarising the data using a Pivot Table can be a quick and easy way of accessing the data you are looking for.

This example is a summary of the AUDIT_JOURNAL table for ACCOUNT_REF (nominal code) '5000' by DEPT_NAME.

It is filtered to show a single year 2017 (as circled) and shows the calendar months 5 to 7 (ie May to July – as circled)

Drilling down on this table will reveal the transactions making up the relevant balance.



	A	B	C	D	E	F	G
1	YEAR	2018					
2							
3					Month		
4	Values	Description	ACCOUNT_REF	Department	May	June	July
5	Actual	Cost of Sales:Gross Wages	5000	Central Costs	-	-	-
6				Area Office 1	78,739	82,084	76,118
7				Area Office 2	57,468	59,388	56,305
8				Area Office 3	23,090	17,536	12,841
9				Area Office 4	38,173	36,523	30,432
10			5000 Total		197,470	195,530	175,696
11		Cost of Sales:Gross Wages Total			197,470	195,530	175,696
12	Budget	Cost of Sales:Gross Wages	5000	Central Costs	-	-	-
13				Area Office 1	79,608	77,557	79,345
14				Area Office 2	73,847	72,206	74,129
15				Area Office 3	25,159	26,471	29,016
16				Area Office 4	37,488	37,053	38,432
17			5000 Total		216,102	213,287	220,922
18		Cost of Sales:Gross Wages Total			216,102	213,287	220,922
19	Total Actual				197,470	195,530	175,696
20	Total Budget				216,102	213,287	220,922

The same report can also be achieved using a Pivot Table of the FINANCIAL_BUDGET table.

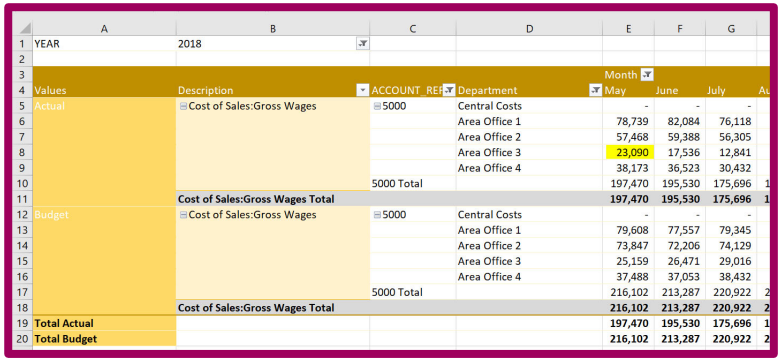
In this case the Actual figures can be shown alongside the Budget numbers as both are in the same table.

Drilling down on this table will only reveal one line per entry.

Notes:
In the top Pivot Table the period data is derived from the transaction date so period 5, 2017 is May 2017.

In the lower Pivot Table the period data is derived from the periods in the Financial Year. In this case May 2017 is the first period of FY2018. You will see that the highlighted balances in the lower table are the same for each period but the circled year and period references are different.

EgoPetram
Shaping Up Businesses



	A	B	C	D	E	F	G
1	YEAR	2018					
2							
3					Month		
4	Values	Description	ACCOUNT_REF	Department	May	June	July
5	Actual	Cost of Sales:Gross Wages	5000	Central Costs	-	-	-
6				Area Office 1	78,739	82,084	76,118
7				Area Office 2	57,468	59,388	56,305
8				Area Office 3	23,090	17,536	12,841
9				Area Office 4	38,173	36,523	30,432
10			5000 Total		197,470	195,530	175,696
11		Cost of Sales:Gross Wages Total			197,470	195,530	175,696
12	Budget	Cost of Sales:Gross Wages	5000	Central Costs	-	-	-
13				Area Office 1	79,608	77,557	79,345
14				Area Office 2	73,847	72,206	74,129
15				Area Office 3	25,159	26,471	29,016
16				Area Office 4	37,488	37,053	38,432
17			5000 Total		216,102	213,287	220,922
18		Cost of Sales:Gross Wages Total			216,102	213,287	220,922
19	Total Actual				197,470	195,530	175,696
20	Total Budget				216,102	213,287	220,922

GETPIVOTDATA() is a useful formula for extracting data so that it can be reported elsewhere.

In this example, the Pivot Table starts in cell A3 so the formula to return the highlighted number is:

=GETPIVOTDATA("ACTUAL.",A3,"ACCOUNT_REF","5000", "Month","May","Department","Area Office 3","Nominal Account","Cost of Sales: Gross Wages")

Variables can be used to replace the Item values; a valuable approach when copying a formula into multiple cells.

In this example the Item values are "5000", "1", "Temp Division: Marlow" and "Cost of Sales: Temp Gross Wages"

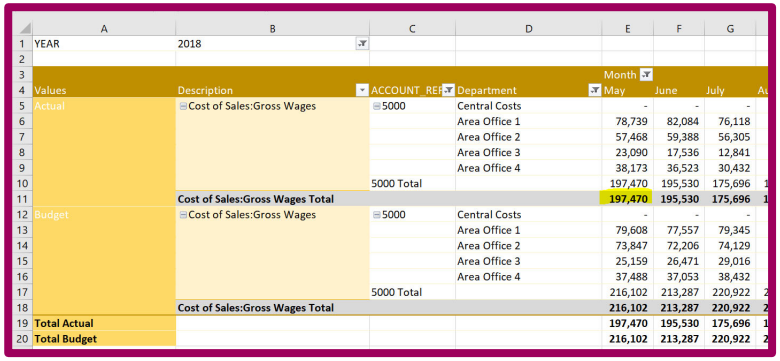
Creating an entire pivot table might be an unnecessary use of memory and create an overly large spreadsheet.

SUMIFS() is a useful alternative to creating a Pivot Table and offers more flexibility when combined with other functions.

In the above example, the formula

=SUMIFS(Dept_Data[ACTUAL],Dept_Data[Month],"May",Dept_Data[YEAR],2018,Dept_Data[ACCOUNT_REF],5000,Dept_Data[Department],"Area Office 3")

Returns the same value of £23,090 without use of a Pivot Table. Instead the formula directly summarises the "Dept_Data" table from which the Pivot Table was created. The "Dept_Data" table is the name given in this example to the connection to the FINANCIAL_BUDGET table in Sage.



	A	B	C	D	E	F	G
1	YEAR	2018					
2							
3					Month		
4	Values	Description	ACCOUNT_REF	Department	May	June	July
5	Actual	Cost of Sales:Gross Wages	5000	Central Costs	-	-	-
6				Area Office 1	78,739	82,084	76,118
7				Area Office 2	57,468	59,388	56,305
8				Area Office 3	23,090	17,536	12,841
9				Area Office 4	38,173	36,523	30,432
10			5000 Total		197,470	195,530	175,696
11		**Cost of Sales:Gross Wages Total**			**197,470**	**195,530**	**175,696**
12	Budget	Cost of Sales:Gross Wages	5000	Central Costs	-	-	-
13				Area Office 1	79,608	77,557	79,345
14				Area Office 2	73,847	72,206	74,129
15				Area Office 3	25,159	26,471	29,016
16				Area Office 4	37,488	37,053	38,432
17			5000 Total		216,102	213,287	220,922
18		**Cost of Sales:Gross Wages Total**			**216,102**	**213,287**	**220,922**
19	**Total Actual**				**197,470**	**195,530**	**175,696**
20	**Total Budget**				**216,102**	**213,287**	**220,922**

As the total movement for period 1 also appears in the BALANCE_MTH1 column of the NOMINAL_LEDGER table the monthly movement can be returned without reference to either the FINANCIAL_BUDGET table nor the AUDIT_JOURNAL table.

A common way of achieving this is to use the VLOOKUP() function.

In this case =VLOOKUP("5000",Nominal_Data,4,0) as the BALANCE_MTH1 field is in the 4h column of the "Nominal_Data" table.

The OFFSET() function provides a useful, more flexible alternative to the VLOOKUP() function. Particularly when combined with the MATCH() function.

In this example:
=MATCH("5000",Nominal_Data[ACCOUNT_REF],0) identifies "5000" as being the 75th entry in the ACCOUNT_REF field, and
=MATCH("BALANCE_MTH1",Nominal_Data[#Headers],0) identifies "BALANCE_MTH1" as being the 4th Header

The OFFSET() function uses a base cell reference of (0,0) and so, allowing 1 row for the headers

=OFFSET('Nominal Data'!A2,74,3)

Returns the same £197,470 value from as is totalled by the pivot table for period 1.

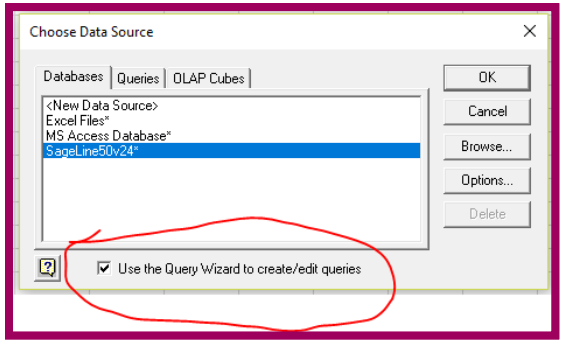
The advantage of the OFFSET() formula is it allows for the total of a range of values to be returned.

=SUM(OFFSET('Nominal Data'!A2,74,3):OFFSET('Nominal Data'!A2,74,5) in this example returns the total of £568,696 that is the total of the movements for the first quarter.

EgoPetram
Shaping Up Businesses

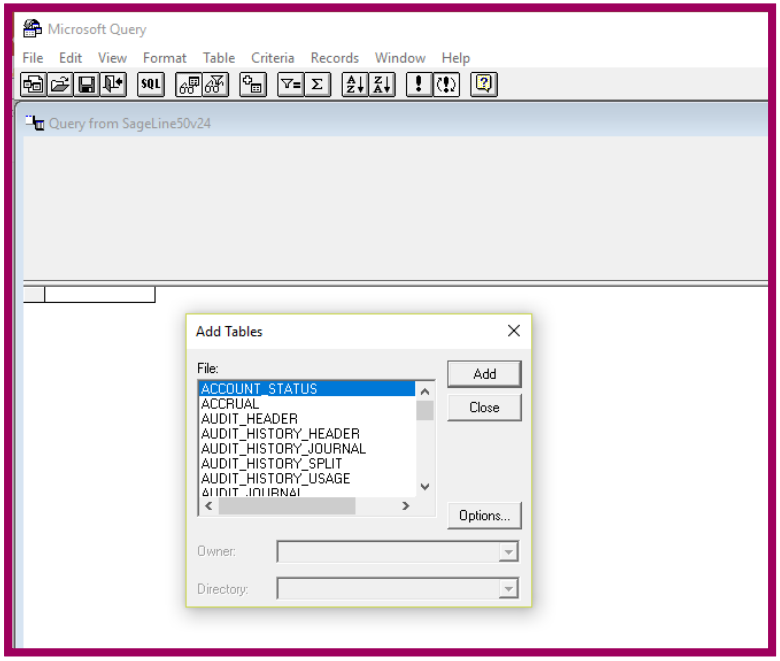
MICROSOFT QUERY

EgoPetram
Shaping Up Businesses



When you become more familiar with the Sage 50 database and the Query Wizard, more useful connections can be created using the Microsoft Query Editor.

To access this you to untick the box that launches the Query Wizard when the "Choose Data Source" message box opens.

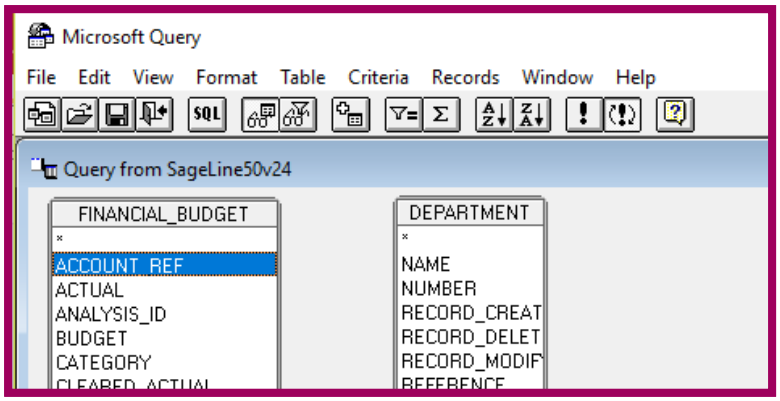


Once the Data Source has been selected, the Microsoft Query window opens and offers a message box permitting selection of the table(s) for inclusion in the query.

This is the best route to building a query with based on multiple tables as it allows the flexibility to choose how the tables are linked.

In this example, the FINANCIAL_BUDGET and the DEPARTMENT tables are to be selected and added to the query.

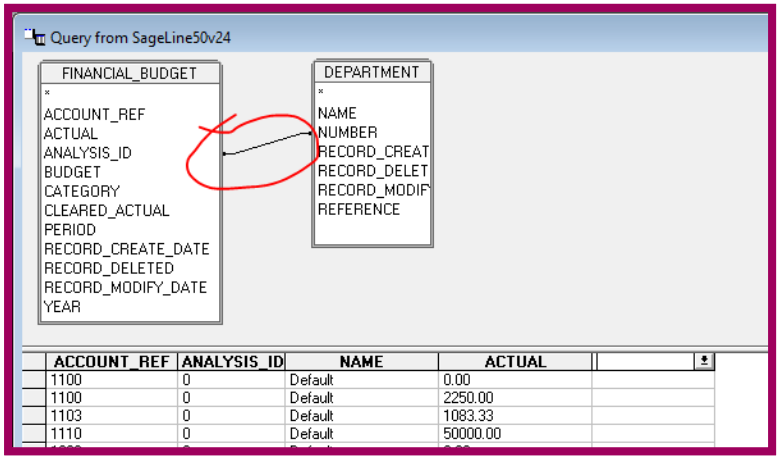
EgoPetram
Shaping Up Businesses



Once you have added the tables you require you need to link them by a common data field.

In this case the ANALYSIS_ID field in the FINANCIAL_BUDGET table cross references to the NUMBER field I the DEPARTMENT table.

The link is created by clicking on the field in one table and dragging it to the cross reference field in the other table.



This creates a join between the two tables.

The basic join matches each entry in one table with one entry in the other on a ONE:ONE basis.

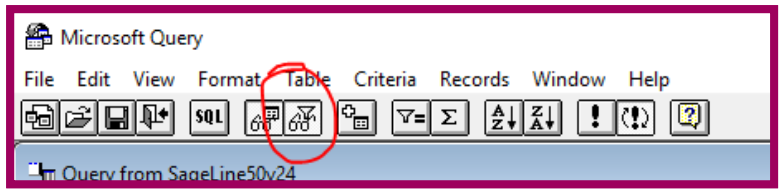
There are occasions when a MANY:ONE or a ONE:MANY match is required. Right click on the line to change the match selected.

Initially no columns are selected in the Query Editor.

Columns can be selected by double clicking the field in the table list or by dragging it to the relevant column in the bottom section of the screen.

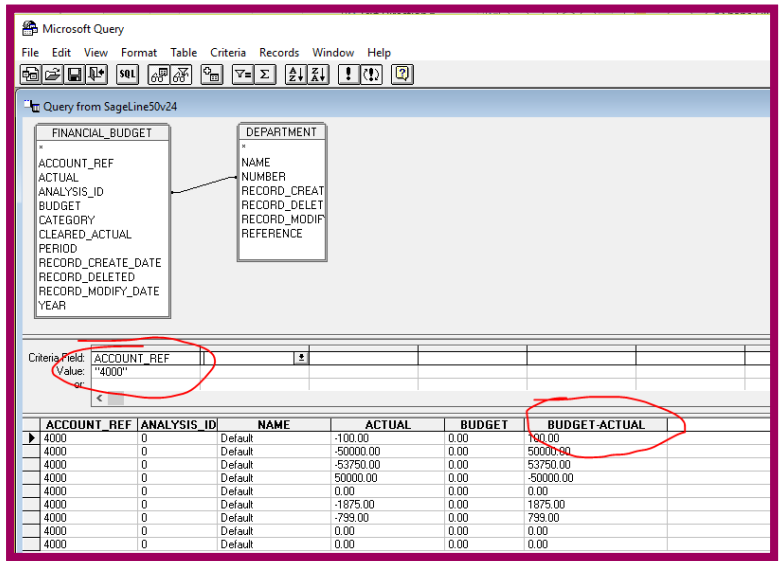
Columns will appear in excel in the order in which they appear when the query is first designed. They can be moved by dragging them to a new position.

EgoPetram
Shaping Up Businesses



To add filters to your query you must click the 'Show Criteria' button on the menu bar.

This reveals a Criteria panel into which selection criteria can be added.



The criterium that has been added in this example is a simple filter on ACCOUNT_REF to show only records with a nominal code of "4000".

More complex criteria added using Boolean logic as you become familiar with the tool.

Notice that this example also includes a calculated field. This has been achieved by typing BUDGET-ACTUAL into a blank column of the data section at the bottom of the window.

Notes:
The nominal codes in Sage 50 are stored as text fields. Remember to enter any values in inverted commas (eg "4000").

In this example the BUDGET and ACTUAL field names only appear in the FINANCIAL_BUDGET table and so there is no conflict arising when the field names are entered into a formula as here. If the names appear in multiple tables the formula must include the table name as part of the field description – the format is tablename.fieldname. In this example that would be FINANCIAL_BUDGET.BUDGET and FINANCIAL_BUDGET.ACTUAL

ONE FINAL POINT

EgoPetram
Shaping Up Businesses

One Final Point

This guide is intended as a basic introduction to linking Excel to Sage 50; its purpose is to provide sufficient insight to get started.

The pointers on how to use the data, particularly the suggested Excel formulae are just a few of the many, many options available. The suggestions here are just suggestions, but are one's we have found very useful over the years.

The introduction to Microsoft Query presented here is intentionally basic; it only touches on the available functionality of the tool. It is, for example, possible to add variable criteria and write macros to automatically change queries but...

It is very easy to forget that often the next user of your spreadsheet may not have the knowledge to edit your query or understand your macro.

For this reason we recommend that you keep your queries simple so that they are unlikely to require editing or amending in the future.

We also recommend that you reserve a sheet in your workbook to document its structure and the key formulae you have used. That way others will be able to use your spreadsheet and change it if required without having to spend long hours understanding how it works first.