

Chapter 5 - Calculated Fields

Overview

In this chapter, you will learn how to perform a variety of calculated field functions. You will also learn how to transform, manipulate, retrieve, encapsulate, or derive a new field from existing data. Lastly, you will be able to use calculated fields to create rich, customized reports using only simple field configurations without coding or programming.

Objectives

By the end of this chapter, you will be able to:

- Create calculated fields that manipulate dates and text, evaluate conditions, work with data from related objects, and perform arithmetic calculations.
- Use a combination of calculated fields to solve complex business requirements.
- Describe the security features that control the creation of, and access to, calculated fields.
- Explain the performance considerations when creating a calculated field.

Calculated Fields Overview

Calculated fields are field definitions that you configure to manipulate, transform, retrieve, and derive values based on existing data. You can use calculated fields to:

- Perform date calculations and formatting.
- Perform arithmetic calculations.
- Manipulate text with concatenate, substring, and formatting functions.
- Convert currency fields.
- Derive range bands from numeric or currency fields.
- Determine if a condition is true or not.
- Drill down and look up levels and values in hierarchies and organizations.
- Sum, count, and aggregate information across related instances.
- Look up values in related objects.

How To Use Calculated Fields

You can use calculated fields in a variety of ways. Use calculated fields in reporting to deliver data that is otherwise unavailable on the primary business object of the report. As an example, you may need to generate a Social Security number excluding dashes. You can use a Concatenate Text calculated field function to link together the numbers extracted using the Substring Text function.

Additionally, you might need a business process condition rule requirement to determine whether an employee's one-time retention bonus requires the vice-president's approval. Using a True/False Condition calculated field function, you can evaluate if a worker's bonus is greater than or equal to a percentage threshold to initiate the approval.

Use Calculated Fields in Reporting

You can add calculated fields to a custom report. For example, you can create a Date Difference calculated field to determine how far an employee's performance review is overdue.

You can also use calculated fields to create custom prompts or filters that affect the report output. For example, you can create a True/False Condition (Boolean) calculated field to display workers with a regular or full-time status and exclude contract workers.

Additionally, you can use calculated fields to access data on the primary business object (PBO). For example, you can make data accessible from a related business object (RBO) on the PBO for use in report types, report functions, and in other calculated fields.

Use Calculated Fields in Business Processes

You can control condition steps in a business process. For example, you can create a Text Length calculated field to determine how many letters are in a new hire's name. You can then add a condition to the Hire business process generating a custom name tag if the new hire's name is more than 20 characters.

Use Calculated Fields in Integrations

Use calculated fields with connectors and in reports that collect data for document transformation. You can also use calculated fields in an integration using Workday Enterprise Integration Builder (EIB). For example, you can use a Format Text calculated field to format employee first names to uppercase. Doing so aligns with the needs of the external system in the integration.

Use Calculated Fields to Schedule Recurring Processes

You can use calculated fields to determine dynamic runtime parameters for recurring scheduled processes. For example, you can run a specific report for each day (from two months ago

through today) using an Increment or Decrement Date calculated field. This calculated field function establishes the date parameters.

Characteristics of Calculated Fields

There are three factors that characterize calculated fields.

- Workday calculates the value of the field based on existing data such as other Workday-delivered fields, other calculated fields, or available custom fields.
- Workday determines the value of a calculated field at runtime so that the system captures real-time data for the calculation. However, keep in mind that your tenant does not store calculated field values.
- Workday associates calculated fields with a business object. The business object determines what data to use in a calculated field and where to use the calculated field. Then, the calculated field becomes a new field on the business object.



Tip: Consider performance when you configure calculated fields. For example, if you build a report that uses a calculated field referencing many different calculated fields, that calculation will significantly impact the runtime for that report.

Based on Existing Data

Existing data within the Workday tenant is the basis for calculated fields. The system does not store a calculated field's value. Instead, the system pulls the value from existing fields at runtime. Changes to the underlying objects and fields you use in the calculation do not affect calculated fields. When values in the Workday tenant change, any derived calculations automatically include the new values.

Calculated at Runtime

All calculated fields resolve at runtime. Workday derives the value based on other Workday-delivered fields, calculated fields, and available custom fields. Calculated fields resolve at runtime because the values of the other fields that make up the calculation of the calculated field vary. When you run the report or execute the condition rule that uses the calculated field, Workday retrieves these field and object instance values.

Associated with Business Object

A business object determines which fields are available within the calculation of a calculated field function. Any calculated field you create for a business object appears and behaves just like any

other Workday-delivered field for that business object. The business object also controls where the calculated field appears in prompts and reports.

The existing fields for the Worker business object are the basis for a calculated field built on that business object. That calculated field becomes a new field on the Worker business object. Wherever you use fields on the Worker business object, use that calculated field.



System-Wide vs. Report-Specific Calculated Fields

You create system-wide calculated fields with the [Create Calculated Field](#) or [Maintain Calculated Fields](#) tasks. Workday automatically enables these system-wide calculated fields across the system.

You can also create calculated fields with limited scope. You create these report-specific calculated fields directly from within the report definitions.

System-wide Calculated Fields

The benefits of using system-wide calculated fields include:

- Availability throughout Workday, not just in reports.
- Availability in multiple reports.
- Possibility of reuse in other reports and processes avoiding duplicated calculated fields.

Report-Specific Calculated Fields

Report-specific calculated fields function in the same way as system-wide calculated fields, except report-specific calculated fields:

- Are only available in a single report definition.
- Do not appear in the list of system-wide calculated fields when you run the [Maintain Calculated Fields](#) report/task.

The benefits of report-specific calculated fields include:

- Reducing clutter in the tenant as you use these fields in only a single report or integration.
- Allowing report authors to define calculations without having to coordinate with others.

Creating Report-Specific Calculated Fields

Follow these steps to create report-specific calculated fields for a given custom report definition:

- From a field prompt in the report definition, select the [Create Calculated Field for Report](#) task to add a report-specific calculated field to your report column, prompt, filter, or sort.
- From a report's Related Actions, you can select:
 - Calculated Field for Report > Create.
 - Calculated Field for Report > Maintain.
- Use the [Maintain Calculated Fields for Report](#) task for a given report to access the add, edit, and delete options.

A warning message displays if you create a report-specific calculated field that you do not reference in the report. It will do so, even if you associate it with and define it for the report.

Maintain Report-Specific Calculated Fields

The [Maintain Calculated Field for Report](#) task works the same as the [Maintain Calculated Fields](#) report/task for system-wide calculated fields. However, results display by custom report. The output only shows report-specific calculated fields in that report. You can edit, delete, and add report-specific calculated fields from this task option.

Convert Report-Specific and System-Wide Calculated Fields

For a report-specific calculated field you want to use in a different report, do not create an additional calculated field. You can convert system-wide and report-specific calculated fields to suit your reporting needs.

To convert a report-specific calculated field to a system-wide calculated field (for reference in other custom reports), follow these steps:

1. Access the [Convert Calculated Field for Report](#) task.
2. Select the report that references the calculated field.
3. Select the business object associated with the calculated field.

4. Select the field.
5. Select **OK** and **Done**.

To convert a system-wide calculated field to a report-specific calculated field (no longer available in other custom reports), follow these steps:

1. Access the **Convert Calculated Field** task.
2. Select the report that references the calculated field.
3. Select the business object associated with the calculated field.
4. Select the field.
5. Select **OK** and **Done**.



Note: If you use a system-wide calculated field in anything other than a report (including business process conditions or calculated fields), it cannot convert to report-specific. You can convert it if you are using it in the report to which you want to tie it.

Create a Calculated Field

The basics for creating a calculated field involve running the Create Calculated Field task and configuring the Calculation and the Additional Info tabs accordingly. You can follow these steps:

- To create system-wide calculated fields, use the Create Calculated Field task.
- Run the Create Calculated Field task:
 - Enter the name for the calculated field.
 - Select a Workday-delivered business object.
 - Choose a function type.

Create Calculated Field

Field Name *	<input type="text"/>
Business Object *	<input type="text"/>
Function *	<input type="text"/>

OK **Cancel**

- In the Calculation tab, specify the calculated field's parameters. Parameters vary depending on the calculated field function selection.

Create Calculated Field - Arithmetic Calculation

Field Name	* CF Test																									
Business Object	Worker																									
<input checked="" type="button"/> Calculation <input type="button"/> Additional info																										
Arithmetic Calculation																										
Field Type	* <input type="text"/> Numeric																									
Rounding	<input type="text"/>																									
Currency Code Field	(empty)																									
Return Zero on Error	<input type="checkbox"/>																									
Arithmetic Expression 1 item <table border="1"> <tr> <td>Order</td> <td>{</td> <td>*Field</td> <td>}</td> <td>Operator</td> </tr> <tr> <td>+/-</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>*</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>/</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table>		Order	{	*Field	}	Operator	+/-					*					/									
Order	{	*Field	}	Operator																						
+/-																										
*																										
/																										

- In the Additional Information tab, specify information necessary to organize, document, locate, and reference the calculated field.

Create Calculated Field - Arithmetic Calculation

Field Name: * CF Test

Business Object: Worker

Calculation Additional Info

Description:

Category: * Uncategorized

Authorized Usage: * Default Areas

Intermediate Calculation:

Do Not Use:

> Advanced

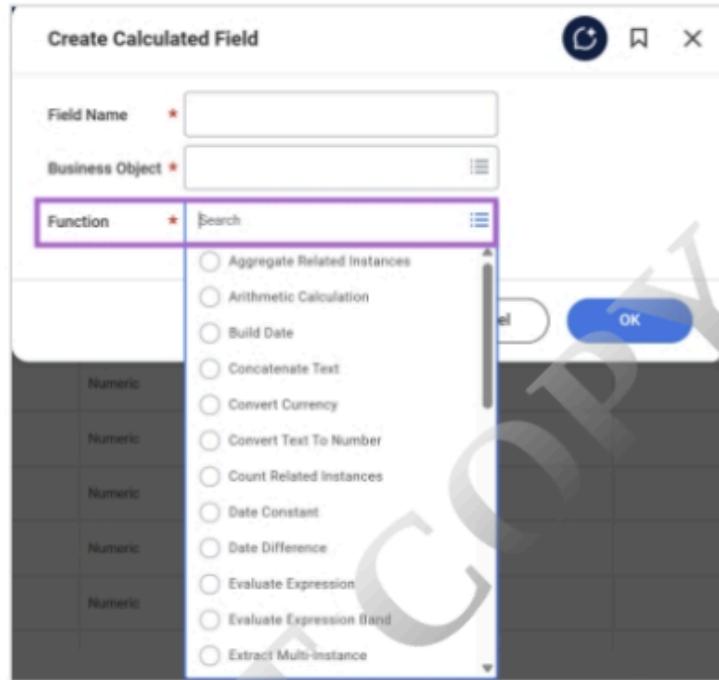
Initial Information

At the prompt, enter a field name, select the business object, and the function. The business object choice determines two major outcomes:

- What fields are available within the calculation? If you choose the Employee business object, all class report fields on the Employee business object are available for use in your calculated field.
- Where can you use the new calculated field? If you choose the Employee business object, your calculated field is available anywhere you use fields on Employee. The calculated field is a new field on the Employee business object and has the same access rules and characteristics as the other fields on Employee.

Calculation Tab

In the Calculation tab, you specify the parameters of the calculated field. The parameters you configure here vary depending on the calculated field function you select.



Additional Information Tab

Use the following fields, as necessary, in the Additional Information tab to provide specifics about your calculated field. These fields help organize, document, locate, and reference your calculated field.

Term	Definition
Description	Helps you and other users to track the purpose of each calculated field.
Category	Helps users locate calculated fields by category in field prompts. "Uncategorized" is the default. You can configure more categories as needed.
Authorized Usage	Allows customers to restrict the areas within Workday where the calculated field will display in prompts. Use "Default Areas" as it is the default. For example, add Benefits Eligibility, Calculation Engine (Payroll/Absence), and Compensation Eligibility in addition to "Default Areas" if needed.

Intermediate Calculation	Drives whether the calculated field shows in main field prompts or not. If checked as Intermediate, it does not show in main prompts and shows up in the "Other" category. If you do not select Intermediate, it shows in main field prompts under the "All" or designated Category. Intermediate calculated fields are searchable and usable. This checkbox only dictates where the field shows in field prompts.
Do Not Use	Appends the "Do Not Use" label to your calculated field name to indicate not to use this field anymore. Use fields shown in the main field prompt only (All or By Category) and do not use fields found under Other > Do Not Use Fields. Note: You cannot delete Calculated fields when they are in use. Mark as "Do Not Use" in development or if you plan to deprecate use.

Advanced Options

The system collapses the Advanced section in the Additional Info tab by default. You can expand this section to access available options.

Create Calculated Field - Arithmetic Calculation

Field Name: * CF Test

Business Object: Worker

Calculation Additional Info

Description:

Category: * Uncategorized

Authorized Usage: * Default Areas

Intermediate Calculation:

Do Not Use:

Advanced

Options:

Reference ID: * Worker - CF Test - 1638530571

WQL Alias: cf_CFTest

Resource	Definition
Options	<ul style="list-style-type: none">Select "Include as default View By field" in drill downs, if appropriate.Use this option with matrix reports. It applies to single instance, text, Boolean, date, and numeric calculated fields.Specifies to include the calculated field in the list of default fields that the user can group data by when drilling down.

Reference ID	<ul style="list-style-type: none"> • The system creates this identification field by default from the business object name and your calculated field name. • You can change the value. • Use this field when building integrations, for operations that need to access this field by its ID.
---------------------	---

Copy a Calculated Field

To copy an existing calculated field, use its Related Actions to select **Calculated Field > Copy**. When you copy an existing calculated field, the new field must use the same business object and function as the original field.

Naming Conventions

The best practice when creating calculated fields is to adhere to a set naming convention for your organization. You can include information in the name of the calculated field to easily identify details such as organization, calculated field function, report author name, and report name.



Example: Company-CF-RPT-Date Difference-MD-Phase1-Months since last pay increase

Alternatively, you can name your calculated fields like any other field in Workday. You can include the details in the Category and Description sections of the Additional Info tab.



Example: Months since last pay increase

Make sure your organization has a set naming convention and method of organizing your calculated fields and reports. Adhere to these conventions to avoid redundancy.

Access to Calculated Fields

Security Permission to Create Calculated Fields

To create, edit, or delete system-wide calculated fields, you must belong to a security group authorized for the Custom Field Management domain. To ensure control and consistency, and avoid duplicate field definitions, limit access to that domain to key individuals in your organization.

The following domains control who can access and create calculated fields:

- Custom Field Management (for system-wide calculated fields)
- Private Calculated Fields Management (for report-specific calculated fields)

Access to Values for a Calculated Field

Users with access to the underlying secured fields can access the values for calculated fields. Underlying secured fields are Workday-delivered report fields or custom fields, not calculated fields.

Domains secure Workday-delivered report fields and custom fields. Users with access to a domain can access any calculated field built using the underlying fields in that domain. Therefore, underlying secured fields determine security access to a calculated field definition.

You can view the security for a calculated field along with the underlying secured fields and security configurations. Using the calculated field's Related Actions, select Calculated Field > View Security Groups to access the underlying secured fields and configured security groups.

View Security Groups for Calculated Field

Worker Count ...

To access the calculated field above, the user must have access to all the fields referenced in the calculation. The table below lists all of the fields referenced and the corresponding security groups. For each Report Field listed, the user must be a member of one or more of the associated security groups. If no fields are listed, then all users can access the calculated field.

Fields Referenced in the Calculation and Related Security Groups 1 item	
Report Field	Security Groups
Current Workers Assigned to Project	Finance Auditor HR Partner HR Partner (By Location) Implementers Manager More (5)

To determine which domain or domain security policies to configure, use the calculated field's Related Actions. Select Security > View Security to access the underlying domain's or domains' security policies.

View Security for Calculated Field

Worker Count ...

Type	Calculated Field			
Permission Required	View			
1 item				
Report Field	Security Policy	Domain	Functional Areas	Permitted Security Groups
Current Workers Assigned to Project	Worker Data: Project Tracking	Project Tracking		Finance Auditor HR Partner HR Partner (By Location) Implementers Manager More (5)

Maintain Calculated Fields Report

You can use the [Maintain Calculated Fields](#) report as a control center for managing system-wide calculated fields. With this report you can:

- View all the system-wide calculated fields in the tenant.
- Create a new calculated field using the Add New button.
- Edit a calculated field using the Edit button or using a field's Related Actions to select Calculated Field > Edit.
- Delete a calculated field using the Delete button or using the field's Related Actions to select Calculated Field > Delete. You cannot delete calculated fields already in use until you remove all references within Workday for that calculated field. View the Where Used tab in the calculated field definition to determine calculated field instances in your tenant.
- Copy a field using its Related Actions to select Calculated Field > Copy.
- Check all the security requirements to access a field by using its Related Actions to select Calculated Field > View Security Groups. The security of the Workday-delivered (or custom) fields determines the list of security groups that can access that field.

You can also run the following tasks directly from the search box to edit, create, and delete calculated fields:

- [Create Calculated Field](#)
- [Edit Calculated Field](#)
- [Delete Calculated Field](#)

Use the *cf:* prefix in the search box to locate a calculated field by name.

Scenario: Date Functions



Logan McNeil needs a report showing the payout for approved expense reports by date. Also, she needs to know if the payouts meet the expected deadline of 30 days after approval.

The image below displays the fields she needs to render in the report:

WICT CF Expense Report Payment Dates					
Company Global Modern Services, Inc. (USA)					
Expense Report	Expense Report Date	Approval Date	Expected Paid Date	Actual Paid Date	# of Days Before Payment
Expense Report: EXP-00003301	01/25/2008	04/22/2008	05/22/2008	02/16/2009	300
Expense Report: EXP-00003302	01/25/2008	04/22/2008	05/22/2008	02/16/2009	300
Expense Report: EXP-00003303	01/24/2008	04/22/2008	05/22/2008	02/16/2009	300
Expense Report: EXP-00003304	01/31/2008	04/24/2008	05/24/2008	02/16/2009	298
Expense Report: EXP-00003305	02/20/2008	04/24/2008	05/24/2008	06/09/2010	776
Expense Report: EXP-00003306	02/22/2008	04/24/2008	05/24/2008	06/09/2010	776
Expense Report: EXP-00003307	02/29/2008	04/24/2008	05/24/2008	06/09/2010	776

Global Calculated Fields

Global calculated fields exist on the Global business object. The Global business object contains the fields that are global in nature and can be Workday-delivered fields and calculated fields you define. Fields associated with the Global business object are available for use with any business object and are visible to all users. They can represent constants such as 1, 23, single space, is true, and USD.

They can also represent variables that vary over time such as today, last day of this month, and current user.

To create additional global fields, select Global as the business object when creating a calculated field. Global fields display in their own Global field prompt category.



Example: A global calculated field that returns a date field value of one year before the variable value of Today (review the image below).

Calculated Field - Increment or Decrement Date

One Year Ago ... 914

Field Name One Year Ago
Business Object Global

[Calculation](#) [Additional Info](#) [Where Used](#)

Increment or Decrement Date

Date Field Today

Years to Add or Subtract -1
Months to Add or Subtract (empty)
Days to Add or Subtract (empty)
Hours to Add or Subtract (empty)
Minutes to Add or Subtract (empty)
Seconds to Add or Subtract (empty)
Milliseconds to Add or Subtract (empty)

Return Blank Date on Error Yes

Increment or Decrement Date

The Increment or Decrement Date function computes a date that's a specific number of days, months, or years before or after the date field. You can also add or subtract time to or from a date field that captures time data. The Increment or Decrement Date function calculates a date that is a specific number of:

- Years
- Months
- Days
- Hours
- Minutes
- Seconds
- Milliseconds

This function returns a Date field type. In the example below, we add one year to the Credit Verification Date field to derive a credit verification renewal date.

The screenshot shows the configuration for a calculated field named "Credit Verification Renewal Date" under the "Customer" business object. The "Calculation" tab is selected. The "Date Field" is set to "Credit Verification Date". The "Years to Add or Subtract" is set to 1. All other time units (Months, Days, Hours, Minutes, Seconds, Milliseconds) are set to (empty). The "Return Blank Date on Error" option is checked. A watermark "anand.tripathi@wgu.edu" is visible across the form.

Field Name	Credit Verification Renewal Date	
Business Object	Customer	
Calculation	Additional Info	Where Used
Increment or Decrement Date		
Date Field	Credit Verification Date	
Years to Add or Subtract	1	
Months to Add or Subtract	(empty)	
Days to Add or Subtract	(empty)	
Hours to Add or Subtract	(empty)	
Minutes to Add or Subtract	(empty)	
Seconds to Add or Subtract	(empty)	
Milliseconds to Add or Subtract	(empty)	
Return Blank Date on Error	Yes	

You commonly use the Increment or Decrement Date calculated field function with variable Global date fields such as Today and Report Effective Date. This function is also useful for change detection. It allows you to derive a date/time field; you can then compare the value of another field as of a time period. For example, today versus two years ago.

If the date field you are using is blank for an instance, a runtime error occurs. However, if you select the Return Blank Date on Error option, the suppressed error returns a blank date. Best practice is to make sure to check this box every time you create an Increment or Decrement Date field.

Other examples of incrementing and decrementing dates using calculated fields include:

- Calculate two months from now for use in a report filter to select the employee certifications that expire in the next two months.
- Dynamically calculate the "First day of the month, three months ago" for use as a parameter for scheduling a recurring report.

- Subtract one millisecond from the date/time that a business process event completed to access the value of a field before the business process change.
- Use a pair of calculated fields that return all customers without current credit verifications from the last year. Then, create a report that shows all the overdue customer credit verifications.

Date Difference

The Date Difference function computes the number of years, months, days, hours, minutes, or seconds between two dates. The calculation subtracts the end date from the start date. If the calculation requires a date constant, it must exist before you create the calculated field. The system rounds the results down to the nearest whole number depending on the value returned. This function returns a numeric field type.

Calculated Field - Date Difference

* Project Expected Duration (in months)

Field Name	Project Expected Duration (in months)
Business Object	Project
Calculation	<input checked="" type="radio"/> Date Difference <input type="radio"/> Project Start Date <input type="radio"/> Project End Date
Start Date Field	<input type="checkbox"/>
End Date Field	<input checked="" type="checkbox"/>
Value Returned	(empty) <input type="radio"/> In Years <input checked="" type="radio"/> In Months <input type="radio"/> In Days <input type="radio"/> In Hours <input type="radio"/> In Minutes <input type="radio"/> In Seconds
Return Zero on Error	<input checked="" type="checkbox"/> Yes
Additional Info <input type="checkbox"/> Ignore Lower Level Date Components	

By default, the year difference calculation (In Years) considers the month and day in computing the number of years. By default, the month difference calculation (In Months) considers the day.

Select the Ignore Lower-Level Date Components checkbox if you want the system not to consider the lower-level components in the date difference calculation. For example, the year difference between April 1, 2016 and Feb. 1, 2018 results in one year if you do not select the Ignore Lower-Level Date Components checkbox. If you do select the checkbox, the result is two years. The difference in months between May 15, 2017 and Sept. 14, 2017 results in three months if you do not select the Ignore Lower-Level Date Components checkbox. If you do select the checkbox, the result is four months.

	Apr. 1, 2016 – Feb. 1, 2018	May 15, 2017 – Sep. 14, 2017
Ignore Lower Level Date Components <input type="checkbox"/>	1 Year and 10 Months = Only 1 Full Year	3 Months and 30 Days = Only 3 Months
Lower-Level Date Components	10 Months	30 Days
Ignore Lower Level Date Components <input checked="" type="checkbox"/>	2016 to 2018 = 2 Full Years	May to September = 4 Months

Here are examples of calculating the difference between two dates:

- Days remaining before an employee returns from leave.
- Number of months that a position is unfilled.
- Number of backdated months in an expense report or forward-dated from the current month.

Activity 5.1 - Date Functions

Business Case: Logan McNeil needs a report showing the payout for approved expense reports by date. Also, she needs to know if the payouts meet the expected deadline of 30 days after approval.

Task #1: Create a Custom Report

1. Sign in as Logan McNeil (*/mcneil*).
2. Run the **Create Custom Report** task.
3. Enter the following information:

Field Name	Entry Value
Report Name	WICT Expense Report Payment Dates
Report Type	Advanced
Data Source	Expense Reports for Company
Optimized for Performance	(clear)



Important: Double check the right data source was selected. Do not choose Expense Report **Lines** for Company.

4. Select **OK**.
5. Verify that the data source filter is **Expense Reports Filter**.
6. In the Columns tab, add the **Expense Report**, **Expense Report Date**, and **Approval Date** fields to the report using the Expense Report business object.
7. Select **OK**.
8. Run the report using **Global Modern Services, Inc. (USA)** as the Company.

Task #2: Create an Increment or Decrement Date Field

- Run the **Create Calculated Field** task and enter the following information:

Field Name	Entry Value
Field Name	CF IDD Expected Paid Date
Business Object	Expense Report
Function	Increment or Decrement Date

- Select **OK**.
- Enter the following information:

Field Name	Entry Value
Date Field	Approval Date
Days to Add or Subtract	30
Return Blank Date on Error	(select)

- In the **Additional Info** tab, enter the following information:

Field Name	Entry Value
Description	Calculates the expected payment date for approved Expense Reports.
Category	Customer Defined Expense Report

- Select **OK**.

Task #3: Edit Custom Report

1. Navigate to the **Edit Custom Report** task and select the **WICT Expense Report Payment Dates** report.
2. Select **OK**.
3. On the Columns grid, add a row after the Approval Date field and add the newly created calculated field **CF IDD Expected Paid Date** to the report definition.
4. Add another row after the last row in the Columns grid and add the **Paid Date** field to the report definition.
5. Select **OK** to save the updated report definition.
6. Run the report for **Global Modern Services, Inc. (USA)**. Notice the expected date for each expense report, as well as the actual payment date.

Task #4: Create a Date Difference Field

Next, Logan needs to determine how many days it takes for each of the expense reports to pay out. Since she already knows the approval date and paid date, she can use a Date Difference calculated field to return how many days pass between those two dates.

1. Run the **Create Calculated Field** task, and enter the following information:

Field Name	Entry Value
Field Name	CF DD Days Before Payment
Business Object	Expense Report
Function	Date Difference

2. Select **OK** and enter the following information:

Field Name	Entry Value
Start Date Field	Approval Date
End Date Field	Paid Date

Value Returned	In Days
Return Zero on Error	(select)

3. In the Additional Info tab, enter the following information:

Field Name	Entry Value
Description	Returns the number of days it takes to pay out an approved expense report.
Category	Customer Defined Expense Report

4. Select **OK**.

Task #5: Edit Custom Report

1. Navigate to the **Edit Custom Report** task and select the **WICT Expense Report Payment Dates** report.
2. Select **OK**.
3. On the Columns grid, add a row after the Paid Date field and add the newly created calculated field **CF DD Days Before Payment** to the report definition.
4. In the Column Heading Override column, enter the following information:

Field	Column Heading Override
CF IDD Expected Paid Date	Expected Paid Date
Paid Date	Actual Paid Date
CF DD Days Before Payment	# of Days Before Payment

5. Select **OK** and run the report for **Global Modern Services, Inc. (USA)**.

CHALLENGE QUESTION:

How can you build a calculated field to determine how many days overdue an expense report payment was from the expected paid date?

Activity Complete

DO NOT COPY
anand.tripathi@wgu.edu

Activity Answer - Date Functions

Task #5: Edit Custom Report

CHALLENGE QUESTION:

How can you build a calculated field to determine how many days overdue an expense report payment was from the expected paid date? Create a Date Difference field using the Expense Report business object, and use the following information:

Field Name	Entry Value
Start Date Field	CF IDD Expected Paid Date
End Date Field	Paid Date
Value Returned	In Days
Return Zero on Error	(select)

Substring Text

The Substring Text function extracts the specified portion of text within a field. You can use it to extract a substring from a text or single-instance field. You can base substrings on fixed positions or delimiters allowing you to search a string forward or backward.

For example, you use a report for an integration to another system. You need to extract only the Cost Center Name from a field that contains both Cost Center Number and Name.

In the example, the Cost Center Number is always five characters long followed by the Cost Center Name: *12345CostCenterName*. Use a fixed position substring type to extract just the name from the field. Start at position six in the field and go forward, left to right, to the end of the field:

Example:

12345CostCenterName → CostCenterName



You can also use a delimiter substring type to convert text that contains a special character, such as a dash, forward slash, or even a blank space.

The following are examples of the different delimiter substring types:

Before Delimiter Examples (Forward):

Before Delimiter	Original String	Returned Substring
-	ABC-XYZ	ABC
BC	ABC-XYZ	A
/	07/04	07

After Delimiter Examples (Forward):

After Delimiter	Original String	Returned Substring
BC	ABC-XYZ	-XYZ

W	ABC-XYZ	
/	07/04	04

After Delimiter Examples (Backward):

After Delimiter	Original String	Returned Substring
-	802-555-1212	1212
-- (blank space)	Senior Clerk (Pleasanton)	(Pleasanton)
-- (blank space)	Sales and Marketing 7283	7283

Between Two Delimiters Examples:

Start Delimiter	End Delimiter	Original String	Returned Substring
[]	ABC-[XYZ]	XYZ
()	2010 Actuals (Global Modern Services (USA))	Global Modern Services (USA)

Convert Text to Number

The Convert Text to Number function enables you to extract a number that currently exists as a text field and convert it to a numeric data type. Once you convert a field to a numeric data type, you can use it in arithmetic operations and operations that require a numeric source field. For example, you can convert the Date of Birth - Year text field to a numeric field. You can then use that numeric field to create a Lookup Range Band calculated field that returns the worker's generation (e.g. Millennial, Generation X, Boomer) using the worker's birth year.

The Convert Text to Number function includes an option to return zero if the result is not valid, instead of displaying a runtime error message.

Count Related Instances

The Count Related Instances function dynamically counts related instances on a related business object. The instances can be all for a certain field or only instances that meet a condition. If adding a condition, define the condition on the related business object. If a condition is not pre-existing in the tenant, you can use the True/False Condition calculated field to create the condition. This function counts only instances you have security access to view.

Calculated Field - Count Related Instances Employee Involuntary Terminations Count ...

Field Name	Employee Involuntary Terminations Count
Business Object	Organization
Calculation	Additional Info Where Used
Count Related Instances	
Related Field	<input checked="" type="checkbox"/> Termination Events
Related Business Object	Termination Event
Count	(empty) <input type="radio"/> All instances <input checked="" type="radio"/> Instances where condition is true <input type="checkbox"/> Is Involuntary Employee Termination

The following diagram outlines how the Count Related Instances function works:



Examples of Count Related Instances include counting the number of:

- Sick days occurring on Monday or Friday during the last year for each employee.
- Employee hires during the last year by location.
- Open positions in an organization.
- Expense lines per employee.
- Training-related expense reports.
- Completed amortization line installments as of a prompted date.
- Approved expense reports for any given worker.

Sum Related Instances

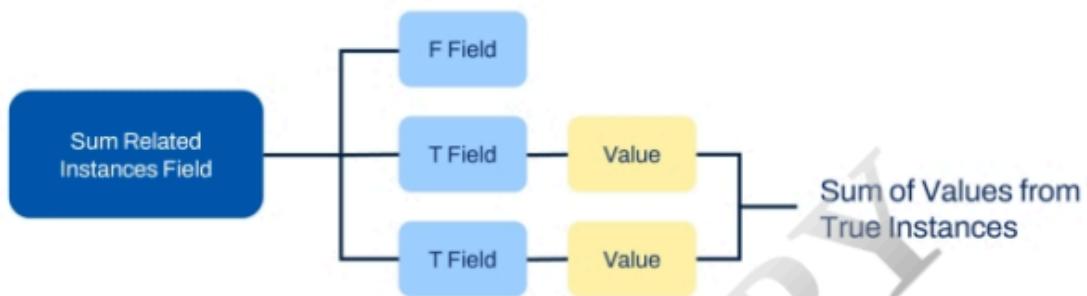
The Sum Related Instances function dynamically sums related instances on a related business object. This function works with numeric and currency fields. All instances must use the same currency code for the system to generate the sum of currency fields.

The summed instances can include all instances of a certain field or of fields that meet a condition. If adding a condition, you must define the condition for the related business object. If a condition does not pre-exist in the tenant, you can use the True/False calculated field to create the condition. This function counts only instances you have security access to view.

Calculated Field - Sum Related Instances Training Expense Total in USD for Expense Report ...

Field Name	Training Expense Total in USD for Expense Report	
Business Object	Expense Report	
Calculation	Additional Info	Where Used
Sum Related Instances		
Related Field	<input checked="" type="checkbox"/> Expense Report Lines	
Related Business Object	Expense Report Line	
Sum	(empty)	
<input type="radio"/> All instances <input checked="" type="radio"/> Instances where condition is true <input checked="" type="checkbox"/> Is Training Expense		
Field to Sum	<input checked="" type="checkbox"/> Expense Line Extended Amount in USD	

The diagram below outlines how the Sum Related Instances function works:



Examples of Sum Related Instances include the sum of:

- Performance review ratings for each organization for use in a calculation to find the average.
- Expense line items for meals in an expense report to compare against an approval limit.
- Taxable invoice lines to calculate the VAT amount for each supplier invoice.

Arithmetic Calculation

The Arithmetic Calculation function uses numeric and currency field types. This function performs simple arithmetic.

- Addition (+)
- Subtraction (-)
- Multiplication (*)
- Division (/)

If the arithmetic calculation requires numeric constants, they must already exist in the system. You can use parentheses, when necessary, to specify the evaluation order of the calculations. If you use a currency field, all values must have the same currency type.

If the field tries to divide by zero, a runtime error returns. If you select the Return Zero on Error option and division by zero occurs, the error suppresses, and a zero amount returns.

Calculated Field - Arithmetic Calculation

Vacation Days Remaining

Field Name Vacation Days Remaining

Business Object Worker

Calculation Additional Info Where Used

Arithmetic Calculation

Field Type Numeric

Rounding (empty)

Currency Code Field (empty)

Return Zero on Error Yes

Arithmetic Expression 2 items

(Field)	Operator
	<input checked="" type="checkbox"/> Vacation Entitlement Days from Accrual		- (Subtract)
	<input checked="" type="checkbox"/> Vacation Days Taken		



Note: Consider using parentheses to make your calculation run accurately. Arithmetic Calculation fields use the standard order of operations in its calculations: Parentheses, Exponents, Multiplication, Division, Addition, Subtraction (PEMDAS).

Examples of using an Arithmetic Calculation include:

- Calculating an employee's hourly, daily, weekly, or monthly rate.
- Calculating 20% of an employee's salary to compare against the employee's bonus amount.
- Calculating an employee's salary difference from this year compared to last year.
- Determining revenue per headcount in a given year/quarter/month.
- Finding the average cost of meal per attendee on an expense report.
- Calculating total projected salary through the end of the year/quarter/month.

- Identifying revenue recognition for projects by percentage complete.
- Calculating the average daily ending balance of a journal.
- Calculating the number of days in a fiscal period plus one.



Tip: For Teresa's scenario, use the Arithmetic Calculation function. Take the Sum of the meal lines and divide it by the Count of the meal lines, returning an average amount.

Convert Currency

To perform an arithmetic calculation on currency fields, the fields included in the calculation must use the same currency. Workday delivers many fields that return amounts in a specific currency, like USD, or in the base currency for a company. For currency fields without a Workday-delivered conversion, you can use the Convert Currency calculated field function to determine the equivalent value of field in a different currency.

All monetary fields have an associated currency code. You can also convert the source currency to the user's preferred currency. Enter the global variable Users Preferred Currency in the Target Currency Code Field prompt. The value can vary from user to user, and each user can change their preferred currency using the Change Preferences task.

To convert currencies, use the Maintain Currency Conversion Rates task to set up conversion rates for all valid Source Currency/Target Currency/Currency Rate Type combinations. Complete this setup before using this calculation function. If no conversion rate exists at runtime, the function returns an error.

Calculated Field - Convert Currency  Compensation Midpoint in USD 

Field Name	Compensation Midpoint in USD
Business Object	Worker
Calculation	
Convert Currency	
Source Currency Amount Field	<input checked="" type="checkbox"/> Compensation Range - 50th Percentile
Target Currency Code Field	<input checked="" type="checkbox"/> USD
As Of Date Field	<input checked="" type="checkbox"/> Report Effective Date
Currency Rate Type	Current
Return Zero on Error	

Converts source field to new target currency

When the report or condition that uses this calculated field runs, the system evaluates the Currency Rate Type and As Of Date. The appropriate exchange rate calculates and returns. If the As of Date Field has a blank value at runtime, the resulting value is zero and a runtime error returns. If there is no data for the combination of Source Currency Amount Field, Target Currency Code Field, and Currency Rate Type, the resulting field is zero. A runtime error returns.

Examples of converting currencies include:

- Displaying an employee's annual salary in euros, regardless of its current currency type.
- Determining all employee bonuses in U.S. dollars, using the exchange rate from the end of last year.
- Converting an expense report total to a common currency (such as USD) for business process approval rules based on an amount.
- Converting Canadian employee benefit contributions into USD for consolidated reporting with U.S. benefit data.

Scenario



Logan McNeil needs to modify the WICT CF Employee Details Report by categorizing employee Total Base Pay into a Low, Medium, or High value. The word "High" is used if the employee makes over \$100,000. "Low" is used if the employee makes less than \$50,000, and "Medium" is used for anything in between those two categories.

This image displays the fields she needs in the report:

WICT CF Employee Comp Levels ...					
506 items					
Employee	Hire Date	Supervisory Organization	Management Level	Total Base Pay - Amount	Compensation Level
Adam Carlton	06/14/2010	Payroll Department	8 Individual Contributor	\$66,116.80	Medium
Adesh Shah	08/06/2014	Consulting Services - UK Group	8 Individual Contributor	\$40,065.74	Low
Adrian Martin	02/01/2012	Marketing Communications Group	6 Manager	\$131,870.00	High
Aesha Pillay	01/23/2018	Field Sales - EMEA Group	8 Individual Contributor	\$21,633.81	Low
Aidan Mitzner	01/01/2000	Finance & Administration	8 Individual Contributor	\$107,022.80	High
Ainsley Carey	01/01/2016	Field Sales - Emerging Markets Group	8 Individual Contributor	\$13,506.83	Medium
Ajey Mokashi	01/04/2010	Global Support - Greater India Group	8 Individual Contributor	\$8,514.80	Low
Alain DuBois	01/01/2000	Global Support - North America Group	6 Manager	\$82,482.82	Medium
Alanna Haywood	01/01/2022	Consulting Services - Australia Group	8 Individual Contributor	\$57,152.42	Medium
Alberto Bassani	01/01/2000	Global Support - Southern Europe Group	8 Individual Contributor	\$42,325.07	Low

To construct this report, use these three calculated field functions:

- Text Constants as values to display in the report.
- True/False Conditions that assess which category an employee falls into.
- Evaluate Expression that connects the True/False Conditions to the Text Constant values.

Constant Fields

Workday reporting uses three types of constant fields on the Global business object:

- Numeric Constant
- Text Constant
- Date Constant

Each constant represents a static value that you can use in reporting and calculations. These fields are useful in Evaluate Expression calculated fields. You can access and use them in the same way as a Global field, including in reports and calculated fields.

Numeric Constant

Use Numeric Constant fields when you need a constant numerical value for reference, such as in a mathematical equation. These numeric values are Workday-delivered fields but you can create additional numeric constants as required.

Text Constant

Use the Text Constant calculated field value when you need a constant text value for reference, such as in capturing measurements.

For example, in this scenario you generate a report that evaluates and categorizes employee salaries as low, medium, or high. Workday does not deliver these category markers. But, you can build each of these text strings as Text Constants. Then, you can use these text constants with other report data to assign these terms meaning.

Date Constant

Use the Date Constant calculated field when you need a constant date value for reference. You can also use when you need a specific date to compare other dates against. For example, you can create a field that compares how many days an employee works before a company-wide, January 1, 2021, review date. In this example, that date constant value needs to exist in the system.

True/False Condition

The True/False Condition calculated field function determines if a condition is true or not. The return is a Boolean type field that you can use in custom reporting, condition rules, and as a condition for other calculated fields. You can use parentheses, when necessary, to group the conditions for evaluation.

Calculated Field - True/False Condition

Flight Risk

Field Name Flight Risk

Business Object Worker

Calculation

Additional Info Where Used

True/False Condition

7 Items

And/Or	Field	Operator	Comparison Type	Comparison Value	
And	<input checked="" type="checkbox"/> Employee	is not empty	Value specified in this filter	Yes	x        
And	(<input type="checkbox"/> 45 Years or Older	equal to	Value specified in this filter	Yes	
And	<input type="checkbox"/> Has Dependents	not equal to	Value specified in this filter	Yes	
And	<input type="checkbox"/> Marital Status = Married	not equal to	Value specified in this filter	Yes	
Or	(<input type="checkbox"/> Is High Retention Risk	equal to	Value specified in this filter	Yes	
Or	(<input type="checkbox"/> More than 50K Stock Options	equal to	Value specified in this filter	Yes	
And	<input type="checkbox"/> Percent of Stock Options Vested	greater than or equal to	Value specified in this filter	0.01	



Important: When using True/False fields, put conditions that exclude the greatest number of instances first for optimal report performance.

Examples of True/False Condition calculated fields include:

- A field that indicates if an employee is highly compensated.
- A field that confirms if an employee is enrolled currently in a Health Savings Account (HSA).
- A field that verifies if an employee has one year of service or less.
- Grouping expenditures and revenue into various categories (with Evaluate Expression).
- Grouping expense reports by approval status and date (with Evaluate Expression).



Reminder: Where possible, remember to use filters or subfilters in reports instead of calculated fields for optimal report performance.

Evaluate Expression

The Evaluate Expression calculated field function groups and transforms data. This function evaluates a series of conditions and returns the value associated with the first condition that is true. At runtime, the function applies the default value and then starts testing conditions starting with the first condition defined. If a condition is true, Workday sets the corresponding return value and does not test any subsequent conditions. Otherwise, Workday tests the second condition, and if true, sets the corresponding return value, and so on.

The screenshot shows the 'Calculated Field - Evaluate Expression' configuration screen. The 'Field Name' is set to 'CF EE Comp Level_Act 5.2'. The 'Business Object' is 'Worker'. The 'Calculation' tab is selected, showing the 'Evaluate Expression' section. The 'Default Value' is set to a placeholder value. Below it, there are three conditions listed under 'Condition': 'CF TF Low Comp_Act 5.2', 'CF TF Medium Comp_Act 5.2', and 'CF TF High Comp_Act 5.2'. To the right, a list of 'Return Value If Condition is True' is shown, corresponding to each condition: 'CF TC Low_Act 5.2', 'CF TC Medium_Act 5.2', and 'CF TC High_Act 5.2'. The entire screenshot is overlaid with a large diagonal watermark reading 'anand.tripathi@wgu.edu'.

The evaluated conditions must be a Boolean field that returns a true or false value at runtime. If you use Evaluate Expression, you can capture information that evaluates existing data across different fields and returns values in one calculated field. Also, with an Evaluate Expression function, don't forget to place the most likely conditions first to optimize report performance.

Examples of grouping data include:

- Evaluate employees and group by employment status such as active, leave, terminated, or contractor.
- Evaluate employees and return a management status such as management or staff.
- Evaluate a list of expenditures and group them into categories.
- Evaluate journal data to categorize spending such as employee burden, contingent labor, travel and expense, or goods and services.

Calculation Hierarchy

View Calculation Hierarchy

The View Calculation Hierarchy feature allows you to explore all underlying fields in a calculated field. Access the View Calculated Fields Hierarchies action from the Related Actions of a report definition.

The screenshot shows the 'View Custom Report' interface for a report named 'WICT Employee Comp Levels_Alt 5.2'. The report details are as follows:

Report Name	WICT Employee Comp Levels_Alt 5.2
Report Type	Advanced
Data Source	P Workers for HCM Reporting
Data Source Filter	All Employees
Data Source Type	Indexed
Primary Business Object	Worker

Below the report details, there is an 'Additional Info' section with tabs for 'Columns', 'Sort', 'Filter', 'Prompts', 'Output', 'Share', and 'Actions'. The 'Actions' tab is selected, showing a list of related actions. The 'Custom Report' section of the context menu is expanded, and the 'View Calculated Field Hierarchies' option is highlighted with a purple box.

Calculated Field Usage Hierarchy

Calculated Field Usage Hierarchy is a Workday-delivered report that assesses the impacts of changes to a calculated field you use in reports.

This report provides you useful information such as:

- The number of instances that use a particular calculated field across your tenant.
- The nested level in which a calculated field exists.
- The usage details for calculated fields, such as the areas referenced.

Calculated Field Usage Hierarchy

The screenshot shows a table titled "Calculated Field Usages" with the following columns: Calculated Field, Level, Usage, and Area. There are 11 items listed, all corresponding to the "Worker Status" calculated field at Level 1. The "Usage" column lists various reports and analyses, and the "Area" column indicates they are all "Custom Report".

Calculated Field	Level	Usage	Area
Worker Status	1	CBASE All Workers	Custom Report
Worker Status	1	BLP All Workers	Custom Report
Worker Status	1	Hiring Success Analysis	Custom Report
Worker Status	1	Benefits – Married Employees - EE Coverage	Custom Report
Worker Status	1	Worker and Cost Center Details Lookup Expanded	Custom Report
Worker Status	1	All Contingent Workers	Custom Report
Worker Status	1	Low Performer Analysis	Custom Report
Worker Status	1	Benefits – Single Employees - EE Coverage	Custom Report
Worker Status	1	Find Basic Worker Data	Custom Report
Worker Status	1	AI GMS - Terminated Workers	Custom Report
Worker Status	1	EWF All Workers	Custom Report

Planning Your Calculated Fields

In the current scenario, you need to categorize employee compensation into levels of low, medium, and high. The following graphic represents the design process for this scenario.



- Create the Text Constant values to return in the report output.
- Create True/False Conditions to group compensation amounts into individual categories.
- Create an Evaluate Expression calculated field to evaluate an employee's compensation against the True/False Condition categories.
- Return the Text Constants.



Tip: In Workday, you have the flexibility to create calculated fields within calculated fields using an available prompt as you build. You have a choice to build the individual components as you go or build the calculated field from the bottom up.

Activity 5.2 - True/False Condition and Evaluate Expression

Business Case: Logan McNeil needs to modify her Employee Details Report categorizing Total Base Pay into a low (\$50k or less), high (\$100k or more), or medium (anything in between) value. Use the word "High" if the employee makes \$100,000 or more. She needs to:

1. Create a text constant calculated field.
2. Create a true/false condition calculated field.
3. Create an evaluate expression calculated field.
4. Add the new calculated field into report.

Task #1: Copy Custom Report

1. Sign in as Logan McNeil (*/mcneil*).
2. Search for and run the **Employee Details Report**.
3. Use the report's Related Actions to select **Custom Report > Copy**.
4. Name the new report *WICT Employee Comp Levels* and verify the **Optimized for Performance** checkbox remains clear.
5. Select **OK**.
6. Confirm the data source filter is **All Employees**.
7. Select **OK and Run**.

Task #2: Create Text Constants

First, Logan needs to create text constant calculated fields indicating whether the pay is low, medium, or high.

1. Run the **Create Calculated Field** task.
2. Enter the following information:

Field Name	Entry Value
Field Name	CF TC Low
Business Object	Global

Function	Text Constant
----------	---------------

3. Select **OK**.
4. In the Text Constant field, enter *Low*.
5. Optionally, fill out the Additional Info tab.
6. Select **OK**.
7. Use the new field's **Related Actions** to select **Calculated Field > Copy**. Copying this field will retain the same business object and function on the new field. This action can be a valuable time-saving shortcut when creating multiple calculated fields using the same function and business object.
8. Enter the following information:

Field Name	Entry Value
Field Name	CF TC Medium
Text Constant	Medium

9. Optionally, fill out the Additional Info tab.
10. Select **OK**.
11. **Copy** this new field.
12. Enter the following information:

Field Name	Entry Value
Field Name	CF TC High
Text Constant	High

13. Optionally, fill out the Additional Info tab.
14. Select **OK**.

Task #3: Create True/False Conditions

Now, Logan needs to create True/False calculated fields for the Compensation Ranges needed.

1. Select the **Create Another Calculation** button or run the **Create Calculated Field** task.
2. Enter the following information:

Field Name	Entry Value
Field Name	CF TF Low Comp
Business Object	Worker
Function	True/False Condition

3. Select **OK**.
4. Enter the following information:

Field Name	Entry Value
Field	Total Base Pay Annualized in Reporting Currency - Amount
Operator	less than or equal to
Comparison Type	Value specified in this filter (<i>default</i>)
Comparison Value	50000

5. Optionally, fill out the Additional Info tab.
6. Select **OK**.
7. Use the new field's **Related Actions** to copy it.
8. In the Field Name field, enter *CF TF Medium Comp*.
9. On the calculation tab, enter the following information:

Field Name	Entry Value
Field	Total Base Pay Annualized in Reporting Currency - Amount (<i>defaults from copied field</i>)
Operator	greater than
Comparison Type	Value specified in this filter (<i>default</i>)
Comparison Value	50000 (<i>defaults from copied field</i>)

10. Add a new row in the Calculation tab.

11. Enter the following information:

Field Name	Entry Value
Field	Total Base Pay Annualized in Reporting Currency - Amount
Operator	less than
Comparison Type	Value specified in this filter (<i>default</i>)
Comparison Value	100000

12. Optionally, fill out the Additional Info tab.

13. Select OK.

14. Use the new field's **Related Actions** to copy it.

15. In the Field Name field, enter *CF TF High Comp*.

16. Delete the first row on the grid on the Calculation tab.

17. Edit the remaining row using the following information:

Field Name	Entry Value
Field	Total Base Pay Annualized in Reporting

	Currency - Amount
Operator	greater than or equal to
Comparison Type	Value specified in this filter (<i>default</i>)
Comparison Value	100000

18. Optionally, fill out the Additional Info tab.

19. Select **OK**.

Task #4: Create an Evaluate Expression

Now, Logan needs to create an Evaluate Expression calculated field to associate the True/False Conditions with her new text constants.

1. Select the **Create Another Calculation** button or run the **Create Calculated Field** task.
2. Enter the following information:

Field Name	Entry Value
Field Name	CF EE Comp Level
Business Object	Worker
Function	Evaluate Expression

3. Select **OK**.

4. Enter the following information:

Field Name	Entry Value
Field Type	Text
Default Value	Global Fields > Text > -

5. Use the **Add Row** icon to add two additional rows in the calculation section for a total of

three rows.

6. Enter the following information:

Condition	Return Value If Condition is True
CF TF Low Comp	CF TC Low
CF TF Medium Comp	CF TC Medium
CF TF High Comp	CF TC High

7. Optionally, fill out the Additional Info tab.

8. Select **OK**.



Note: Consider performance when configuring this type of calculated field. Put the condition that impacts the largest population first in the list to make your calculation run quickly and efficiently.

Task #5: Edit Report and Add Evaluate Expression

1. Run the **Edit Custom Report** task.
2. Select the **WICT Employee Comp Levels** report.
3. Select **OK**.
4. **Add a row to the bottom of the Columns grid.**
5. Enter the following information:

Field Name	Entry Value
Field	CF EE Comp Level
Column Heading Override	Compensation Level

6. Select **OK** and **Run** to view the report output. Now, each employee has a Compensation Level in accordance with their Total Base Pay Amount.
7. Try filtering the results by Compensation Level.
 - A. Select the **Compensation Level** column header and filter employees in the High compensation level.
 - B. Select the **Compensation Level** column header and filter employees in the Medium compensation level.

Task #6: Optional – View a Financials Example

1. From the search box, search for *cf: talent source*.
2. Select the **Talent Source** Evaluate Expression field definition.



Note: This calculated field exists on the Journal Line business object. It checks a series of conditions that, if true, will return a configured talent source text constant.

3. Review each condition definition.
4. From the **Where Used** tab, notice where the system uses the Talent Source calculated field.
5. Select the **Talent Acquisition Cost by Source** custom report.



Note: This is a matrix report that groups journal lines by the Talent Source calculated field, showing acquisition costs by talent source .

6. Review the report definition and **run** the report to examine the result groupings.

Activity Complete

Lookup Range Band

The Lookup Range Band calculated field function accesses a specific value and determines where it falls in relation to a set of defined ranges. This function is like the Evaluate Expression function, except it only evaluates numeric or currency fields. This function returns a single-instance field type that corresponds to the range band (the range of values) in which the source field falls. The calculated field returns a blank instance if the value of the field does not meet a defined band.

The screenshot shows the configuration of a calculated field named "Days Opened Category" for the "Job Requisition" business object. The "Calculation" tab is selected. Under "Source Field", "Days Opened" is chosen. The "Rounding" option is set to "Round down". The "Where Used" tab is visible at the top right. Below the configuration area is a table titled "8 items" showing the range bands:

From Value	To Value	Return Value
0	30	0 - 30 Days
31	60	31 - 60 Days
61	90	61 - 90 Days
91	120	91 - 120 Days
121	150	121 - 150 Days
151	180	151 - 180 Days
181	365	181 - 365 Days
366	1,000,000	> 1 Year

Example uses for this field type include:

- Salary increase percentage
- Length of service
- Days overdue
- Aging of customer and supplier invoices

Evaluate Expression Band

You can use the Evaluate Expression Band calculated field to specify values for True/ False (Boolean) conditions and return instances.

Evaluate Expression Band works like the Evaluate Expression calculated field in that it evaluates a series of True/False conditions. However, Evaluate Expression Band always returns a single instance type field whose values are defined when creating the field.

In the example below, the Evaluate Expression Band calculated field determines who has initiated a business process. The system will identify the True conditions - "Employee" or "Manager." If any instance does not meet any of the True conditions, those instances will default to a specified value - "NOT initiated by Manager or Employee Self-Service."

Calculated Field - Evaluate Expression Band Calculated Field

CF EEB Comp Level_Act 5.2 101

Report Name: WICT Employee Comp Levels_Act 5.2
 Field Name: CF EEB Comp Level_Act 5.2
 Business Object: Worker

Calculation Additional Info Where Used

Evaluate Expression Band

Condition	Return Value If Condition Is True
CF TF Low Comp_Act 5.2	Low
CF TF Medium Comp_Act 5.2	Medium
CF TF High Comp_Act 5.2	High

Default Value Check Conditions Default value for "False" conditions

"True" conditions

Evaluate Expression vs. Lookup Range Band vs. Evaluate Expression Band

You may notice that Evaluate Expression, Lookup Range Band, and Evaluate Expression Band functions return similar report outputs. While these two calculated fields perform similar functions, there are some key differences.

Calculated Field	Functions
Evaluate Expression	<ul style="list-style-type: none"> Evaluates a series of conditions and returns value associated with the first true condition. Returns any field type. Configure default value for when no conditions are met.

Lookup Range Band	<ul style="list-style-type: none"> Evaluates a numeric or currency field against defined ranges and returns a range band. Enter "From/To" ranges manually when creating field. Returns a single-instance field. If a value does not match any range, the system returns a blank instance.
Evaluate Expression Band	<ul style="list-style-type: none"> Evaluates a series of conditions and returns text associated with the first true condition. Returns a single instance field. Type in a default value for when no conditions are met.

Scenario



Logan wants to start a college recruiting program. She needs a report that lists the top 10 schools in the United States from which active GMS employees have earned their degrees. She wants to be able to drill down by employee name, hire date, management level, supervisory organization – primary position, school attended, and degree obtained.

The following image displays the fields she needs on the report:

WICT CF Top 10 Schools in the USA							
WICT CF Most Recent School Attended	B.S.	B.A.	MBA	M.S.	A.A.	M.A.	Count
Northwestern University	2	2	5	0	0	0	9
Massachusetts Institute of Technology	3	0	0	0	0	0	3
University of California-Berkeley	1	1	0	1	0	0	3
Baylor University	2	0	0	0	0	0	2
Cornell University	1	1	0	0	0	0	2
Fordham University	1	0	0	1	0	0	2
Georgetown University	2	0	0	0	0	0	2
San Francisco State University	1	0	0	0	1	0	2
Stanford University	0	0	2	0	0	0	2
University of California-Los Angeles	1	1	0	0	0	0	2
Other	20	13	3	1	0	1	36
Total	34	18	10	3	1	1	67

To build this report, use two different calculated field functions:

- The **Lookup Related Value** function pulls a value from a field that exists on a related business object. For this scenario, you pull most recent school attended and degree received as an attribute into a report using the Education business object.
- The **Extract Single Instance** function isolates a single instance to extract a specific value from a group of related instances. For this scenario, you isolate the instance where the employee attended a school in the United States and received a degree.



Reminder: Matrix reports cannot use fields from related business objects. You can use a calculated field function that uses related business objects.

Lookup Related Value

The **Lookup Related Value** function retrieves a value from a field on a related business object. This function promotes that value from the related business object to the primary business

object. Once the value is available on the primary business object, you can use it for a calculation, condition rule, or reporting.

Here are some examples of looking up a value on a related business object:

- Promote the value of Worker's Manager from the Worker business object to the Competency business object for a matrix report using Competency as its primary business object.
- Make a field from Worker available to a compensation event condition rule.
- Promote a field from a related business object to the primary business object to make it available for grouping and totaling on an advanced report.
- Promote a field from a related business object to a primary business object to use it in combination with a field on the primary business object to create a calculated field.

To use the Lookup Related Value calculated field function, there must be a 1:1 relationship between the primary business object and related business object.

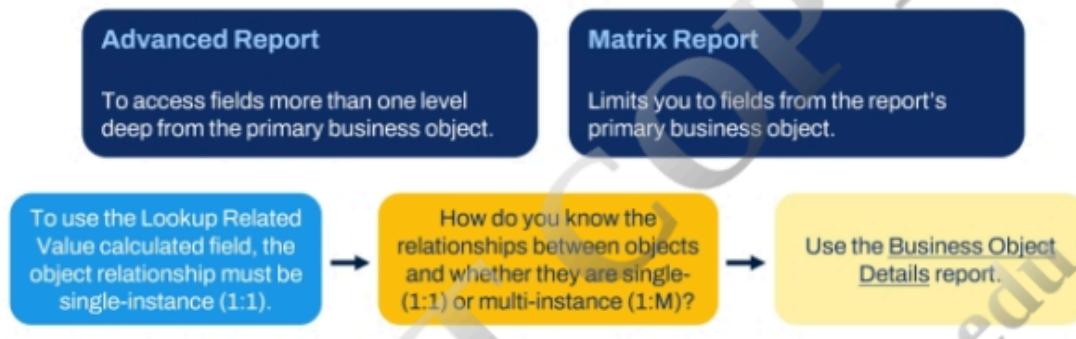
In the example below, we use the Lookup Related Value function to create a new field on the Competency business object. The new field retrieves the Worker's Manager from the Worker business object and relates it back to the Competency business object. We can now use the value in Worker's Manager in a report that uses the Competency business object.



Using the Business Object Details Report

The two business objects in a Lookup Related Value calculated field must have a 1:1 relationship. You can use the Business Object Details report to determine whether the relationship between your two objects is 1:1 (single instance) or 1:M (multi-instance).

When designing needed calculated fields, you can use the Business Object Details report to research the relationships between business objects. The image below demonstrates how valuable this report can be when planning and configuring the Lookup Related Value calculated field function in advanced and matrix reports.



Extract Single Instance

The Extract Single Instance function returns a single instance from a group of related instances on a related business object. The single instance is one of many from a multi-instance field that has a 1:M relationship with the primary business object. Use the Extract Single Instance function to retrieve the last, first, or "nth" instance from a set of instances that meet a specific condition. Depending on the condition, the field will return one or zero values. The sort field and direction enable you to select the occurrence that meets your needs.

Examples of Extract Single Instance uses include:

- First Master's degree an employee receives.
- Last complete performance review for an employee in 2022.
- Second time-off request of 2023.
- Last Payroll Result completed.
- Oldest unapproved expense report for a worker.

Use the Extract Single Instance function when using the Lookup Related Value. Do so on a business object that has a one-to-many (1:M) relationship with the calculated field's business

object. The function creates a one-to-one (1:1) relationship. Then, use the Lookup Related Value function to extract the value.

How Extract Single Instance Determines Return Value

The Extract Single Instance calculated field function determines return value as follows:

- Advanced Report - Accesses related business object instances associated with the multi-instance field you specify.
- Applies Conditions - Applies a condition to the instances by the source field. You can configure the conditions from the [Create Calculated Field](#) task when creating the Extract Single Instance calculated field.
- Sorts the Data - Sorts any related business object instances that satisfy the condition in ascending or descending order.
- Select Single Instance - Identifies and extracts the first, last, or "-nth" occurrence of the instances sorted.

For example, the following represents a list of expense reports for one worker.

Expense Report	Expense Report Date	Created On	Total Amount	Expense Report Status
Expense Report: EXP-00009433	10/17/2022	11/24/2022	7,726.37	Approved
Expense Report: EXP-00008798	10/15/2021	12/07/2021	7,722.51	Approved
Expense Report: EXP-00008232	10/16/2020	02/08/2021	7,721.73	Approved
Expense Report: EXP-00009205	02/08/2022	07/26/2022	2,138.00	Approved
Expense Report: EXP-00009384	11/17/2022	11/24/2022	1,681.26	Approved

If you want to extract the oldest approved expense report, you can use an ESI field and reference the following data:

Expense Report	Expense Report Date	Created On	Total Amount	Expense Report Status
Expense Report: EXP-00008232	10/16/2020	02/08/2021	7,721.73	Approved

Remember to use the most specific and targeted source field to extract an instance from. To optimize performance, your report should only process the minimal number of instances.

When creating an Extract Single Instance calculated field, you must specify the sorting options that determine which instance to extract. In the following example, the calculated field will return the most recent worker calibration event for each employee.

Calculated Field - Extract Single Instance

Most Recent Calibration Event

Field Name: Most Recent Calibration Event
Business Object: Employee

Calculation Additional Info Where Used

Extract Single Instance

Source Field: Worker Calibration Events
Related Business Object: Worker Calibration Event

Condition: Is True
Sort Field: Date and Time Initiated
Sort Direction: Descending (Z to A)
Instance to be Returned: First occurrence

Condition

In the Condition field, specify the filter condition for instances on the related business object. Doing so narrows down results. If you do not need a condition, select a global field like "Is True" to accept all instances of that field.

Sort Field

The sort field determines the value the system uses to sort the instances. In this case, the values sort by date and time initiated. The Sort field is a required field. If the condition eliminates all but one field, or you do not need a sort, use a global field like "Any Field" to return the instance.

Sort Direction

Choose the direction in which to evaluate the instances. To optimize performance, the field should evaluate the fewest possible instances. For example, if you sort by date and want to return the most recent instance. Start from the latest value and select the first available instance. Doing so is quicker than starting with the first or earliest date and selecting the last value available.

Instance to be Returned

Finally, you specify the instance to return - first occurrence, last occurrence, or specific occurrence.

Activity 5.3 - Lookup Related Value and Extract Single Instance

Business Case: Logan needs a report that lists the top 10 schools in the United States from which employees have earned their degrees. On a matrix report, she wants to be able to drill down by:

- Worker
- Hire Date
- Last School Attended
- Degree Obtained

Task #1: Create T/F Condition and Extract Single Instance Field

First, Logan needs to isolate a single-instance where the employee attended a school in the United States and received a degree.

1. Sign in as Logan McNeil (*/mcneil*).
2. Run the **Create Calculated Field** task.
3. Enter the following information:

Field Name	Entry Value
Field Name	CF TF School in the USA and Degree Received
Business Object	Education
Function	True/False Condition

4. Select **OK**.
5. Enter the following information:

Field Name	Entry Value
Field	Country
Operator	Frequently Used > in the selection list
Comparison Type	Value specified in this filter (<i>default</i>)

Comparison Value	United States of America
------------------	--------------------------

6. Select the **Add Row** icon to add a second row.
7. Enter the following information:

Field Name	Entry Value
Field	Degree Received
Operator	Frequently Used > in the selection list
Comparison Type	Value specified in this filter (<i>default</i>)
Comparison Value	Yes

8. Optionally, you may add a Description and update the Category on the Additional Info tab.
9. Select **OK** to save.
10. Select **Create Another Calculation** or run the **Create Calculated Field** task.
11. Enter the following information:

Field Name	Entry Value
Field	CF ESI Most Recent School in the USA
Business Object	Worker
Function	Extract Single Instance

12. Select **OK**.
13. Enter the following information:

Field Name	Entry Value
Source Field	Education
Condition	CF TF School in the USA and Degree Received

Sort Field	Last Year Attended
Sort Direction	Descending (Z to A)
Instance to be Returned	First occurrence

14. Select **OK** to save this field.

Task #2: Create Lookup Related Value Fields

Now, Logan needs to bring in the School and Degree associated with the most recent education record.

1. Select the **Create Another Calculation** button or run the **Create Calculated Field**.
2. Enter the following information:

Field Name	Entry Value
Field Name	CF LRV Most Recent School Attended
Business Object	Worker
Function	Lookup Related Value

3. Select **OK**.
4. Enter the following information:

Field Name	Entry Value
Lookup Field	CF ESI Most Recent School in the USA
Return Value	School

5. Select **OK** to save this field.
6. Select the **Create Another Calculation** button or run the **Create Calculated Field**.



Tip: Keep in mind that when creating multiple Lookup Related Values in a row, you can also copy the first LRV using the copy function in the Related Actions to create the remaining LRVs.

7. Enter the following information:

Field Name	Entry Value
Field Name	CF LRV Most Recent Degree
Business Object	Worker
Function	Lookup Related Value

8. Select **OK**.

9. Enter the following information:

Field Name	Entry Value
Lookup Field	CF ESI Most Recent School in the USA
Return Value	Degree

10. Select **OK** to save this field.

Task #3: Create a Custom Report

1. Run the **Create Custom Report** task.
2. Enter the following information:

Field Name	Entry Value
Report Name	WICT Top 10 Schools in the USA
Report Type	Matrix

Data Source	Workers for HCM Reporting
Optimized for Performance	(clear)



Important: Be sure to clear the **Optimized for Performance** checkbox. For the purpose of this training activity, we need to filter on and group by nonindexed fields in this report. If you only need to use indexed fields, you can always change this setting from the Advanced tab of the report definition after creating the report.

3. Select **OK** to continue.
4. In the Data Source Filter field, change the selection to **All Employees**.
5. On the Row Grouping column, in the Group by Field column, select **CF LRV Most Recent School Attended**.
6. In the Maximum Number of Rows field, enter **10**.
7. On the Column Grouping (Optional) grid, in the Group by Field column, select **CF LRV Most Recent Degree**.
8. On the **Drill Down** tab, add three rows on the Drillable Fields grid, and enter the following information:

Field Name	Sort
CF LRV Most Recent Degree	Alphabetical - Ascending
CF LRV Most Recent School Attended	Alphabetical - Ascending
Worker	Alphabetical - Ascending

9. Add four rows to the Detail Data Columns grid.
10. Select the following fields:
 - Worker
 - Hire Date
 - CF LRV Most Recent School Attended
 - CF LRV Most Recent Degree

11. Select the **Filter** tab.
12. Add a row on the Filter on Instances grid and configure the row as follows:

Field Name	Sort
And/Or	And (<i>default</i>)
*Field	CF LRV Most Recent School Attended
*Operator	Frequently used > is not empty

13. Select the **Prompts** tab.
14. Select the **Populate Undefined Prompt Defaults** checkbox.
15. Select **Do Not Prompt at Runtime** for all three prompts.
16. Select **OK** and **Run**.

Activity Complete

Extract Multi-Instance

Extract Multi-Instance extracts specific instances from a multi-instance field. The calculated field returns zero, one, or multiple instances based on the number of instances that satisfy the condition.

Examples of the Extract Multi-Instance calculated field include:

- Trainings each employee completes within the last 12 months.
- Current enrollment elections for an employee.
- Invoices currently due, but unpaid.
- Accounts with negative balances.
- A specific set of companies for bursting report output.
- Projects that are both high-risk and have a status of open.

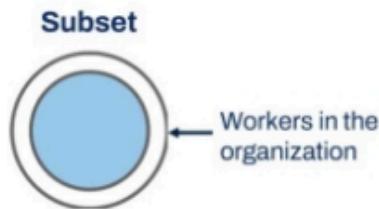
The following breaks down how the Extract Multi-Instance calculated field function determines its return value:

- Selects operation type determining how many source fields you will select.
- Accesses the related business object associated with the multi-instance field you specify.
- Applies the condition to the instances selected by the source fields. Make sure to define the condition on the related business object before creating the Extract Multi-Instance calculated field.
- Identifies and extracts instances that meet criteria and operation type.

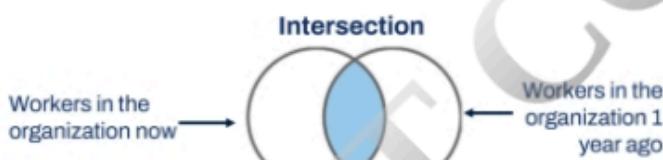
Extract Multi-Instance Operation Types

The Extract Multi-Instance function extracts data in several ways:

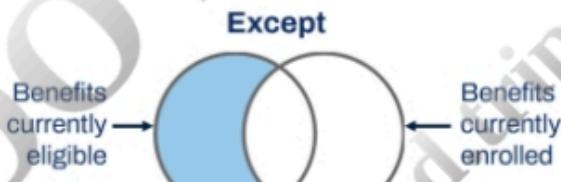
- As a subset, it filters instances from the specified source field (Source Field 1) and returns the results. For example, return workers in the organization who are "high potential."



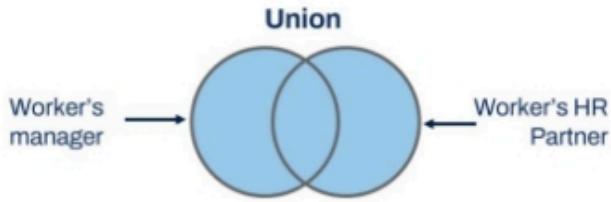
- As an intersection, it evaluates the filtered instances from both fields (Source Field 1 and Source Field 2) and returns only the instances that are common to both fields. For example, return workers who are in the organization now and were in the organization one year ago.



- As an exception, it evaluates the filtered instances from both specified source fields (Source Field 1 and Source Field 2) and returns the instances from the first field minus the instances from the second field. Except is useful for exception reporting. For example, return a list of benefits in which an employee is currently eligible, but for which the employee is not currently enrolled.



- As a union, it combines the filtered instances from both specified source fields (Source Field 1 and Source Field 2) and returns the results. For example, return a list for each employee consisting of their manager and HR Partner.



In functions using two source fields, both source fields must be on the same business object.

Chapter Summary

Key Takeaways:

- Consider performance when creating fields that evaluate conditions.
- Fields on the Global business object are viewable on any report by all users.
- Use the Date Difference function to return the amount of time between two dates.
- Use the Increment or Decrement Date function to compute a date that is a specified amount of time before or after an existing date field.
- Use the True/False Condition calculated field to return a Boolean value you can use in other calculated fields.
- Use Evaluate Expression fields to evaluate a series of conditions and return the value associated with the first true condition.
- Use the Lookup Related Value calculated field to look up the value of a field from a related business object and promote it to the primary business object.
- Use the Extract Single Instance calculated field to return a single-instance field from a related business object.

Chapter 5 Knowledge Check

1. Which calculated field function can be used to add or subtract a certain number of days, months, or years to a specific date?
 - a. Format Date
 - b. Increment or Decrement Date
 - c. Date Difference
 - d. Date Constant
2. Which calculated field function is the best option to create a field that captures the last four digits of an employee's Social Security number?
 - a. Format Number
 - b. Substring Text
 - c. Format Text
 - d. Extract Single Instance
3. Which calculated field function is the best option to evaluate and categorize the number of days off that an employee has taken over the past year?
 - a. Evaluate Expression
 - b. True/False Condition
 - c. Lookup Range Band
 - d. Lookup Related Value