



ASR Analytics, LLC

Prepared by: Kate Connolly
E-mail: kate.connolly@asranalytics.com
Web: <http://www.asranalytics.com/>

SAN JOSE EVERGREEN COMMUNITY COLLEGE DISTRICT: ADVANCED REPORTING AND QUERY TRAINING FOR RESEARCHERS

SAP BusinessObjects Web Intelligence Training

Version: 1.2

Last updated: October 17, 2023

DOCUMENT CHANGE HISTORY

Document Version	Date	Description of Change	Affected Sections
1.0	November 20, 2019	Created with baseline content for San Jose Evergreen Community College District	All
1.1	November 21, 2019	Added Combined Query creation demo and practice exercise.	Adding a Combined Query
1.2	October 17, 2023	Added updated name of new unified universe (BI Student Success)	Introduction

CONTENTS

Document Change History ii

Contents..... 3

Introduction..... 6

Understanding Data Flow 7

 Conventions used in this document..... 9

Report Design Patterns and Best Practices 10

 Simple Design Pattern 10

 Separate Selection Design Pattern 10

 Combined Result Data Design 10

 Report Design Best Practices 10

Migrating Reports to Another Universe..... 11

 Learning Objective 11

 High-level Overview 11

 Practice Steps..... 11

Understanding and Working with Nested Filters and Subqueries..... 17

 Query Filters 18

 Nested Query Filters 19

 Nested Report Filters 21

 Filtering Data Using Subqueries..... 22

 Learning Objective 25

 High-level Overview 25

 Practice Steps for Nested Query Filters 25

 Practice Steps for Building a Sub Query 26

Creating and Using Variables..... 27

 Using If-Then-Else Logic in Report Variables..... 28

 Building a Dynamic Title..... 29

 Time Functions 29

 Character Functions..... 30

 Learning Objective 32

 High-level Overview 32

 Practice Steps..... 32

Calculation Contexts 34

 Using the IN Context Operator 37

 Using the ForAll Context Operator..... 39

 Using the Where Conditional Context Operator 40

 Using the In Report Context Operator 41

 Learning Objective 43

 High Level Overview 43

 Practice Steps..... 43

Advanced Formatting 46

 Formatting Tables 46

 Conditional Formatting..... 49

 Breaks..... 50

 Sections 53

 Outline (fold/unfold)..... 53

 Hands-on Exercise Learning Objective..... 54

 High Level Overview 54

Practice Steps.....	54
Linking and Drill-Through Reporting	57
Element Linking	57
Intra-Document Linking	60
Document Linking to a Webpage	62
Drill Through Reporting	63
Hands-on Exercise Learning Objective.....	63
High Level Overview	63
Practice Steps.....	63
Using Web Intelligence’s Interactive Viewer.....	67
Navigating the Home Screen.....	67
Accessing Reports	69
Effective Sharing of Reports and Analyses.....	70
Setting Up Web Intelligence Reports to Refresh on Open.....	70
Exporting Web Intelligence Content to PDF or Excel Format.....	72
Working with The Web Intelligence Scheduler.....	74
Looking at Historic Instances of Reports	78
Using Publications to Automate Report Distribution	79
Adding a Combined Query	86
Unifying prompts	90
Learning Objective	91
High Level Overview	91
Practice Steps.....	91
Additional References	93
Independent Materials	93
SAP Documentation.....	93
SAP online learning courses	93
Ellucian Documentation.....	93
Acronyms and Glossary.....	93

This Page Intentionally Left Blank for Notes

INTRODUCTION

This document provides an overview of specific reporting and query activities, as requested by San Jose Evergreen Community College District (SJECCD), in the BusinessObjects Web Intelligence reporting tool. The material is designed for structured training and as a reference when working independently. This training assumes familiarity with how to login, access and view reports, and handle basic navigation tasks.

The practice steps are followed by sections intended to improve the understanding of data in the BI Unified Universe – dev v1 universe. This universe was in development in the S11 test folder at the time this training was initially delivered and was ultimately moved to production following sign-off from SJECCD. It was renamed BI Student Success; please use BI Student Success from the S11 Production folder rather than the dev universe in the exercise examples below. The goal of this training is to provide users with the skills necessary to continue to explore reports and data and make changes to the content and layout to suit their reporting needs within the constraints of the Operational Data Store (ODS). This training does not cover modifications to the ODS or Universes to add additional data elements that may exist in Colleague. Those customizations are normally an IT function.

UNDERSTANDING DATA FLOW

The following diagram (Figure 1) depicts how data flows from the source to the end user viewing it. The most common way to view data is in a Web Intelligence report in the BusinessObjects BI Launch Pad environment. Data can come from any number of sources; in the SJECCD implementation, it is largely from Ellucian Colleague. Data is extracted and transformed from the Colleague database into the ODS database. From there, it is accessed by the BusinessObjects environment.

TIP: Because the data in the BusinessObjects reporting environment stems from this data flow, most Universes are not updated in “real time,” but is refreshed nightly, and therefore current as of the previous refresh unless otherwise stated by IT. This should be adequate for most types of reporting.

It is also important to understand that at any layer of this flow, data can be filtered or manipulated for a given reporting need. Editing filters for blocks, reports, and documents (which contain the query definition for the Universe). This document does not cover editing filters for Universes and databases.

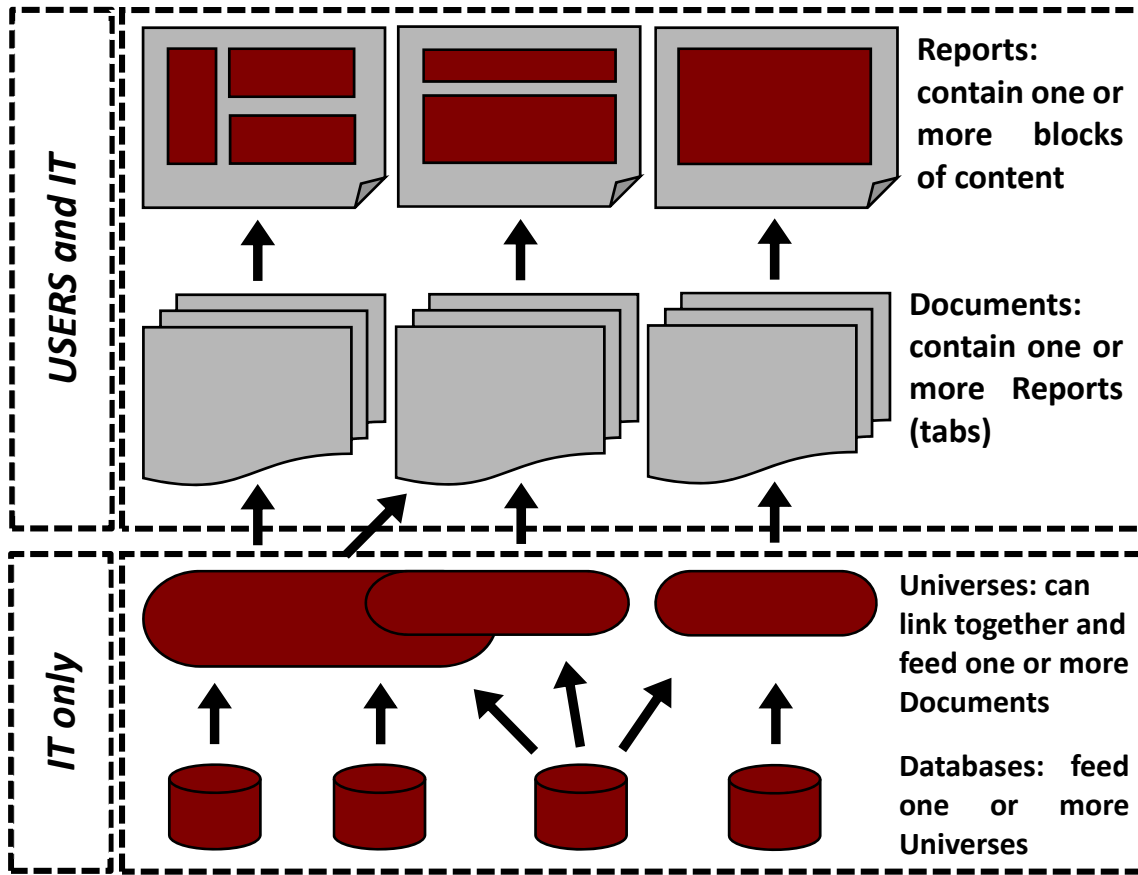


Figure 1: Data flow from source to report

TIP: The terms Web Intelligence “document” and “report” are often used interchangeably, since it is common for a document to only contain one report (tab). Despite the mixed usage, it is important to understand that the data query is defined at the document level, and the result set is used by one or more reports (tabs) in a document.

Conventions used in this document

The following formatting styles are used to identify different components of the reporting tool. They are supported with screen captures of the actions and results.

- Underlined text within single quotes. For example: 'OK', 'Equal To'.
These represent buttons, drop down list values, radio button values, or other types of functions that you can work with in the Web Intelligence reporting tool.
- Italicized text within double quotes. For example: "*Student ID*", "*Query 2*".
These represent object names such as a universe, report, tab, data element, variable, text box, dimension, measure, or detail names. These names are unique to what is being created and could be different from report to report. These are the names chosen by the report designer and are not fixed as part of the reporting tool. They may have default values when being used to create a new component in a report.
- The → symbol. For example: 'Analysis' → 'Data Providers' → 'Edit'
This indicates a series of steps or clicks from a menu to perform in sequence.

REPORT DESIGN PATTERNS AND BEST PRACTICES

There are essentially three report design pattern approaches that cover most reporting needs. Identifying which design pattern best suits your reporting need will aid in efficient and effective query design. For each pattern, consider the following:

Simple Design Pattern

- All data objects are compatible, in related classes, and at the same level of granularity
- Filtering is on data that is part of the desired output result
- Requires only one query

Separate Selection Design Pattern

- Filtering is on data which is separate from the desired output result
- Requires more than one query
- Employs the “SavedList” concept of using one query to return a list of IDs and one or more queries to return data about those IDs

Combined Result Data Design

- All data objects are not compatible or are not at the same level of granularity
- Requires multiple queries that need to be merged on a common data object and the creation of detail objects

Report Design Best Practices

Building reports using a new data model can seem like a daunting task. This section provides report building and data validation tips and techniques that will help you gain experience and confidence with both the data model and the reporting tool, in this case, SAP BusinessObjects Web Intelligence (WebI). Whether you are building class exercise reports or ad hoc reports for your own needs, consider following these best practices as a guideline:

- Start small – adding only a few query objects at a time for a small selection of IDs and verify data accuracy, cardinality, and aggregated measure object values if present
 - To verify cardinality, you can use ‘All’ in your COUNT function. If you were only expecting one row per ID, the overall count should not differ between distinct and non-distinct

The screenshot shows a formula bar at the top with the formula `=Count([CID];All)` highlighted in a red box. Below the formula bar is a table with three rows of data. The first two rows have the value 'STU' in the right column, and the third row has the value 'FAP' in the right column. The bottom row of the table shows a total count of 5,489, which is also highlighted in a red box.

	STU
	FAP
	STU
5,489	

- Add in additional fields from one class at a time and make sure each refresh does not change the cardinality or aggregated measure object values if present
 - If including aggregated measure objects, add in fields to check the underlying detail of the aggregation
- Create a validation tab for each query to ensure data accuracy
- When merging dimensions and creating detail objects, create the details using objects from one query, ensuring that query only returns one row per record/ID (i.e. you cannot create a reliable detail object using a multivalued dimension)
- SAVE early and often!

MIGRATING REPORTS TO ANOTHER UNIVERSE

As part of routine maintenance, ASR enhancements to BI universes, or new releases from Ellucian for their delivered universes, the universe(s) your reports rely on will occasionally require updated versions. Reports may be migrated to the new universe version by your system administrator during the update process. Generally, however, you will need to migrate your reports manually.

Learning Objective

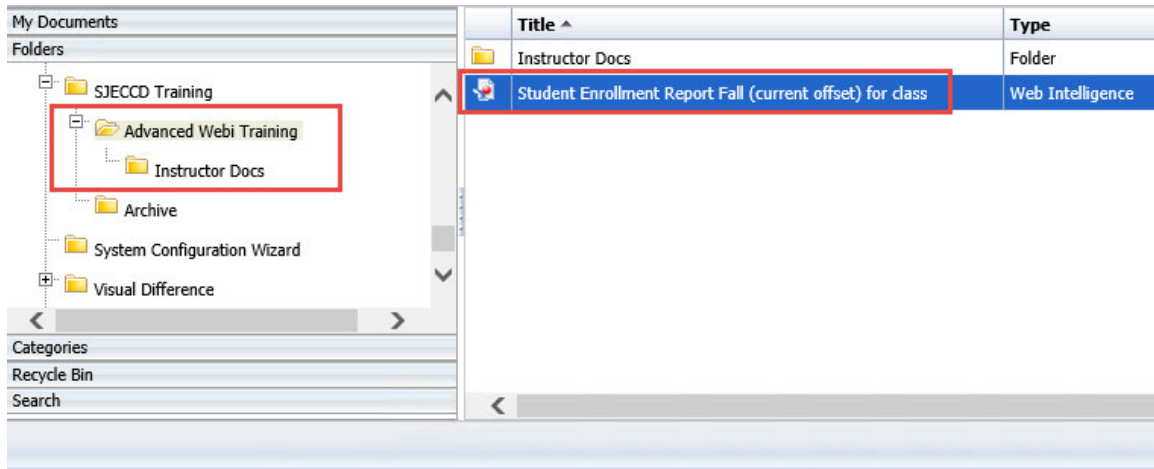
This section will help you understand how to migrate a report from one universe to another.

High-level Overview

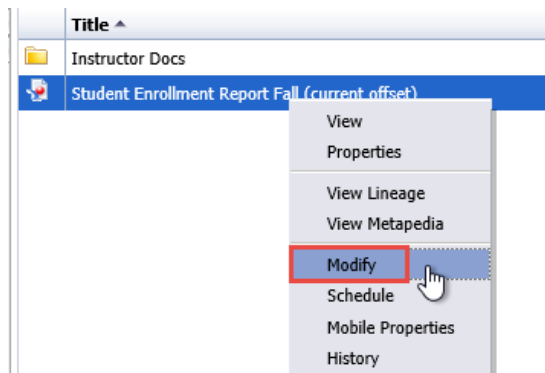
1. Navigate to the report requiring migration in BI Launch Pad
2. Open the report in design mode
3. Navigate to the 'Change Source Wizard'
4. Specify a new data source (universe)
5. Select the 'Strategy' to apply for object mapping
6. Verify the accuracy of the object mapping, re-map, or delete objects as necessary
7. Run the query and verify results
8. Save and close

Practice Steps

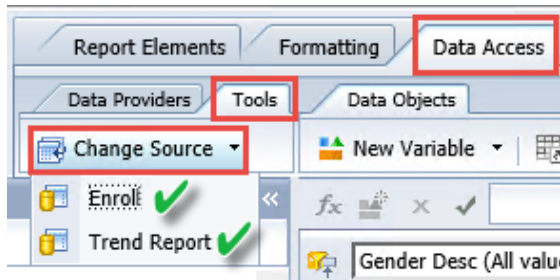
STEP 1: Log in to BI Launch Pad and navigate to the report you want to migrate to another universe. In our example, we are using the *“Student Enrollment Report Fall (current offset) for class”* report found in the *“Public Folders”* → *“SJECCD Training”* → *“Advanced Webi Training”* folder.



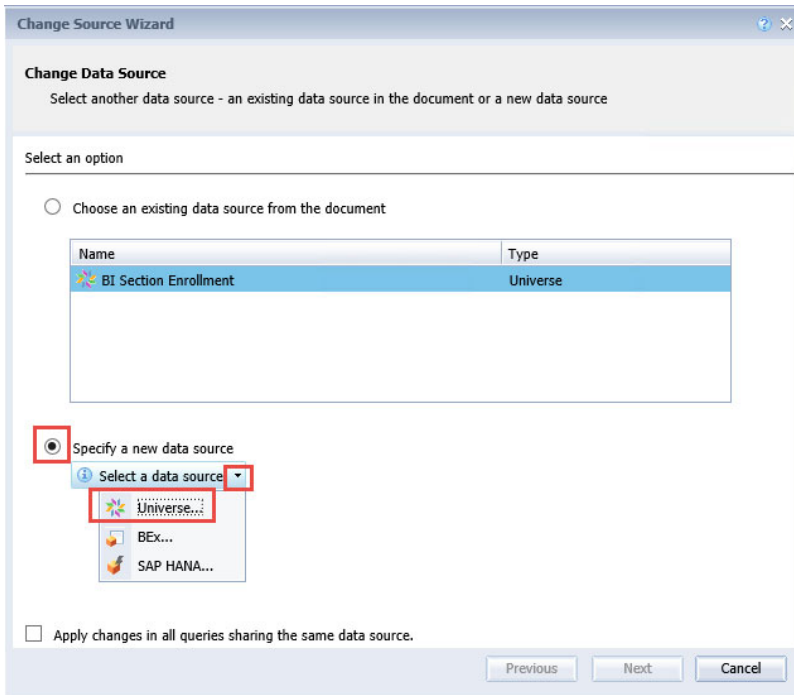
STEP 2: Open the document in design mode by right clicking on it and selecting 'Modify'. Once the document opens, it is good practice to create a copy of the report to work from. That retains the original for reference, and as a backup in case the migration process doesn't go smoothly!



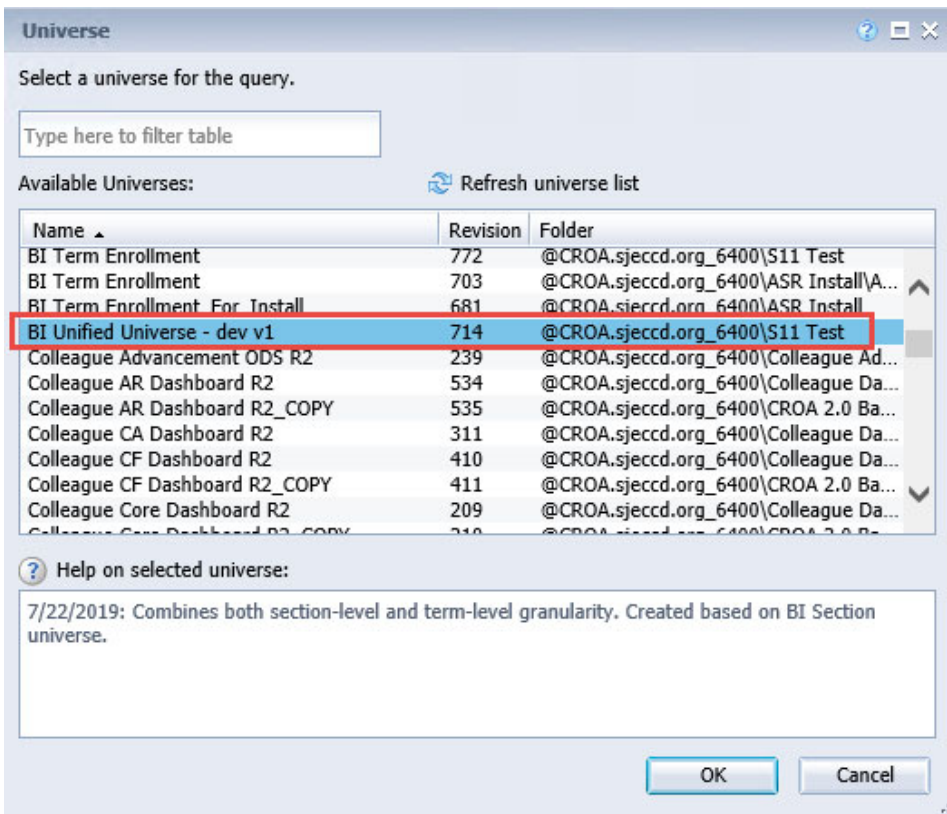
STEP 3: Select the 'Data Access' tab -> 'Tools' sub-tab -> 'Change Source' dropdown and then select the query whose data source you want to change. We will change both query sources, starting with "Enroll" query.



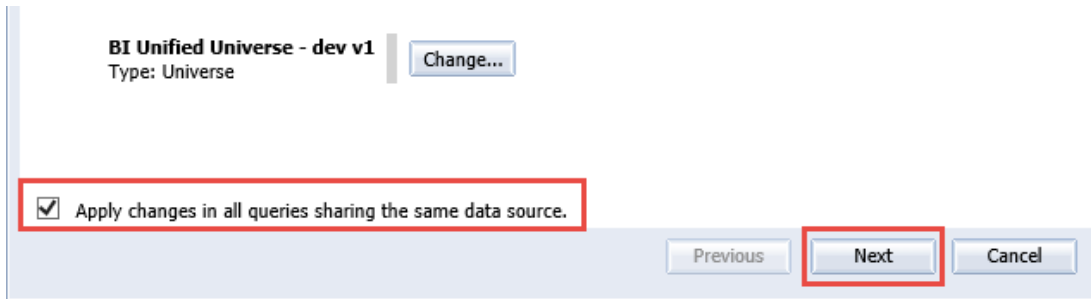
STEP 4: In the 'Change Source Wizard', select the 'Specify a new data source' radio button, click the down arrow next to 'Select a data source', and select 'Universe...'.



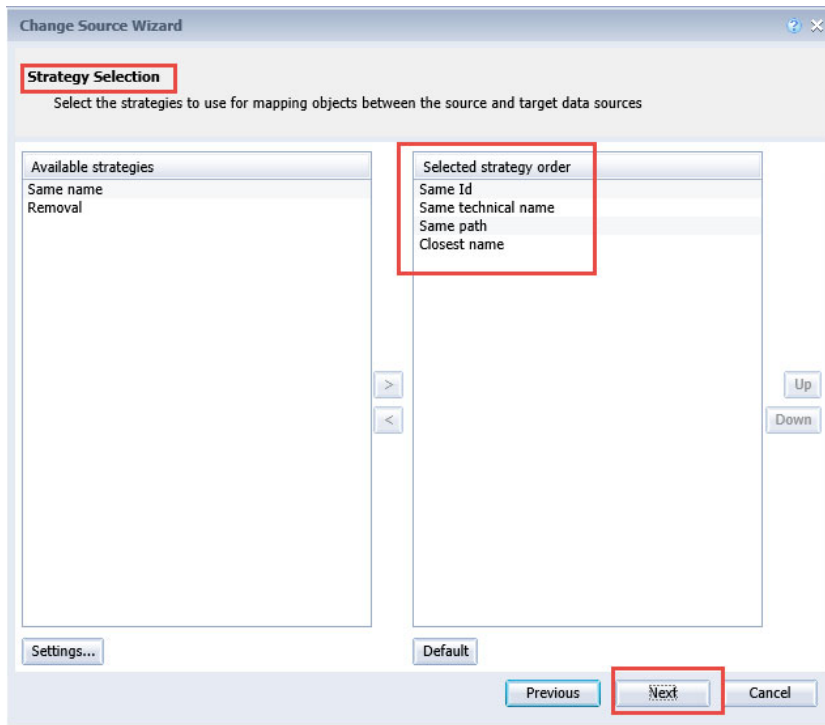
STEP 5: In the 'Universe' window, select the Universe you want to use for migrating this report. Not all users will necessarily have access to all Universes shown here. For this example, use "BI Unified Universe – dev v1" from the "@CROA.sjecd.org_6400\S11 Test" folder. Click 'OK'.



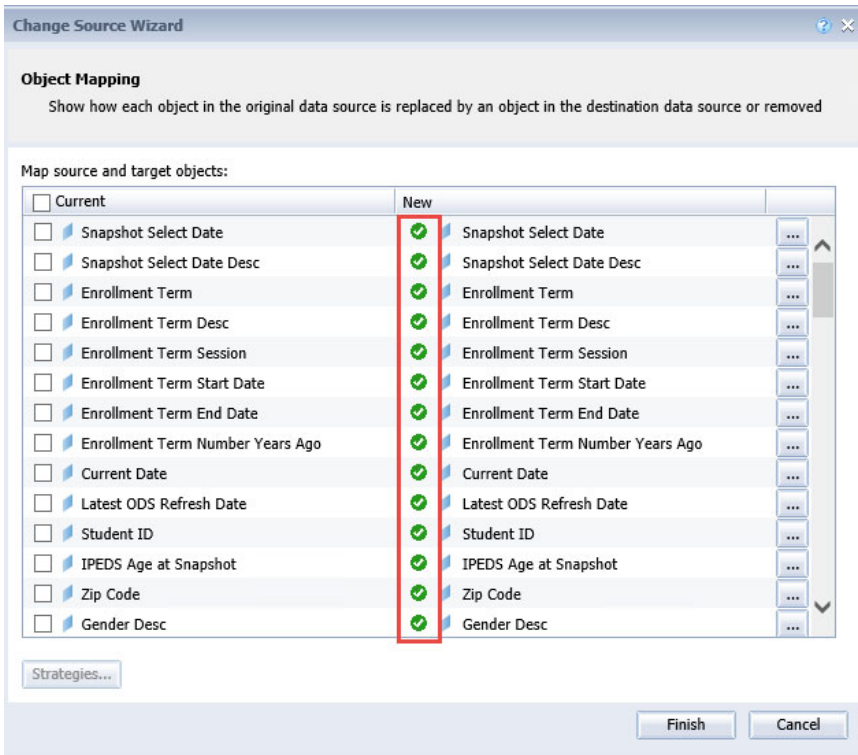
Back in the 'Change Source Wizard' window, click the check box for 'Apply changes in all queries sharing the same data source' located at the bottom of the window and click 'Next'.



STEP 6: Accept the default 'Selected strategy order' in the 'Change Source Wizard' window under 'Strategy Selection' by clicking 'Next'.

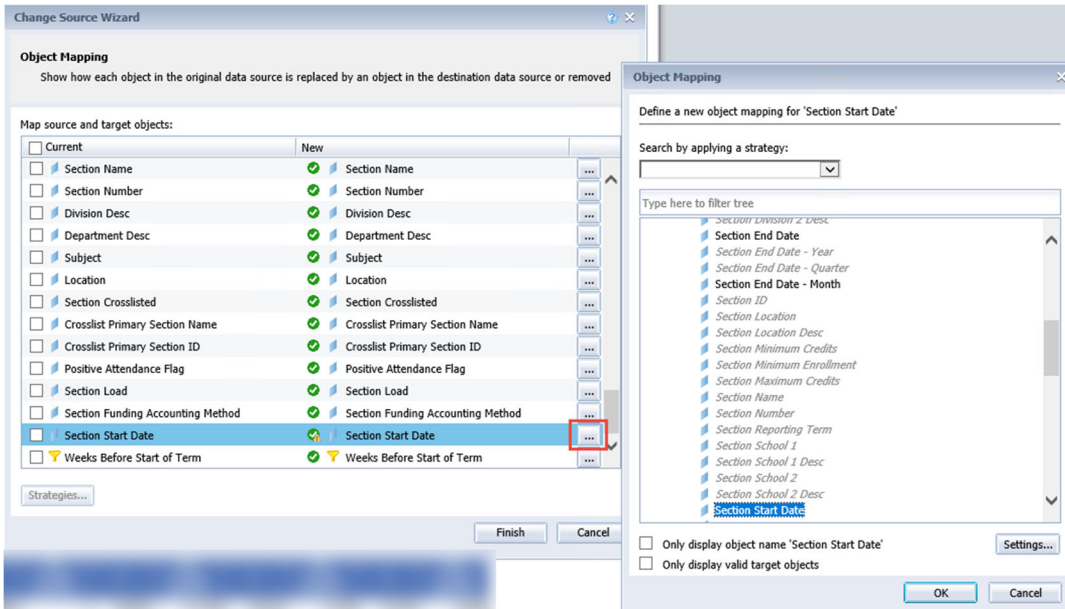


STEP 7: In the 'Change Source Wizard' window, under 'Object Mapping' ensure that all the objects have a green check mark. This means that all objects mapped correctly between the source (current) and target (new) universe. Note that if you are migrating multiple queries using the same original source, objects from all queries will be listed together in this window.

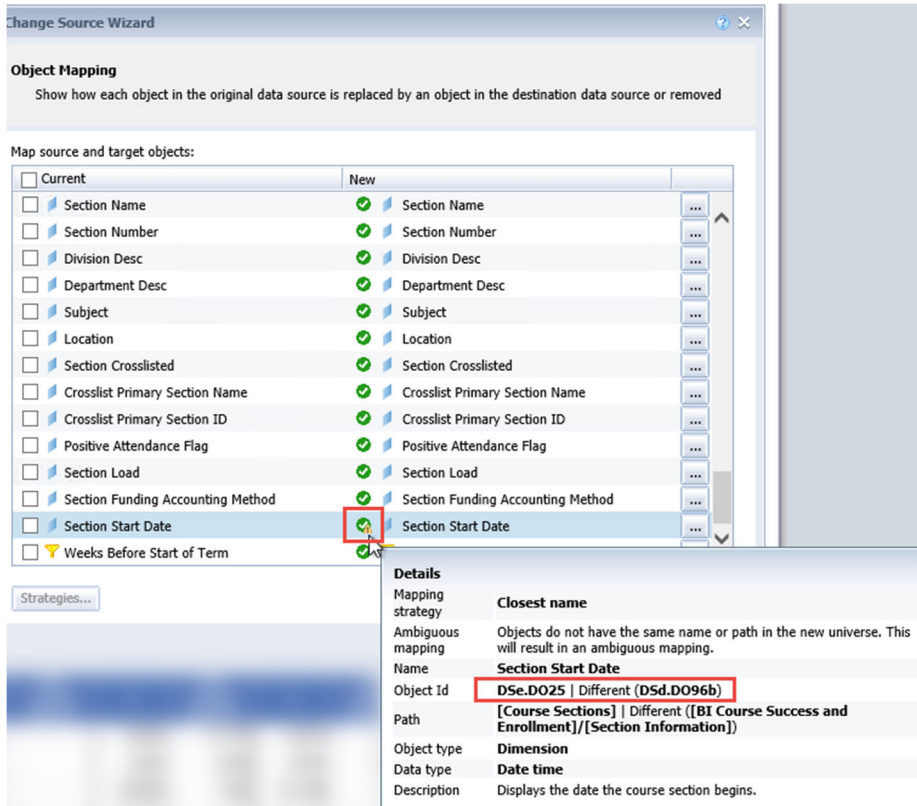


Click 'Finish'.

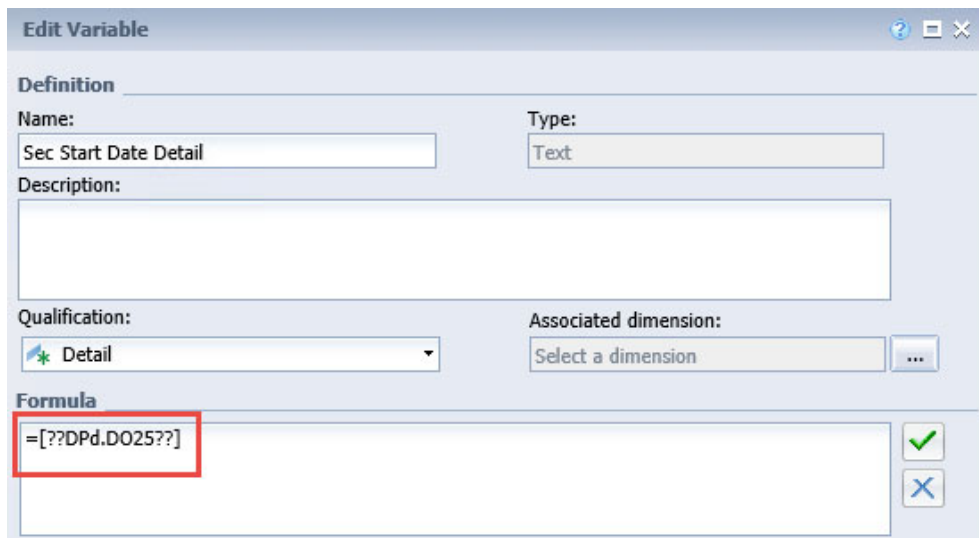
NOTE: For any objects that may not have mapped automatically and do not have a green check mark, click on the ellipsis to manually map the object. In some cases, an object may no longer exist in an updated universe, or the object has been recreated in the new universe and has a new object ID. We recommend removing these objects prior to migrating to a new universe and adding them back in after the migration.



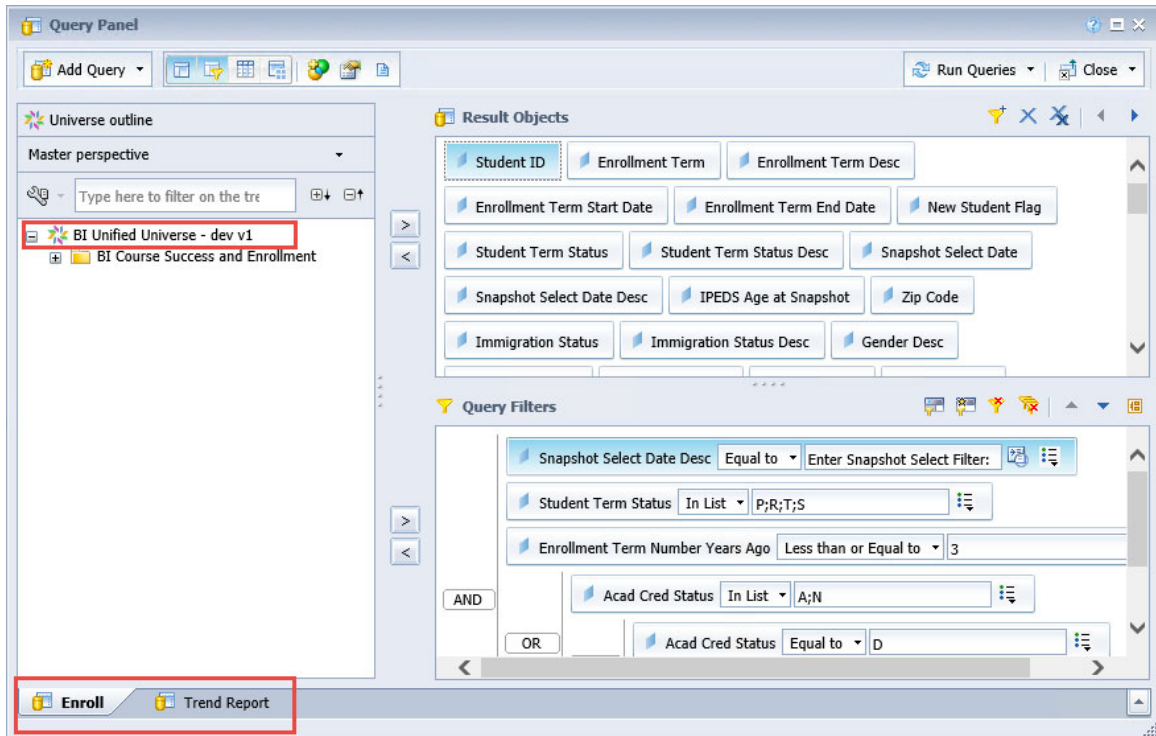
In this case, remapping will not fix this object as it has a new object Id.



If you remove an object that is used in a user defined variable, once it is added back into your migrated report, you will need to also add it back into any dependent variables.



STEP 8: The 'Query Panel' will open. Verify that each query is now using the "BI Unified Universe – dev v1" universe.



STEP 9: Run the queries, answer the query prompts, validate the data, and save the report. Add any objects to the query that you may have removed prior to the migration and check to see if there are any user defined variables dependent on those objects as they will need to be fixed.

UNDERSTANDING AND WORKING WITH NESTED FILTERS AND SUBQUERIES

Data can be filtered at many levels in the flow as shown in Figure 1. The decision to filter data at a given level can depend on several factors including reporting need, performance impact, and space requirements. In general, it is best to query as much data as possible to allow for the greatest flexibility in creating “variations on a theme” reports and “slice and dice” cross tabs – provided the required space and performance expectation are not negatively impacted.

At the database level, data is typically filtered before it is stored so that it is limited to only the data necessary for reporting and analytic needs.

At the universe level, data may be further filtered for specific types of reporting so that it makes sense in the context of the report. For example, the snapshot filters are created in the universe to automatically select the corresponding snapshot name:

The screenshot displays a software interface for defining filters. On the left, a tree view shows a folder named 'Snapshot Filters' containing a list of filter options, each preceded by a yellow funnel icon:

- Weeks Before Start of Term
- Equivalent Offset from Today
- First Day of Registration
- 21 Days Before Start of Term
- 14 Days Before Start of Term
- 7 Days Before Start of Term
- 75 Pct After Start of Term
- 10 Days After End of Term
- 1 Month After End of Term
- First Day of Course Section
- Census After Start of Course Section
- Census by Accounting Method
- Last Day of Course Section
- Current Date

On the right, a detailed view for the 'Current Date' filter is shown. It includes a 'Definition' tab with a funnel icon, a 'Name' field containing 'Current Date', a 'Description' field containing 'Snapshot of the latest version of the data', and a 'Where' field containing the SQL expression: `BI_DIM_SNAPSHOT_DATES.DATE_RANGE = 'Current Date'`.

This section will focus on:

- nested filters, which rely on the use of the AND and OR conditions, within the filter panel both at both the query level and report level
- using subqueries at the query level for more sophisticated filtering needs

Query Filters

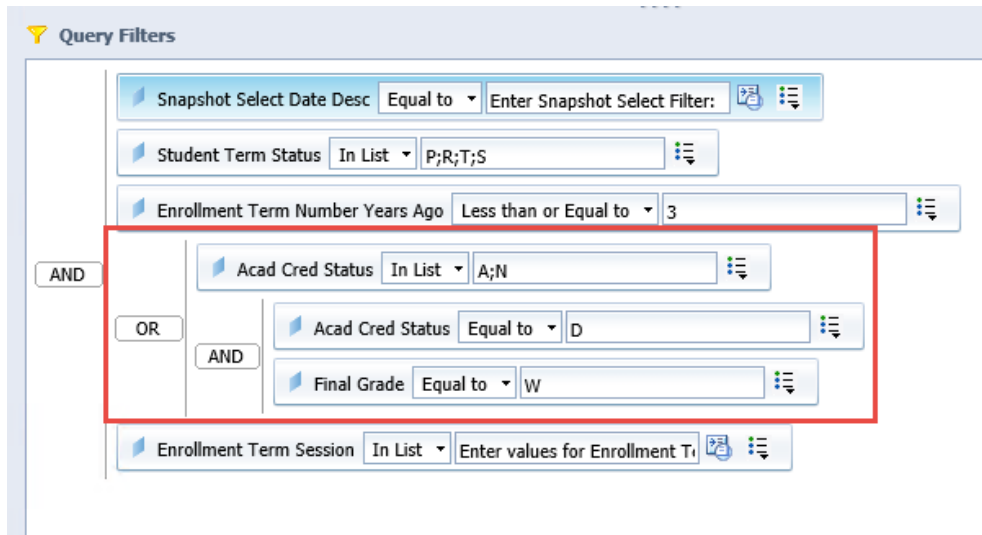
'Query Filters' control what data is selected from the Universe (and ultimately from the source database) and therefore the total scope of data available for the entire Web Intelligence document and all its report tabs and report blocks. The tendency is to think of ONE report result and filter here, when in fact, it can be useful to select all dimension values and create "variations on a theme" reports or "slice and dice" reports that allow analysis of data from many angles. However, query filters have the additional flexibility of being able to filter a value dynamically, based on the value of another data element. This CANNOT be done with report, block, or quick filters.

Remember, data can be included in a query but then filtered or hidden at the report or block level. Use 'Query Filters' carefully - they can powerfully filter the data, but the filter will always apply to this data set for the *entire* document.

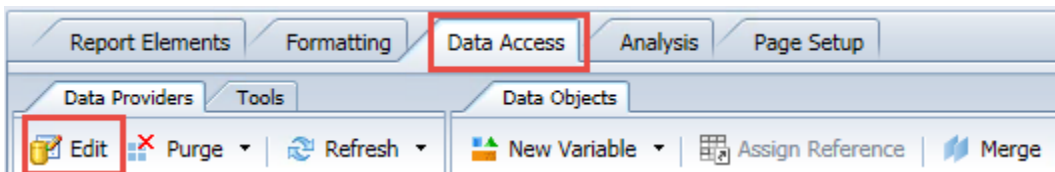
Nested Query Filters

Nested query filters allow the creation of more complex filter conditions than is possible when filters are combined at the same level. When filters are nested, the order is set in which the filters are evaluated.

For example, WebI can restrict the returned data by the two query filters that are nested and joined in an AND relationship (where both filter conditions must be true) and then evaluate the data further by the filter within the OR condition. In this case, both the nested filters in the AND relationship must be true or the condition in the OR relationship.

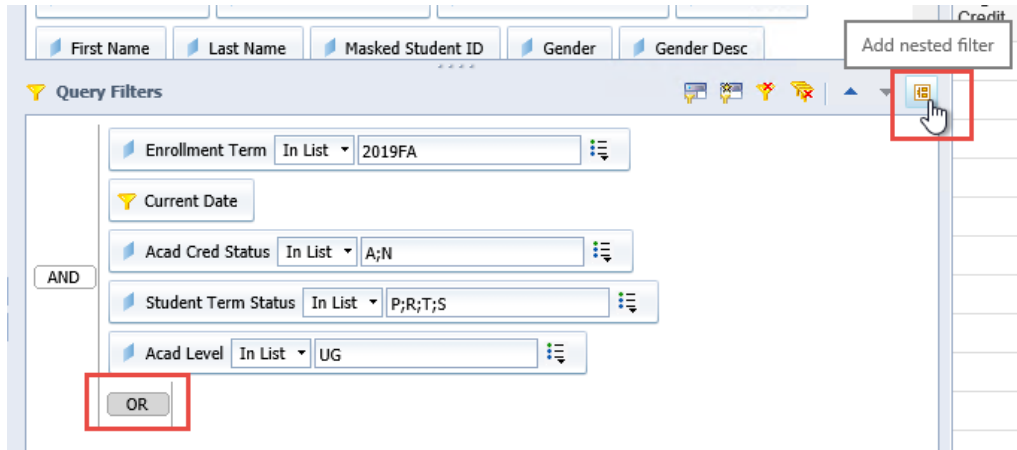


STEP 1: Select the Web Intelligence document you wish to modify. In our example, we are using the “*Sample Enrollment Report*” report found in the “*Public Folders*” → “*SJECCD Training*” → “*Advanced Webi Training*” folder. If you have not already done so, copy this report to your “*My Favorites*” folder. Open the document in ‘Design’ mode, navigate to the ‘Data Access’ tab, and click the ‘Edit’ button.

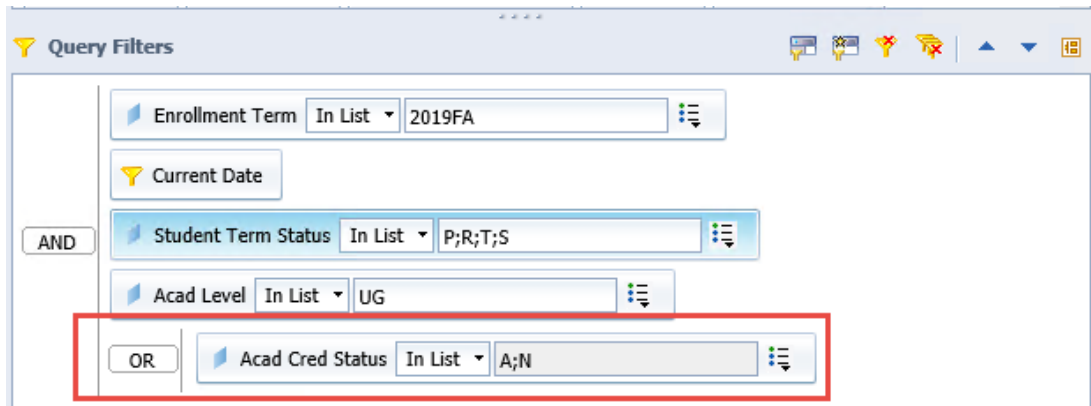


A dialog box titled ‘Query Panel’ will appear displaying the selected objects in the ‘Result Objects’ pane and filters in the ‘Query Filters’ pane.

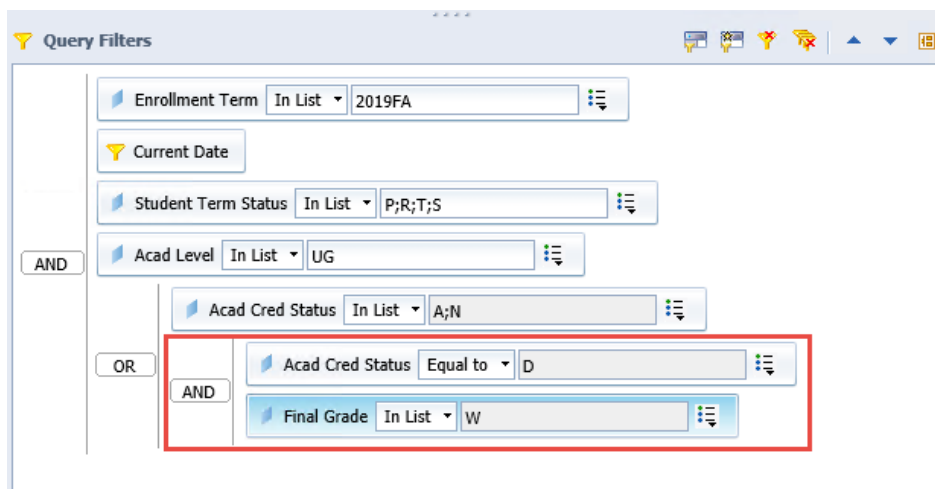
STEP 2: Click on the last filter in the ‘Query Filters’ pane and select ‘Add nested filter’ by clicking on the last button in the ‘Query Filters’ pane. The nested conditions should now be available.



Add the "Acad Cred Status" query filter to the 'OR' condition by selecting and dragging it in front of the 'OR' condition or by selecting the 'OR' condition, deleting the existing "Acad Cred Status" query filter, and re-adding it from the 'Universe' pane.



STEP 3: Click on the "Acad Cred Status" query filter and another nested filter. Add the following filters in front of the 'AND' condition:



You can change the 'And'/'Or' conditions by double clicking on them.

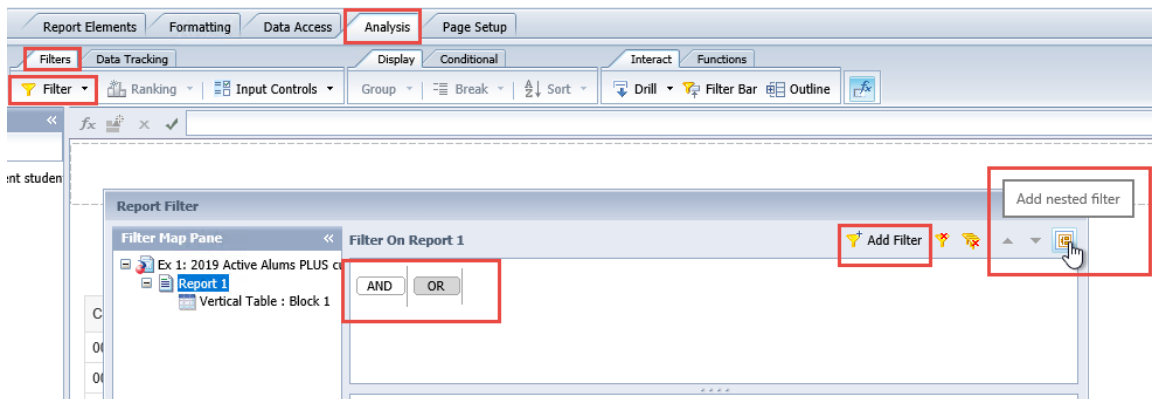
STEP 4: Click 'Run Query' and the report will be refreshed with the filtered data. Click the 'Save' icon and then close. **Save early and often!**

Nested Report Filters

A Report filter controls what data is available and shown on an entire report (tab) and all its blocks, even if there are more data values available from the resulting query. This is a useful way to make multiple reports in the same document but show slightly different versions of the same basic data or display. Nested report filters are set up and behave in a similar way to nested query filters, but they are happening after the query is run and only restricted the data you see in a report document or tab.

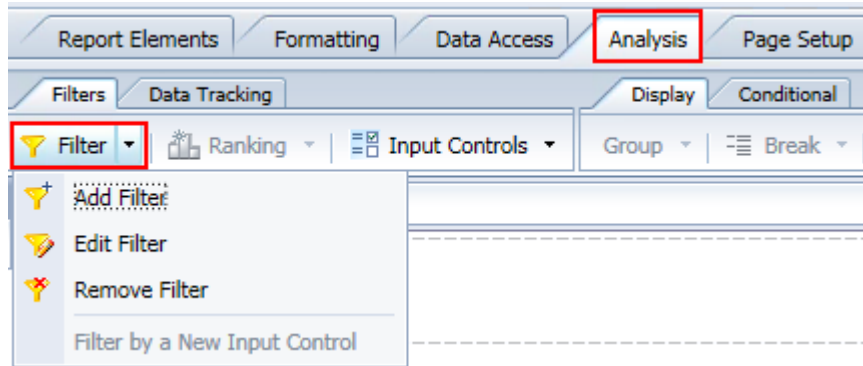
STEP 1: In any report, under the 'Analysis' tab, navigate to the mini-tab 'Filters', and click the 'Filter' button on the far left of the toolbar. A 'Report Filter' dialog box will appear. **NOTE:** This is NOT the same as the 'Filter Bar' button on the right side of the toolbar. That is used for simple report filters.

STEP 2: As with the 'Query filter' pane, at the right end of the 'Report Filter' window, select the 'Add nested filter' option. Click the 'Add Filter' button and an 'Available objects' dialog box will appear allowing you to select data objects to filter on and nest as needed.



STEP 3: Add any additional data objects and select the Boolean comparison of 'And', 'Or' by double clicking on the word to change it, if necessary. Like query filters, it is important to understand the logic involved and how including, excluding, negating, etc. impacts the resulting data.

STEP 4: To edit or remove the 'Report Filter', just select the dropdown next to the 'Filter' button ('Analysis' tab → 'Filters' mini-tab) and choose the option desired.

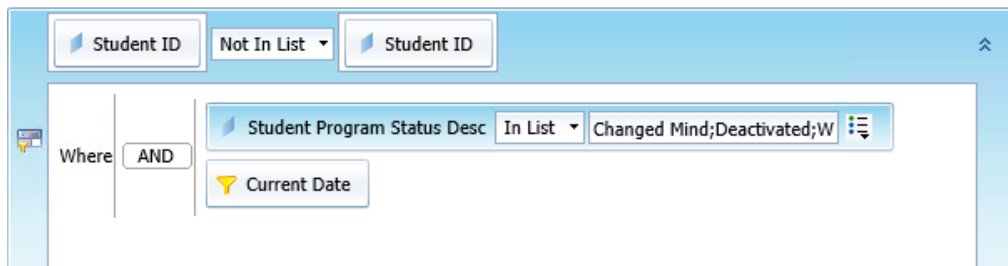


STEP 5: Click the 'Save' button and close. **Save early and often!**

Filtering Data Using Subqueries

A subquery is a flexible query filter that allows you to restrict data in a more sophisticated way than is possible with an ordinary query filter. Subqueries are more powerful than ordinary query filters as:

- They allow you to restrict the values returned by the subquery with a WHERE clause
- They allow you to compare object values used to restrict the query with values from other result objects (i.e. 'Student ID' **Not In List** of 'Student ID' Where 'Student Program Status Desc' has **In List** Changed Mind; Deactivated; Withdrawn; Graduated)
- You can also limit the subquery with additional filter criteria in order to limit the pool of possible records for comparison and increase performance



- They allow you to create complex query filter scenarios that are difficult, if not impossible, to formulate with an ordinary query filter

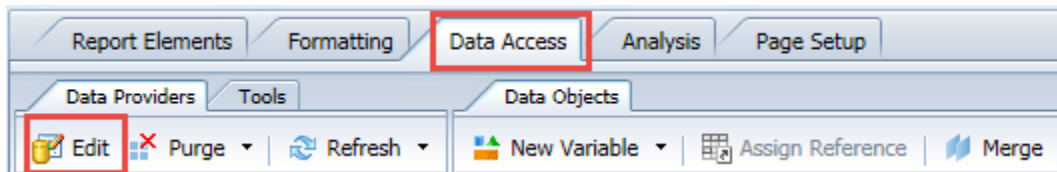
In this example, students could have more than one program status and if they have any records with a status of Changed Mind, Deactivated, Withdrawn, or Graduated we want to eliminate the student completely from our report results.

Student ID	Enrollment Term	Last Name	Acad Program of Study	Acad Program of Study Desc	Student Program Status Desc
1000000001	2019FA	SMITH	2BUS.AA.1	Business Administration, AA (EVC)	Active
1000000002	2019FA	SMITH	2GEN.AA.5	General Studies: Sociology, AA (EVC)	Active
1000000003	2019FA	SMITH	2GEN.AA.3	General Studies: Health Science, AA (EVC)	Active
1000000004	2019FA	SMITH	2GEN.AA.5	General Studies: Sociology, AA (EVC)	Active
1000000005	2019FA	SMITH	1LIB.AA.3	Liberal Arts: Social and Behavioral Sciences, AA (SJCC)	Active
1000000006	2019FA	SMITH	2CADD.AS.1	CADD Tech: Computer Aided Design and Drafting, AS (EVC)	Active
1000000007	2019FA	SMITH	2GEN.AA.5	General Studies: Sociology, AA (EVC)	Active
1000000008	2019FA	SMITH	2PSYCH.AA.1	Psychology, AA (EVC)	Active
1000000009	2019FA	SMITH	2MATH.AS.1	Mathematics AS-T CSU (EVC)	Active
1000000010	2019FA	SMITH	2UND	Undecided	Changed Mind
1000000011	2019FA	SMITH	1GLOB.AAT.1	Global Studies	Active

Results with the sub query in place showing both records removed for the student highlighted above:

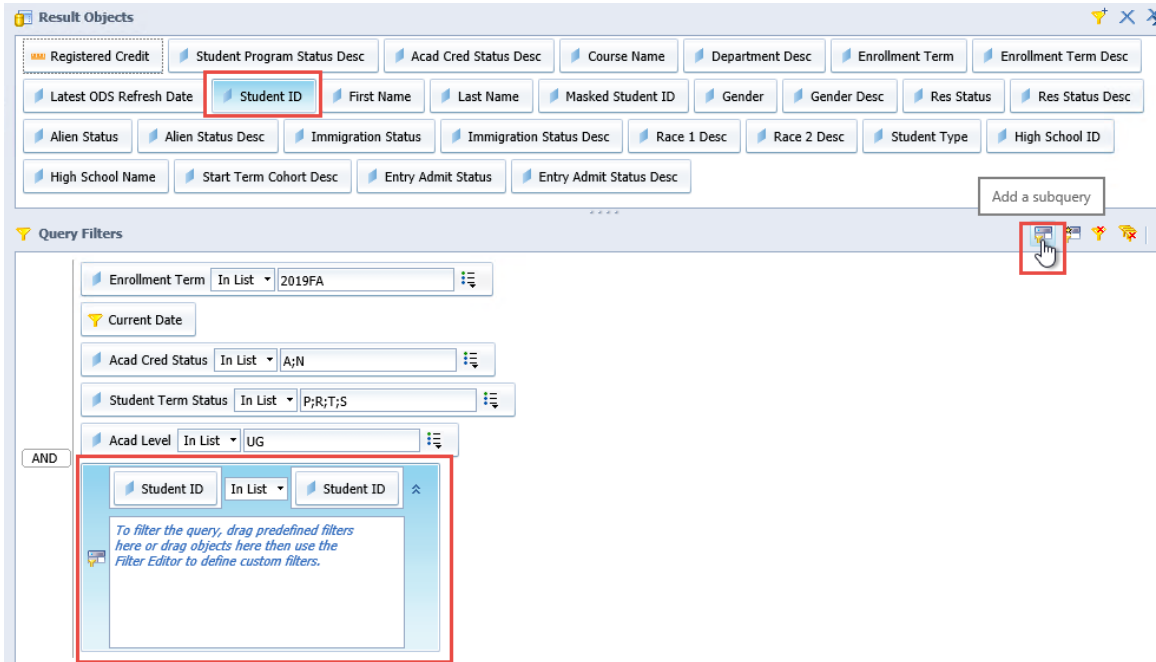
Student ID	Enrollment Term	Last Name	Acad Program of Study	Acad Program of Study Desc	Student Program Status Desc
1000000001	2019FA	SMITH	2BUS.AA.1	Business Administration, AA (EVC)	Active
1000000002	2019FA	SMITH	2GEN.AA.5	General Studies: Sociology, AA (EVC)	Active
1000000003	2019FA	SMITH	2GEN.AA.3	General Studies: Health Science, AA (EVC)	Active
1000000004	2019FA	SMITH	2GEN.AA.5	General Studies: Sociology, AA (EVC)	Active
1000000005	2019FA	SMITH	1LIB.AA.3	Liberal Arts: Social and Behavioral Sciences, AA (SJCC)	Active
1000000006	2019FA	SMITH	2CADD.AS.1	CADD Tech: Computer Aided Design and Drafting, AS (EVC)	Active
1000000007	2019FA	SMITH	2GEN.AA.5	General Studies: Sociology, AA (EVC)	Active
1000000008	2019FA	SMITH	2PSYCH.AA.1	Psychology, AA (EVC)	Active
1000000011	2019FA	SMITH	1GLOB.AAT.1	Global Studies	Active

STEP 1: Select the Web Intelligence document you wish to modify. In our example, we are using the “Subquery Example” report found in the “Public Folders” → “SJECED Training” → “Advanced Webi Training” folder. If you have not already done so, copy this report to your “My Favorites” folder. Open the document in ‘Design’ mode, navigate to the ‘Data Access’ tab, and click the ‘Edit’ button.



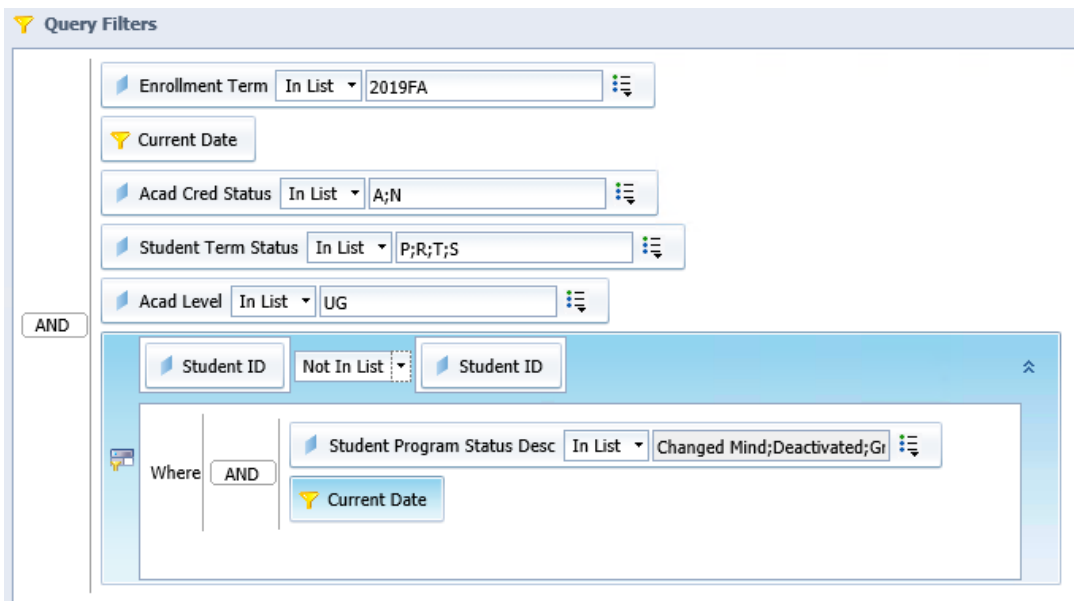
A dialog box titled ‘Query Panel’ will appear displaying the selected objects in the ‘Result Objects’ pane and filters in the ‘Query Filters’ pane.

STEP 2: Click on the “Student ID” object in the ‘Result Objects’ pane to select it and then select ‘Add a subquery’ by clicking on the first button in the ‘Query Filters’ pane. The subquery filter box should now be available.



Change the operator in the subquery to 'Not In List' and add in the "Student Program Status Desc" object, from the "Student Information/Multivalued Student Dimensions" sub class using the 'Universe' pane, to the bottom of the subquery. Select "Changed Mind; Deactivated; Graduated; Withdrawn" from the 'Value(s) from list' define filter type option.

Add the "Current Date" filter from the "Snapshot Filters" class to your sub query as well. Your 'Query Filters' pane should look like something this:



Learning Objective

In this section you will work existing reports to add nested query filters and a subquery filter.

High-level Overview

1. Navigate to the report you wish to access
2. Open the document
3. Navigate to the 'Query Panel' and add nested query filters or a subquery
4. Run query to get the data
5. Save and close

Practice Steps for Nested Query Filters

STEP 1: Log in to BI Launch Pad, select the 'Documents' tab, and navigate to "*Public Folders*" → "*SJECCD Training*" → "*Advanced Webi Training*". In this part of the exercise, we will use the "*Sample Enrollment Report*" document.

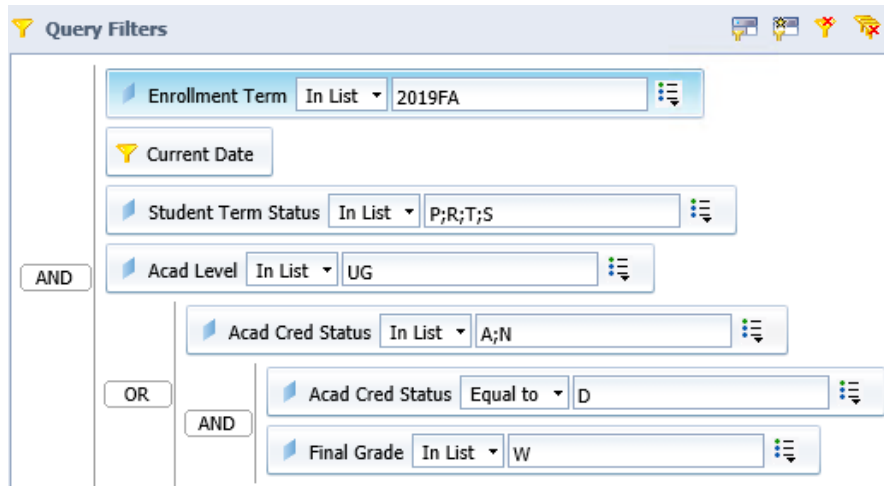
STEP 2: Open the document in 'Design' mode, navigate to the 'Data Access' tab, and click the 'Edit' button.

STEP 3: Add a nested 'Or' condition filter at the bottom of the current set of query filters and place the current filter for "*Acad Cred Status*" in front of it. Add a second nested 'And' condition filter.

Add the following filters to the nested 'AND' condition:

- "*Acad Cred Status*" from the "*Student Enrollment Information*" class and set the operator to 'Equal to' and setting the operand to "*D*"
- "*Final Grade*" from the "*Student Enrollment Information*" class leaving the default operator of 'In List' and setting the operand to "*W*"

Your 'Query Filters' pane should look something like this:



STEP 4: Click the 'Run Query' button and review the results.

STEP 5: Save and close the report.

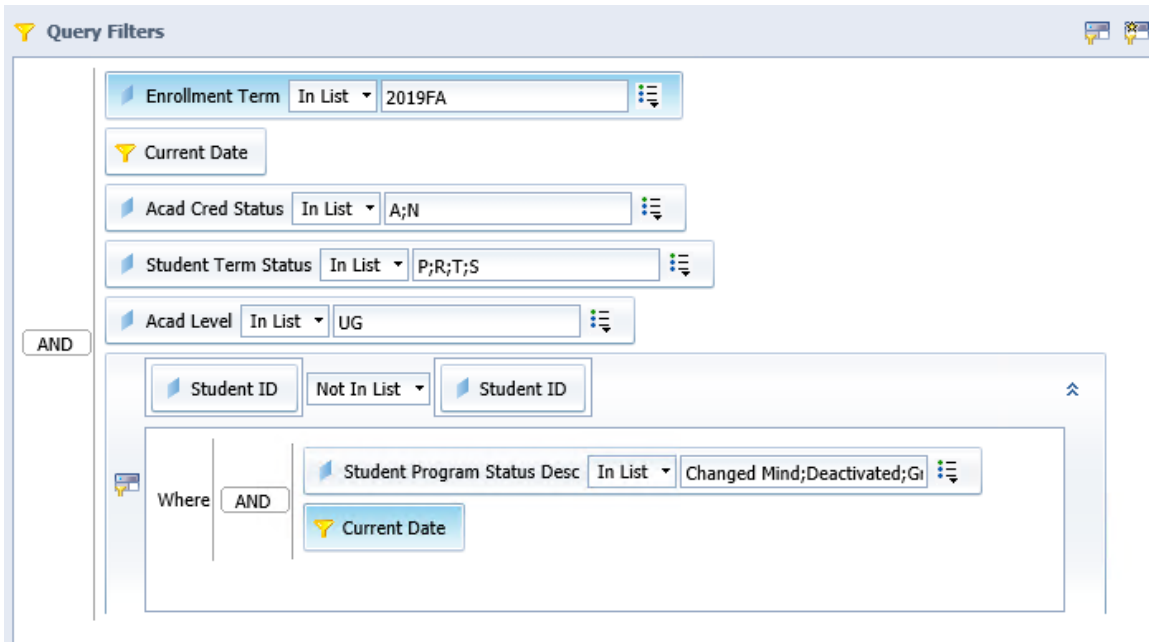
Practice Steps for Building a Sub Query

STEP 1: Log in to BI Launch Pad, select the 'Documents' tab, and navigate to "Public Folders" → "SJECCD Training" → "Advanced Webi Training". In this part of the exercise, we will use the "Subquery Example" document.

STEP 2: Open the document in 'Design' mode, navigate to the 'Data Access' tab, and click the 'Edit' button.

STEP 3: Add a subquery for the "Student ID" object that selects records that are Not In List of those student ID's with a program status description of Changed Mind, Deactivated, Graduated or Withdrawn. Add a second query filter to the subquery that restricts the list of student ID's to those that are in the current date snapshot.

Your 'Query Filters' pane should look something like this:



STEP 4: Click the 'Run Query' button and review the results.

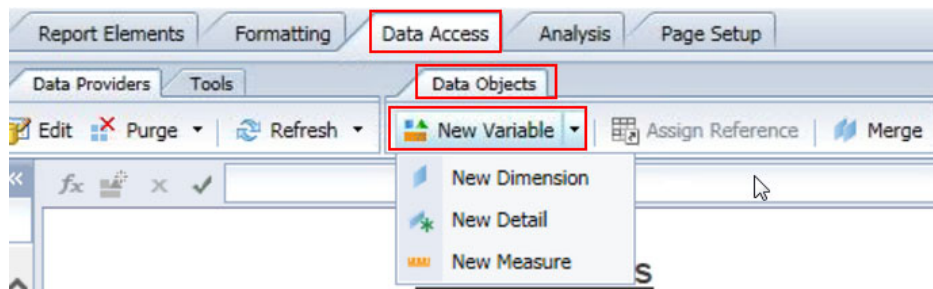
STEP 5: Save and close the report.

CREATING AND USING VARIABLES

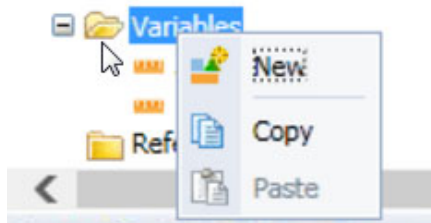
In this section you will learn how to create user defined custom variables that can be used throughout the document. Custom variables are a powerful tool and enable you to manipulate and calculate data. There are numerous functions and potential formulas that you can create including using one custom variable within another custom variable. This section will focus on formulas that we use frequently.

There are several ways to create a new variable. You can choose any of the following:

- From the 'Data Access' tab -> 'Data Objects' tab click on the dropdown arrow next to 'New Variable' and select the type that best suits your needs



- Right-click on the 'Variables' folder and select 'New'



- Select an existing column that contains an in-place formula and turn it into a reusable variable, view the expression in the 'Formula Bar', and click on the 'Create Variable' button

Enrollment Term	Department Desc	Course Name	Registered Credit	Student Count
2019FA	EVC Accounting	ACCTG-001A	864	216
2019FA	EVC Accounting	ACCTG-001B	500	125

Using If-Then-Else Logic in Report Variables

When creating formulas in Web Intelligence reports, it is frequently necessary to do conditional logic, that is perform a logical test that determines what the output of the formula will be. The most straightforward way to accomplish this is to use the 'If' function. The format of this function is as follows:

If <Condition> **Then** <Expression1> **Else** <Expression2>

In this definition, <Condition> is a logical True/False expression and <Expression1> and <Expression2> are any formulas that return numeric, string, or date results. Note that you can also embed another 'If' formula surrounded by parenthesis in place of either <Expression1> or <Expression2> to express more complex logical constructs.

For example, we may want to create a variable that characterizes students according to their credit load as either full-time or part-time. In this case, we'll assume that 12 or more credits is considered full-time.

This could be accomplished by creating a variable called "*PT/FT Status*" which would have the following definition:

=If [Registered Credit] >= 12 Then "FT" Else "PT"

Dragging this new variable into the report would result in the following:

Admit Status Desc	Acad Cred Status Desc	Registered Credit	PT/FT Status
First Time Student	New	15	FT
Returning Student	New	7	PT
First Time Student	New	3	PT

Another very useful application of conditional logic is to use it to conditionally count or aggregate values. As an example of this technique, imagine that we wanted a count of students based on their residency status. To do this, you would embed the 'If' statement WITHIN the 'Count' function call, as below:

For In-State Residency Status:

=Count (if [Res Status] = "INST" Then [Student ID])

For Out-of-State Residency Status:

=Count (if [Res Status] = "OST" Then [Student ID])

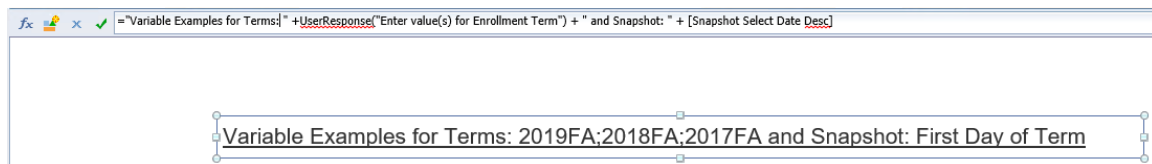
Note that since there is no 'Else' clause on our 'If' statement within the 'Count' function, in the case where the *"Res Status"* object does not equal "INST" for our In State Variable or "OST" for our Out of State variable, there is no *"Student ID"* value returned and hence no value added to the overall count.

Building a Dynamic Title

Anytime a query has a filter that prompts the user for a response, this response can be added to the title of the document creating a dynamic title that will automatically refresh when a prompt value changes. This requires the UserResponse() function and the format used is as follows:

UserResponse("Enter value(s) for Enrollment Term")

Note: the text within the parentheses must be identical to the text in the query filter.



Time Functions

There are several functions that deal specifically with date and time. This section will highlight a few of the functions we use most often.

- CurrentDate(): Returns today's date
=CurrentDate()
- DayName(): Turns a date into a day of the week
=DayName([Enrollment Term Start Date])
- DaysBetween(): Calculates the number of days between two dates
=DaysBetween([Enrollment Term Start Date];[Snapshot Select Date])
- MonthsBetween(): Calculates the number of months between two dates; add *"/12"* to return years
=MonthsBetween([Snapshot Select Date];[Current Date])
=MonthsBetween([Snapshot Select Date];[Current Date])/12
- DatesBetween(): Calculates the number of days, weeks, months, or years between two given dates, based on the time period parameter included in the formula. Functions like

DaysBetween() and MonthsBetween() but is more flexible with the addition of the parameter.

=DatesBetween([Session Start Date];[Snapshot Select Date];WeekPeriod)

- Round(): Rounds a number to the number of decimal places specified by the user (1 in the example below). Follows standard rounding rules.

=Round(MonthsBetween([Snapshot Select Date];[Current Date])/12;1)

- Ceil(): Ceiling function; rounds a number up to the next nearest integer (not standard rounding)

=Ceil(MonthsBetween([Snapshot Select Date];[Current Date])/12)

- Floor(): Opposite of Ceil(); rounds a number down to the next nearest integer (not standard rounding)

=Floor(MonthsBetween([Snapshot Select Date];[Current Date])/12)

Enrollment Term	Enrollment Term Year	Enrollment Term Start Date	Snapshot Select Date	Current Date	Snapshot Select Date Desc	Days Between Enrollment Term Start Date and Census	Months between Census and Now
2017FA	2017	9/5/17	9/18/17	10/24/19	Census After Start	13	25
2018FA	2018	9/4/18	9/17/18	10/24/19	Census After Start	13	13
2019FA	2019	9/3/19	9/16/19	10/24/19	Census After Start	13	1

Character Functions

As there are several functions that deal specifically with date and time, there are also several character or string functions available as well. This section will highlight a few of the functions we use most often.

The + is used concatenate two string together, for example:

= [Department Desc] + " - " + [Course Name]

The screenshot shows a spreadsheet interface. The formula bar at the top contains the formula: `= [Department Desc] + " - " + [Course Name]`. Below the formula bar is a table titled "Course Enrollment Summary Report".

Enrollment Term	Department Desc	Department and Course name	Course Name	Registered Credit	Student Count
2019FA	EVC Accounting	EVC Accounting - ACCTG-001A	ACCTG-001A	864	216
2019FA	EVC Accounting	EVC Accounting - ACCTG-001B	ACCTG-001B	500	125
2019FA	EVC Accounting	EVC Accounting - ACCTG-030	ACCTG-030	51	17

Substr(): Returns part of a string; user provides the string, the starting point, and the length to return.

=Substr([Enrollment Term];1;4)

Enrollment Term	Enrollment Term Year
2017FA	2017
2018FA	2018
2019FA	2019

POS(): Returns the starting position of specified character(s) in a string

=POS([High School Name]; "HIGH")

HS POS	High School Name
0	MARIA AUXILIADORA
9	PEGASUS HIGH
0	HARGRAVE MILITARY ACADEMY
0	GUAJOME PARK ACADEMY
0	No Value Entered
11	AGUA FRIA HIGH SCHOOL
14	SILVER CREEK HIGH SCHOOL
18	EVERGREEN VALLEY HIGH SCHOOL
18	EVERGREEN VALLEY HIGH SCHOOL
16	PIEDMONT HILLS HIGH SCHOOL
9	OAKLAND HIGH SCHOOL
14	INDEPENDENCE HIGH SCHOOL

You can create variables that use previously created variables. This method allows you to create more complex variables one step at a time ensuring that each piece is working the way you would expect. For example, in the following variable, we will use the POS() variable in a SUBSTR() variable. Using this technique, the length to be returned is determined dynamically.

=Substr([High School Name];1;[HS Pos]-1)

This will return everything found before "HIGH" in the high school name. The "-1" ensures that it doesn't return the space before "HIGH".

AGUA FRIA HIGH SCHOOL	AGUA FRIA
SILVER CREEK HIGH SCHOOL	SILVER CREEK
EVERGREEN VALLEY HIGH SCHOOL	EVERGREEN VALLEY
EVERGREEN VALLEY HIGH SCHOOL	EVERGREEN VALLEY
PIEDMONT HILLS HIGH SCHOOL	PIEDMONT HILLS
OAKLAND HIGH SCHOOL	OAKLAND
INDEPENDENCE HIGH SCHOOL	INDEPENDENCE

Learning Objective

In this section you will work with an existing report to create a dynamic title.

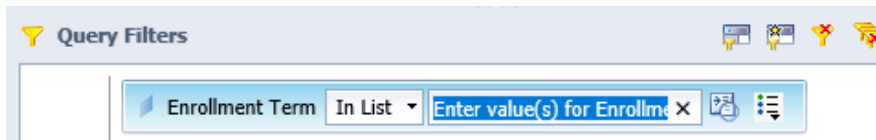
High-level Overview

1. Navigate to the report you wish to access
2. Open the document
3. Create a dynamic title in the report
4. Create custom variables
5. Save and close

Practice Steps

STEP 1: Log in to BI Launch Pad, select the 'Documents' tab, and navigate to "Public Folders" → "SJECCD Training" → "Advanced Webi Training". We will use the "Sample Enrollment Report" that we worked with in an earlier exercise.

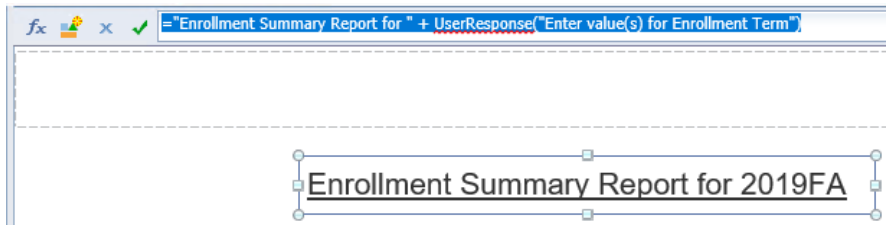
STEP 2: Navigate to the 'Query Panel' and copy the text used in the prompt query filter for "Enrollment Term" and close the 'Query Panel'.



STEP 3: Click on the title and create the following formula:

= "Enrollment Summary Report for " + UserResponse("Enter value(s) for Enrollment Term")

Your resulting title should look something like this:



STEP 4: Save your report.

STEP 5: If it's not already in the table, add the *"High School Name"* object.

STEP 6: Create a variable to determine the position of "HIGH" in the returned values for the *"High School Name"* object.




STEP 7: Create a variable that only returns the first part of the high school name (minus anything after finding the word "HIGH" and any spaces before the word "HIGH").

STEP 8: Add your new variables to the report and validate the results.

STEP 9: Save and close your report.

CALCULATION CONTEXTS

By default, Web Intelligence determines the result of a measure using (1) the dimension objects placed in the same table block and (2) the level of placement in a report. The measure will default to the lowest level of aggregation depending on where it is placed. The set of relevant dimensions for which a measure is aggregated, based on its position in a block on a report, is called the “Calculation Context” for the measure. As an example, if a measure is included as a column of a ‘List’ block with a set of dimensions, then the measure will be aggregated by each combination of values of those dimensions (i.e. for every row in the list) and that set of dimensions is the calculation context for the measure in the block. If the measure is placed on the summary line of a ‘List’ block, then the measure is aggregated independent of the dimensions in the list, which are no longer part of its calculation context, as below:

fx    =[Registered Credit]

Enrollment Term	Gender	IPEDS Race Ethnic Classification	Registered Credit
2019FA	F	American Indian	439.5
2019FA	F	Asian	21,541
2019FA	F	Black or African American	2,217.5
2019FA	F	Hawaiian/Pacific Islander	235
2019FA	F	Hispanic	36,217.5
2019FA	F	Two or More Races	1,750.5
2019FA	F	Unknown	11,607.5
2019FA	F	White	3,896.5
2019FA	M	American Indian	227.5
2019FA	M	Asian	18,085
2019FA	M	Black or African American	2,034.5
2019FA	M	Hawaiian/Pacific Islander	194
2019FA	M	Hispanic	25,281.5
2019FA	M	Two or More Races	1,425
2019FA	M	Unknown	10,853
2019FA	M	White	3,479
		Sum:	139,484.5

In the example above, when the “Registered Course Credits” measure is placed at position 1, its calculation context is the combination of values of the “Enrollment Term”, “Gender”, and “IPEDS Race Ethnic Classification” dimensions. When it is placed at position 2 (in the summary row of the ‘List’ table), the calculation context doesn’t contain any dimensions, and it is aggregated for all values of all dimensions.

For most cases, the default calculation context determined by Web Intelligence is the one you probably want to use. However, there are several reporting circumstances where you may want to alter or specifically define the calculation context to be used for a measure in your report.

Examples of when you might want to do this might include:

- You want each line of your report to compare the enrollment for a specific term with the average enrollment for all terms in the report (i.e. are we above or below the average for this term)
- You want to create a report with a single row with the summarized enrollment for a term but also include columns providing data about subtotals of that enrollment, such as the total enrollment of certain student types
- You want to calculate a delta value between the total enrollment in the first term included in a report with the enrollment for the last term included in the report

A complete review of the capabilities that Web Intelligence provides to work with calculation contexts would require a separate course in of itself and is well out of the scope of this one. However, this section will include a brief overview of the functions provided for report designers to define and modify calculation contexts and will present a couple of specific examples of their use.

Calculation context specifications are expressed in Web Intelligence as additional keywords and query object expressions in the aggregation functions you are already used to, such as ‘Count’ or ‘Sum’. The following table summarizes the options for these calculation context expressions, along with the meaning and a simple example:

Operator	Definition
In – can be used for both input and output contexts	Used to specify dimensions explicitly – must include dimensions in the formula even if they are in the table/block. Can be used with extended syntax words (Report, Section, Break, block, body – see Keywords Table below for definitions) Formula: <code>=Max([Registered Course Credits] In ([Enrollment Term] ; [Enrollment Department 1 Desc]))</code>
Where	Specifies limiting conditions on the data:

	<p>Formula:</p> <pre>=Sum([Total Invoice amt including taxes]) Where([Invoice Item AR Code] = "BKSUP")</pre>
ForEach – use as an input context	<p>Adds dimensions to the context (Automatically takes dimensions into consideration that are already in the table/block)</p> <p>Formula:</p> <pre>=Max([Registered Course Credits] ForEach ([Enrollment Term]))</pre>
ForAll – use as an output context	<p>Removes dimensions from the context – provides a more global calculation at a granular, row-by-row level</p> <p>Formula:</p> <pre>=Average([Student Academic Level Cum Gpa]) ForAll ([Student ID])</pre>

The 'in' calculation context keyword allows you to express the complete list of dimensions for which you are aggregating the measure. In many cases, you want to specify that the context is that of an enclosing report structure, such as a report break or section, without specifying a list of dimensions. To support this, the 'in' calculation context keyword supports the following options:

Environment Level	Effect
Report	The projected value of the measure is aggregated for all dimensions contained within the page of the report
Section (Same result as Block)	The projected value of the measure is aggregated for all dimensions contained within the section of the report
Break	The projected value of the measure is aggregated for all dimensions contained within the break of the table **must use as input context (syntax) in order to get this to work**
Block (Same result as Section)	The projected value of the measure is aggregated across all values for dimensions contained within the block
Body	The projected value of the measure is aggregated for all dimensions at the level in the report that it is placed

Finally, there are two basic types of calculation contexts: Input Contexts and Output Contexts. The difference is subtle and somewhat technical. An input calculation context consists of any dimension objects that need to be included directly IN the calculation itself and cause aggregation to happen at a finer level of detail than what is in the table/block. An output calculation takes dimension objects out of the default context calculation and causes the aggregation of the results at a higher or more global level of detail.

The syntax of the two types is slightly different, but very important as it will cause very different behaviors of the aggregating measure. The key syntax difference between input and output contexts is the position of the operator relative to the parenthesis.

- Input context:
 - o Operator is placed **within** the parenthesis of the aggregate:
 - Aggregate (measure **OPERATOR** (dimension))
 - =Max([Registered Course Credits] **In** (Enrollment Term) ; [Enrollment Department 1 Desc])
- Output context:
 - o Operator is placed **outside** of the parenthesis that contain the aggregate:
 - Aggregate (measure) **OPERATOR** (dimension)
 - =Average ([Student Academic Level Cum Gpa] **ForAll** ([Student ID])

We are going to look at the following specific applications of the calculation context functionality:

- Using the 'IN' context operator – input context
- Using the 'ForAll' context operator – output context
- Using the 'Where' conditional context operator – output context
- Using the 'In Report' context operator – output context

Using the IN Context Operator

You would use the 'IN' context operator when you want to aggregate a measure by additional dimensions beyond what is in your table/block. If the dimension is available from your query it can be used in your calculation. The IN operator requires all dimensions needed in the calculation context to be listed in the formula even if they already exist in the table/block.

To see the use of this context operator, consider a simple report where we have a list of registered credit totals for departments by term. We have a break on department and have a descending sort on registered credits:

Department Desc	Enrollment Term	Registered Credit
EVC Accounting	2019FA	1,813
	2018FA	1,745
	2017FA	1,653
EVC Admin of Justice	2019FA	1,320
	2018FA	1,110
	2017FA	1,080
EVC Anthropology	2019FA	114
	2017FA	81
	2018FA	78

We want to see for each department, for 3 consecutive fall terms, the highest number of registered credits in any one term. To do this, create a new table in the report to the right of the current table which will contain the department object and an object you will create with the following formula:

Create Variable ? □ ×

Definition

Name: Type:

Description:

Qualification:

Formula

✓
✗

The resulting report should look like the following:

Department Desc	Enrollment Term	Registered Credit
EVC Accounting	2018FA	3,156
	2017FA	3,147
	2019FA	2,963
EVC Admin of Justice	2019FA	1,930
	2018FA	1,804
	2017FA	1,774
EVC Anthropology	2019FA	267
	2017FA	216
	2018FA	192

Department Desc	Dept's Most Reg Creds
EVC Accounting	3,156
EVC Admin of Justice	1,930
EVC Anthropology	267
EVC Art	3,093
EVC Arts	912
EVC Automotive Technology	2,014
EVC Basic Skills-Math	6,407
EVC Basic Skills-Reading	347.5
EVC Basic Skills-Writing	1,932
EVC Biology	7,429
EVC Building Info. Modeling	97

The table on the left serves as a truth set to ensure our calculation context variable is working as expected.

Using the ForAll Context Operator

You would use the 'ForAll' context operator where you want to aggregate a measure for ALL values of a dimension in a place in a report where you would typically aggregate to a SPECIFIC value of that dimension. In other words, you are removing the dimension from the calculation context for the measure.

To see the use of this context operator, consider a simple report where we have a list of the registered credit totals for enrolled students in the 2019FA term categorized by the student's IPEDS Race Ethnic Classification:

Enrollment Term	IPEDS Race Ethnic Classification	Registered Credit
2019FA	American Indian	1,028.5
2019FA	Asian	65,474
2019FA	Black or African American	6,816
2019FA	Hawaiian/Pacific Islander	671.5
2019FA	Hispanic	95,437.5
2019FA	Two or More Races	5,273.5
2019FA	Unknown	35,027.5
2019FA	White	12,152.5

We want to figure out what the average registered credits are per IPEDS Race Ethnic Classification, so we can easily determine whether each IPEDS Race Ethnic Classification has more or less than the average. To do this, create a new column in the report to the right of the current column. In the formula box at the top, specify an average of the “*Registered Credit*” measure over ALL the values of the “*IPEDS Race Ethnic Classification*” dimension, using the following formula:

=Average ([Registered Credit]) ForAll ([IPEDS Race Ethnic Classification])

The resulting report looks like the following:

Enrollment Term	IPEDS Race Ethnic Classification	Registered Credit	Average Credits
2019FA	American Indian	1,028.5	27,735.13
2019FA	Asian	65,474	27,735.13
2019FA	Black or African American	6,816	27,735.13
2019FA	Hawaiian/Pacific Islander	671.5	27,735.13
2019FA	Hispanic	95,437.5	27,735.13
2019FA	Two or More Races	5,273.5	27,735.13
2019FA	Unknown	35,027.5	27,735.13
2019FA	White	12,152.5	27,735.13

Using the Where Conditional Context Operator

The ‘Where’ context operator works in a slightly different way than all the other operators. Basically, the ‘Where’ operator allows the user to specify a logical expression which must evaluate to true for a row of data to be added to the specified aggregate. An example would be if you wanted to break out the “*Registered Credit*” measure into two columns according to whether the student was male or female. Each of these new columns would contain a measure with a conditional context operator with the appropriate test of the “*Gender*” dimension.

To do this, starting with the previous example, create a new column to the right of the list. Specify the following formula for the new cell:

=Sum ([Registered Credit]) Where ([Gender] = "F")

The resulting report shows a new column only for registered credits summed for female students:

Enrollment Term	IPEDS Race Ethnic Classification	Registered Credit	Average Credits	Credits - Female
2019FA	American Indian	1,028.5	27,735.13	638.5
2019FA	Asian	65,474	27,735.13	36,626.5
2019FA	Black or African American	6,816	27,735.13	3,622.5
2019FA	Hawaiian/Pacific Islander	671.5	27,735.13	379.5
2019FA	Hispanic	95,437.5	27,735.13	57,777.5
2019FA	Two or More Races	5,273.5	27,735.13	2,952.5
2019FA	Unknown	35,027.5	27,735.13	18,385.5
2019FA	White	12,152.5	27,735.13	6,633.5

After adding another column for male students, and adjusting column headers:

Enrollment Term	IPEDS Race Ethnic Classification	Registered Credit	Average Credits	Credits - Female	Credits - Male
2019FA	American Indian	1,028.5	27,735.13	638.5	390
2019FA	Asian	65,474	27,735.13	36,626.5	28,691.5
2019FA	Black or African American	6,816	27,735.13	3,622.5	3,186
2019FA	Hawaiian/Pacific Islander	671.5	27,735.13	379.5	292
2019FA	Hispanic	95,437.5	27,735.13	57,777.5	37,438.5
2019FA	Two or More Races	5,273.5	27,735.13	2,952.5	2,301
2019FA	Unknown	35,027.5	27,735.13	18,385.5	16,358.5
2019FA	White	12,152.5	27,735.13	6,633.5	5,444.5

Using the In Report Context Operator

Another alternative to the 'ForAll' context operator is the 'In Report' operator. When using the 'ForAll' operator, you need to explicitly identify one or more dimensions which you want to drop from the calculation context. The 'In Report' operator will always simply aggregate a measure to the highest level of aggregation on a report.

As an example, consider a report listing the distinct student enrollment counts for a set of years, as follows:

Enrollment Term Academic Year	Student Enrollment Count
2017	20,920
2018	21,080
2019	20,392

Assume that we want to compare each year's enrollment count with the highest aggregate student enrollment count in the report. If the variable "*Student Enrollment Count*" is defined as the distinct count of "*Student ID*", we can add a column to the report with the following formula:

=Max ([Student Enrollment Count]) In Report

This will result in the following report:

Enrollment Term Academic Year	Student Enrollment Count	Max Enrollment
2017	31,167	32,767
2018	32,675	32,767
2019	32,767	32,767

The aggregate maximum calculation will always return the maximum individual total within the report. Adding in another dimension, such as the "*Enrollment Term*" query object, will cause the expression to now reflect whatever the maximum aggregate value in the report, as below:

Enrollment Term Academic Year	Enrollment Term	Student Enrollment Count	Max Enrollment
2017	2017FA	20,920	21,080
2017	2017SP	17,724	21,080
2017	2017SPI	3,050	21,080
2017	2017SU	9,898	21,080
2018	2018FA	21,080	21,080
2018	2018SP	19,150	21,080
2018	2018SPI	3,571	21,080
2018	2018SU	10,830	21,080
2019	2019FA	20,392	21,080
2019	2019SP	18,923	21,080
2019	2019SPI	3,987	21,080
2019	2019SU	11,628	21,080

Learning Objective

At the end of this section, you will understand how to apply several of the calculation context modification operators in Web Intelligence to change the way aggregation happens within a Web Intelligence report. This section will allow you to apply the 'ForAll', 'In Report', and 'Where' operators to build a simple report analyzing student enrollment by departments across multiple academic terms and years.

High Level Overview

1. Create a new Web Intelligence report
2. Select needed dimensions and measures into report query
3. Build a list report comparing enrollment by department compared to the overall student enrollment for each term
4. Save the document

Practice Steps

STEP 1: Create a new Web Intelligence document. Use the *"BI Unified Universe – dev v1"* as the universe data source for the report.

STEP 2: In the query for the report, add the following objects from the source universe:

From the *"BI Course Success and Enrollment"* folder:

- *"Enrollment Term"*

From the *"Student Information"* folder:

- *"Student ID"*

From the *"Student Enrollment Information"* folder:

- *"Registered Credit"*

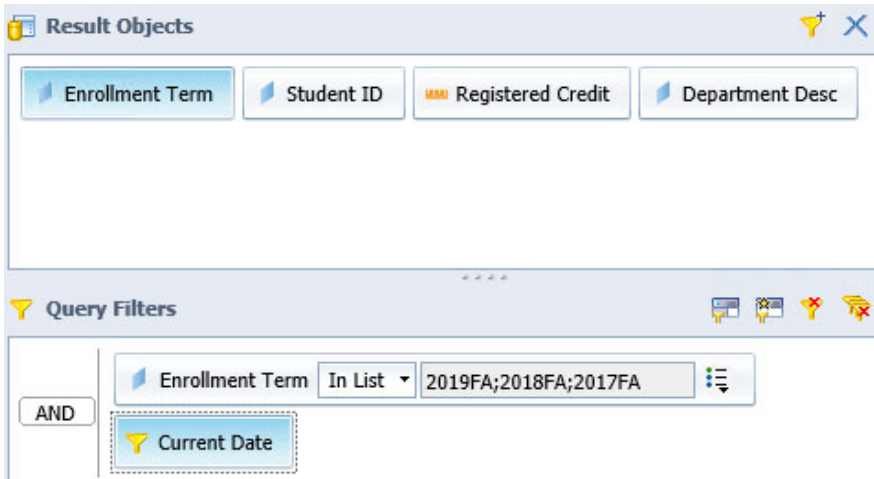
From the *"Section Information"* folder:

- *"Department Desc"*

STEP 3: Create the following filters for the report:

- A filter specifying that the *"Enrollment Term"* is *"2019FA; 2018FA; 2017FA"*
- Add the pre-defined snapshot filter for *"Current Date"*

When complete, your query panel should look something like this:



STEP 4: Run the report query. Modify the default 'List' block so that it only contains the following fields, in this order:

- "Enrollment Term"
- "Enrollment Department 1 Desc"
- "Registered Course Credits"

STEP 5: Change the name of the new report tab to be "Dept Enrollment Percentage".

STEP 6: Select the "Enrollment Term" column and create a report break on this column. At this point your report should look something like this:

Department Enrollment Percentage

Enrollment Term	Department Desc	Registered Credit
2017FA	EVC Accounting	3,147
	EVC Admin of Justice	1,774
	EVC Anthropology	216
	EVC Art	2,475
	EVC Arts	912
	EVC Automotive Technology	2,014
	EVC Basic Skills-Math	6,407
	EVC Basic Skills-Reading	347.5
	EVC Basic Skills-Writing	1,932
	EVC Biology	6,894



STEP 7: Create a new report variable in the report called “*Student Count*” providing the distinct count of enrolled students. Use the following formula for this variable:

$$=\text{Count}([\text{Student ID}])$$

STEP 8: Create a new report variable in the report called “*Student Count for All Departments*”. This variable will use the ‘ForAll’ calculation context modification keyword to provide the distinct count of students independent of the department. Use the following formula for this variable:

$$=\text{Count}([\text{Student ID}] \text{ ForAll } ([\text{Enrollment Term}];[\text{Department Desc}]))$$

STEP 9: Create a new report variable in the report called “*Department % of Total Students*”. This variable shows the proportion of the total distinct number of enrolled students that are enrolled in a course for specific department. Use the following formula for this variable:

$$=[\text{Student Count}] / [\text{Student Count for All Departments}]$$


STEP 10: Drag the “*Student Count*”, “*Student Count for All Departments*”, and “*Department % of Total Students*” variable into the ‘List’ report.

STEP 11: Format the last column of the report as a percentage.

The completed report tab should look something like the following:

Department Enrollment Percentage

Enrollment Term	Department Desc	Registered Credit	Student Count	Student Count for All Departments	Department % of Total Students
2017FA	EVC Accounting	3,147	632	41,187	1.53%
	EVC Admin of Justice	1,774	444	41,187	1.08%
	EVC Anthropology	216	69	41,187	0.17%
	EVC Art	2,475	680	41,187	1.65%
	EVC Arts	912	258	41,187	0.63%
	EVC Automotive Technology	2,014	343	41,187	0.83%
	EVC Basic Skills-Math	6,407	1,403	41,187	3.41%
	EVC Basic Skills-Reading	347.5	242	41,187	0.59%
	EVC Basic Skills-Writing	1,932	472	41,187	1.15%


Department Enrollment Percentage

STEP 12: Save your report with the name “*Enrollment Analysis by Department*”.

ADVANCED FORMATTING

At this point, we have the data in place within several reports. However, there are several additional formatting techniques available to us that we can apply to improve the overall readability and usability of our reports.

Formatting Tables

There are several formatting steps for list tables that can greatly enhance the clarity and readability of the data presented. This section will cover ones that we use most frequently.

To open the 'Format Table' window, right click on the outer edge of the table and select 'Format Table...'. There are several table properties available here.

Alternate Row Color

In the 'Format Table' window, navigate to the 'Appearance' panel, and set the 'Alternate color' properties.

Masked Student ID	Enrollment Term
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA
*****00	2019FA

Format Table

General
Border
Appearance
Layout

Background

Color: No color

Pattern: None

Skin:

Image from address

Image from file

Display:

Position:

Spacing

Horizontal: inches Vertical: inches

Alternate color

Frequency: Color:

Relative Positioning

Setting relative positioning when a report contains two or more tables on the same tab ensures consistent spacing between the tables and prevents overlapping of tables when data is changed or refreshed. In the 'Format Table' window, navigate to the 'Layout' panel, to set the 'Position' properties. In this example, the table on the left is named "Dual Sections" and we want to set the table on the right, "Enrolled Students" relative (0.2" to the right and even with the top) to the "Dual Sections" table.

Click on a column in this table to filter the Roster Table:

Roster Table:

Enrollment Term	Department Desc	Section Title	Section ID	Section Synonym	Course Type	Enrollment Term	Section ID	Student ID	Last Name
2019FA	SJCC Admin of Justice	Street Law							
2019FA	SJCC Business	Introductio							
2019FA	SJCC Construction Technology	Tools, Mat							
2019FA	SJCC Early Childhood Ed	Literacy Di							
2019FA	SJCC Ethnic Studies	Asian Paci							
2019FA	SJCC Ethnic Studies	Chicana/o							
2019FA	SJCC Ethnic Studies	Mexican A							
2019FA	SJCC Ethnic Studies	Mexican A							
2019FA	SJCC Family Consumer Studies	Child Deve							
2019FA	SJCC Guidance	Orientalior							
2019FA	SJCC Psychology	General P							
2019FA	SJCC Psychology	General P							
2019FA	SJCC Sign Language	ASL 1							
2019FA	SJCC Sign Language	ASL 1							
2019FA	SJCC Theatre Arts	Chicanx/L							

Format Table

General
Border
Appearance
Layout

Horizontal
 New page: Start on a new page Page layout: Avoid page breaks in table
 Repeat on every page

Vertical
 New page: Start on a new page Page layout: Avoid page breaks in table
 Repeat on every page Repeat header on every page
 Repeat footer on every page

Position
Within the Report
 The left of this table is 0.2 inches to
 Right side of... Dual Sections
 The top of this table is 0 inches to
 Top side of... Dual Sections

OK Cancel Apply

Repeating Column Headers on each Page

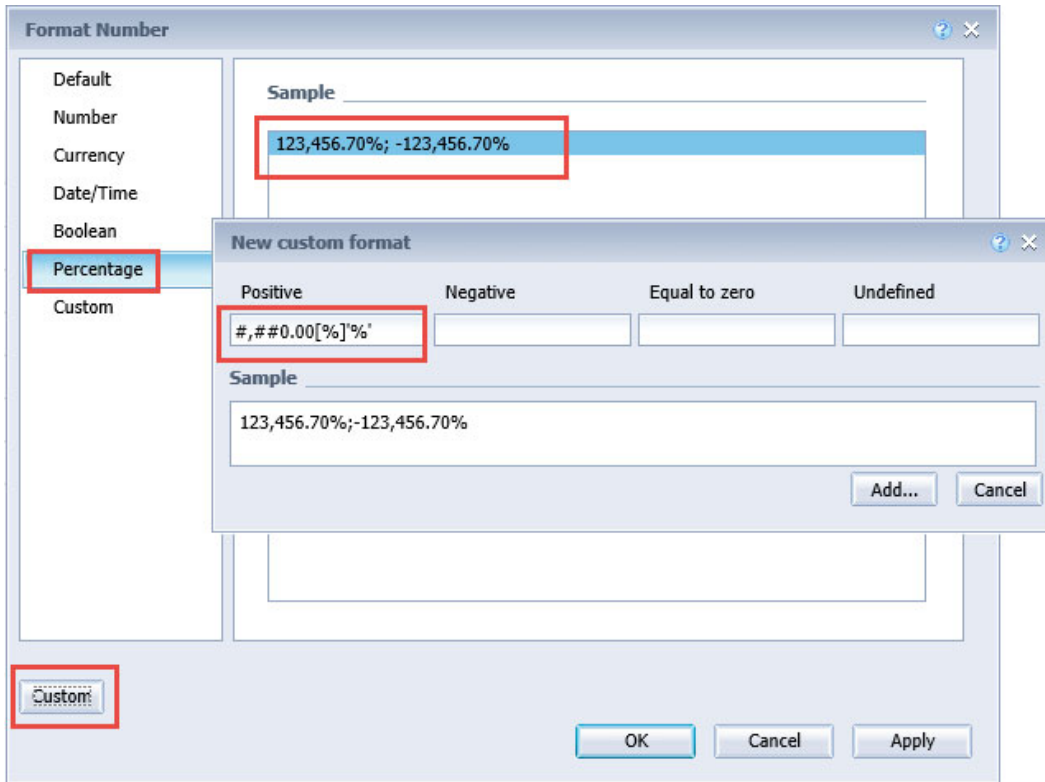
To increase readability for tables that span multiple pages, you can set the column headers to repeat on each page. Navigate to the 'Report Elements' tab -> 'Behaviors' sub tab, click on the dropdown arrow next to 'Repeat' and select 'Header on every page'.

Formatting Percentages

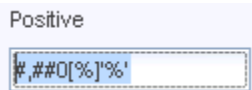
Another necessary formatting step in any Web Intelligence report is to format fractions as percentages. To format "Percentage of Students" as a percentage, right click on the cell and select 'Format Number...' then select 'Percentage' from the left panel in the window.

The default percentage presentation provided shows two digits to the right of the decimal point. If that is the representation you need, simply select that row in the selection list and click the 'OK' button. However, in many cases, this overstates the level of precision needed for your visualization.

With 'Percentage' selected from the left panel in the window, highlight the 'Percentage' selection from the sample box on the right, then click the 'Custom' button at the bottom left to bring up the 'New Custom Format' sub-window:



Adjust the template format in the highlighted box. To eliminate the decimal points, delete the '.00' string in the format as follows:

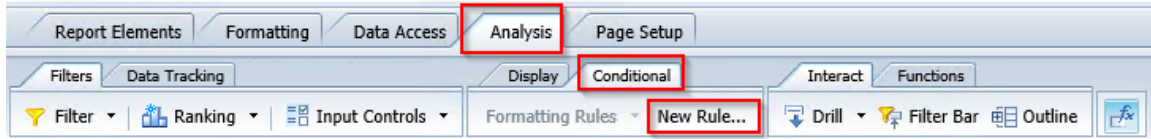


Click the 'Add...' button to add this altered format as a new custom format in your report, then click 'OK' to close the 'Format Number' window. The newly formatted field will now show as follows:

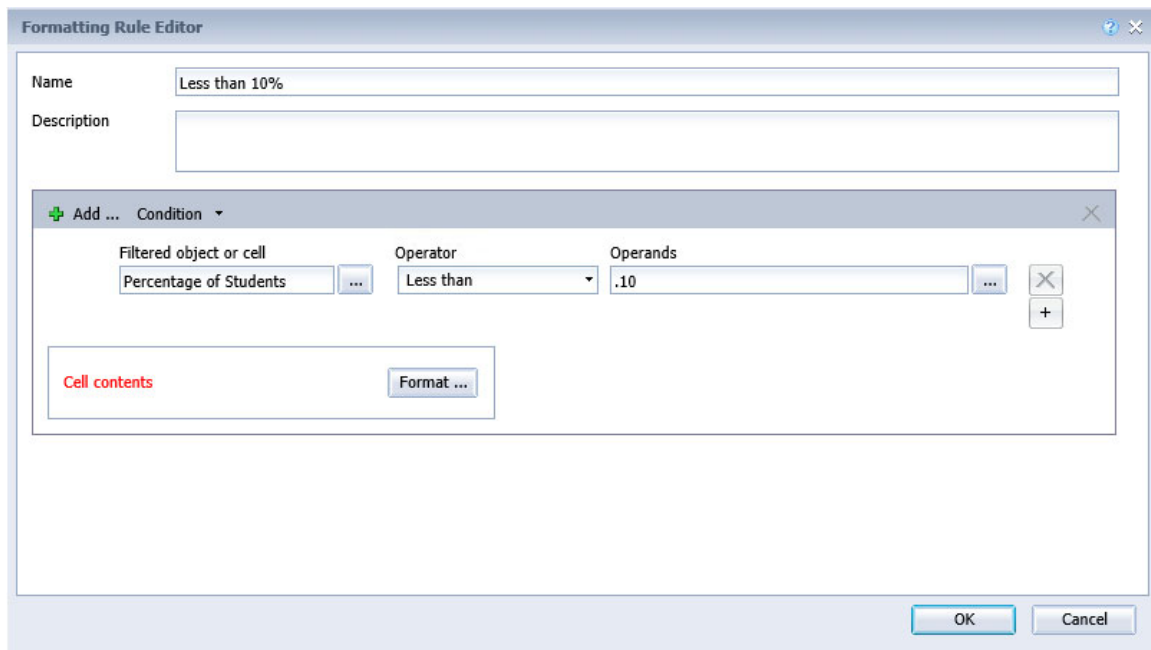
Enrollment Term	IPEDS Age - Current	Gender Desc	Registered Credit	Student Count	Total Students	Percentage of Students
2019FA	17 & Below	Female	4,138.5	794	16,948	5%
2019FA	17 & Below	Male	3,206.5	663	16,948	4%
2019FA	18-24	Female	46,356.5	4,843	16,948	29%
2019FA	18-24	Male	41,115.5	4,440	16,948	26%
2019FA	25-39	Female	18,673.5	2,526	16,948	15%
2019FA	25-39	Male	12,172	1,715	16,948	10%
2019FA	40 & Over	Female	8,099	1,257	16,948	7%
2019FA	40 & Over	Male	4,497.5	710	16,948	4%

Conditional Formatting

There is a powerful feature in Web Intelligence called 'Conditional Formatting' that can be used to highlight data, change the colors or formatting of data based on certain conditions, or even display different data. We are going to use 'Conditional Formatting' to highlight the "*Percentage of Students*" when it drops below 10 percent. Under the 'Analysis' tab, select the 'Conditional' sub-tab, and then click on 'New Rule...'



In the 'Formatting Rule Editor' window, name the rule "*Less than 10%*". Click the '...' next to 'Filtered object or cell' and select 'Select object or variable'. From the 'Available objects' pop-up window, select "*Percentage of Students*". Select 'Less than' for the 'Operator'. Type ".10" in the "Type a value" 'Operands' box.



Click the 'Format...' button. Located on the left panel of the box, click on the 'Text' link. Select 'Bold' for the 'Font style'. Notice that "*Cell Contents*" in the 'Preview' window now shows bold and red font.

Click 'OK' to on the 'Formatting Rules Display' and 'OK' again on the 'Formatting Rule Editor' to finish.

Now we must apply this formatting rule to the “Percentage of Students” cells. Highlight the “Percentage of Students” column on the table. Click the ‘Formatting Rules’ button and you should see a check box beside the new “Less than 10%” format rule. Click the check box to apply. The resulting report looks like the following, with all percentages less than 10% highlighted in bold, red text.



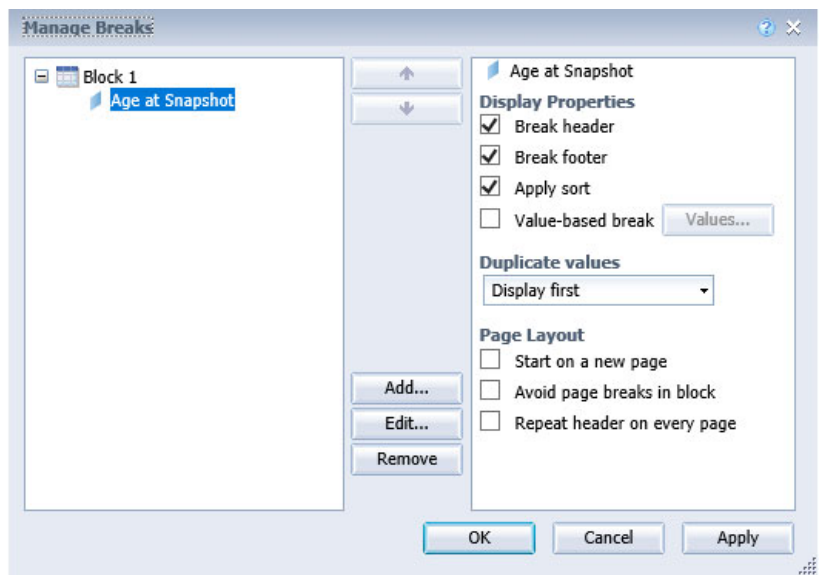
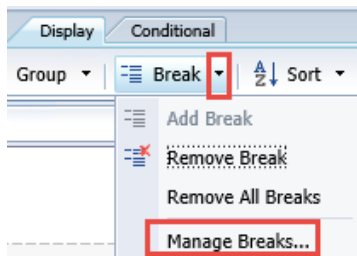
Enrollment Term	IPEDS Age - Current	Gender Desc	Registered Credit	Student Count	Total Students	Percentage of Students
2019FA	17 & Below	Female	4,138.5	794	16,948	5%
2019FA	17 & Below	Male	3,206.5	663	16,948	4%
2019FA	18-24	Female	46,356.5	4,843	16,948	29%
2019FA	18-24	Male	41,115.5	4,440	16,948	26%
2019FA	25-39	Female	18,673.5	2,526	16,948	15%
2019FA	25-39	Male	12,172	1,715	16,948	10%
2019FA	40 & Over	Female	8,099	1,257	16,948	7%
2019FA	40 & Over	Male	4,497.5	710	16,948	4%

Breaks

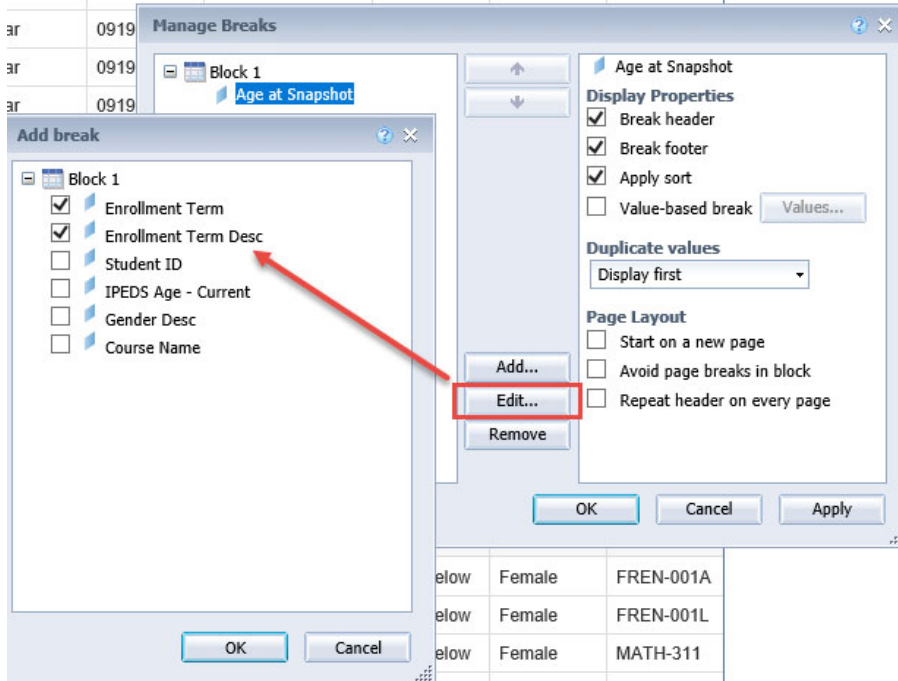
You can use breaks to split tables up by unique values of an object and to apply subtotals and other aggregations. Each unique value of a dimension becomes a sub-group within the table. A break automatically inserts a blank row after each sub-group to insert aggregations; you can hide or delete this row. To set up a single-object column break, click on the column, navigate to the ‘Analysis’ tab -> ‘Display’ sub tab -> ‘Break’ button.

Enrollment T	Enrollment Term Desc	Student ID	Age at Snapshot	IPEDS Age -	Gender Des	Course Nam
2018FA	Fall 2018 Regular		11	17 & Below	Male	CIS-054
	Count:	1	11			
2019FA	Fall 2019 Regular		12	17 & Below	Female	MATH-013
	Count:	1	12			
2018FA	Fall 2018 Regular		13	17 & Below	Female	FREN-001A
2018FA	Fall 2018 Regular		13	17 & Below	Female	FREN-001L
2018FA	Fall 2018 Regular		13	17 & Below	Female	MATH-311
2018FA	Fall 2018 Regular		13	17 & Below	Female	SL-001A
2019FA	Fall 2019 Regular		13	17 & Below	Male	PHYS-002A
2019FA	Fall 2019 Regular		13	17 & Below	Female	MUSIC-050E
2019FA	Fall 2019 Regular		13	17 & Below	Female	SPAN-001A

Access the ‘Manage Breaks...’ window to fine-tune the display of your table. You can set up and edit breaks, combine breaks, and set display properties. You can set one break with multiple objects that all share the same properties or can set up and order individual breaks and set different properties for each.

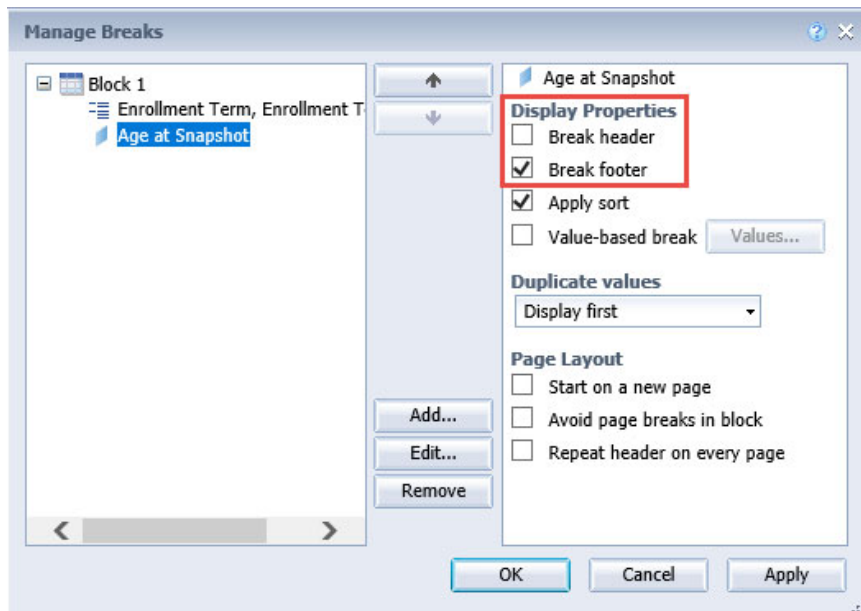


Edit the break to add multiple objects. All objects will have the same display properties.



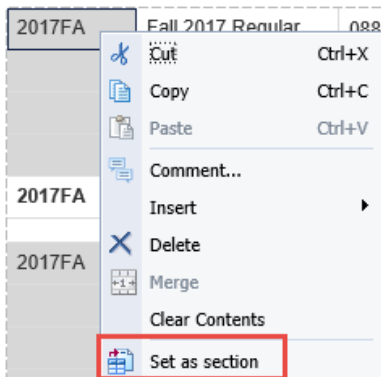
2017FA	Fall 2017 Regular		14	17 & Below	Female	MATH-063
				17 & Below	Male	MATH-013
				17 & Below	Female	MATH-013
				17 & Below	Male	ASTRO-010
2017FA	Fall 2017 Regular	4	14			

Set up separate breaks with individual properties for controlling whether they show break headers and footers.



Sections

While a break groups data inside the table, a section groups data outside the table, sectioning the entire report. Therefore, report elements (tables, charts, etc.) added to one section will repeat in all sections, with the unique value(s) for that section. Unlike breaks, sections can be set on any dimension, regardless of whether it is displayed in a report element. Breaks can be set on a table within a section. To set a column as a section, right click on it and select 'Set as section'.

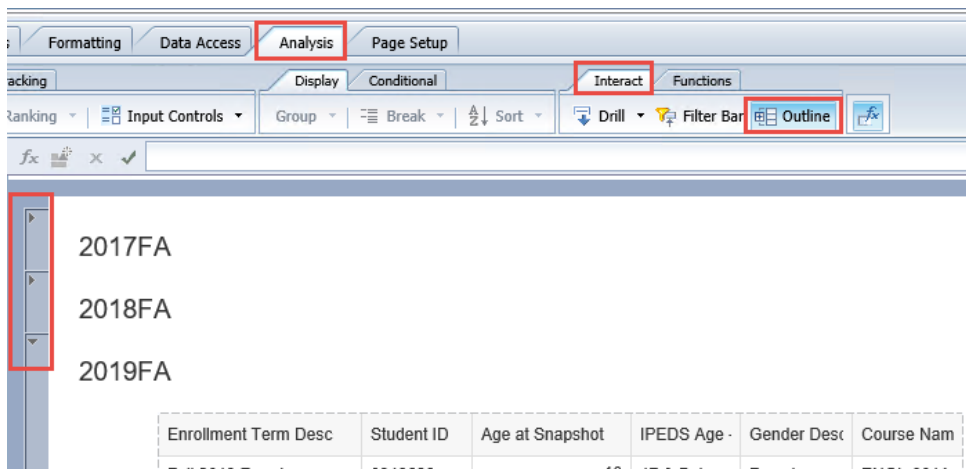


2017FA

Enrollment Term Desc	Student ID	Age at Snapshot	IPEDS Age	Gender Desc	Course Nam
Fall 2017 Regular		14	17 & Below	Female	MATH-063
Fall 2017 Regular			17 & Below	Male	MATH-013
Fall 2017 Regular			17 & Below	Female	MATH-013
Fall 2017 Regular			17 & Below	Male	ASTRO-010
Count:	4	14			

Outline (fold/unfold)

The Outline functionality allows you to fold and unfold a table based on sections or breaks. navigate to the 'Analysis' tab -> 'Interact' sub tab -> 'Outline' button and click on a report element to reveal the folding/unfolding arrows in the margin.



Hands-on Exercise Learning Objective

At the end of this section you will know how to apply several advanced formatting techniques to reports.

High Level Overview

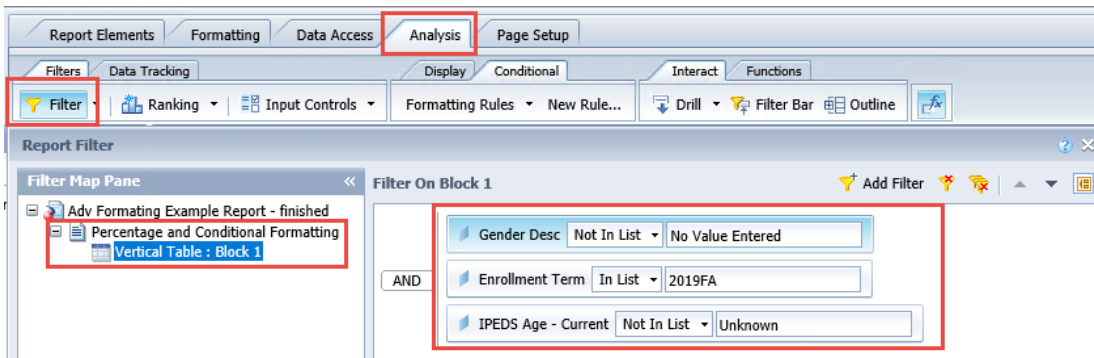
1. Format cells as percentage and create a custom format
2. Add conditional formatting
3. Set alternate row color and repeat headers on every page
4. Work with the breaks and sections
5. Save the document

Practice Steps

STEP 1: Open the “Adv Formatting Example Report” and make sure you are working with the “Percentage and Conditional Formatting” report tab.

STEP 2: Format the “Percentage of Students” column as a percentage with no decimal point.

STEP 3: Add report filters to remove “Unknown” IPEDS ages, gender descriptions of “No Value Entered”, and to ensure you are only viewing data for 2019FA.



STEP 4: Add conditional formatting of bold, red text for any values in the “Percentage of Students” column that are less than 10%.

At this point, your table should look something like this:

Enrollment Term	IPEDS Age - Current	Gender Desc	Registered Credit	Student Count	Total Students	Percentage of Students
2019FA	17 & Below	Female	4,138.5	794	16,948	5%
2019FA	17 & Below	Male	3,206.5	663	16,948	4%
2019FA	18-24	Female	46,356.5	4,843	16,948	29%
2019FA	18-24	Male	41,115.5	4,440	16,948	26%
2019FA	25-39	Female	18,673.5	2,526	16,948	15%
2019FA	25-39	Male	12,172	1,715	16,948	10%
2019FA	40 & Over	Female	8,099	1,257	16,948	7%
2019FA	40 & Over	Male	4,497.5	710	16,948	4%

Save your report.

STEP 5: Move to the “*Enrollment Summary Report*” report tab.

STEP 6: Set the alternate row color to light gray.

STEP 7: Set the column headers repeat on every page.

Your table should look something like this:

Enrollment Summary Report

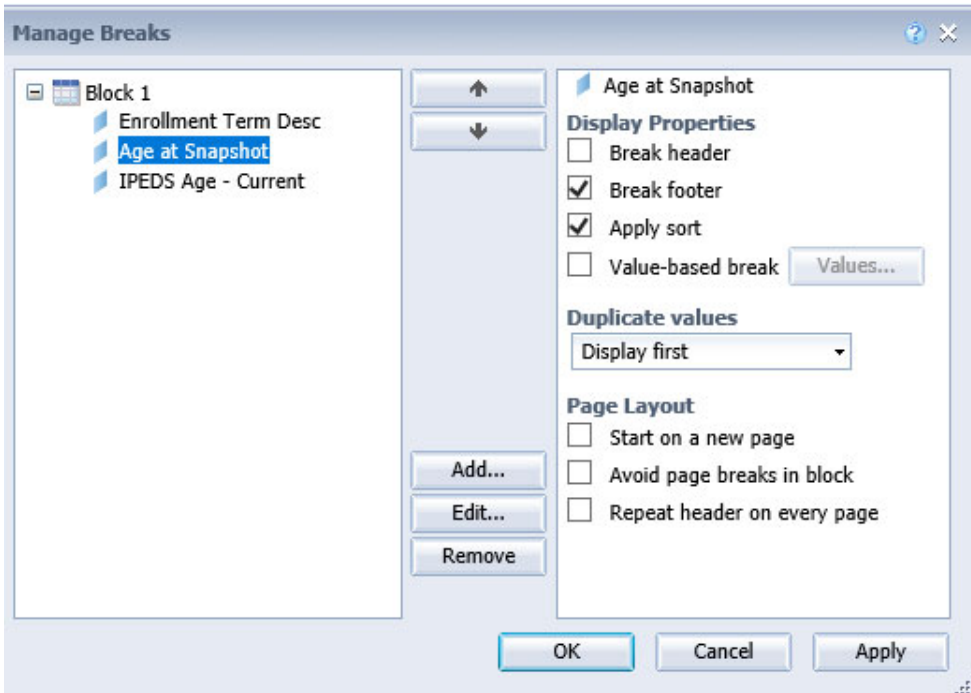
Masked Student ID	Enrollment Term	Start Term Cohort Desc	Res Status Desc	Immigration Status Desc	Gender Desc	Entry Admit Status Desc	Registered Credit
*****00	2017FA	2000 Fall	In State	U.S. Citizen	Female	New, First Time Student	10
*****00	2017FA	2000 Fall	In State	U.S. Citizen	Male	Returning Student	7
*****00	2017FA	2001 Fall	In State	U.S. Citizen	Female	New, First Time Student	5
*****00	2017FA	2002 Spring	In State	U.S. Citizen	Female	New Transfer Student	7
*****00	2017FA	2003 Fall	In State	U.S. Citizen	Female	Returning Student	3
*****00	2017FA	2004 Fall	In State	Permanent Resident	Male	New, First Time Student	4

Save your report.

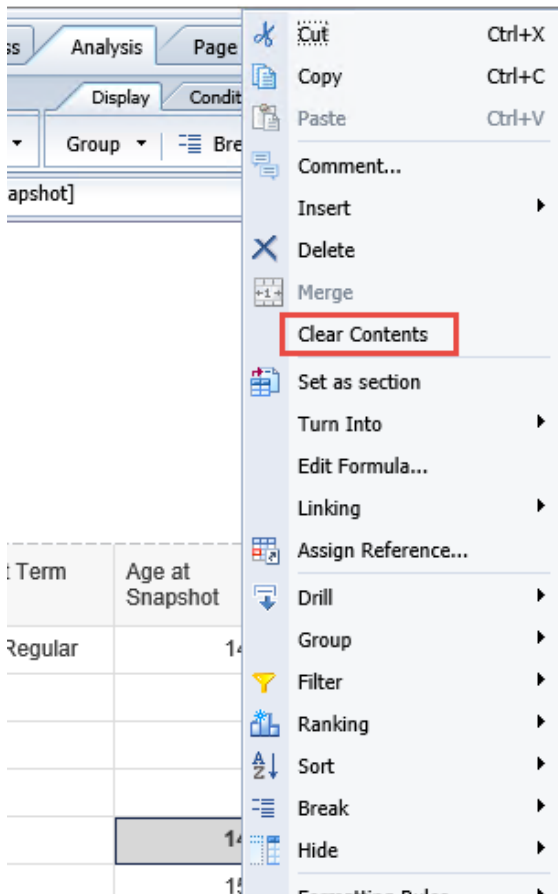
STEP 8: Move to the “*Adv Formatting Practice*” report tab.

STEP 9: Set the “*Enrollment Term*” column as a section.

STEP 10: Set up breaks for the columns with repeating values. Only leave one break footer row for the “*Age at Snapshot*” column. Turn off break headers for all breaks created.



STEP 11: Clear the contents in the footer under the “Age at Snapshot” column.



STEP 12: Add a count for Student ID and make sure all aggregations are in a single footer row.

Your report should look something like this:

2017FA

Enrollment Term Desc	Age at Snapshot	IPEDS Age - Current	Gender Desc	Student ID	Course Name	Registered Credit
Fall 2017 Regular	14	17 & Below	Female	[REDACTED]	MATH-063	3
			Female		MATH-013	5
			Male		MATH-013	5
			Male		ASTRO-010	3
			Count:		4	16

Save your report.

LINKING AND DRILL-THROUGH REPORTING

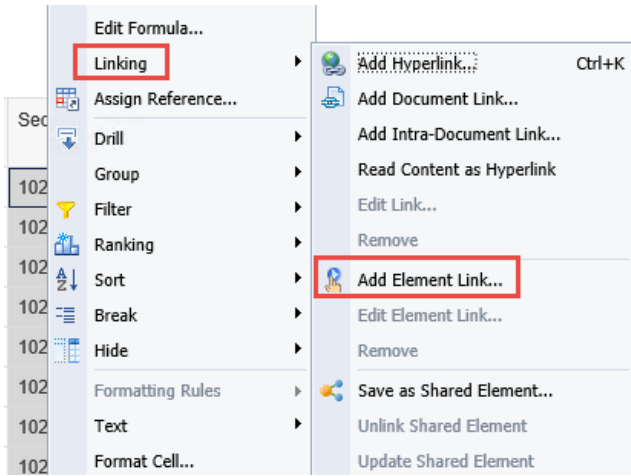
Web Intelligence offers several methods for creating hyperlinks within a document. In this section, we will review using element linking, document linking to a web page or as drill through reporting, and intra-document linking.

Element Linking

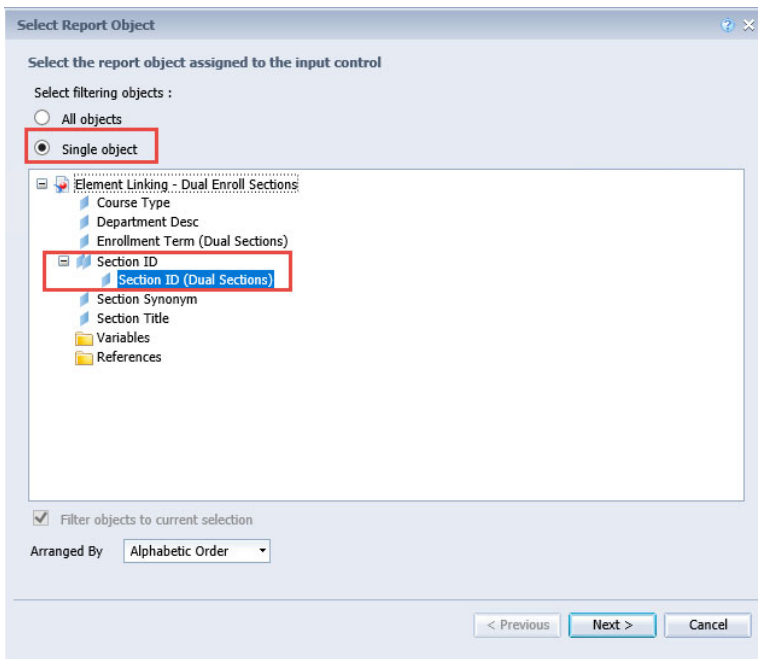
An element link is a feature designed to pass a value from one report object (table or chart) to another report object. Element links transform the values in a table or chart into dynamic values that can filter other data elements.

For this example, we will use the Web Intelligence document “*Element Linking – Dual Enroll Sections*” found in the “*Public Folders*” → “*SJECCD Training*” → “*Advanced Webi Training*” folder. If you have not already done so, copy these reports to your “*My Favorites*” folder.

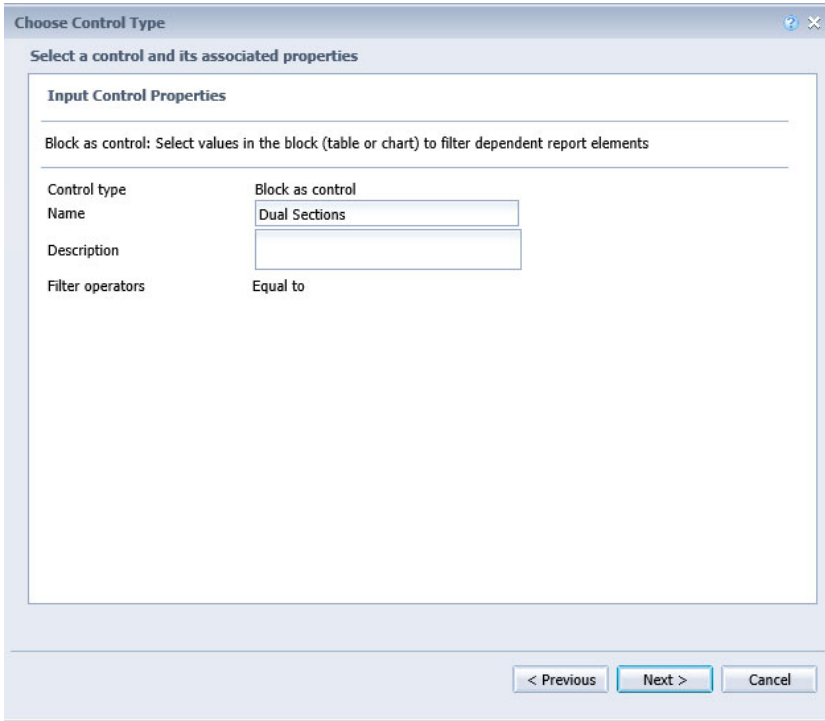
STEP 1: To set up element linking, right click on the object in the table you want to use to affect another report object, select ‘Linking’ -> ‘Add Element Link...’.



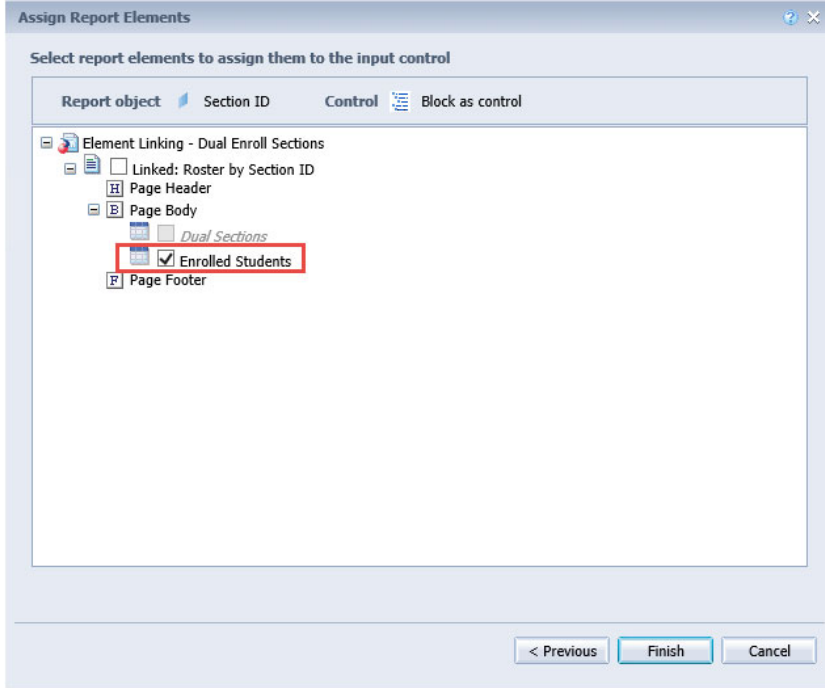
STEP 2: A link can be set between one object or all objects. In this example, we will set it for a single object to return all Section ID's for a given term.



STEP 3: The 'Choose Control Type' window indicates that the link will act as an input control and it provides the information on the block which controls the dependent report elements. In this example, our controlling block has the name of "Dual Sections" and the filter operator is set to 'Equal to'.



STEP 4: The ‘Assign Report Elements’ window allows you to decide which block will be the target. Our target block is named “*Enrolled Students*”.



STEP 5: To activate the filter created by the link, click on a cell in the controlling table, and observe the changes in the dependent table.

Linked: Roster by Section ID

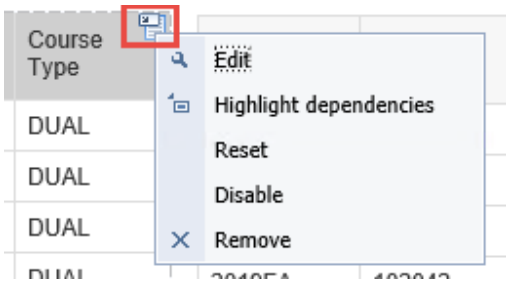
Click on a column in this table to filter the Roster Table:

Enrollment Term	Department Desc	Section Title	Section ID	Section Synonym	Course Type
2019FA	SJCC Admin of Justice	Street Law	102052	100525	DUAL
2019FA	SJCC Business	Introduction to Business	102046	100519	DUAL
2019FA	SJCC Construction Technology	Tools, Materials, and Safety	102061	100534	DUAL
2019FA	SJCC Early Childhood Ed	Literacy Dev	102121	100592	DUAL
2019FA	SJCC Ethnic Studies	Asian Pacific American Culture	102054	100527	DUAL
2019FA	SJCC Ethnic Studies	Chicana/o Culture	102159	100630	DUAL
2019FA	SJCC Ethnic Studies	Mexican American History I	102051	100524	DUAL

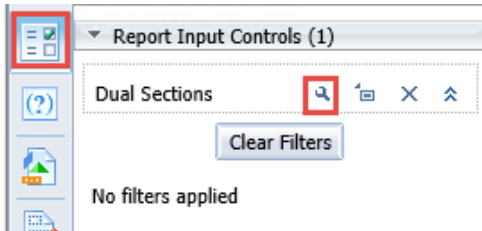
Roster Table:

Enrollment Term	Section ID	Student ID	Last Name	First Name
2019FA	102121	0		
2019FA	102121	0		
2019FA	102121	0		
2019FA	102121	0		a
2019FA	102121	0		tte
2019FA	102121	0		e
2019FA	102121	0		

STEP 6: To reset the dependent table, click any column header in the controlling table. To make changes to the element link, right-click the small white box in the top right corner of the table and choose an option from the menu.



You can also access editing options from the input control panel.

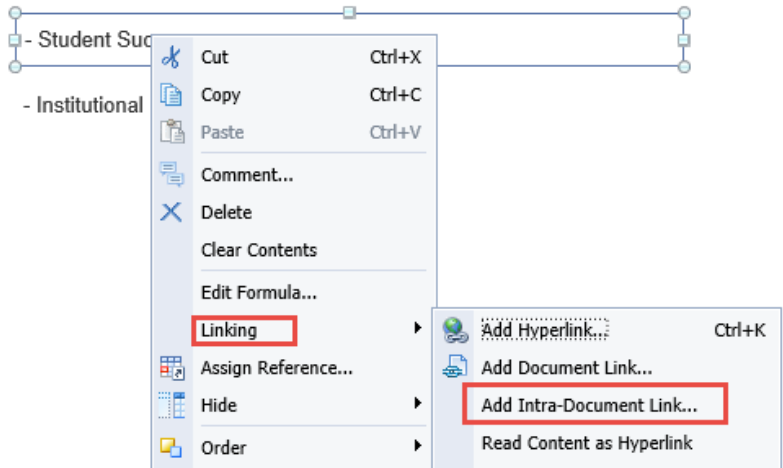


Intra-Document Linking

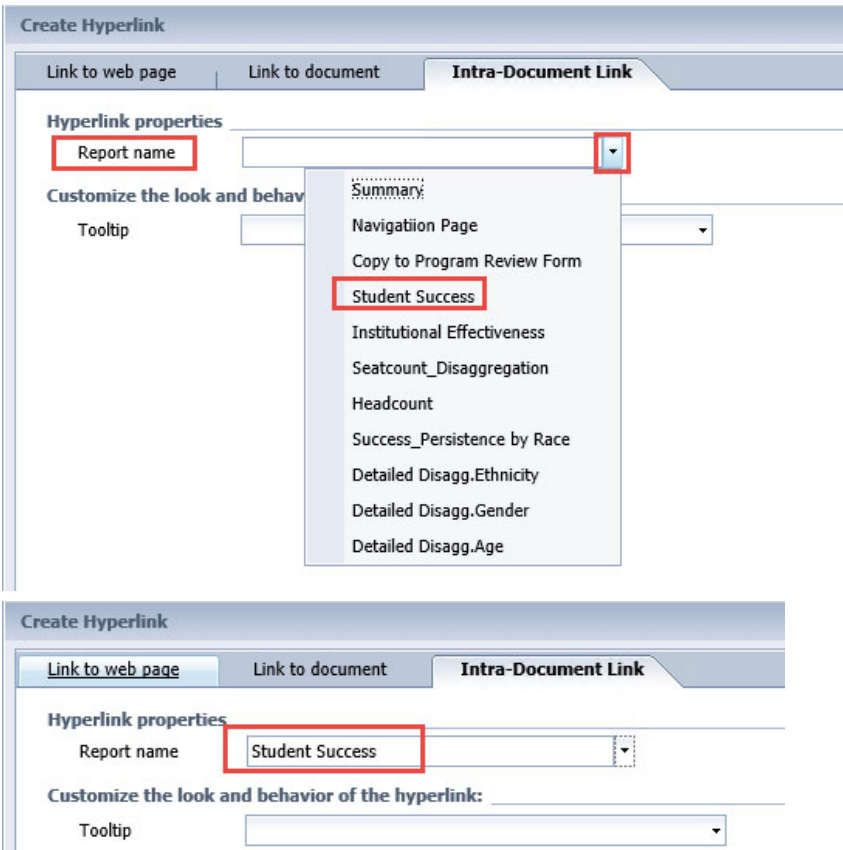
Intra-document linking allows you to link data from one report document to another. Report creators can set up a link between two report tabs in a single document using cell contents as a hyperlink. For example, you can organize a complex document containing several report tabs by creating a page with a link to each tab that simplifies navigation within the document.

For this example, we will use the Web Intelligence document “*Program Review Intra-document Linking*” found in the “*Public Folders*” → “*SJECCD Training*” → “*Advanced Webi Training*” folder. If you have not already done so, copy these reports to your “*My Favorites*” folder.

STEP 1: To set up intra-document linking, navigate to the report tab you where you want to set up your links, right click on the intended cell or column that will serve as the link to the desired reporting tab, and select 'Linking' -> 'Add Intra-Document Link...'.



STEP 2: In the 'Create Hyperlink' window, select the appropriate tab name from the 'Report name' dropdown.



Click 'OK'.

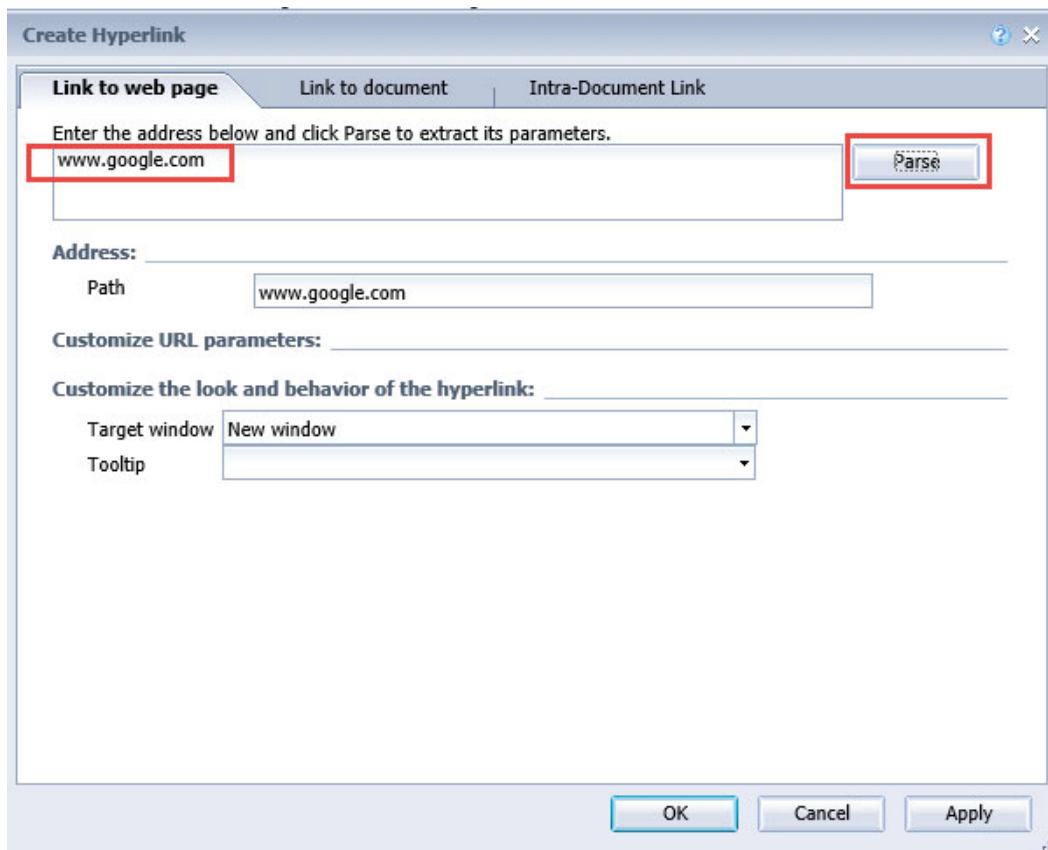
STEP 3: Click on the new hyperlink and you should automatically jump to the linked report tab.

STEP 4 OPTIONAL: Continue to link the remaining tabs to the available cells adding additional cells for remaining report tabs.

Document Linking to a Webpage

You can link a Web Intelligence document to a web page similarly to element and intra-document linking. Select the cell or column where you want to apply your link and navigating to the 'Create Hyperlink' window.

Click on the 'Link to web page' tab and enter the web address in the available space and click 'Parse'.



The screenshot shows the 'Create Hyperlink' dialog box with the following details:

- Tab: **Link to web page** (selected)
- Instruction: Enter the address below and click Parse to extract its parameters.
- Address field: **www.google.com** (highlighted with a red box)
- Parse button: **Parse** (highlighted with a red box)
- Address section:
 - Path: **www.google.com**
- Customize URL parameters: (empty)
- Customize the look and behavior of the hyperlink:
 - Target window: **New window**
 - Tooltip: (empty)
- Buttons: **OK**, **Cancel**, **Apply**

Click on the hyperlink and the web page should launch.

Visit our webpage for further information:





Google Search

I'm Feeling Lucky

Drill Through Reporting

The drill through capability in Web Intelligence takes you from one report (source report) to another (target report) that is relevant to the data being analyzed. Using drill-through access, you can move from one report to another within a session while maintaining focus on the same piece of data. Drill through target reports are separate reports and are opened via hyperlinks. Target reports typically contain parameters that are passed to them by the main report.

Hands-on Exercise Learning Objective

At the end of this section you will know how to build and set up drill through reporting between two Web Intelligence documents.

High Level Overview

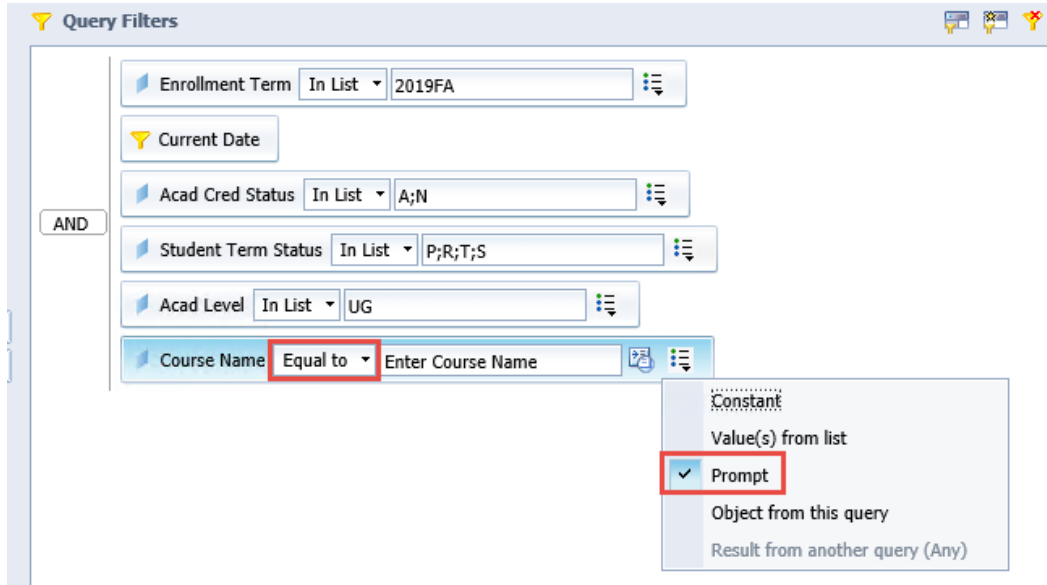
1. Work with existing documents
2. Add a query filter to the target report
3. Set up drill through reporting from source report to target report
4. Save the documents

Practice Steps

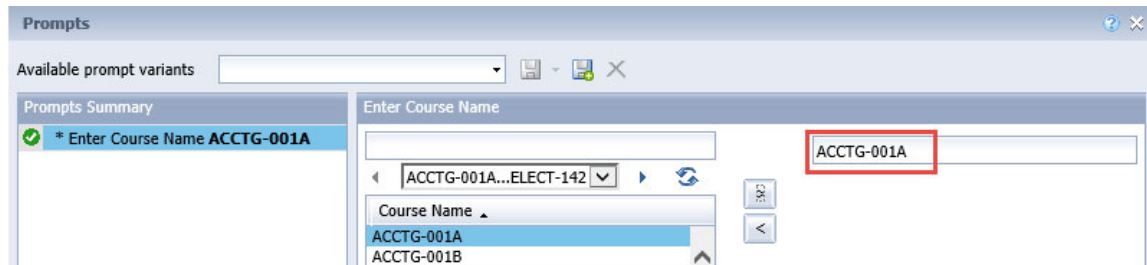
STEP 1: For this exercise, we will use the Web Intelligence documents “*Enrollment Summary Report - Source*” and “*Enrollment Summary Report - Target*” found in the “*Public Folders*” → “*SJECCD Training*” → “*Advanced Webi Training*” folder. If you have not already done so, copy these reports to your “*My Favorites*” folder.

STEP 2: Open the “*Enrollment Summary Report - Target*” and navigate to the ‘Query Panel’.

STEP 3: Add a query filter that prompts for and returns only one “Course Name”.



STEP 4: Run the query and use the value in the following screenshot to answer the prompt.



Your report should look like this:

Course Enrollment Detail Summary Report

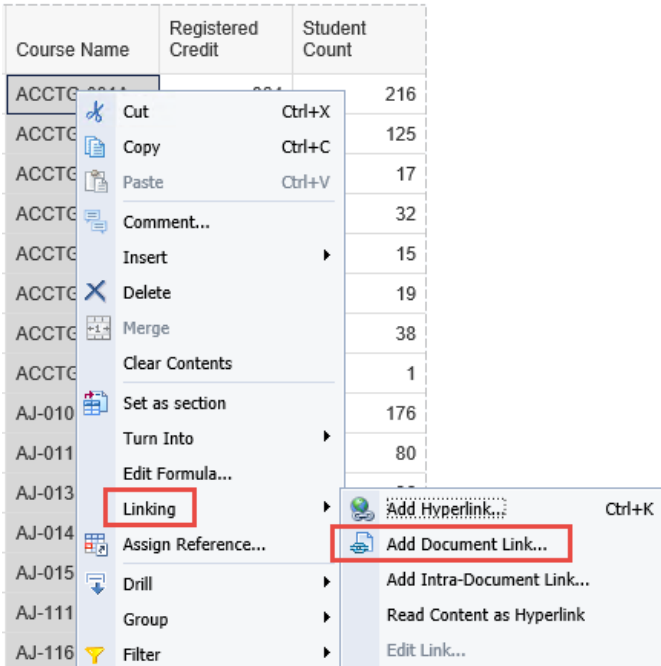
ACCTG-001A

Student ID	Last Name	First Name	Enrollment Term	Start Term Cohort Desc	Res Status Desc	Immigration Status Desc	Gender Desc	Entry Admit Status Desc	Acad Cred Status Desc	Registered Credit	Student Count
02			2019FA	2000 Fall	In State	U.S. Citizen	Male	Do Not Use - Return Trans Stu.	New		216
04		nh	2019FA	2003 Fall	In State	U.S. Citizen	Male	Returning Student	New		4
04			2019FA	2000 Fall	In State	Refugee/Asylee	Female	Returning Student	Add		4
05		ha	2019FA	2003 Fall	In State	U.S. Citizen	Male	New, First Time Student	New		4
06			2019FA	2014 Spring	In State	U.S. Citizen	Female	New Transfer Student	New		4
06		s	2019FA		In State	U.S. Citizen	Male	High School Stu.-R40 Rqd.	New		4
06		a	2019FA	2006 Fall	In State	U.S. Citizen	Female	New, First Time Student	Add		4
06		i	2019FA	2005 Fall	In State	U.S. Citizen	Female	Returning Student	New		4
06		l	2019FA		In State	U.S. Citizen	Female	High School Stu.-R40 Rqd.	New		4
06		d	2019FA	2006 Fall	In State	U.S. Citizen	Female	New, First Time Student	New		4
06		a	2019FA	2012 Spring	In State	U.S. Citizen	Female	New Transfer Student	New		4

STEP 5: Save and close your report.

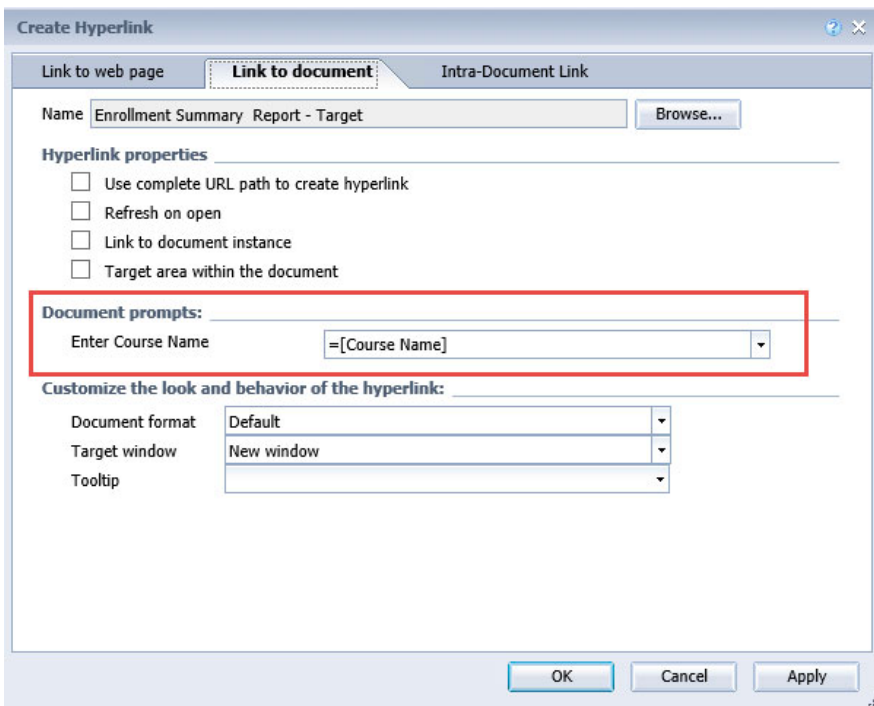
STEP 6: Open the source Web Intelligence document, “Enrollment Summary Report - Source”.

STEP 7: Right click on the “Course Name” column and select ‘Linking’ -> ‘Add Document Link...’.



STEP 8: Click ‘Browse...’ and browse to your target report named “Enrollment Summary Report - Target”.

STEP 9: Click ‘OK’ to accept the defaults in the ‘Create Hyperlink’ window.



The list values for “Course Name” should now be hyperlinks and look like this:

Enrollment Term	Department Desc	Course Name	Registered Credit	Student Count
2019FA	EVC Accounting	ACCTG-001A	864	216
2019FA	EVC Accounting	ACCTG-001B	500	125
2019FA	EVC Accounting	ACCTG-030	51	17
2019FA	EVC Accounting	ACCTG-062	160	32
2019FA	EVC Accounting	ACCTG-095	75	15
2019FA	EVC Accounting	ACCTG-097	57	19
2019FA	EVC Accounting	ACCTG-101	114	38

STEP 10: Save your report.

STEP 11: Click on a hyperlink value within the list to launch your target report.

Your target report should launch and return student-level detail data for the course name represented in the list value you selected in the source report.

Course Enrollment Detail Summary Report

ACCTG-063

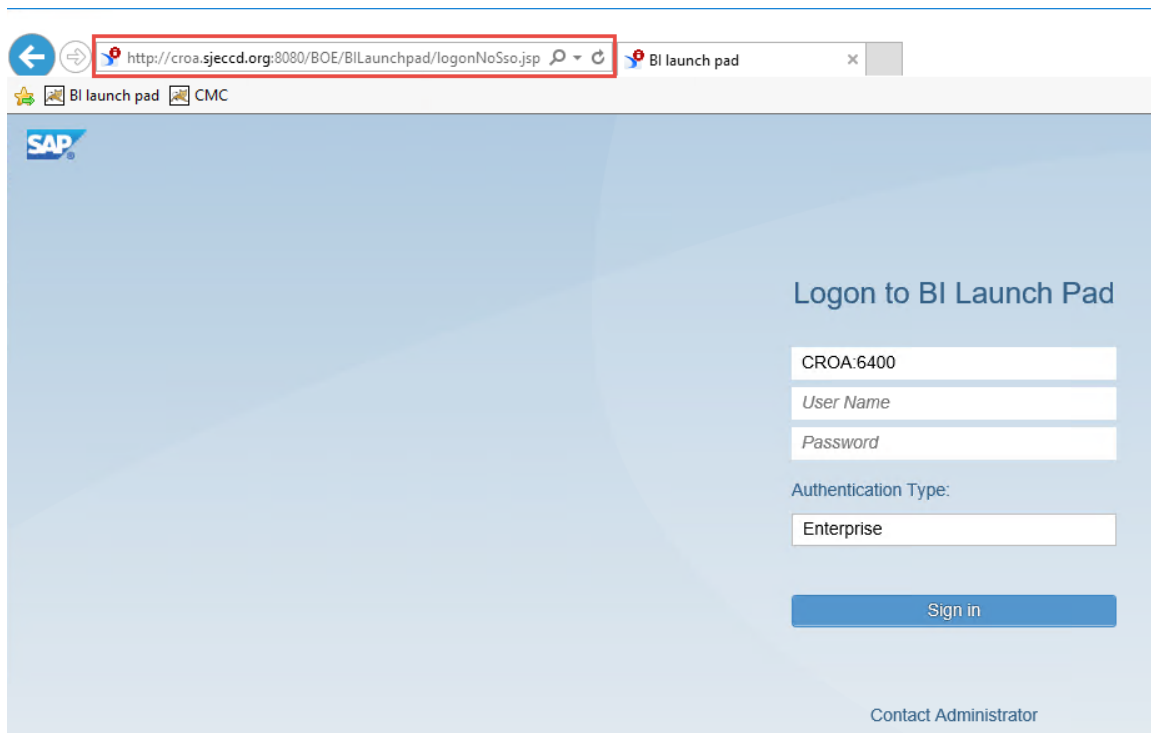
Student ID	Last Name	First Name	Enrollment Term	Start Term Cohort Desc	Res Status Desc	Immigration Status Desc	Gender Desc	Entry Admit Status Desc	Acad Cred Status Desc	Registered Credit	Student Count
0		ia	2019FA	2000 Fall	In State	U.S. Citizen	Female	New Transfer Student	New	5	15
0			2019FA	2018 Fall	In State	U.S. Citizen	Male	Returning Student	New	5	
0			2019FA	2012 Fall	In State	Permanent Resident	Female	New, First Time Student	New	5	
0		y	2019FA	2011 Fall	In State	U.S. Citizen	Female	New Transfer Student	Add	5	
0		ng	2019FA	2013 Spring	In State	U.S. Citizen	Female	New, First Time Student	New	5	
0			2019FA	2019 Spring	In State	U.S. Citizen	Female	New, First Time Student	New	5	
0			2019FA	2015 Spring	In State	Permanent Resident	Female	New Transfer Student	New	5	
0			2019FA	2015 Fall	In State	Permanent Resident	Female	New, First Time Student	Add	5	
0		h	2019FA	2015 Fall	Foreign Country	Student Visa (F-1,M-1)	Female	New, First Time Student	New	5	
0		cer	2019FA		In State	U.S. Citizen	Male	High School Stu.-R40 Rqd.	New	5	
0		at	2019FA	2016 Fall	In State	Permanent Resident	Female	New, First Time Student	New	5	
0			2019FA	2016 Fall	Foreign Country	No Value Entered	Female	New, First Time Student	New	5	
0			2019FA	2016 Fall	In State	Permanent Resident	Female	New, First Time Student	New	5	
0			2019FA	2018 Spring	Foreign Country	No Value Entered	Male	New Transfer Student	New	5	
0		y	2019FA	2018 Spring	In State	U.S. Citizen	Female	New Transfer Student	New	5	
	15							Sum:		75	

USING WEB INTELLIGENCE'S INTERACTIVE VIEWER

The Web Intelligence Interactive Viewer is a modern version of WebI which allows you to consume documents in a new way. The interface is simplified in order to favor a more intuitive user experience while reducing the learning curve as much as possible. This interface is ideal for the casual user or someone who is primarily viewing content.

To access the interactive viewer use the following URL:

<https://croa.sjeccd.org:8080/BOE/BILaunchpad/logonNoSso.jsp>

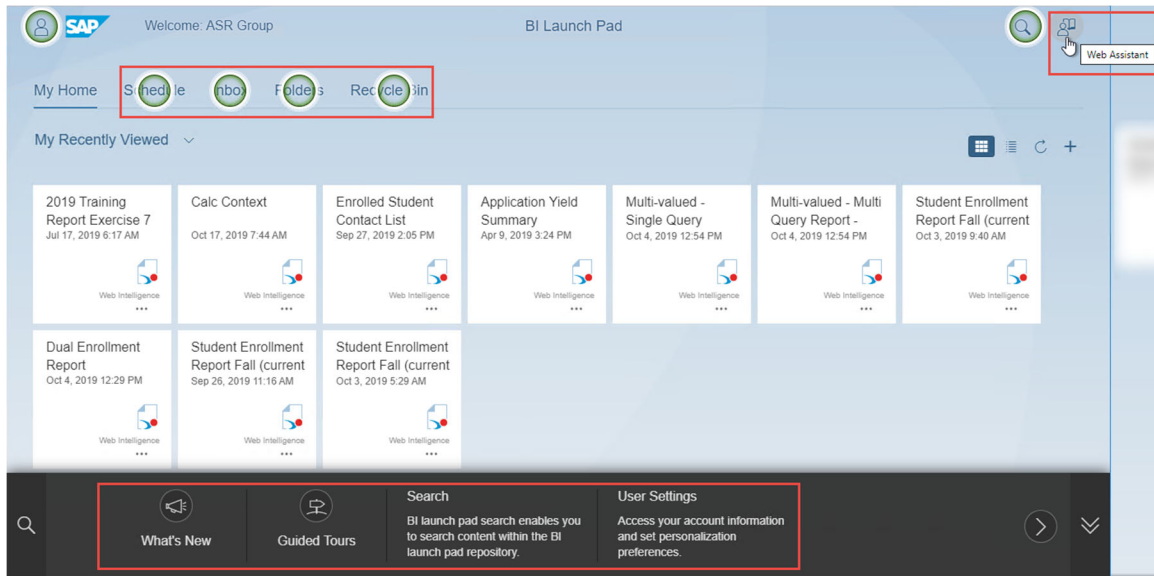


Navigating the Home Screen

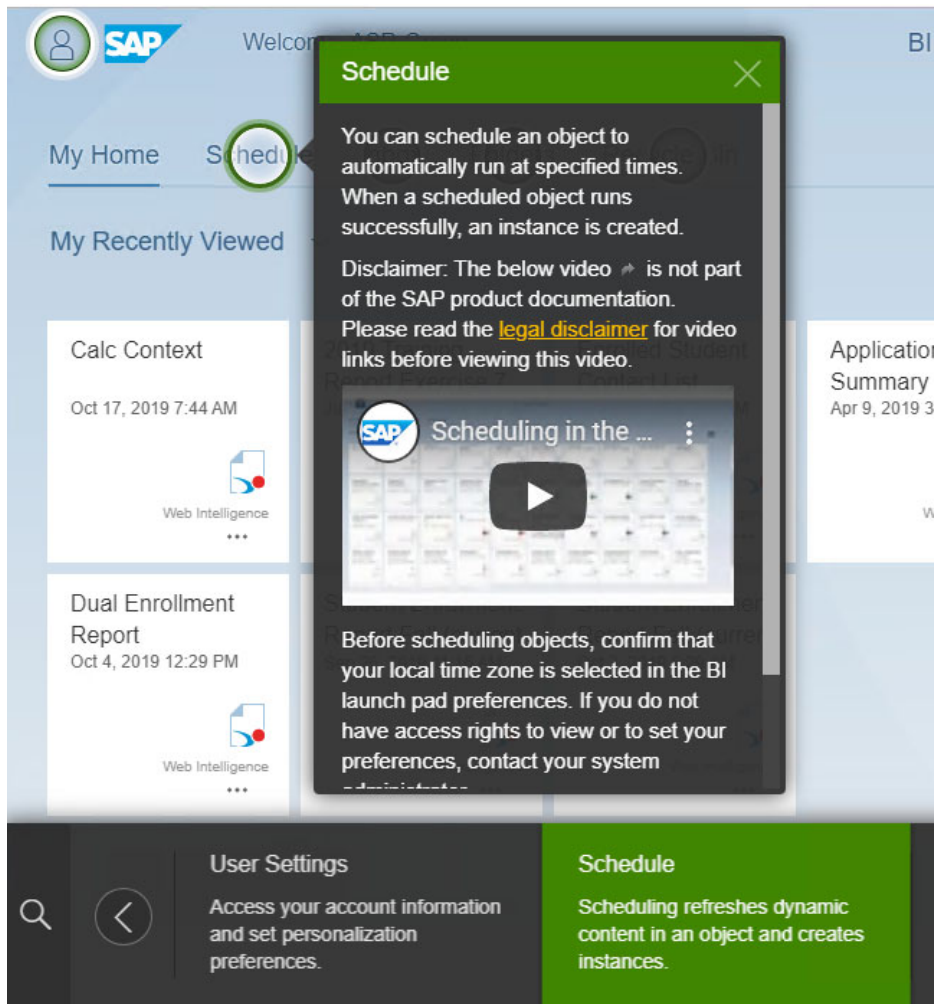
Once you are logged into the interactive viewer the default home screen offers navigation to recently viewed documents and links to documents you have scheduled, your BI Inbox, personal and public folders, and your recycle bin as well as the ability to customize and refresh your view of documents, search documents and web assistance for understanding all available features.

If you are in web assistance mode, you will see green circles on top of each feature which, when clicked, will open a window with more information and it is linked to the informational banner at the bottom of the screen. Click on '[Web Assistant](#)' to exit this mode and enable the ability to navigate within BI Launch Pad.

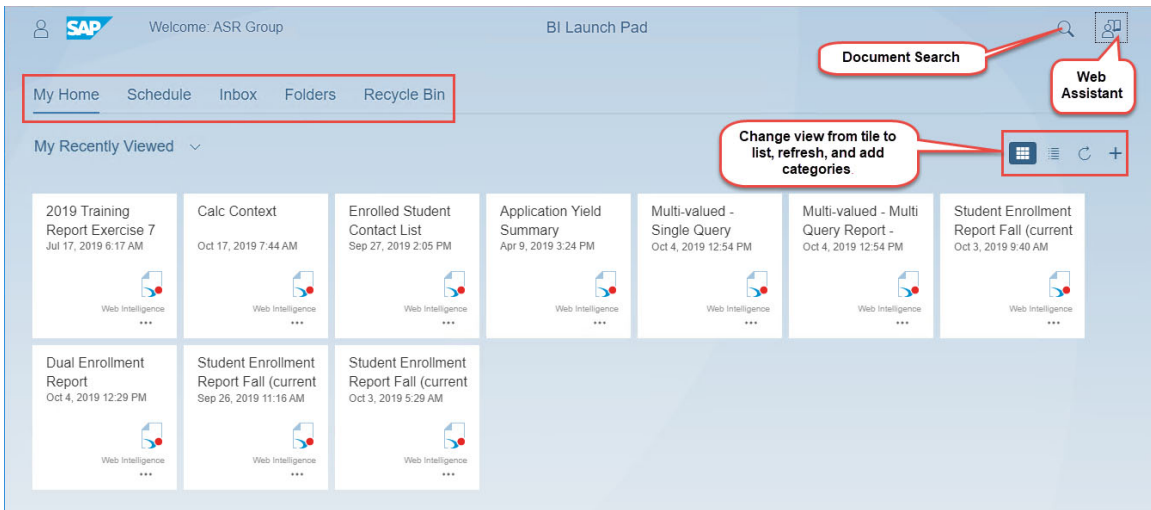
Web Assistant Mode:



In the following screenshot, we've clicked on the green circle for 'Schedule'.

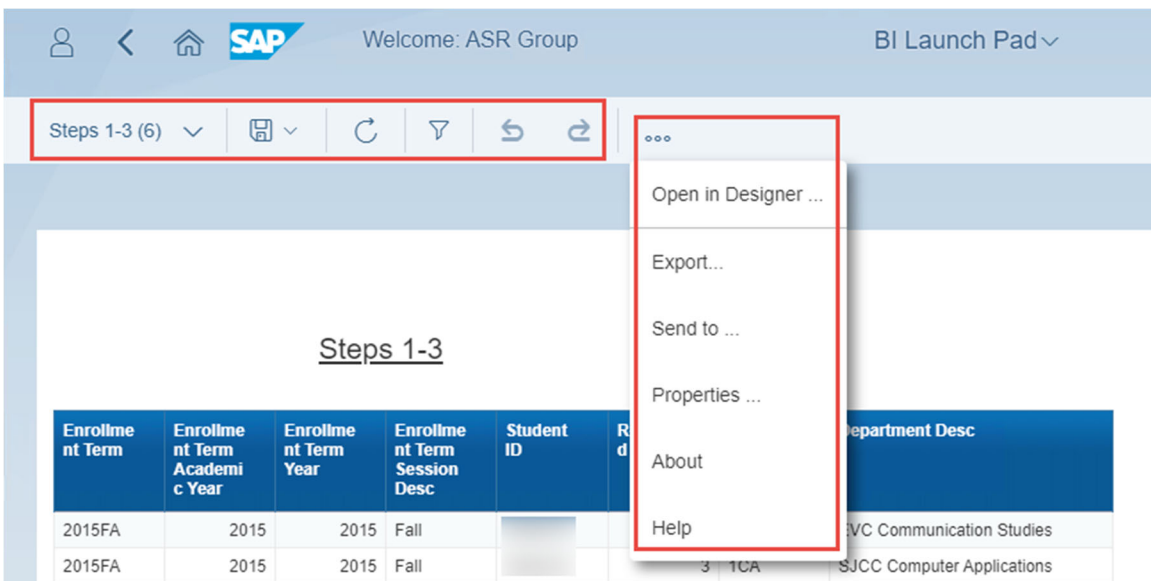


Interactive Mode:

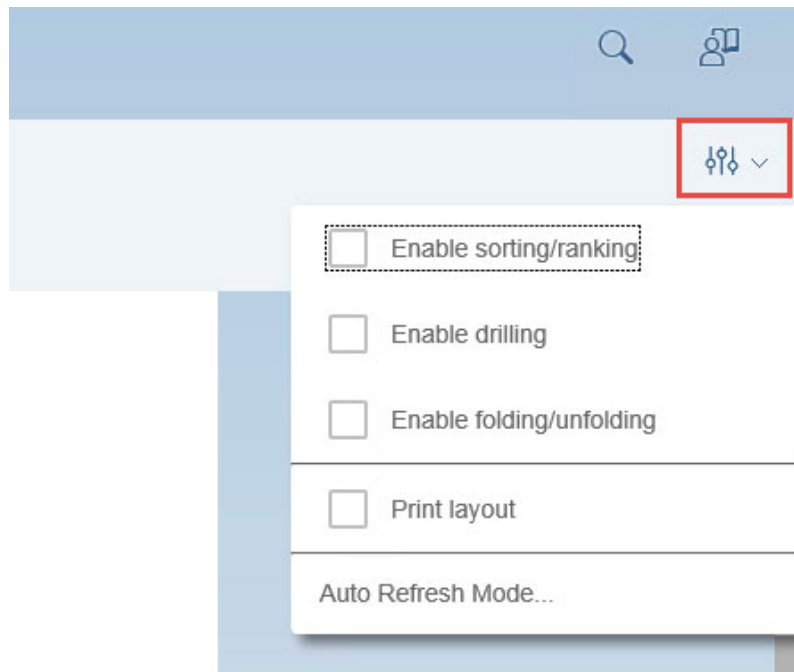


Accessing Reports

Click on any report in your favorites or available public folders to open in the interactive viewer. The top left menu of options allows the viewer to switch between report tabs, save or save as, refresh the document, view input controls, undo or redo last actions, and offers a menu for opening in designer, exporting sending etc.



Additional interactive options are available on the far right:



EFFECTIVE SHARING OF REPORTS AND ANALYSES

Once you develop Web Intelligence reports to provide the visualizations and analyses that you need to manage your institution's business processes, there are several factors you need to consider to effectively share this data across the institution. These include:

- Ensuring that users are seeing up-to-date data when they look at reports
- Ensuring that long-running reports are refreshed ahead of time
- Ensuring that reports are scheduled on a regular basis
- Allowing inspection of previous executions of reports
- Ensuring the users have reports and analyses delivered to them in the most convenient method and format

The following specific techniques to address these factors are explained in this section:

- Setting up reports to automatically refresh on open
- Exporting Web Intelligence report content to PDF and Excel format
- Working with the Web Intelligence report scheduler
- Looking at historic instances of reports

Setting Up Web Intelligence Reports to Refresh on Open

When business users access Web Intelligence reports through the BI Launchpad, there is always the risk that the information in each report will be out of date. There are several techniques you can use to help alleviate this problem.

The first of these is to clearly display the latest refresh date on your report, using the function that Web Intelligence provides to display the last execution date and time of the report. To create this display in your report, first drag an empty text cell into the header area of the report. Then enter the following formula for the new cell:

=“Last Refresh Date/Time: “ + FormatDate (LastExecutionDate ());”MM/dd/yyyy hh:mm:ss A”)

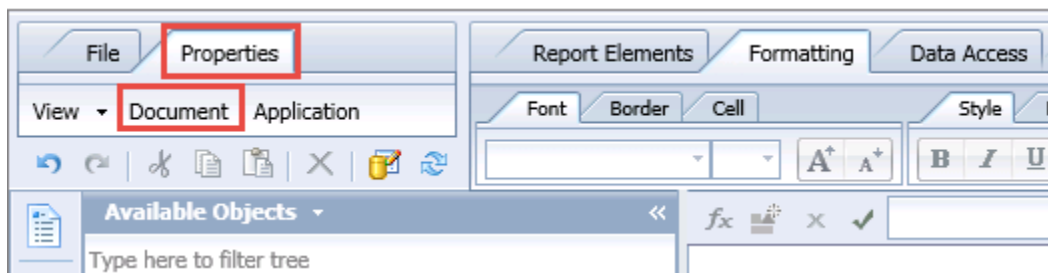
Note that this formula also uses the ‘FormatDate’ function which allows you to format a date/time value (which is returned from the ‘LastExecutionDate’ function) however you want, as specified by a passed formatting string. This will result in the following text notification being displayed on your report:

Last Refresh Date/Time: 10/24/2019 01:26:10 PM

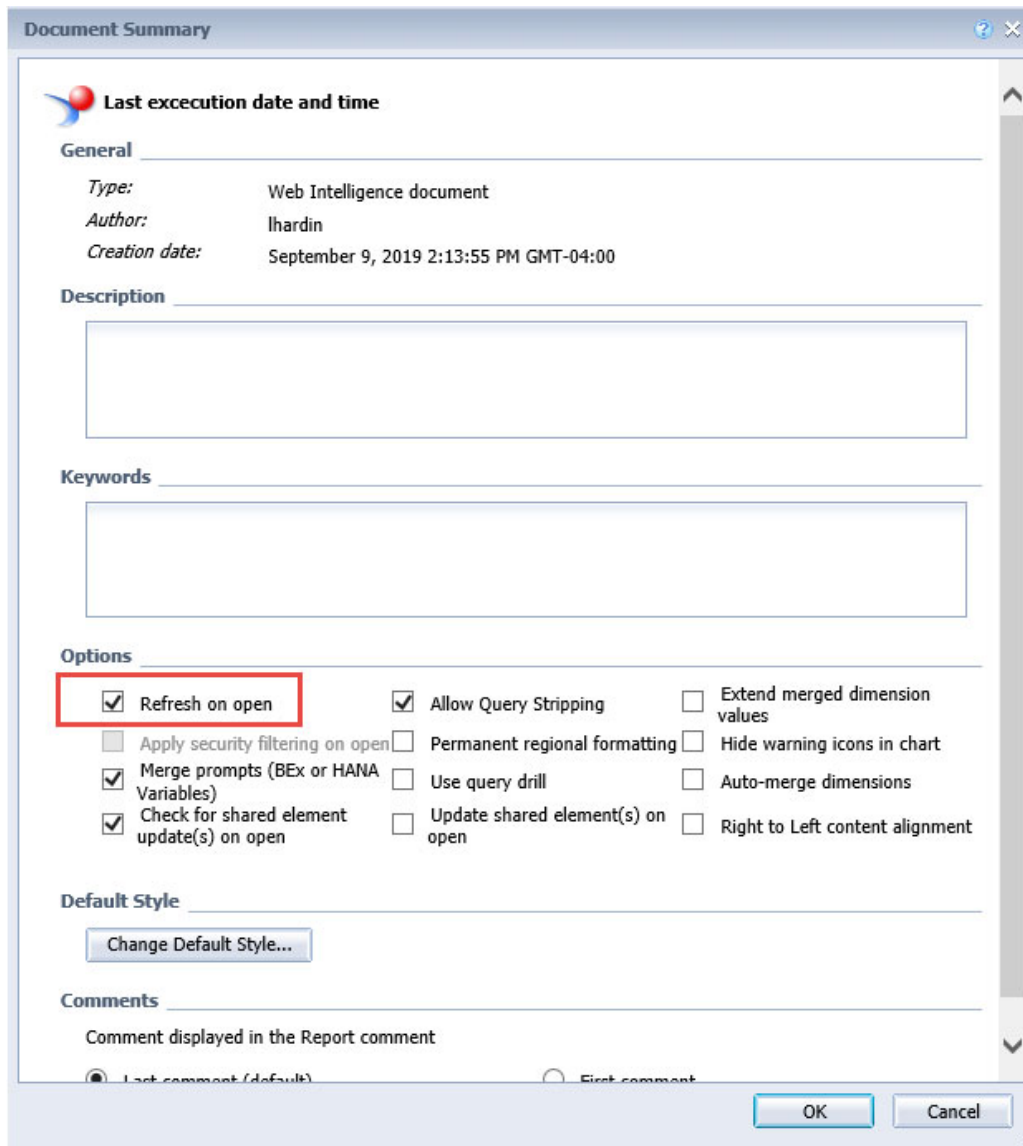
Enrollment Term	2017FA		201
Immigration Status Desc	In State Count	Out of State Count	In State Count
No Value Entered	858	27	969
Other Status	105	95	110
Permanent Resident	2,154	93	2,251
Refugee/Asylee	63	24	86

However, this merely informs the user of the last refresh, but still allows an unobservant user to continue with “stale” data. A second technique that is very effective is to set up each Web Intelligence report so that whenever it is accessed through the BI Launchpad, it automatically causes the report to be refreshed. Obviously, this requires that the execution time of the report be reasonable. But in this case, this absolutely guarantees that users are looking at the latest data available in the underlying ODS database.

To set this up, open the report you want to auto-refresh, and select the ‘Properties’ tab above the left panel of the report window. Then select the ‘Document’ tab on the sub-menu:



This will bring up the 'Document Summary' window:



Click on the 'Refresh on Open' checkbox on this window, click 'OK' and save your report.

Now, when you open your report, either for editing or simply to view it, the queries associated with the report will be automatically run before the user is shown the report content. This guarantees that the reporting user will see only report data that is up to date.

Exporting Web Intelligence Content to PDF or Excel Format

One of the key challenges in any BI environment is making reports and analyses available to information consumers who don't have access to the BI environment. Therefore, it can be

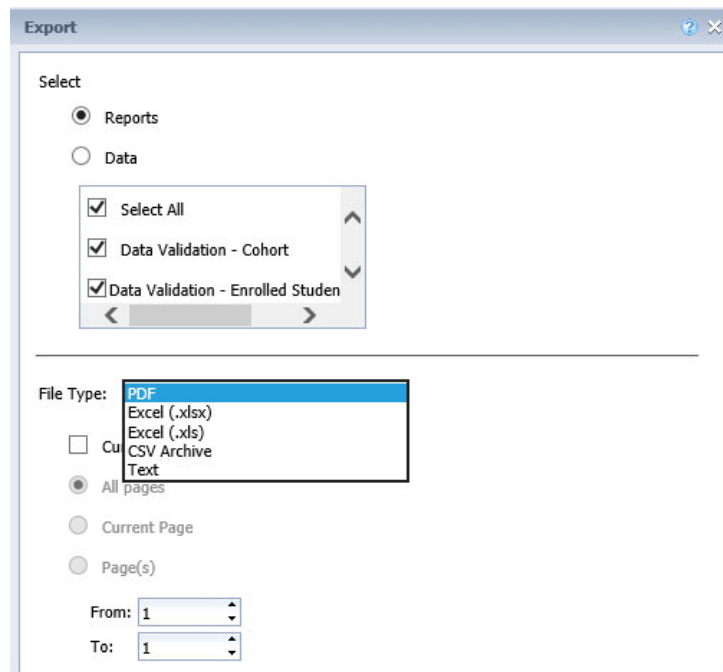
essential to provide the information from Web Intelligence reports in externally accessible formats. Web Intelligence provides the ability to export report content to PDF, Excel and comma-delimited data files.

It is strongly recommended that this capability only be used to share the results of analysis done within Web Intelligence reports with non-BusinessObjects users. It is NOT recommended that these exported datasets be used for further analyses.

At any rate, to do this, select the 'Export' button in the 'File' tab:



Selecting this provides several options:



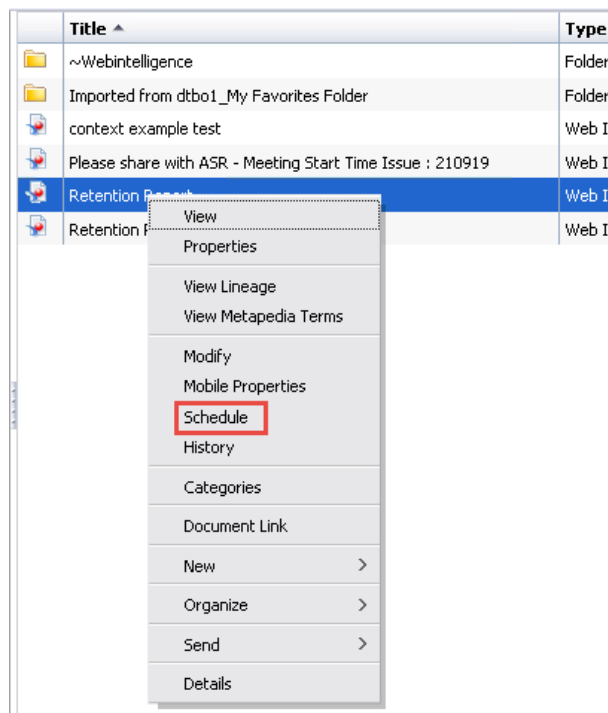
You can export the reports or just the data, the entire document (as a multi-page PDF or multi-spreadsheet Excel document) or just the current report tab, as Excel, CSV, PDF, or text file.

All these options will result in the creation of a file on the default location of the computer which is running the browser being used to access Web Intelligence. You can experiment with the different output formats to see how Web Intelligence maps the formatting and data into the various formats.

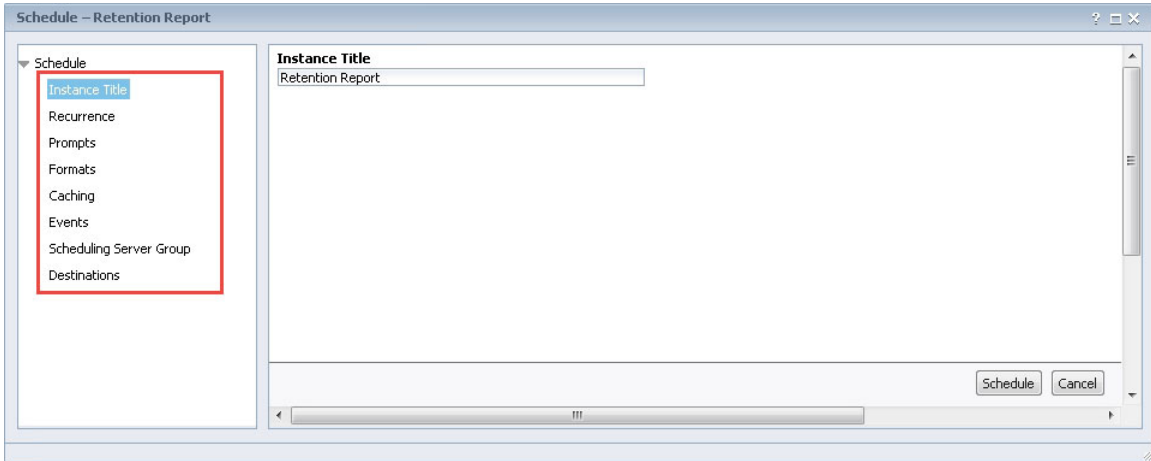
Working with The Web Intelligence Scheduler

BusinessObjects provides a very convenient report execution scheduling tool that you can use to run reports immediately or on a regular scheduled basis. It also provides the ability to output the results of the reports in various supported formats and a variety of destinations for report output, including email to specific user email addresses. Note that the capabilities of Web Intelligence in this regard are quite extensive; we are only going to address some basic capabilities in this section.

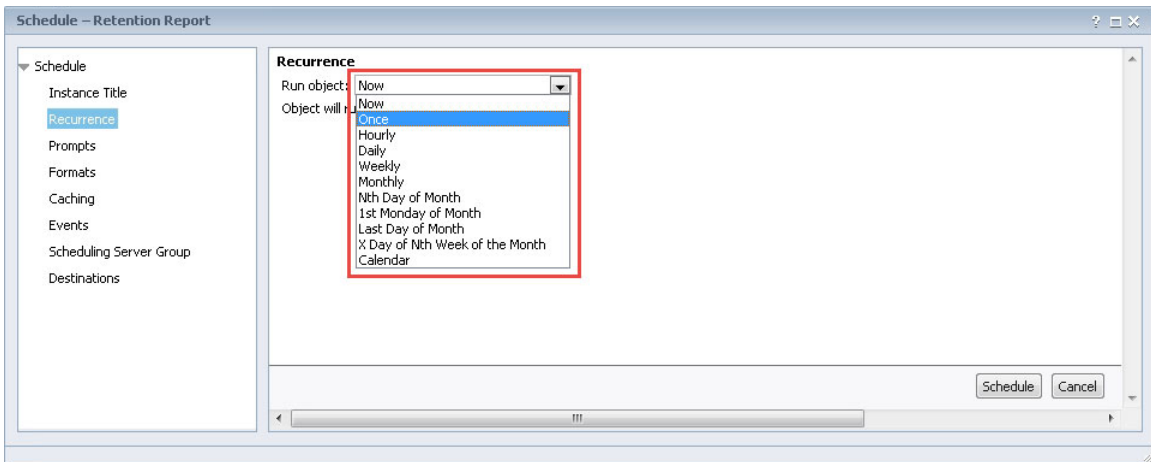
To schedule the execution of a Web Intelligence report, right click on a report in the BI Launchpad and select 'Schedule':



This will bring up the 'Schedule' window, which has a set of sub-window selections on the left-hand side:



The most important of these is the 'Recurrence' sub-window, which you can bring up by clicking on the 'Recurrence' selection. This window allows you to either run the report immediately or set up a recurring execution at a specified interval:



The parameters for each selection vary, but those for the 'Daily' interval show a typical set:

Recurrence

Run object:

Object will run once every N days.

Days(N) =

Start Date/Time:

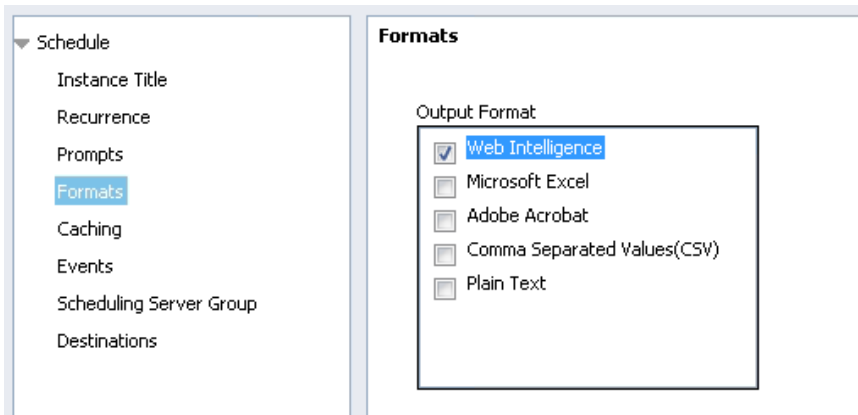
End Date/Time:

Selecting the 'Now' selection and then the 'Schedule' button, will immediately start the execution of the report. It will place the user immediately into the 'History' screen, which will allow you to monitor the current status of the execution. Click the refresh icon to update the listed status for the report:

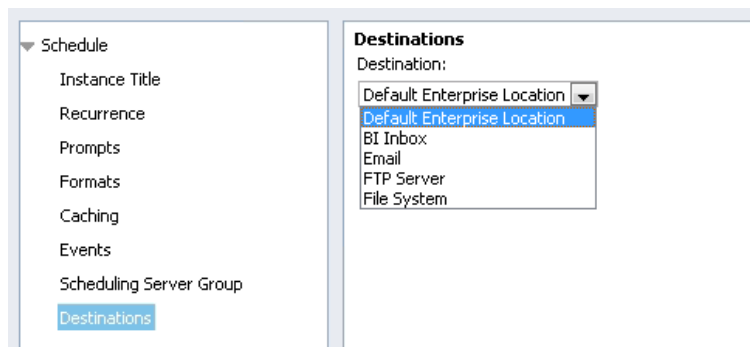
Instance Time	Title	Status	Created By	Type	Parameters
Jun 8, 2017 11:38 AM	Retention Report	Success	JMarsh	Web Intelligence	2009

Eventually the 'Status' field will show 'Success' or 'Failure'. Clicking on the status value will provide further information on the execution of the report. Note that the parameters supplied to the report are listed in the 'Parameters' column.

Another important sub-window selection of the 'Schedule' window is the 'Formats' sub-window. This allows you to specify the form of the output of the report:



Finally, the 'Destinations' sub-window allows you to specify where the selected output should be stored or delivered:

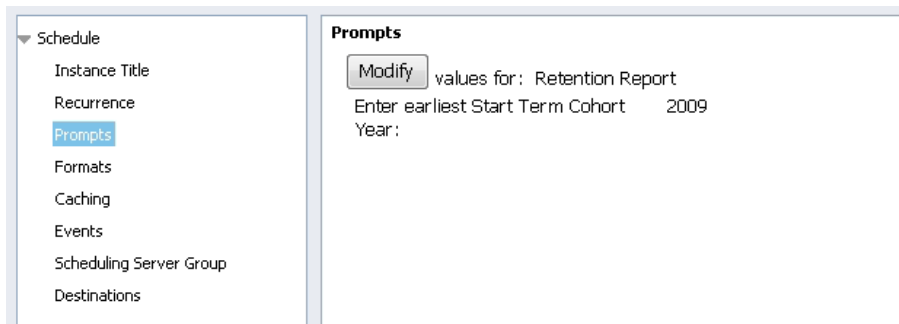


The options all depend somewhat on the 'Format' selected in the previous window, but general are the following:

- Default Enterprise Location – the BI Launchpad folder where the original report is stored, typically

- BI Inbox – The BI Launchpad ‘Inbox’ folder of one or more selected users in the BusinessObjects system
- Email – The output can be included in an email sent to one or more addresses. For the Web Intelligence format, this would send a URL in the email which links into the BusinessObjects system. For the various file formats (i.e. PDF, Excel, Text), this would include the report as an attachment to the email
- FTP Server – sends the output to an FTP server connection that has been configured for the system
- File System – A specific folder on a file system accessible from the BusinessObjects server using the network user id/password provided. Note that this will NOT default to the user’s workstation as does the ‘Export’ functionality

The parameters used to run the report will be whatever the saved prompt values are from the saved report. If you want to change those parameters for the scheduled run, select the ‘Prompts’ sub-window, which will display the saved values of each prompt and allow you to modify those values for the scheduled report.



When all settings are entered, click on ‘Schedule’ button to initiate the scheduled execution of the report.

After a report has been run as a scheduled job, you will see that its display in the BI Launchpad will have changed a little. You will see values in the ‘Last Run’ column and the ‘Instances’ column:

Title ^	Type	Last Run	Instances
~Webintelligence	Folder		
Imported from dtbo1_My Favorites Folder	Folder		
context: example test	Web Intelligence		0
Please share with ASR - Meeting Start Time Issue : 210919	Web Intelligence		0
Retention Report	Web Intelligence	Jun 8, 2017 11:38 AM	2
Retention Report EMB	Web Intelligence		0

When you right click on the report itself, you will now see ‘View Latest Instance’ as a new selection:

	Title ^	Type
	~Webintelligence	Folder
	Imported from dtbo1_My Favorites Folder	Folder
	context example test	Web Intelligence
	Please share with ASR - Meeting Start Time Issue : 210919	Web Intelligence
	Retention Report	Web Intelligence
	Rete	Web Intelligence

View
View Latest Instance
Properties
View Lineage
View Metapedia Terms
Modify

Clicking on this selection will bring up the most recently refreshed version of the report in 'View' mode. Note that this is a "report instance", something a little different than the editable version of the report we have been working with up to now in this course. It is a read-only version of a specific execution of the report and is not an editable version that we can use to modify the original report in any way. You can think of this "report instance" as analogous to a PDF created from a Microsoft Word file.

Looking at Historic Instances of Reports

As we have seen in the previous section, when reports are scheduled, we typically look at the most recent instance of that report, showing the report with the most recently refreshed data. However, the question frequently arises as to what previous instances of the report looked like, typically to understand whether exception situations reflect a current event or have occurred in the past.

BusinessObjects provides the ability to save a set of previous instances of a report, complete with all the data generated when the report was run at the previous time. The number of these historic instances that are maintained can be set on a report or system basis.

To access the previous instances of the report, right click on the report and click on the 'History' menu selection:

	Title ^	Type
	~\Webintelligence	Folder
	Imported from dtbo1_My Favorites Folder	Folder
	context example test	Web Intelligence
	Please share with ASR - Meeting Start Time Issue : 210919	Web Intelligence
	Retention Report	Web Intelligence
	Retention	Web Intelligence

- View
- View Latest Instance
- Properties
- View Lineage
- View Metapedia Terms
- Modify
- Mobile Properties
- Schedule
- History
- Categories
- Document Link

This will bring up the same form that was displayed when we scheduled a report to run immediately. Note that all the historical instances of the report are listed here and that the values in the date field are active links. Clicking on any of the values in the date field will bring up the corresponding instance of the report in ‘View’ mode.

Instance Time	Title	Status	Created By	Type	Parameters
Jun 8, 2017 12:16 PM	Retention Report	Success	JMarsh	Web Intelligence	2009
Jun 8, 2017 12:04 PM	Retention Report	Success	JMarsh	Web Intelligence	2009
Jun 8, 2017 11:38 AM	Retention Report	Success	JMarsh	Web Intelligence	2009

Using Publications to Automate Report Distribution

In addition to working with the report scheduler, BusinessObjects provides an additional method, called publications, for automating the delivery of several different personalized reports at once to different recipients to multiple destinations, including BI Inboxes and email, based on different defining characteristics.

A publication is a collection of documents intended for distribution to a mass audience. Before the documents are distributed, the publisher defines the publication source or Web Intelligence document, its recipients, and the personalization to apply.

Enterprise or Dynamic Recipients

You can choose to send your publication to enterprise and/or dynamic recipients.

Enterprise recipients are users who are part of the BI platform system and that have a login to SAP BusinessObjects BI Launchpad. As an enterprise recipient, published documents can be sent to their BI Inbox, via email, or any of the other destination options available which are the same as the options available through the report scheduler explained in an earlier section.

Dynamic recipients are non-Enterprise users that are either outside of your network or are not configured with an enterprise user account. Published document can only be sent through email if they are not on your network. Also, keep in mind that a BI Inbox is not a valid destination for dynamic recipients.

Personalization

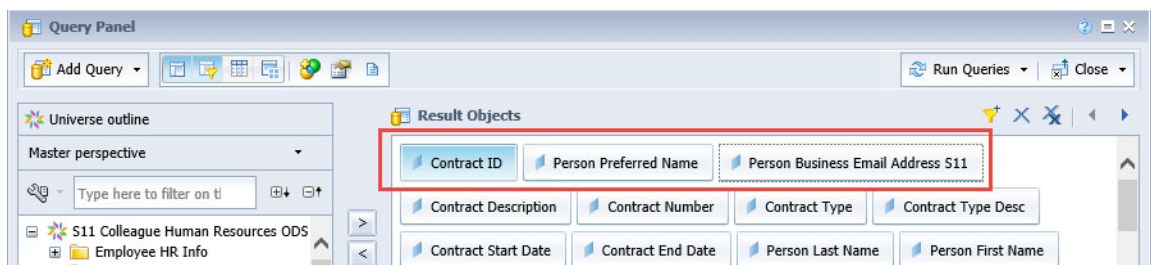
Personalization is the process of filtering data in source documents so that only relevant data appears to publication recipients. When personalization is applied, users only receive information that is relevant to them. For example, a published document for each faculty member with only their specific advisees, a published document for the deans so they are receiving program review reports for their specific programs, or a published document containing contracts for employment whereas each recipient only receives their contract information.

Creating a personalized publication for dynamic recipients

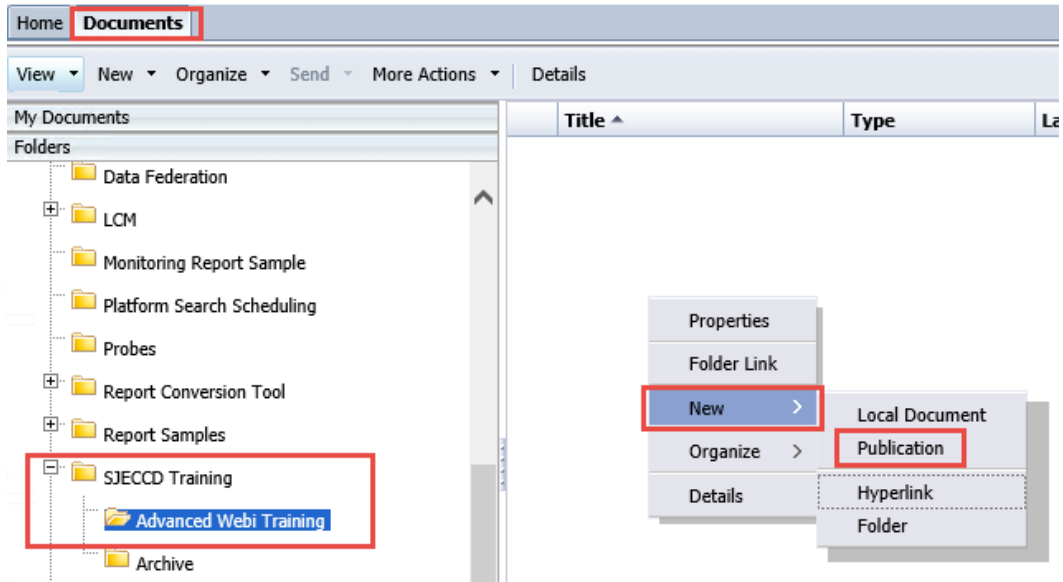
The following are steps for setting up a publication for personalized report distribution to dynamic recipients.

STEP 1: Locate the source document for your publication and ensure that it contains the necessary objects for dynamic recipient mapping:

- Recipient ID (Person ID, Student ID, Contact ID etc.)
- Full name of recipient
- Email address

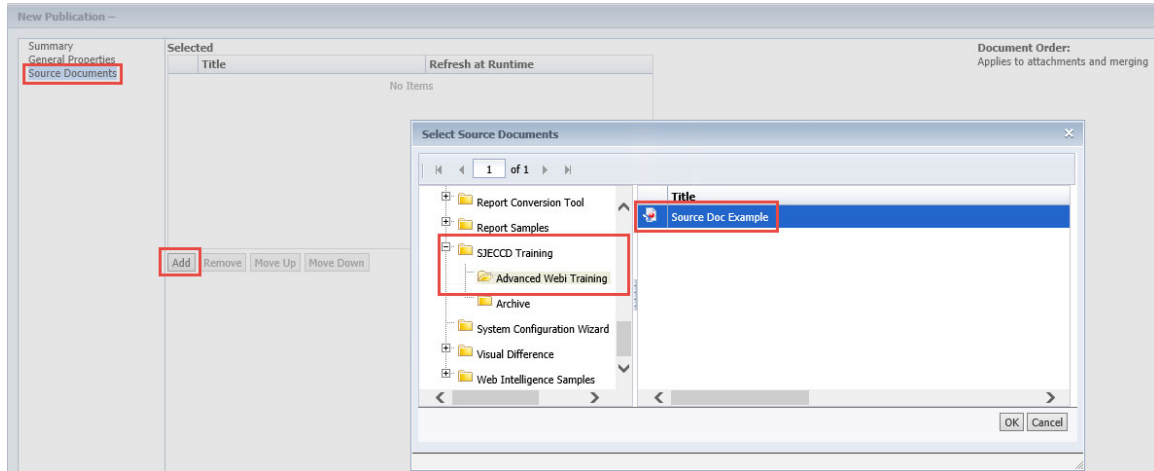


STEP 2: Create a new publication in BILaunchpad. Navigate to the 'Documents' tab, expand the folder drawer, locate the folder you intend to create the publication in, right-click, and select 'New' -> 'Publication'.

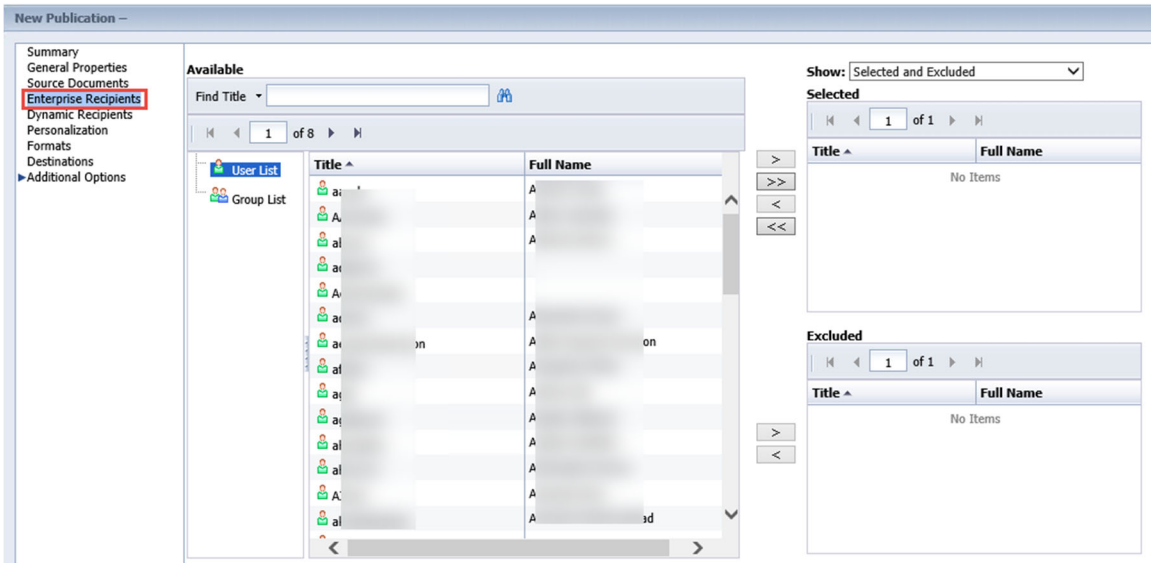


STEP 3: In the 'New Publication' window, enter the required title for your publication. Optionally, you can enter a description or keywords.

STEP 4: Click 'Source Documents' in the navigation list on the left and then click 'Add', in the 'Select Source Documents' window select a source document(s) to add to the publication and click 'OK'.

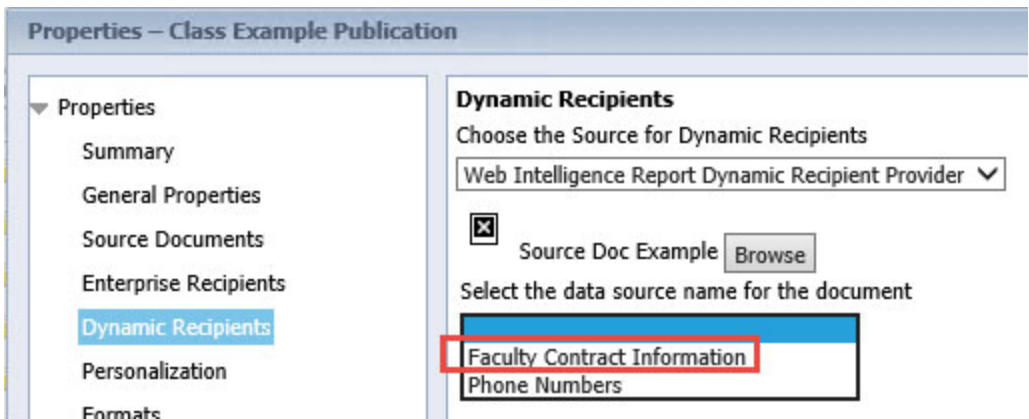


STEP 5: Specify any Enterprise Recipients that you would like to select or exclude from this publication.

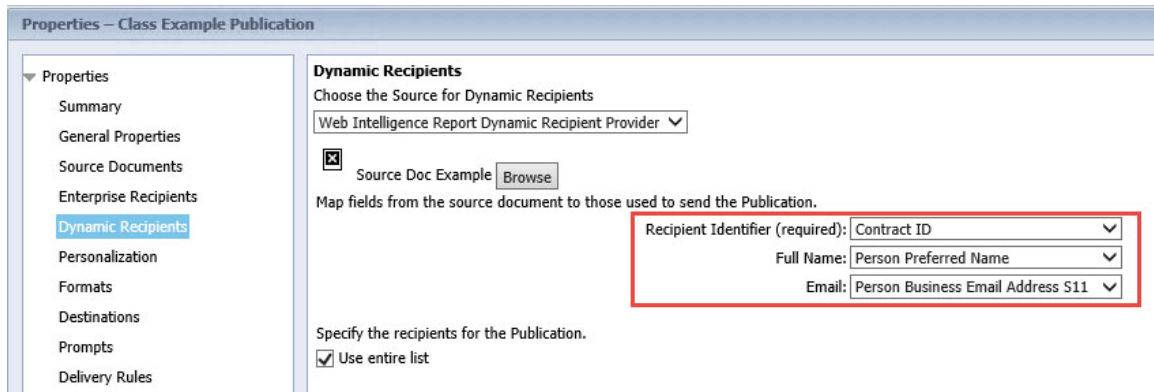


STEP 6: For Dynamic Recipients, select 'Web Intelligence Report Dynamic Recipient Provider' under 'Choose the Source for Dynamic Recipients' and navigate to your source document and click 'Ok'.

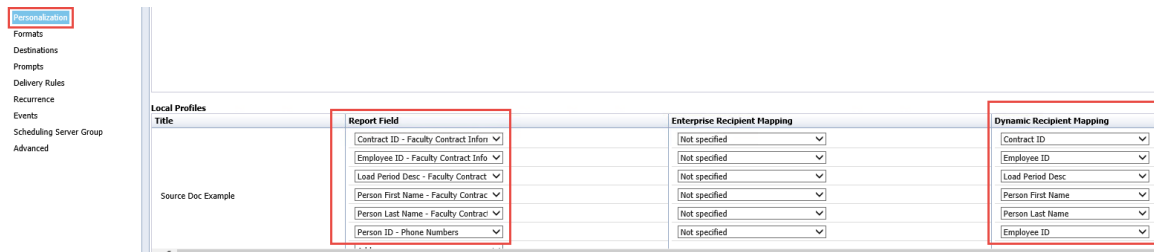
STEP 7: If your source document contains more than one query, you will be asked to 'Select the data source name for the document'. Choose the query which contains the recipient ID, name, and email objects.



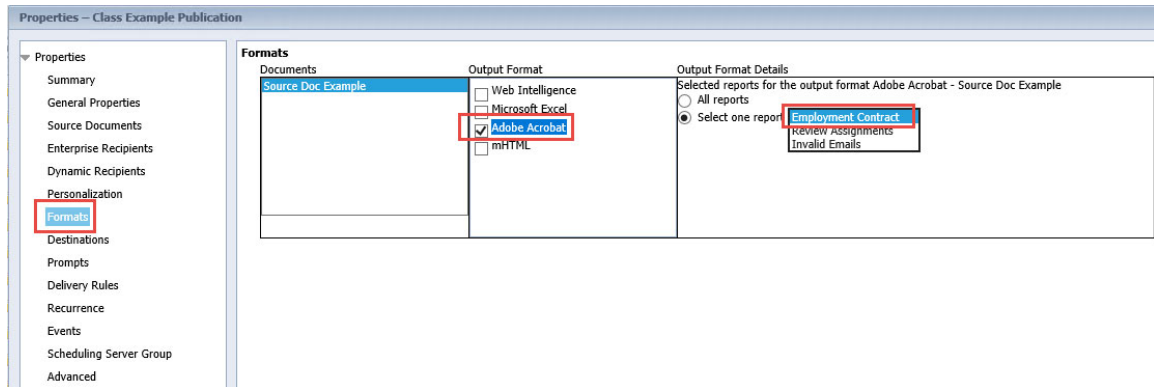
STEP 8: Select the appropriate objects in the dropdowns under 'Map fields from the source document to those used to send the Publication'.



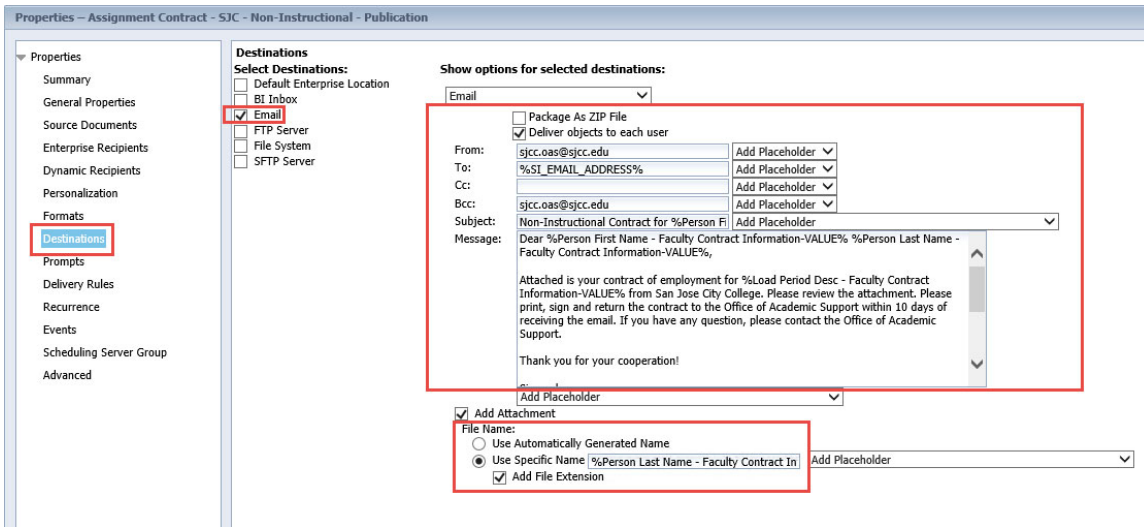
STEP 9: To personalize the Webi document by filtering fields for dynamic recipients, select 'Personalization' and add a report field and a corresponding dynamic recipient mapping for each required filtered field.



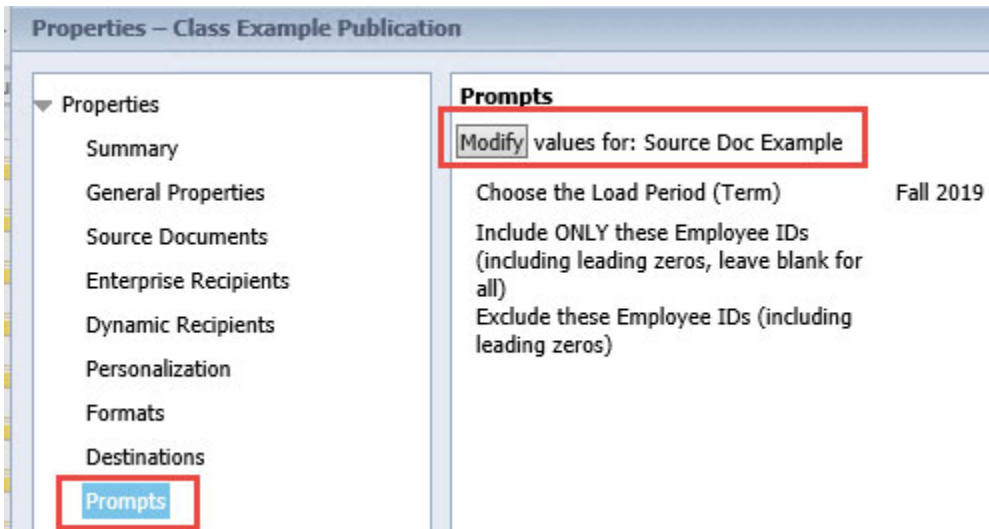
STEP 10: Select 'Formats' and then place a check mark next to the desired output format. Under 'Output Format Details' select an individual report for publishing or allow all reports to publish.



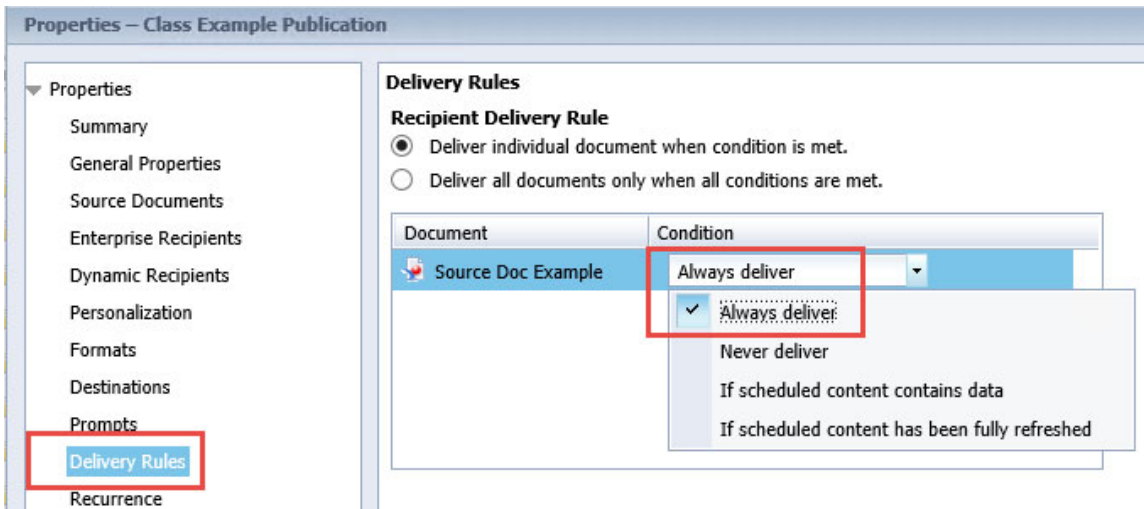
STEP 11: Select 'Destinations' and then place a check mark next to the desired destination choice. If 'Email' is selected, fill out the columns that appear under 'Show options for selected destinations'. The following screenshot is an example of using placeholders for dynamic text and creating a dynamic file name for each recipient.



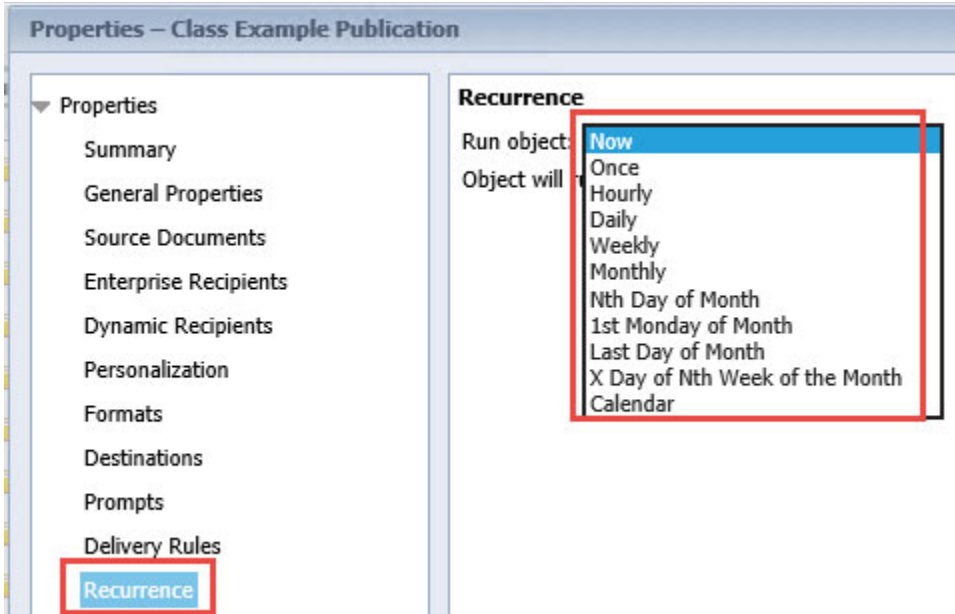
STEP 12: If your report contains any prompt filters, select 'Prompts' and modify as necessary.



STEP 13: Select 'Delivery Rules' and set to meet your requirements.



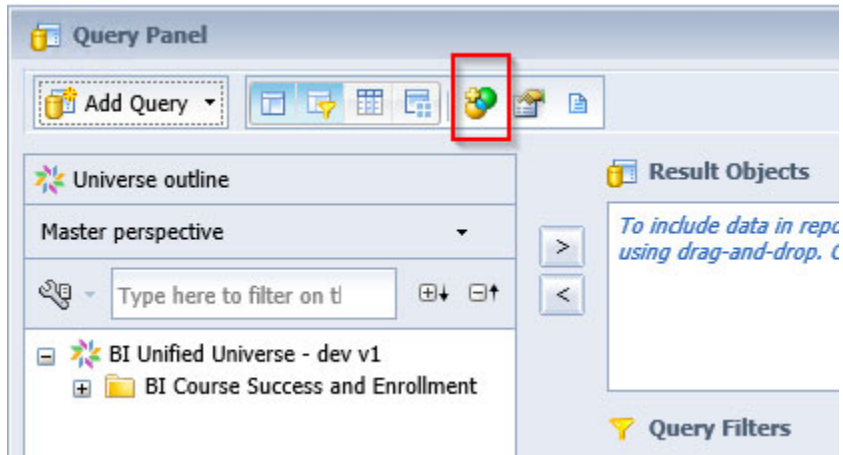
STEP 14: Select 'Recurrence' and set to meet your requirements for scheduling the published document.



STEP 15: Save and test your publication.

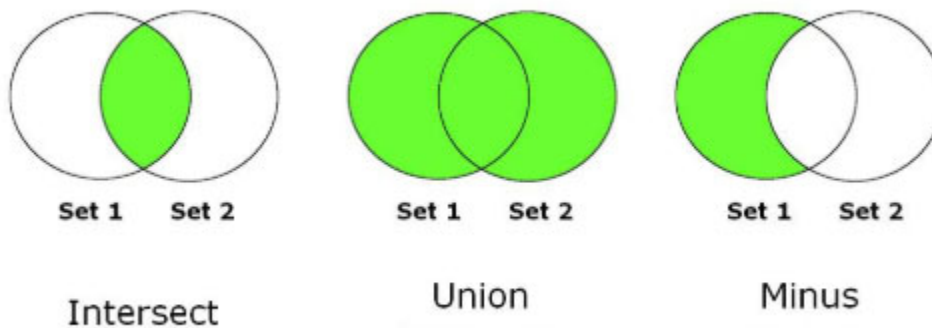
ADDING A COMBINED QUERY

Combined queries, like subqueries, are accessed through the **Query Panel**.



Combined filters can perform 3 functions: union, intersection, or minus. This language might be familiar to those with some SQL background. Like subqueries, the objects referenced in the combined queries must all come from the same data source (universe, Excel file, etc.). Unlike subqueries, the relationships between the objects do not necessarily have to be compatible.

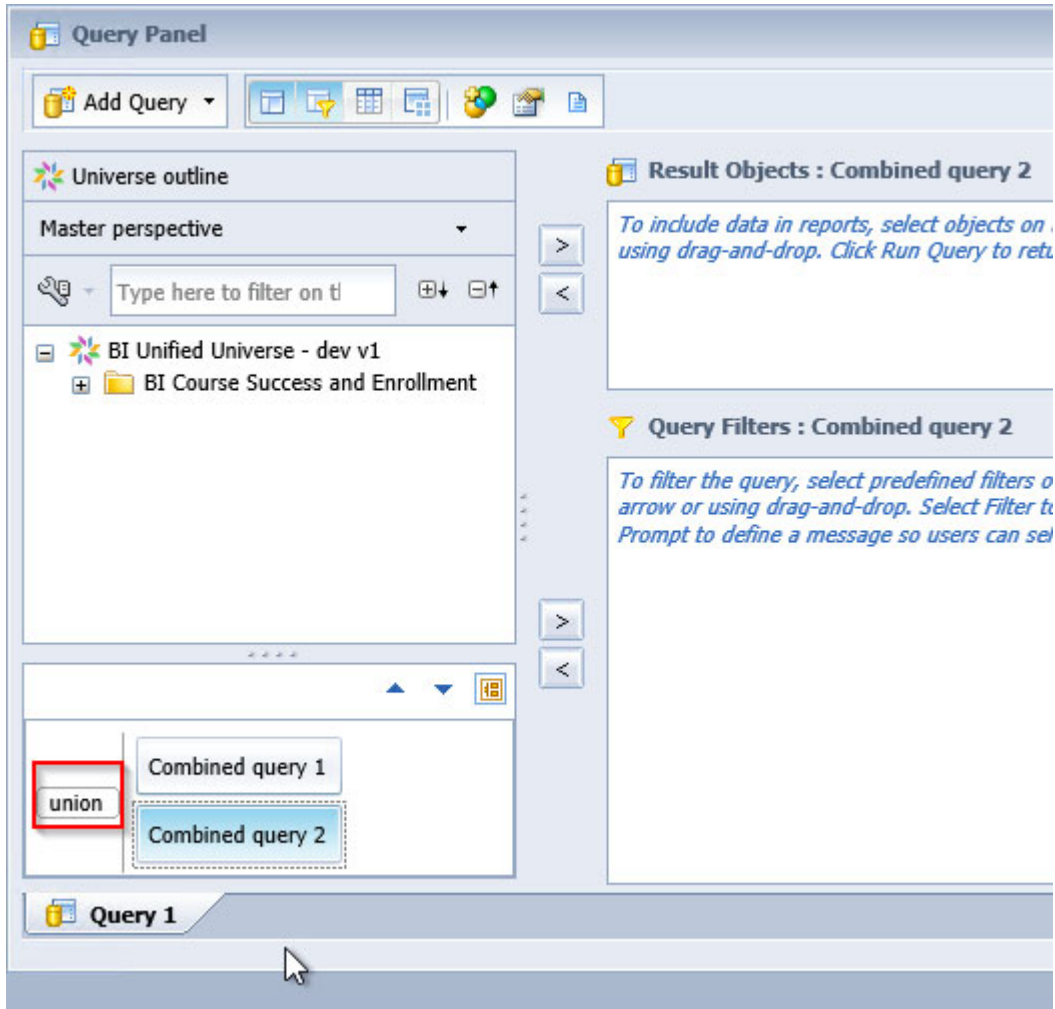
The number of objects in the **Result Objects** pane in all combined queries must be equal and the datatypes must be compatible. The filters in each query can (and should) be different. Combined queries are a way of stacking datasets with the same structure on top of each other and performing set operations on them. **Intersection** will provide only the results common to both datasets. **Union** will provide all the results from both datasets. **Minus** will provide the results that are in the first dataset but not in the second dataset.



An important note about **Intersect**: The results values *have* to be in common or it will return no results! For example, if you are querying course-taking behavior, looking for students who took

ENGL-001A in one term and ACCTG-001A in a future term, you will return no results if you include the course names or enrollment terms in the result objects, because they do not match!

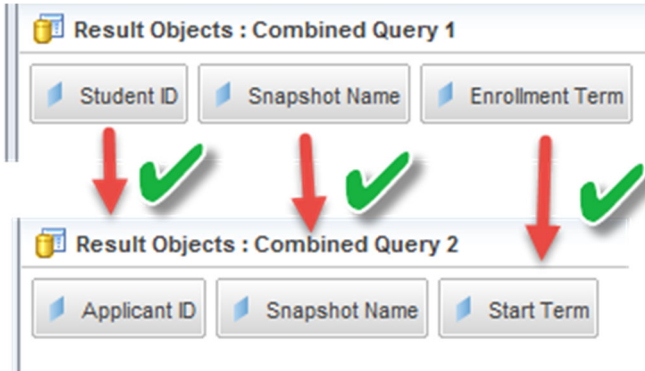
The default set operation in WebI is a union but clicking on the word in the bottom left of the **Query Panel** will toggle it through the different options.



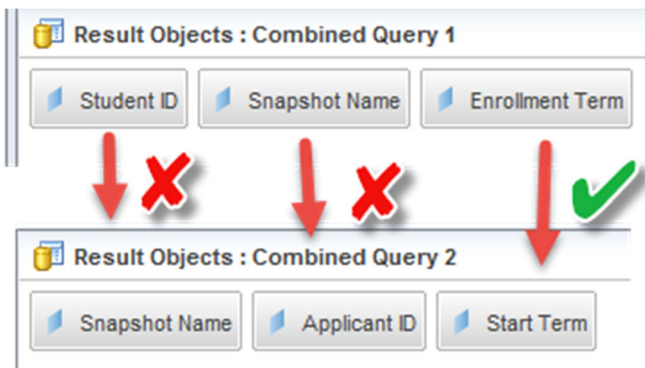
The user can also create more than two combined queries, though we advise caution here. Adding several combined queries increases the complexity and maintenance of the report.

When creating a combined query, WebI will automatically generate the same **Result Objects** as are set in the first query but it will not duplicate the query filters. Sometimes it is desirable to keep the same objects in the **Result Objects** pane, sometimes it is not. It is okay to remove and replace the default objects generated by WebI, provided that you are very careful when making replacements. Replaced objects must be of the same data type and those data types must align with the same order of the first query.

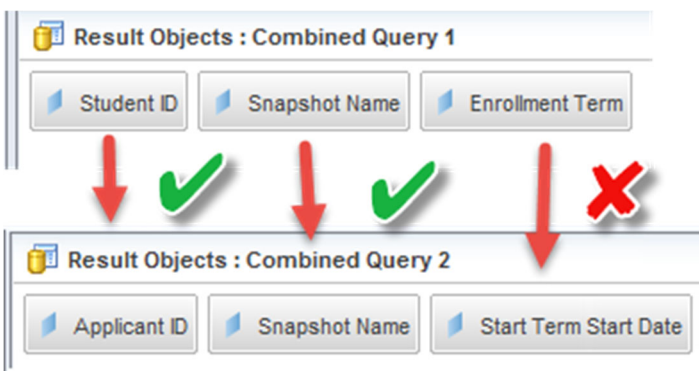
If we replace the default objects from WebI with similar objects, we must make sure to match the order in the first query. As an example, if the first query has Student ID, Snapshot Name, and Enrollment Term, the second query can have Applicant ID, Snapshot Name, and Start Term, but they *must* be in that order:

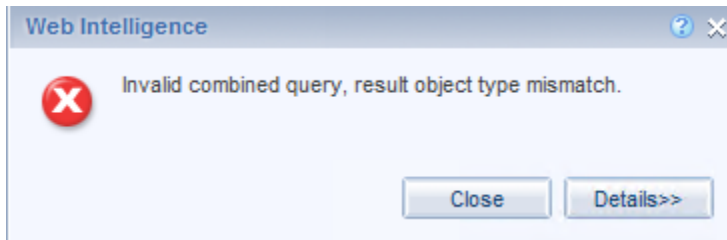


Because the datatypes match, WebI will still run the query if you reorder the results objects so that Student ID, Snapshot, and Enrollment Term are stacked on top of Snapshot, Applicant ID, and Start Term but the resulting dataset will not make any sense!

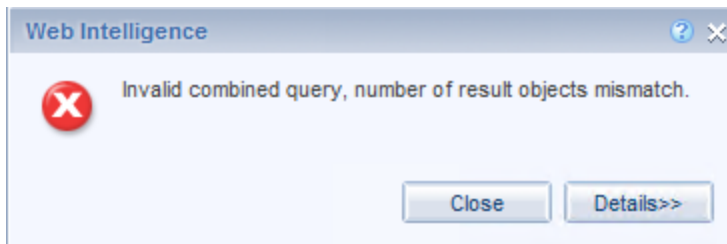
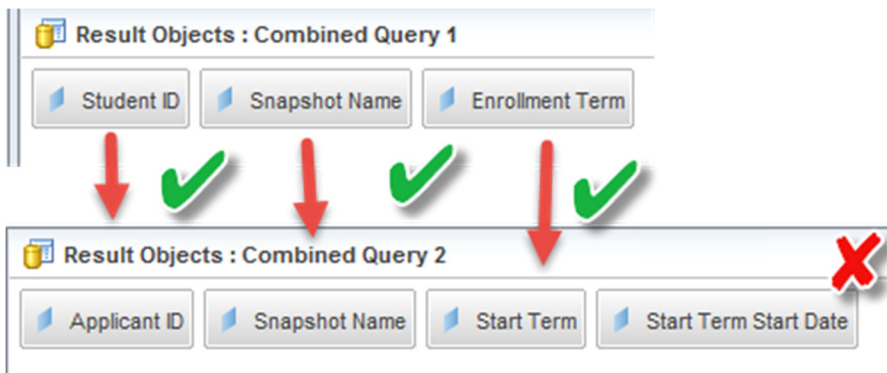


In addition, if we replaced Start Term with Start Term Start Date in the second query, WebI would return an error because the data types do not line up. Enrollment Term is a character field and Term Start Date is a date field. WebI cannot combine different datatypes in the same column.





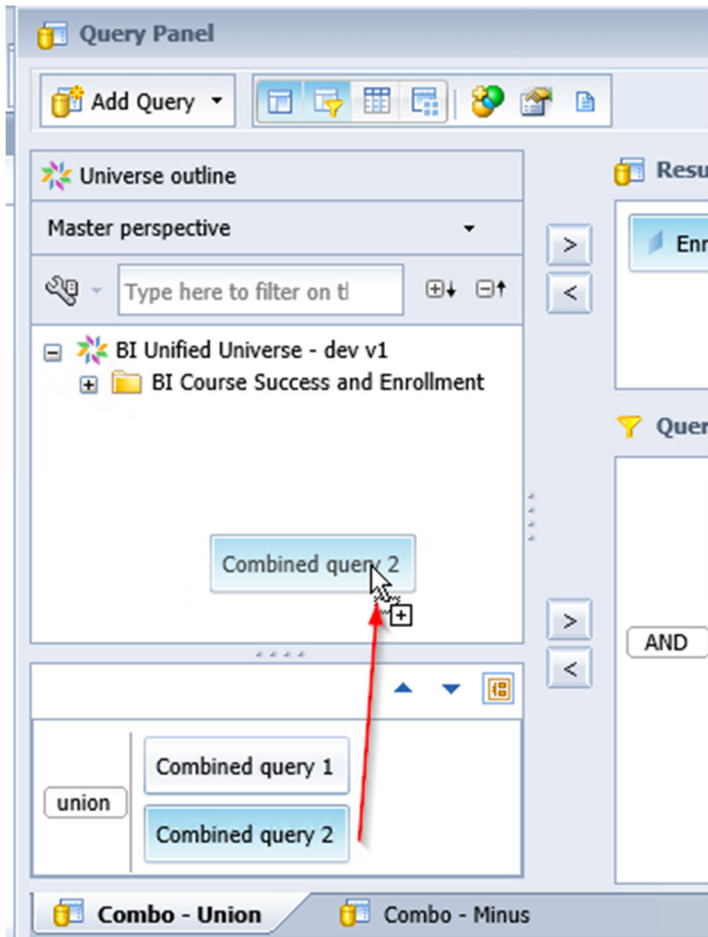
Finally, not only must the order and the data type match in all combined queries, the total number of objects must match as well. If one of the queries has fewer objects than the other, WebI will return an error.



After carefully structuring your **Result Objects** for the second combined query, you are free to filter the query using any criteria desired.

This functionality can be very useful if you want to compare similarly structured sets of data from the same data source that do not necessarily have mapped relationships in the universe and if you do not need a lot of objects in your **Result Objects** pane. The more objects you add to your **Result Objects** pane, the harder it is to maintain the integrity of a combined query. We suggest limiting the use of this functionality to only a few objects to prevent confusion/unexpected results.

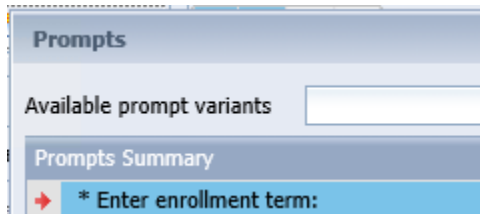
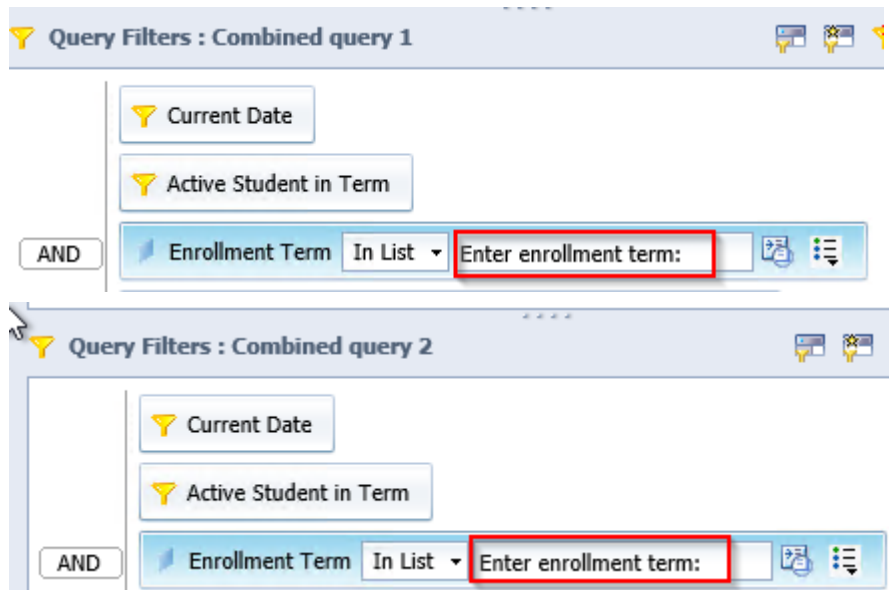
To remove a combined query, click the **Combined Query** object in the bottom left of the **Query Panel** and drag it onto the **Universe outline** pane above:



Unifying prompts

When working with subqueries, combined queries, or multiple queries, a report will often end up with several overlapping prompts. In the example above, we would likely want a prompt on both queries for *Enrollment Term*. If the same value is needed for both queries, using the same prompt text ensures that the user only sees one prompt when they refresh the report, and only has to enter the value once. For unified prompts, the text has to match *exactly* in both places. A space or a missing colon will keep the prompts from unifying.

Conversely, if different values are needed in each query, different prompt text should be entered to separate them, as the prompt text will match by default. This can be done from the **Query Panel** by typing directly into the text box next to the operator dropdown.



Now the user only has to enter the object value(s) one time, to satisfy both prompts.

Learning Objective

In this exercise, you will create a combined query using the unified universe and adjust prompt text to meet the report design needs.

High Level Overview

1. Create a combined query
2. Toggle between union and minus operators to compare results on report
3. Customize prompt text to unify prompts for end user

Practice Steps

STEP 1: Create a new Web Intelligence document using the BI Unified Universe – dev v1 as your source.

STEP 2: Add *Enrollment Term*, *Student ID*, *Course Name*, *Acad Cred Status*, and *Verified Grade* to the Results Objects pane.

STEP 3: Add the *Current Date* snapshot filter, *Active Student* filter, *Enrollment Term*, *Acad Level*, and *Course Name* to the Query Filters pane. Set the *Enrollment Term* to prompt the user and

enter "Enter ENGL term:" (without the quotes) as the prompt text. Set *Acad Level* equal to UG. Set *Course Name* equal to ENGL-001A.

STEP 4: Click on the appropriate icon to create a combined query. Note that adding results objects prior to creating the combined query ensures that those objects are copied over to the second combined query. The query filters are *not* copied. Leave the default "union" operator.

STEP 5: Add the same query filters to Combined Query 2 as exist in Combined Query 1, with two changes. For the *Enrollment Term* prompt text, enter "Enter subsequent course term:" (without the quotes). Set *Course Name* equal to ACCTG-001A.

STEP 6: Run the query, selecting 2017FA as the ENGL term, and all following major terms for the subsequent term prompt.

STEP 7: Record the distinct count of students returned: _____.

STEP 8: Return to **Query Panel**, toggle from the default **union** set operation to **minus**.

STEP 9: Re-run the query and record the count of students returned with the minus query and compare it to the union query: _____.

STEP 10: Return to the **Query Panel** and toggle from **minus** to **intersect**. Remove all but *Student ID* from the results objects (for both combined queries!) and re-run the report. Record the count of students returned; compare to the results from the previous runs: _____.

ADDITIONAL REFERENCES

Independent Materials

[SAP BusinessObjects BI 4.0: The Complete Reference 3/E](#), Cindi Howson

[SAP BusinessObjects Web Intelligence \(Webi\) 4.2: The Comprehensive Guide \(4th Edition\)](#), Jim Brogden

[BusinessObjects Designer XI v3](#), Robert Schmidt

SAP Documentation

[SAP BusinessObjects Web Intelligence User's Guide 4.1](#)

[Using Functions, Formulas and Calculations in SAP BusinessObjects Web Intelligence 4.1](#)

Check [here](#) for updates to these documents.

SAP online learning courses

[\(BOE315\) SBOP BI PLATFORM 4.1 – Business Intelligence Platform: Administration and Security](#)

[\(BOW310\) SBOP BI PLATFORM 4.1 – Web Intelligence: Report Design 1](#)

Ellucian Documentation

[Using the DataOrchestrator ODS](#)

[Reporting from the DataOrchestrator ODS Data Models](#)

Acronyms and Glossary

BI – Business Intelligence: a term used to describe all the activities and technologies involved in providing query and reporting and analysis to answer the business questions of an organization

Class – a folder in a Universe that is a collection of related data elements

Hierarchy – an order of precedence of data elements that implies a drill path such as Year → Month → Week → Day

Measure – a predefined calculation of a Universe class, such as a sum, average, etc.

Object – a data element in a Universe class

ODS – Operational Data Store: a database containing data from one or more source systems organized in a way to support reporting and analysis

Universe(s) – file(s) in the BusinessObjects platform that provide a representation of data for reporting that hides the technical complexities behind the scenes and makes reporting easier.