

# Training Guide

## Web Intelligence Advanced Queries



### **Appropriate Use and Security of Confidential and Sensitive Information**

Due to the integrated nature of the various Human Resources, Finance, and Student modules in Banner and the reporting information in the Enterprise Data Warehouse (EDW), you may have access to information beyond what you need to perform your assigned duties. Your access to Banner and the EDW has been granted based on business need, and it is your responsibility to ensure the information you access is used appropriately.

Here are some reminders of good data stewardship to help you carry out your responsibility:

- Do not share your passwords or store them in an unsecured manner. Do not leave your workstation unattended while logged on to administrative information systems. You are responsible for any activity that occurs using your logon id.
- Do not share confidential and sensitive information with anyone, including colleagues, unless there is a business reason.
- Retrieve printed reports quickly, and do not leave the reports lying around in plain view.
- Secure reports containing confidential and sensitive information (e.g., FERPA, EEO or HIPAA protected data).
- When disposing of reports containing confidential or sensitive information, shred the documents in a timely manner.

Your responsibilities regarding the protection and security of administrative information are outlined in the University of Illinois Information Security Policy posted at <https://www.obfs.uillinois.edu/bfpp/section-19-business-systems-access-security/section-19-5>. Any violation could subject you to disciplinary action, which could include dismissal or, in those cases where laws have been broken, legal action. You should have signed a compliance form that indicates you have read, understand and agree to comply with the University's Information Security Policy for Administrative Information. If you have not already signed the compliance form, please see your Unit Security Contact, who is responsible for maintaining these forms.

---

# Web Intelligence Advanced Queries

## Table of Contents

<b>About this Course .....</b>	<b>4</b>
<b>Chapter 1: Merge Dimensions .....</b>	<b>5</b>
Examples of Merged Dimensions.....	5
Opening the Initial Query.....	6
Add Second Query.....	7
Merge Dimensions – Joining Data from Two Sources.....	9
Creating Variables.....	12
Adding the Variables to the Report Table.....	14
<b>Chapter 2: Using Excel as a Data Source.....</b>	<b>17</b>
Building a Query from Excel.....	17
Using Merge Dimensions to Add Data from Excel to Existing Reports.....	20
Create a Variable to Add Data .....	22
<b>Chapter 3: Advanced Query Filters.....</b>	<b>25</b>
Using Matches Pattern Operator .....	25
Filter on Results from another Query .....	29
<b>Appendix A – Additional Assistance .....</b>	<b>31</b>
<b>Appendix B – Query Filter Operators.....</b>	<b>32</b>
<b>Appendix C – Query Filter Operand Options .....</b>	<b>34</b>

# About this Course

## Objective

The objective of this course is to teach how to use a variety of advanced querying techniques and features including: Combining data from multiple universes using Merged Dimensions, using Excel as a data source, filtering based on results from another query, and using wildcards in query filters.

## Prerequisites

In order to complete this course, you should have a good understanding of Web Intelligence Reporting Basics, including creating a query, building query filters, and basic report formatting.

## Instructor Led Course

This course is presented in a computer lab with an instructor. The instructor presents the information by completing the examples in each chapter on the screen. The students in the class follow by doing the same steps as the instructor.

## Training Data

This course is not designed to teach about the Data, rather the Web Intelligence Reporting software used to create reports. We will use the **Training – Operating Ledger** and the **Training – CFOAP** universes for all the examples. These universes were created for training purposes only and contain fictitious financial data. Also note that these universes are not organized the same as actual finance universes.

## Training Accounts

To access the training data, we will login using training accounts. These accounts are used in training, and can be used outside of training to practice our training exercises. The account user names are **adhoc1** through **adhoc19**, and the all have the same password: **DStrain1ng**

---

# Chapter 1: Merge Dimensions

The process of linking data from different data providers based on a common object is referred to as Merging Dimensions. This process is broken down into several steps: creating the queries, merging dimensions, creating variables, and adding the variables to the report table.

## Examples of Merged Dimensions

Our training example is a fictitious example of using Merged Dimensions. Below are some common uses for Merged Dimensions we see often at the university:

### Finance Data Examples

- Reporting on operating ledger data with purchase orders.  
Sources: **EDW – Finance Purch Order Invoices** with **Operating Ledger Reporting**
- Reporting on operating ledger data with TEM data.  
Sources: **EDW – Finance Travel & Expense** with **EDW – Finance Posted Transactions**

### Student Data Examples

- Reporting on all currently registered students with their completed courses and grades.  
Sources: currently registered students are found in **EDW – STU Registration Limited (or complete)** and completed courses and grades are found in **EDW – STU Academic Records**.
- Reporting on all currently registered students with their most recent Term GPA.  
Sources: currently registered students are found in **EDW – STU Registration Limited (or complete)** and Term GPA is found in **EDW – STU Academic Records**.
- Reporting on applicants that have enrolled to a specific program.  
Sources: applicants are found in the **EDW – Current Applicant** universe and enrolled students are found in **EDW – STU Registration Limited (or complete)**

### HR Data Examples

- Reporting on job CFOAP information combined with years of service.  
Sources: Job CFOAP information is found in **EDW – HR – Position and Job Hierarchy** and years of service is found in the **EDW – HR – Employee Administration**.

## Open the Initial Document

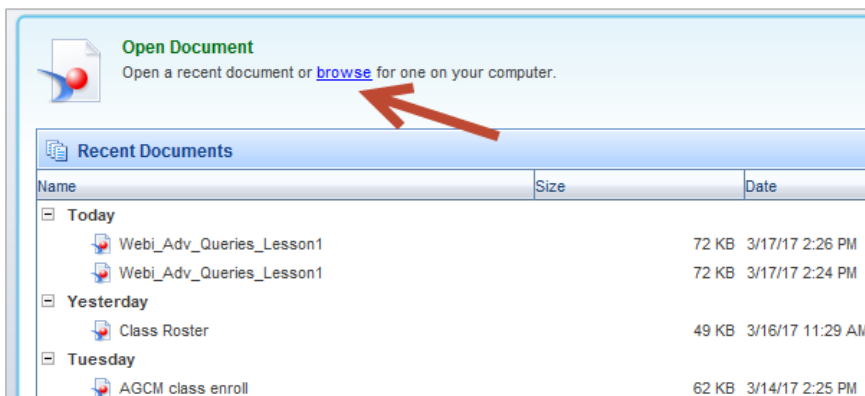
We'll start with an existing document created out of the **Training – Operating Ledger** universe. This document has a single query.

1. Open Web Intelligence Rich Client.
2. Login with your assigned training account.

User Name: **Adhoc (1-19)**

Password: **DStrain1ng**

3. Under Open a Document, click **Browse** link.



4. Open the document **Webi\_Adv\_Queries\_Lesson1** document.

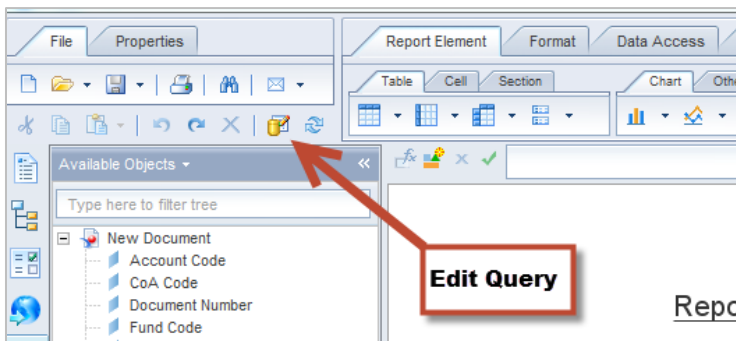
Training Transactions							
Transaction Date	Document Number	Transaction Description	CoA Code	Fund Code	Organization Code	Program Code	Expense Amount
7/1/14	AR034267	Fox-Atkins Development LLC	9	100014	699001	699006	\$0.00
7/1/14	AR034463	ER00656997 Fox-Atkins Development L	9	200201	699001	699006	(\$210.00)
7/1/14	AR034797	HR Payroll 2014 MN 7 1	9	100014	699001	699006	\$0.00
7/1/14	AR034797	HR Payroll 2014 MN 7 1	9	100015	699001	699006	\$0.00
7/1/14	AR034797	HR Payroll 2014 MN 7 2	9	100015	699001	699006	\$0.00
7/1/14	AR034797	HR Payroll 2014 MN 7 2	9	301006	699001	699012	\$0.00
7/1/14	AR034797	HR Payroll 2014 MN 7 5	9	100014	699001	699006	\$0.00
7/1/14	AR034797	HR Payroll 2014 MN 7 6	9	301006	699001	699012	\$0.00
7/1/14	AR035234	Reverse deferral of PR007189	9	100014	699001	699006	\$6,144.98

## Add Second Query

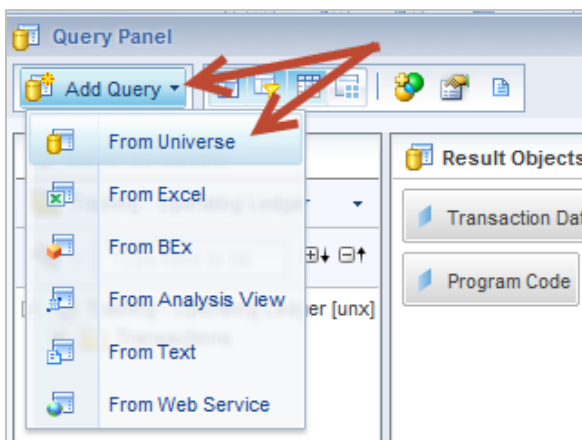
The **Training - Operating Ledger** universe contains only codes for the Fund, Org, Account, and Program information. In order to add the descriptions for these codes, we will add a query from the **Training – CFOAP** universe. *(Note this is a training example only. In our real Operating Ledger Universe, you would find both the codes and their descriptions.)*

When building your second query, always include at least one result object that is also found in your first query. You will need a common dimension object to join the data together in your reports. Often this will be a UIN, or a Transaction Number. In our example, we will join our data together with the Fund Code, CoA Code, and Organization Code objects.

1. Click the **Edit Query** button.



2. Click **Add Query > From Universe**



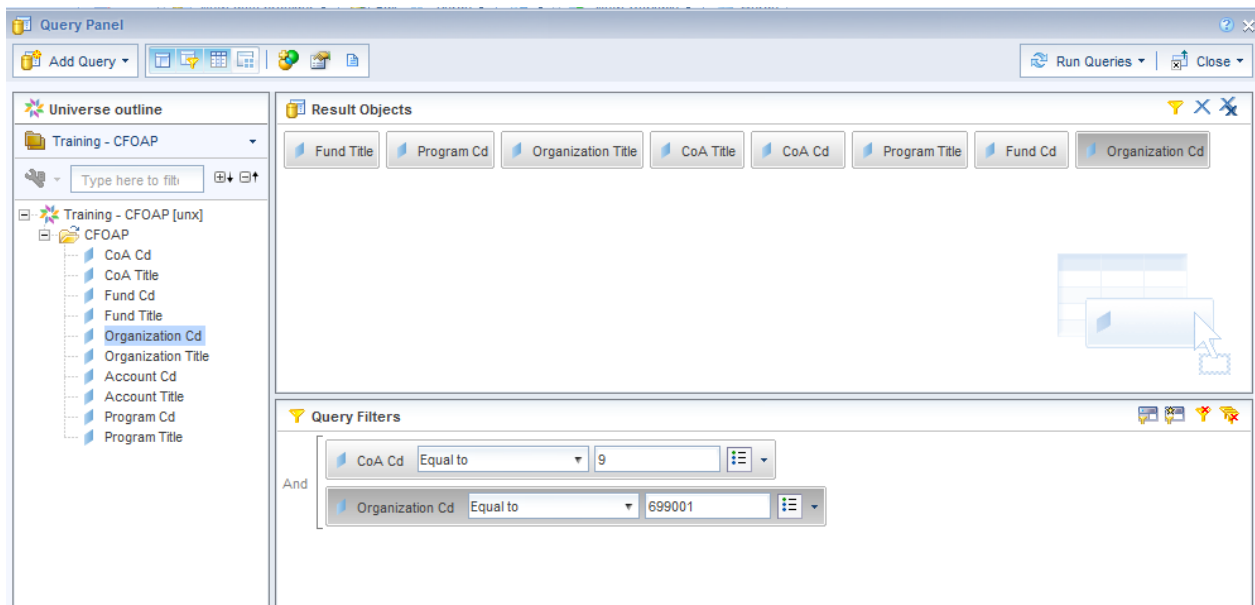
3. Select the **Training – CFOAP** universe.
4. Expand the CFOAP folder.

5. Add the following **Result Objects** to your query:

- CoA Cd
- CoA Title
- Fund Cd
- Fund Title
- Organization Cd
- Organization Title
- Program Cd
- Program Title

6. Add the following Query Filters:

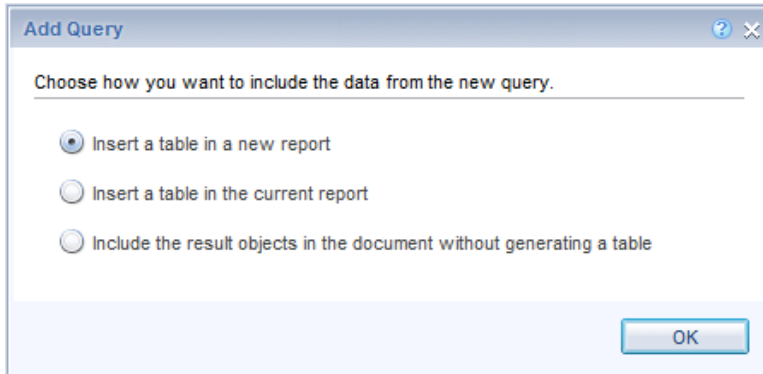
- CoA Code – Equal to – 9
- Organization Cd – Equal to – 699001



7. Click the **Run Queries** button.

8. When prompted to choose how to include the data form the new query, select **Insert a table in a new report** (default), then click **Ok**.





You will now have two reports. Report 2 will contain all of the codes and their descriptions.

## Merge Dimensions – Joining Data from Two Sources

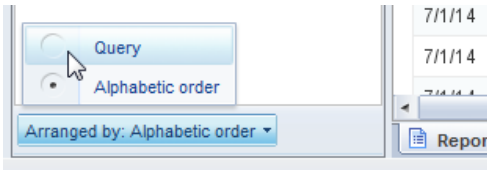
Merge Dimensions allows you to join the two data sources based on a common Dimension object. It is important to join the data sources together with an object they have in common, such as UIN. Note that the objects do not need to have the same name, but the data must be the same.

The available objects list now contains all of the objects from both of our queries. But if you attempt to add one of the objects from Query 2 to Report 1, you will see a message, **“Cannot drop here – the object is incompatible.”**

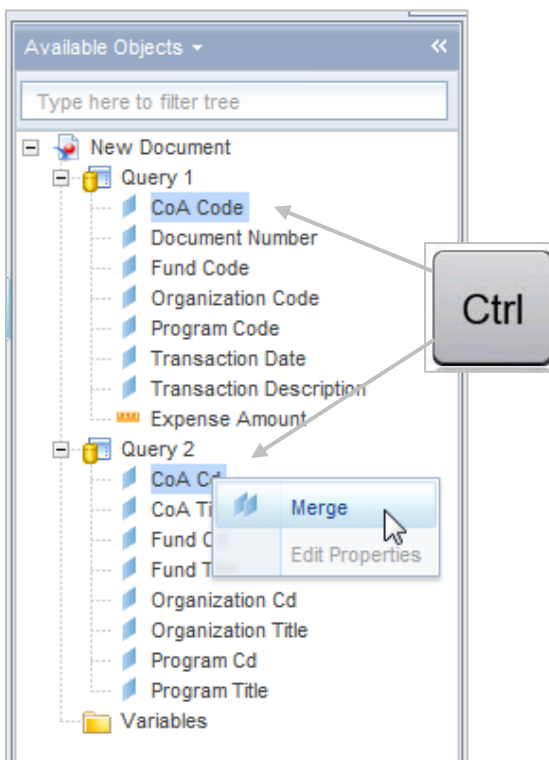
Fund Code	Organization	Program Cod	Expense Amou
100014	699001	699006	0
200201	699001	699006	-210
100014	699001	699006	0
100015	699001	699006	0

In Web Intelligence, you can only display the objects from a single data source in a report block. Since the table in report 1 was generated from Query 1, it can only display data from Query 1. In order to display the data from both queries in a single report table, we must join the data using the **Merge Dimension** feature. Think of the Merged Dimension as the bridge that connects your data providers.

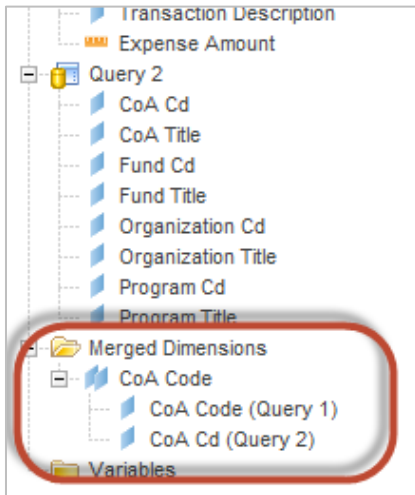
1. Select the Training Transactions report. In the bottom left corner of the report window, click on the **Arranged by: Alphabetical Order** option.
2. Select **Arranged by: Query**.



3. Select the **CoA Code** object from Query 1.
4. Hold down the **CTRL** key on your keyboard.
5. Select the **CoA Cd** object from Query 2.
6. **Right click** on this object.
7. Select **Merge** from the right-click menu.

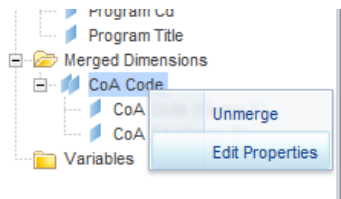


The Merged Dimensions folder is added, and the **CoA Code** merged dimension is added to the folder:



8. Right-click on the new CoA Code Merged Dimension.

9. Select **Edit Properties**



10. Change the Merged Dimension Name to **Merged CoA Code**

**Edit merged dimension**

Select a source dimension to provide default properties for the merged dimension:

**Source Dimension**  
CoA Code (Query 1)

**Merged Dimension Name**  
Merged CoA Code

**Description**  
Contains Chart of Accounts Codes used on transactions. There is a unique code for each University of Illinois campus as well as University Administration. Example valid values are 1, 2, 4, 7 & 9.

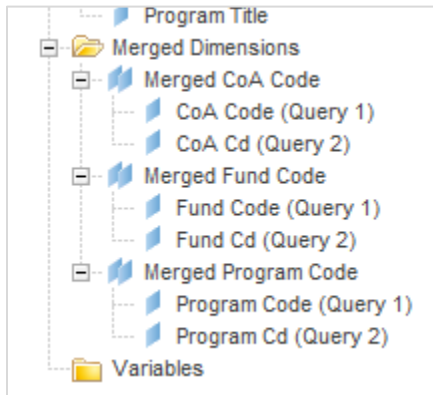
**Data type:** String

**Number Format:**

OK Cancel

**Note:** Renaming the Merged Dimension is optional, but often helps lessen confusion later in the process of joining the data.

11. Repeat steps 3-10 to merge the **Fund Code** objects and the **Program Code** objects.

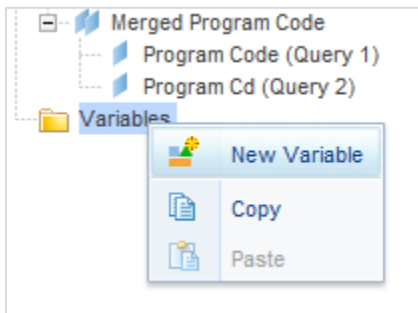


## Creating Variables

The Merged Dimensions we added create a bridge that connects the two data sources. We can now use those Merged Dimension objects to create variables that can be added to the report table. We build the variable as a Detail object and associate it with the Merged Dimension, which will allow us to put it in the Report 1 table.

*Add the Fund Title to the Report 1 table:*

1. Right-click on the **Variables** folder.
2. Select **New Variable**.



3. Enter a unique name for your variable. The name cannot be the same as the object name in any of your queries. For our example, enter **Fund Description**
4. For the Qualification, select **Detail** from the drop-down list.



**Variable Definition**

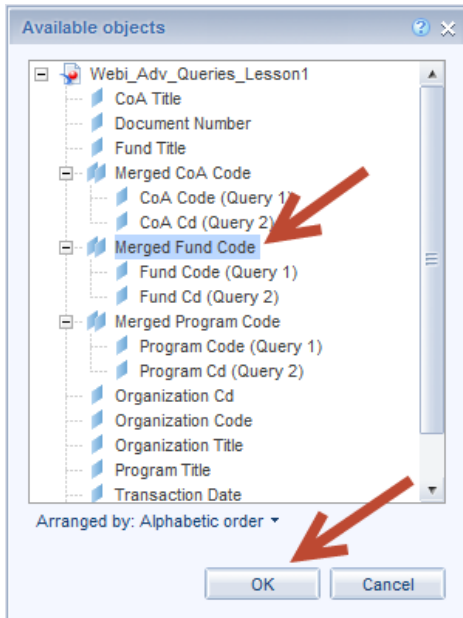
Name:

Qualification:

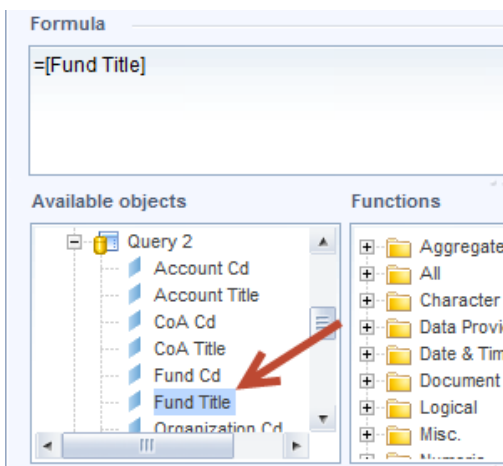
Type:

Formula

5. On the **Associated Dimension** row, click the  button.
6. Select the  **Merged Fund Code** Merged dimension.



7. Click **Ok**.
8. Click in the **Formula** box.
9. Scroll in the **Available Objects** list to Query 2.
10. Double-click the **Fund Title** object (Query 2).



11. The Formula will now read: **= [Fund Title]**
12. Click **OK** to exit the Create Variable window.
13. Repeat steps 1-12 to create a variable for the **Program Description**.

**Create Variable**

**Variable Definition**

Name: Program Description

Qualification: Detail

Associated dimension: Merged Program Code

Type: unknown

**Formula**

= [Program Title]

**Available objects**

- Fund Cd
- Fund Title
- Organization Cd
- Organization Title
- Program Cd
- Program Title
- Merged Dimensions
- Merged CoA Code

**Functions**

- Aggregate
- All
- Character
- Data Provider
- Date & Time
- Document
- Logical
- Misc.

**Operators**

= < <= <> >= > .  
+ - / \* ; ( )

Values...  
Prompts...  
:  
After  
All  
And

**Description**

**Program Title**  
The title associated with the Program Code.

OK Cancel

*Program Description Variable*

## Adding the Variables to the Report Table

Now that we have created variables associated with the Fund Title and Program Title, we can now add these variables to the Query 1 table.

1. From the Variables folder, click on the **Fund Description** variable.
2. Drag into the Report 1 table.
3. Drop to the right of the **Fund Code** column.

**Training Transactions**

Transaction Date	Document Number	Transaction Description	CoA Code	Fund Code	Organization Code	Program Code
7/1/14	AR034267	Fox-Atkins Development LLC	9	100014	699001	699006
7/1/14	AR034463	ER00656997 Fox-Atkins Development L	9	200201	699001	699006
7/1/14	AR034797	HR Payroll 2014 MN 7 1	9	100014	699001	699006
7/1/14	AR034797	HR Payroll 2014 MN 7 1	9	100015	699001	699006
7/1/14	AR034797	HR Payroll 2014 MN 7 2	9	100015	699001	699006
7/1/14	AR034797	HR Payroll 2014 MN 7 2	9	301006	699001	699012
7/1/14	AR034797	HR Payroll 2014 MN 7 5	9	100014	699001	699006
7/1/14	AR034797	HR Payroll 2014 MN 7 6	9	301006	699001	699012
7/1/14	AR035234	Reverse deferal of PR007189	9	100014	699001	699002
7/1/14	AR035570	Accrue I5530693 Information Systems	9	200201	699001	699006
7/1/14	F0274398	HR Payroll 2014 BW 998 0	9	100014	699001	699006
7/1/14	F0274398	HR Payroll 2014 BW 998 0	9	100014	699001	699128
7/1/14	F0274398	HR Payroll 2014 BW 998 0	9	301006	699001	699012
7/1/14	F0275814	HR Payroll 2014 MN 999 0	9	100014	699001	699006
7/1/14	F0275814	HR Payroll 2014 MN 999 0	9	100014	699001	699128

**Training Transactions**

Transaction Description	CoA Code	Fund Code	Fund Description	Organization Code	Program Code
Fox-Atkins Development LLC	9	100014	103 FY14 GRF/EAF/IF State Funds	699001	699006
ER00656997 Fox-Atkins Development L	9	200201	723 Educational and Admin Allow Aux	699001	699006
HR Payroll 2014 MN 7 1	9	100014	103 FY14 GRF/EAF/IF State Funds	699001	699006

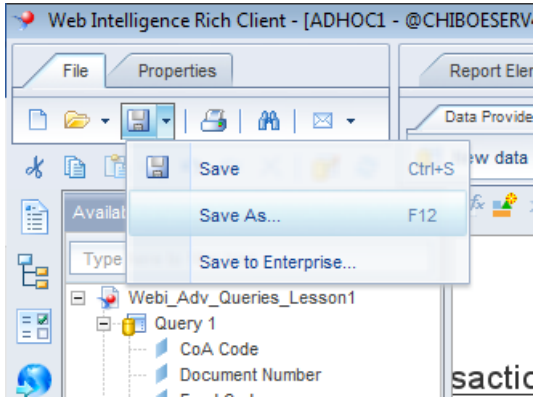
- Select the **Program Description** variable.
- Drag in the Report 1 table.
- Drop to the right of Program Code.

**Training Transactions**

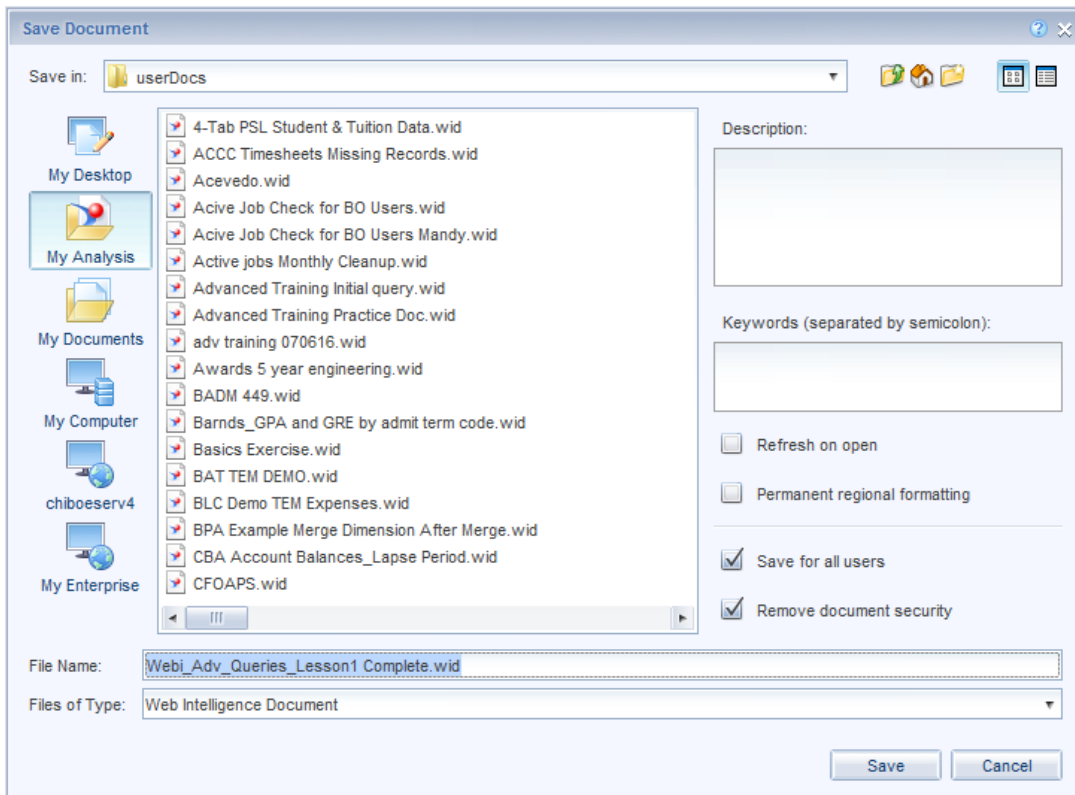
Transaction Description	CoA Code	Fund Code	Fund Description	Organization Code	Program Code	Expense Amount
Fox-Atkins Development LLC	9	100014	103 FY14 GRF/EAF/IF State Funds	699001	699006	\$0.00
ER00656997 Fox-Atkins Development L	9	200201	723 Educational and Admin Allow Aux	699001	699006	(\$210.00)
HR Payroll 2014 MN 7 1	9	100014	103 FY14 GRF/EAF/IF State Funds	699001	699006	\$0.00
HR Payroll 2014 MN 7 1	9	100015	103 FY15 GRF/EAF/IF State Funds	699001	699006	\$0.00
HR Payroll 2014 MN 7 2	9	100015	103 FY15 GRF/EAF/IF State Funds	699001	699006	\$0.00
HR Payroll 2014 MN 7 2	9	301006	699 AITS	699001	699012	\$0.00
HR Payroll 2014 MN 7 5	9	100014	103 FY14 GRF/EAF/IF State Funds	699001	699006	\$0.00
HR Payroll 2014 MN 7 6	9	301006	699 AITS	699001	699012	\$0.00
Reverse deferal of PR007189	9	100014	103 FY14 GRF/EAF/IF State Funds	699001	699002	\$6,144.98
Accrue I5530693 Information Systems	9	200201	723 Educational and Admin Allow Aux	699001	699006	(\$3,911.31)
HR Payroll 2014 BW 998 0	9	100014	103 FY14 GRF/EAF/IF State Funds	699001	699006	(\$32,643.13)
HR Payroll 2014 BW 998 0	9	100014	103 FY14 GRF/EAF/IF State Funds	699001	699128	(\$2,644.50)
HR Payroll 2014 BW 998 0	9	301006	699 AITS	699001	699012	(\$2,338.05)
HR Payroll 2014 MN 999 0	9	100014	103 FY14 GRF/EAF/IF State Funds	699001	699006	(\$578,588.73)
HR Payroll 2014 MN 999 0	9	100014	103 FY14 GRF/EAF/IF State Funds	699001	699128	(\$11,295.32)

The Training Transaction report now displays both the Fund description and Program Description from Query 2.

7. Click the Save Menu button, and select **Save as**



8. Save your Document as **Webi\_Adv\_Queries\_Lesson1 Complete**





## Chapter 2: Using Excel as a Data Source

In addition to pulling data from our Data Warehouse universes, you can also add data from Excel spreadsheets to your Web Intelligence reports. For our training example, we will add additional data related to our transactions that are kept in an Excel document.

**Important Note about using an Excel file as a Data Source:** Prior to adding the Excel file as a query in Web Intelligence, it is important to save the Excel file to a set location on your computer or network drive. In order to run your query, Web Intelligence needs to be able to locate the Excel file, therefore if you move the document, you will have to rebuild your Excel query.

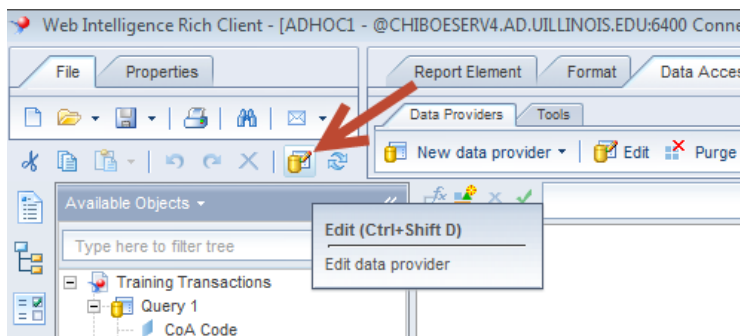
### Prepare your Excel Spreadsheet

Here are some tips for optimizing your Excel Spreadsheet to work well as a data source:

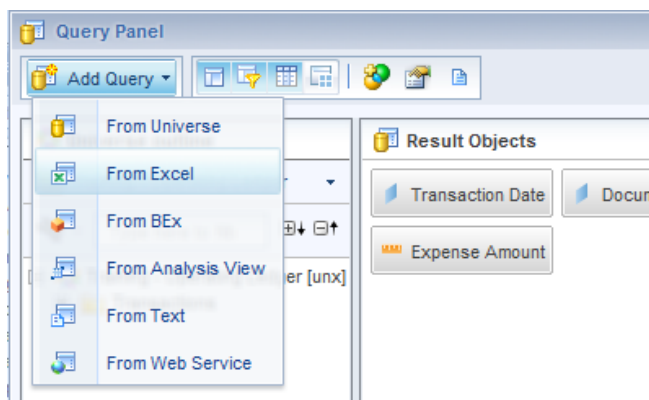
1. Make sure your column headers are in the first row, and that the first column contains data.
2. If your worksheet contains UINs, format the UIN column as text in Excel.
3. Save your Excel document to a location where it will not be moved or deleted.

### Building a Query from Excel

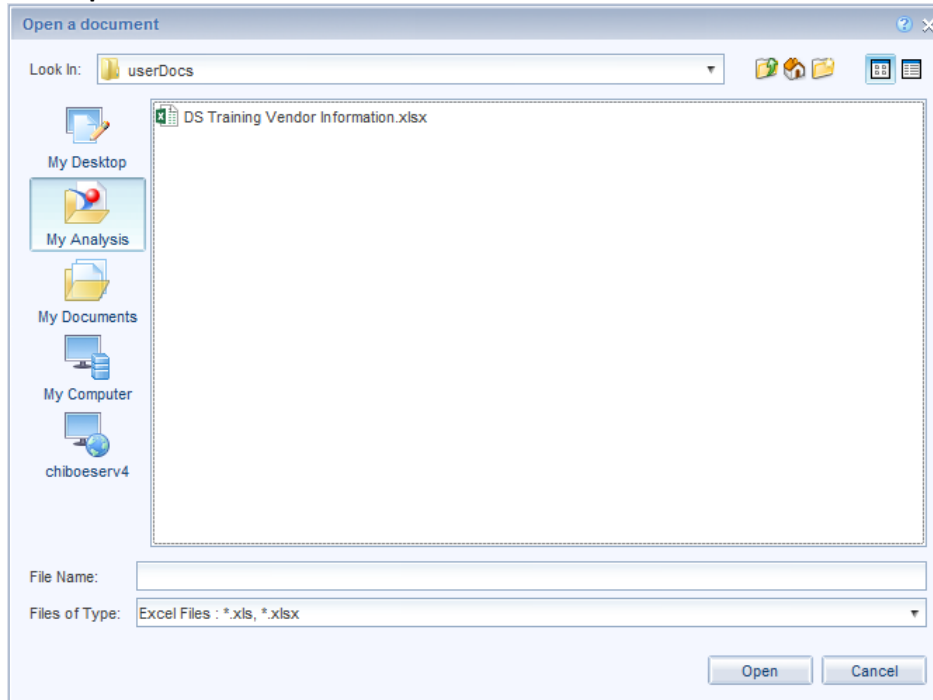
1. Click the **Edit Query** button



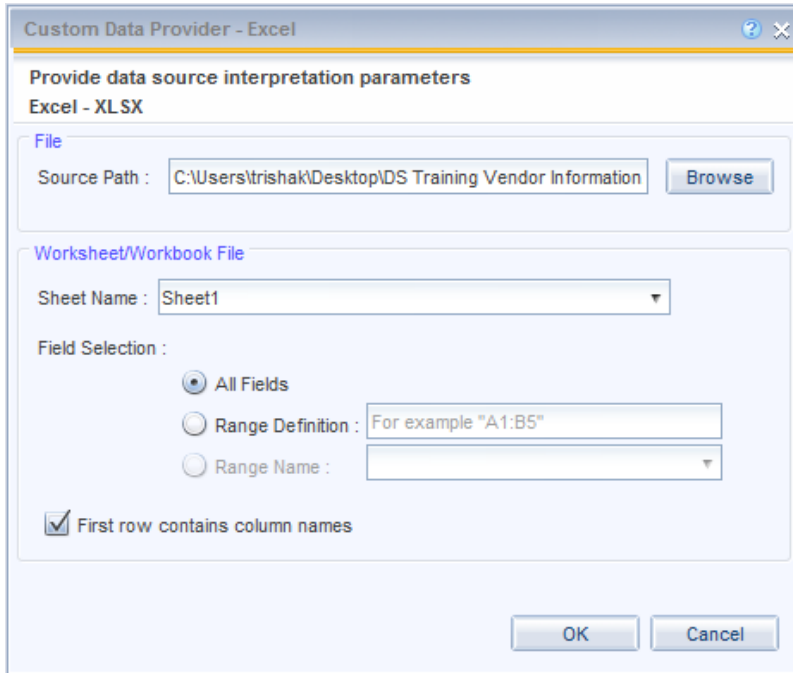
2. Click the **Add Query** button, then select **From Excel**



3. In the **Open a document** box select the **My Analysis icon**.
4. Select the **DS Training Vendor Information** document.
5. Click **Open**

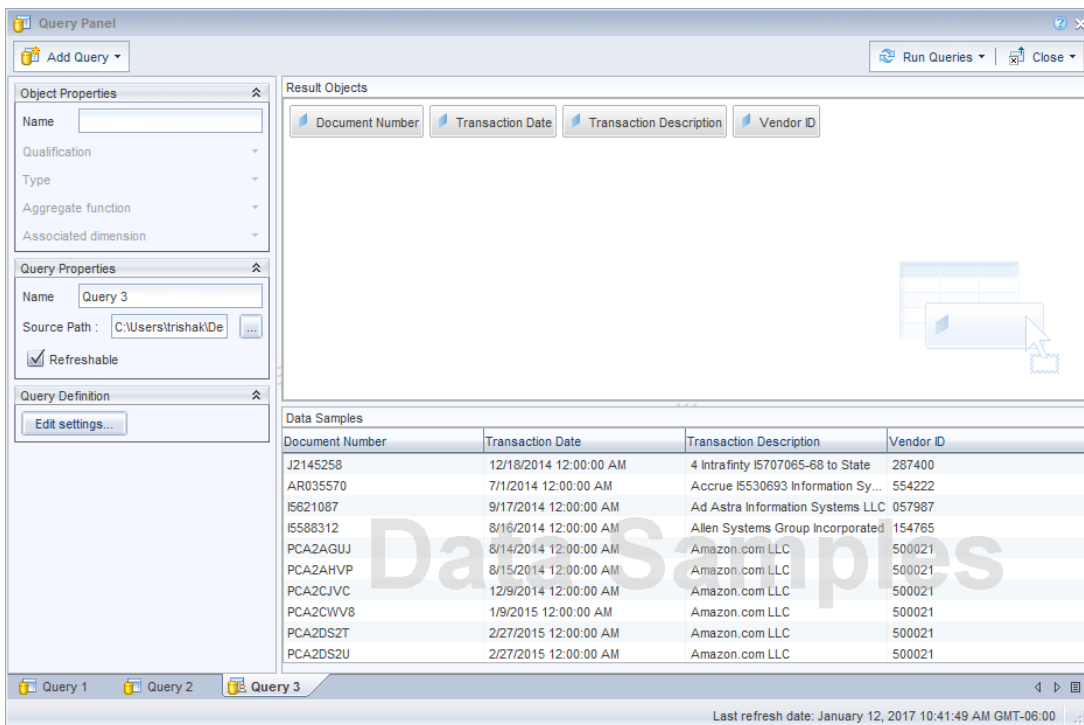


The Customer Data Provider – Excel window is displayed. This window allows you to select specific Worksheets and to specify a field range to include in your query. Checking the **First row contains column names** option will cause the column headers to become the object names in your Excel query.



6. Click **OK**

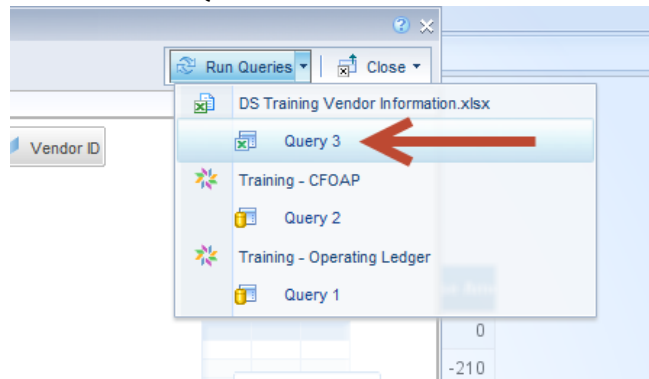
The query based on the Excel file will be displayed:



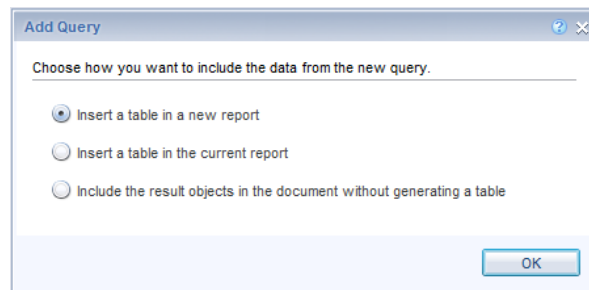
- Review your objects to make sure they have been created correctly. Select each object and verify that the Object Properties are correct.

**Note:** If your Excel spreadsheet contains UINs, make sure the UIN object has a Qualification of Dimension and a Type of String. If UINs have a qualification of Measure objects or a type of Numeric, the UINs will not display properly.

- Select the **Run Queries** menu button. Select to run only **Query 3**



- When prompted to choose how to include the data from the new query, select **Insert a table in a new report** (default), then click **Ok**.



Report 3 will be generated, displaying the data from the Excel spreadsheet.

## Using Merge Dimensions to Add Data from Excel to Existing Reports

Now we will add the Vendor ID information pulled from Excel to Report 1. We will first merge Query 1 and Query 3 using a common object, Document Number.

- Click on the **Report 1** tab
- Holding the **Control** button on your keyboard, select the Document Number object from Query 1 and the Document Number object from Query 3

Report 1

Transaction I	Document Nu	Transaction I	CoA Code
7/1/14			
7/1/14			
7/1/14			
7/1/14	AR034797	HR Payroll 20	9
7/1/14	AR034797	HR Payroll 20	9
7/1/14	AR034797	HR Payroll 20	9
7/1/14	AR034797	HR Payroll 20	9
7/1/14	AR034797	HR Payroll 20	9
7/1/14	AR035234	Reverse defe	9
7/1/14	AR035570	Accrue I5530	9

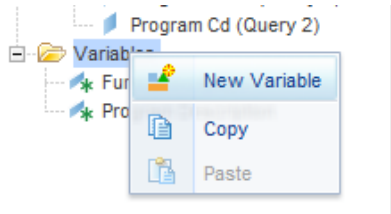
3. Right-click on one of the Document Number objects.
4. Select **Merge** from right-click menu

The Document Number Merged Dimension will be displayed in the Merged Dimensions folder.

## Create a Variable to Add Data

Now we will create a new variable for the Vendor ID and associate it with the Document Number Merged Dimension.

1. Right-click on the **Variables** folder.
2. Select **New Variable**



3. Create a Variable as follows:
  - a. Name: **Vendor Number**
  - b. Qualification: **Detail**
  - c. Associated Dimension: **Document Number** (Merged Dimension)
  - d. Formula: **=Vendor ID (Query 3)**

**Create Variable**

**Variable Definition**

Name:

Qualification:

Associated dimension:

Type:

**Formula**

**Available objects**

- Program file
  - Query 3
    - Document Number
    - Transaction Date
    - Transaction Description
    - Vendor ID
  - Merged Dimensions
    - CoA Code

**Functions**

- Aggregate
- All
- Character
- Data Provider
- Date & Time
- Document
- Logical
- Misc.

**Operators**

= < <= > >= .

+ - / \* ; ( )

Values ...

Prompts ...

:

After

All

And

**Description**

Vendor ID

OK Cancel

4. Add the Vendor Number variable to the Report 1 table to the Right of Document Number

Report 1

Transaction I	Document Nu	Vendor Num	Transaction I	CoA Code	Fund Code	Fund
7/1/14	AR034267		Fox-Atkins De	9	100014	103 F
7/1/14	AR034463	555777	ER00656997	9	200201	723 E
7/1/14	AR034797		HR Payroll 2C	9	100014	103 F
7/1/14	AR034797		HR Payroll 2C	9	100015	103 F
7/1/14	AR034797		HR Payroll 2C	9	100015	103 F
7/1/14	AR034797		HR Payroll 2C	9	301006	699 A
7/1/14	AR034797		HR Payroll 2C	9	100014	103 F
7/1/14	AR034797		HR Payroll 2C	9	301006	699 A
7/1/14	AR035234		Reverse defe	9	100014	103 F
7/1/14	AR035570	554222	Accrue I5530i	9	200201	723 E
7/1/14	F0274398		HR Payroll 2C	9	100014	103 F
7/1/14	F0274398		HR Payroll 2C	9	100014	103 F
7/1/14	F0274398		HR Payroll 2C	9	301006	699 A
7/1/14	F0275814		HR Payroll 2C	9	100014	103 F
7/1/14	F0275814		HR Payroll 2C	9	100014	103 F
7/1/14	F0275814		HR Payroll 2C	9	301006	699 A
7/1/14	F0275814		HR Payroll 2C	9	303134	699 K
7/1/14	F0276008		HR Payroll 2C	9	100014	103 F
7/1/14	F0276008		HR Payroll 2C	9	100014	103 F
7/1/14	F0276008		HR Payroll 2C	9	301006	699 A
7/1/14	I5521300		Tableau Softw	9	100015	103 F

Report 1 now displays the Vendor IDs from the Excel Spreadsheet

5. **Save** the document.





## Chapter 3: Advanced Query Filters

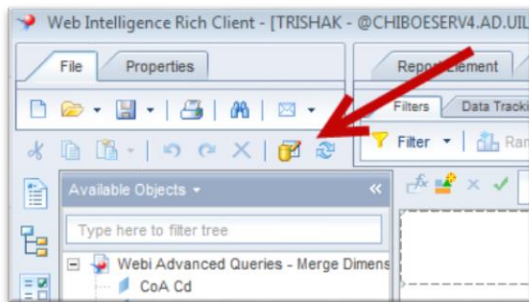
### Using Matches Pattern Operator

The query filter *Matches Pattern* is used to find values that contain the same characters in a set pattern, for example, finding all Last Names that begin with S, or all program codes that begin with ENG. The *Matches Pattern* operator is always used in conjunction with wildcards:

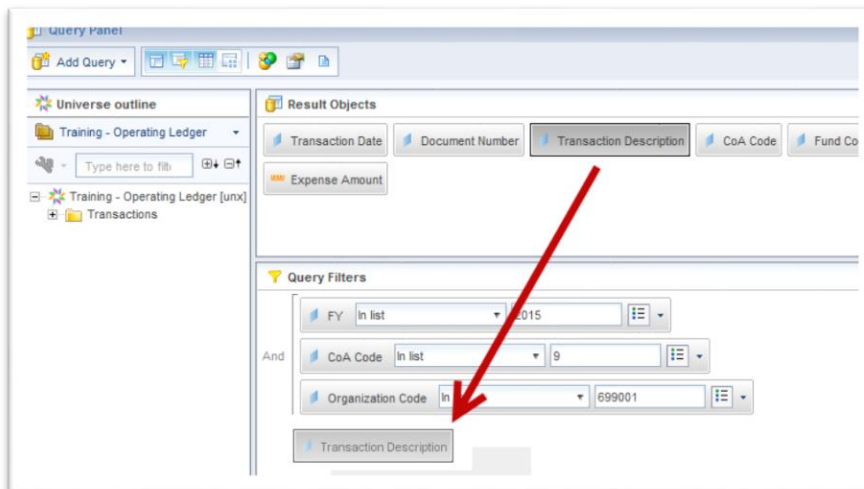
- % Replaces several characters, or in the response to a prompt.  
For example, N% returns all values beginning with an N (New York, Nevada, etc.)
- \_ The underscore character (\_) replaces a single character in a constant.  
For example, S\_ng would find Sang, Sing, Song, Sung, etc.

Now we will create a query filter that shows all transactions with a Transaction Description that contain the text 'Cellular':

1. Click the **Edit Query** button.



2. Drag the Transaction Description object into the Query Filter panel.



3. Select **Matches Pattern** from the operator drop-down list.

**Query Filters**

FY In list 2015

CoA Code In list 9

Organization Code In list 699001

Transaction Description In list

Between

Not Between

Is null

Is not null

Matches pattern

Different from pattern

Both

Except

Data Preview

- In the Operand text box, enter: %Cellular%

Transaction Description Matches pattern %Cellular%

- Click **Run Queries**.

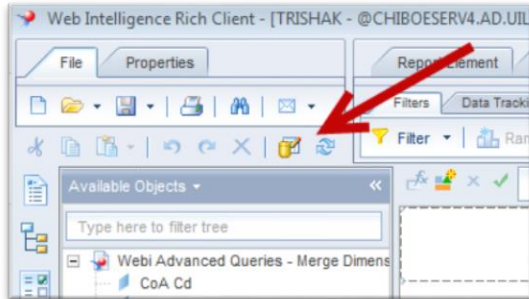
Results are displayed showing only rows where the text “Cellular” is found in the Transaction Description:

Transaction	Document Nu	Transaction I	CoA Code	Fund Code	Organization	Program Cod	Expense Am
8/1/14	CT049910	Cellular Unlir	9	200201	699001	699006	278.25
8/25/14	CT050169	Cellular Unlir	9	200201	699001	699006	278.25
9/25/14	CT050488	Cellular Unlir	9	200201	699001	699006	278.25
10/24/14	CT050795	Cellular Unlir	9	200201	699001	699006	278.25
11/25/14	CT051006	Cellular Unlir	9	200201	699001	699006	278.25
12/23/14	CT051426	Cellular Unlir	9	200201	699001	699006	278.25
1/26/15	CT051687	Cellular Unlir	9	200201	699001	699006	278.25
2/25/15	CT051944	Cellular Unlir	9	200201	699001	699006	278.25
3/25/15	CT052212	Cellular Unlir	9	200201	699001	699006	278.25
4/24/15	CT052526	Cellular Unlir	9	200201	699001	699006	278.25
5/26/15	CT052807	Cellular Unlir	9	200201	699001	699006	9.3

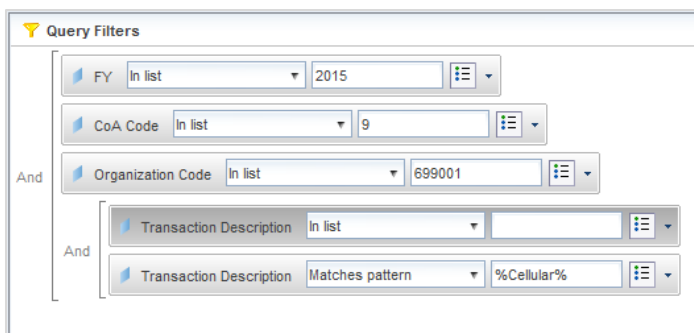
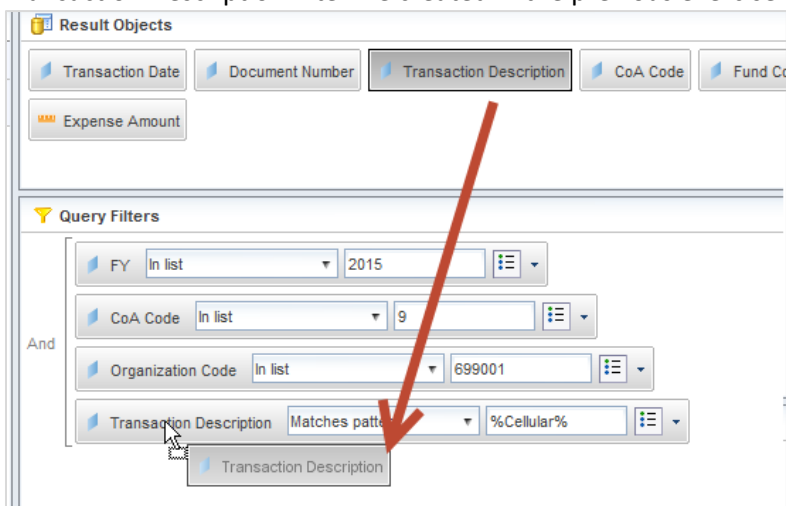
## Using “or” Logical Operator with Matches Pattern

Since operand values are case-sensitive, a Transaction Description containing the text ‘CELLULAR’ would not be displayed in our results. In order to capture these possible variations, we will add another filter to our query.

1. Click the **Edit Query** button.



2. Drag the **Transaction Description** object into your Query Filters. Drop the object **on top of** the Transaction Description filter we created in the previous exercise:



3. Click the **And** logical operator that connects the two Transaction Description Filters to change it to **Or**.

And Organization Code In list 699001

Or Transaction Description In list

Transaction Description Matches pattern %Cellular%

4. Change the Operator for the new filter to **Matches Pattern**:

Or Transaction Description Matches pattern

Transaction Description Not Between

Is null

Is not null

Matches pattern

Different from pattern

Both

Except

Data Preview

5. In the **Operand** text box, type: **%CELLULAR%**

Or Transaction Description Matches pattern %CELLULAR%

Transaction Description Matches pattern %Cellular%

6. Click **Run Queries**

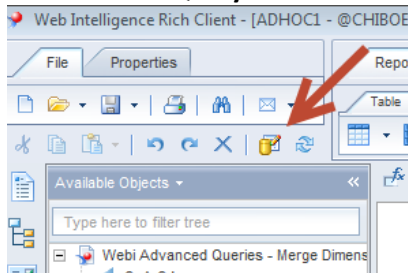
Now results include rows where the transaction description includes the text “Cellular” or “CELLULAR”.

Transaction I	Document N	Transaction Descript	CoA Code	Fund Code	Organization	Program Cod	Expense Am
8/1/14	CT049905	CELLULAR - TELEPH	9	200201	699001	699006	5.25
8/1/14	CT049910	Cellular Unlimited EV	9	200201	699001	699006	278.25
8/25/14	CT050163	CELLULAR - ACTIVAT	9	200201	699001	699006	75
8/25/14	CT050164	CELLULAR - TELEPH	9	200201	699001	699006	5.25
8/25/14	CT050169	Cellular Unlimited EV	9	200201	699001	699006	278.25
9/25/14	CT050342	CELLULAR AIR TIME	9	200201	699001	699006	0.18
9/25/14	CT050358	CELLULAR SURCHA	9	200201	699001	699006	0.97

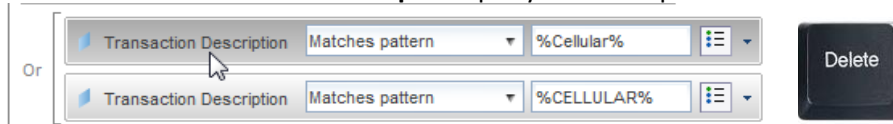
## Removing Query Filters

We will now remove both of our Transaction Description filters prior to our next exercise:

1. Click the **Edit Query** button:



2. Select the first **Transaction Description** query filter and press the **Delete** key on your keyboard:



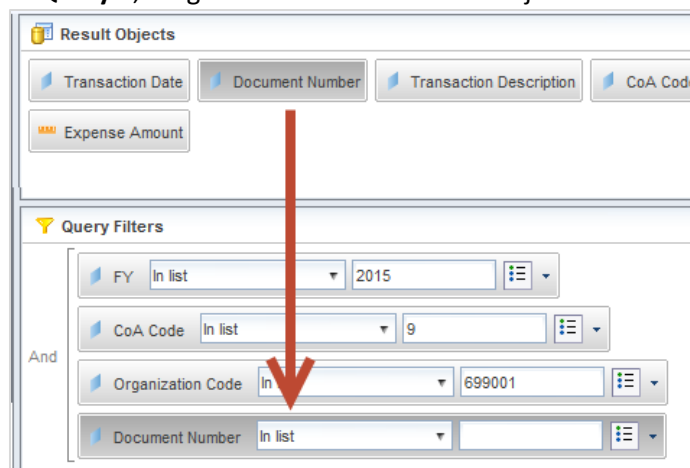
3. Repeat step 2 and delete the other **Transaction Description** query filter.

## Filter on Results from another Query

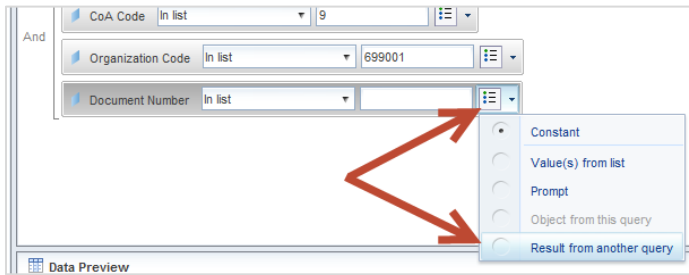
The **Results from Another Query** option when filtering allows you to use a list of values from another query within your Web Intelligence Document as a filter. Examples of this would using a list of UINs from an excel spreadsheet as a filter for a query against a universe. In our example, we will filter Query 1 to show only the transactions that are found in our Excel database (Query 3).

**Note:** Before you can create a filter using the results from another query, you must first run the query.

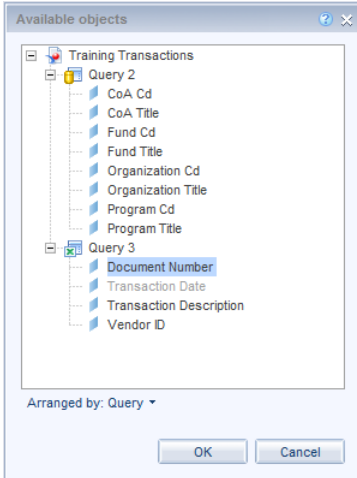
1. In **Query 1**, drag the **Document Number** object into the Query Filter panel.



2. Leave the operator as **In List** (Default value).
3. Click the **Operand Menu Button**.
4. Select **Results from Another Query**.



5. Select Document Number (from Query 3) and click **Ok**.



6. Click **Run Queries**.

Displayed results in Report 1 are now limited to the Transactions that were found in Query 3.

### Report 1

Transaction I	Document Nt	Vendor Numl	Transaction Description	CoA Code	Fund Code	F
7/1/14	AR034463	555777	ER00656997 Fox-Atkins Development L	9	200201	7
7/1/14	AR035570	554222	Accrue I5530693 Information Systems	9	200201	7
7/1/14	I5523588	773001	CollegeSource Inc	9	200201	7
7/1/14	I5525526	287400	Intrafinity Incorporated	9	200201	7
7/1/14	I5530622	800021	Ellucian Support Inc	9	200201	7
7/1/14	I5530693	897789	Information Systems Group Incorpora	9	200201	7
7/1/14	I5536187	224410	FE Moran Inc Alarm & Monitoring Ser	9	200201	7
7/1/14	I5536188	224410	FE Moran Inc Alarm & Monitoring Ser	9	200201	7
7/1/14	I5537626	321123	SSH Communications Security Inc	9	200201	7

**Note:** Web Intelligence queries use Oracle, and Oracle has a list of values limit of 1000 values. So if the list of values from the other query cannot contain more than 1000 rows of data.

7. Save your document.
8. Exit Web Intelligence.

---

## Appendix A – Additional Assistance

### Decision Support Website

Additional resources and online help can be found on the Decision Support website. Resources include user guides and quick tips on using both Web Intelligence and EDDIE.

[https://www.aitis.uillinois.edu/services/reports\\_and\\_data/help\\_and\\_training/](https://www.aitis.uillinois.edu/services/reports_and_data/help_and_training/)

### AITIS Service Desk

For further assistance, or to report a problem, contact the AITS Service Desk:

Email: [servicedeskaitis@uillinois.edu](mailto:servicedeskaitis@uillinois.edu)

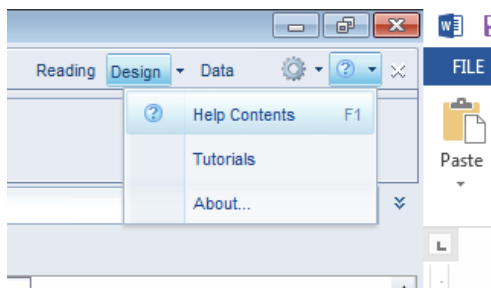
Chicago: (312) 996-4806

Urbana & Springfield: (217) 333-3102

### SAP Web Intelligence Online Help

Web Intelligence provides on-line assistance and information about basic product features, as well as information to help you troubleshoot and solve common problems.

Click the  in the upper right corner of the Web Intelligence window.



#### Help Contents

This will take you to the SAP Web Intelligence online help.

#### Tutorials

This options is not enabled at this time and will take you to the main SAP website. SAP does have a limited amount of video tutorials located on their YouTube Channel.

#### About Web Intelligence

This contains version and licensing information.

## Appendix B – Query Filter Operators

The following table helps you to select the operator you need:

Operator	Description	Example
Equal to	Is equal to <i>one</i> given value	Year Equal to 2005
Not Equal to	Is different from, or not equal to, <i>one</i> given value	Employee Status Code Not Equal to T
Greater than	Is greater than a given value	GL Detail Credit Amount Greater than 100
Greater than or Equal to	Is greater than or equal to a given value	GL Detail Credit Amount Greater than 100
Less than	Is less than a given value	Job Detail Annual Salary Less than 20000
Less than or Equal to	Is less than or equal to a given value	Employees who are 60 or under
Between	Is greater than or equal to the first given value and less than or equal to the second given value	Job Begin Date Between 1/1/2004 12:00:00 AM and 12/31/2004 11:59:59 PM
Not Between	Is less than the first given value or greater than the second given value	Employee Age Not Between 18 and 60
In list	Is equal to any of a list of values	Employee Campus Address Code In list (C1;C2)
Not in list	Is different from all of a list of values	Employee Detail Department Code Not in list (103;714;715)
Is null	Contains empty rows	Employee Campus Email Addr Type CD Is null
Is not null	Does not contain empty rows	Employee Campus Email Addr Type CD Is not null
Matches pattern	Contains the same character(s) as the given pattern	Employee Last Name Matches pattern S%
Different from pattern	Does not contain the same characters as the given pattern	Financial Account Code Different from pattern 9%
Both	Satisfies two conditions on one object	Prior Degree Year Both 2001;2004
Except	Excludes a given value	Employee Status Code Except T



## Using In list

You can type your list of values in the text field of the *Type a constant* text box. If you click the **Operand** button, you can also select **Value(s) from list** to select from a list of values. The maximum number of values allowed in a list is 256.

*Note:* When entering multiple values manually, use a semi-colon (;) to separate each value.

## Using Not equal to; Not in list; and Except

*Not equal to; Not in list; and Except* are all operators that exclude certain data from your query results. Note also that you can only specify one value with *Not equal to*, but multiple values with *Not in list*.

## Using wildcard characters


Conditions with the *Match pattern* and *Different from pattern* operators are great for finding lists of similar values, such as customer names beginning with S.

## Wildcard Description

- % Replaces several characters, or in the response to a prompt.  
For example, N% returns all values beginning with an N (New York, Nevada, etc.)
  
- \_ The underscore character (\_) replaces a single character in a constant.  
For example, GR\_VE returns Grave, Grove, Greve, etc.

## Appendix C – Query Filter Operand Options

The following table helps you select the operand option you need:

Operand Option	Description	Enter by...
Constant	Values that you type	Type the values with a separator (semicolon) between each one. The separator to use is defined in the Windows Control Panel ( <i>Regional Settings</i> > <i>Additional settings</i> .>List separation)
Value(s) from list	Values that you select from the object's list of values	Select <b>Value(s) from list</b> option. Select from the populated list of values. Click <b>Refresh values</b> if list does not automatically populate.
Prompt	Values that you will select when you run the query	Edit default prompt text in text box. Click  <b>Prompt Properties</b> for more options.
Results from another query	Use a list generated by another query, for example: a list of UINs from another spreadsheet.	Select <b>Results from another query</b> then select the data provider from list.