



ORACLE
NETSUITE

SuiteAnalytics Workbook

2023.1

May 24, 2023



Copyright © 2005, 2023, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, and MySQL are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

If this document is in public or private pre-General Availability status:

This documentation is in pre-General Availability status and is intended for demonstration and preliminary use only. It may not be specific to the hardware on which you are using the software. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to this documentation and will not be responsible for any loss, costs, or damages incurred due to the use of this documentation.

If this document is in private pre-General Availability status:

The information contained in this document is for informational sharing purposes only and should be considered in your capacity as a customer advisory board member or pursuant to your pre-General Availability trial agreement only. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described in this document may change and remains at the sole discretion of Oracle.

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle Master Agreement, Oracle License and Services Agreement, Oracle PartnerNetwork Agreement, Oracle distribution agreement, or other license agreement which has been executed by you and Oracle and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <https://www.oracle.com/corporate/accessibility>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Sample Code

Oracle may provide sample code in SuiteAnswers, the Help Center, User Guides, or elsewhere through help links. All such sample code is provided "as is" and "as available", for use only with an authorized NetSuite Service account, and is made available as a SuiteCloud Technology subject to the SuiteCloud Terms of Service at www.netsuite.com/tos.

Oracle may modify or remove sample code at any time without notice.

No Excessive Use of the Service

As the Service is a multi-tenant service offering on shared databases, Customer may not use the Service in excess of limits or thresholds that Oracle considers commercially reasonable for the Service. If Oracle reasonably concludes that a Customer's use is excessive and/or will cause immediate or ongoing performance issues for one or more of Oracle's other customers, Oracle may slow down or throttle Customer's excess use until such time that Customer's use stays within reasonable limits. If Customer's particular usage pattern requires a higher limit or threshold, then the Customer should procure a subscription to the Service that accommodates a higher limit and/or threshold that more effectively aligns with the Customer's actual usage pattern.

Beta Features

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

This documentation is in pre-General Availability status and is intended for demonstration and preliminary use only. It may not be specific to the hardware on which you are using the software. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to this documentation and will not be responsible for any loss, costs, or damages incurred due to the use of this documentation.

The information contained in this document is for informational sharing purposes only and should be considered in your capacity as a customer advisory board member or pursuant to your pre-General Availability trial agreement only. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle Master Agreement, Oracle License and Services Agreement, Oracle PartnerNetwork Agreement, Oracle distribution agreement, or other license agreement which has been executed by you and Oracle and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

Send Us Your Feedback

We'd like to hear your feedback on this document.

Answering the following questions will help us improve our help content:

- Did you find the information you needed? If not, what was missing?
- Did you find any errors?
- Is the information clear?
- Are the examples correct?
- Do you need more examples?
- What did you like most about this document?

Click [here](#) to send us your comments. If possible, please provide a page number or section title to identify the content you're describing.

To report software issues, contact NetSuite Customer Support.

Table of Contents

SuiteAnalytics Workbook Overview	1
Getting Started with SuiteAnalytics Workbook	1
Enabling SuiteAnalytics Workbook in Your NetSuite Account	3
The Analytics Administrator Permission	4
Data Refresh in SuiteAnalytics Workbook	5
Comparison of Standard Data Refresh and Optimized Data Refresh	6
Standard Data Refresh	6
Optimized Data Refresh	8
Accessing and Sharing Workbooks and Datasets	13
Known Limitations in SuiteAnalytics Workbook	14
Analytics Data Source Overview	16
The Analytics Data Source and SuiteAnalytics Workbook	16
Available Record Types	17
Analytics Data Source Changes	18
Guidelines for Joining Record Types in SuiteAnalytics Workbook	25
SuiteQL	31
Using SuiteQL	32
SuiteQL Join Types	36
SuiteQL Syntax and Examples	42
SuiteQL Limitations and Exceptions	44
SuiteQL Supported and Unsupported Functions	44
SuiteQL Supported Built-in Functions	49
Custom Workbooks and Datasets	53
Defining a Dataset	54
Editing a Dataset	56
Formula Fields	57
Hierarchical Fields	75
Advanced Sorting Options	77
Joining Record Types in a Dataset	78
Linking Datasets	78
Dataset Criteria Filters	79
Filter Types	79
Grouping Filters	80
Creating a Workbook	81
Workbook Table Views	84
Workbook Pivot Tables	86
Workbook Charts	92
Workbook Visualization Filters	95
Calculated Measures	100
Conditional Formatting	107
Dataset Linking in SuiteAnalytics Workbook	108
Joining Record Types Versus Linking Datasets	109
Link Datasets in a Workbook	110
Create Visualizations Based on Linked Datasets	113
Currency in Datasets and Workbooks	113
Currency Consolidation in Workbook	114
Currency Conversion in Workbook	115
Formula Fields and Calculated Measures	117
Workbook and Dataset Templates	120
Dataset Templates	120
Workbook Templates	121
Analytical Record Types	122
Workbook-based Portlets	124

SuiteAnalytics Workbook Tutorial	125
Select a Root Record Type	125
Add Fields and Join Record Types	126
Filter Your Dataset	127
Create a Workbook Based on Your Dataset	129
Pivot Your Dataset Query Results	129
Connect a Second Dataset to Your Workbook	132
Set up a Table View	133
Navigating SuiteAnalytics Workbook	134
Analytics Home Page	134
Dataset Builder	135
Table Tab	139
Pivot Tab	141
Chart Tab	144
SuiteAnalytics Workbook Glossary	147
SuiteAnalytics Workbook FAQs	149

SuiteAnalytics Workbook Overview

SuiteAnalytics Workbook is an analytical tool available in NetSuite. With SuiteAnalytics Workbook, you can create highly customizable workbooks that combine datasets, tables, pivot tables, and charts using a single tool that leverages the new analytics data source. The analytics data source is designed to ensure that fields are consistently exposed in Workbook, with consistent results across all workbook visualizations. Additionally, support for multilevel joins has been added to Workbook, enabling you to author workbooks and datasets using field data from multiple record types, including custom records. Custom formula fields are also supported in Workbook, so you can create and add fields with dynamically calculated values to your workbooks. Finally, you can also link datasets in Workbook, enabling you to compare metrics from different datasets in one workbook visualization.

The SuiteAnalytics Workbook user interface has been designed so that even users with limited knowledge of record schemas and query language can still create complex workbooks and datasets through actions such as drag-and-drop editing. When you create a workbook or workbook visualization, you can base it on any dataset that you have access to in your account. You can also connect multiple datasets to the same workbook, enabling you to analyze different metrics without switching browser tabs or workbooks. When you connect multiple datasets to a workbook, you can then link them to analyze their metrics in a pivot table or chart.



Note: You currently cannot edit how datasets are linked using the Workbook user interface, but you can use linked datasets to create new workbook visualizations. For more information, see [Dataset Linking in SuiteAnalytics Workbook](#).

The Workbook user interface makes it ideal for ad-hoc diagnostic analysis, with options for instant formula validation, data refresh, and drilling down through query results. A range of customization options have also been added to the interface to enable rich formatting, filtering, and visualizations of your data.

The latest iteration of SuiteAnalytics Workbook enables you to create workbooks and datasets for a variety of record types. You can also use analytical record types created specifically for Workbook, such as the sales (invoiced) record type.



Important: SuiteAnalytics Workbook is currently not supported by SuiteBundler. To avoid installation errors with your bundles, do not include SuiteAnalytics Workbook objects. For more information about SuiteBundler, see the help topic [SuiteBundler Overview](#).

To help familiarize yourself with SuiteAnalytics Workbook and to better understand the content available in the Workbook Help Center, see [Getting Started with SuiteAnalytics Workbook](#).

Getting Started with SuiteAnalytics Workbook

To help familiarize yourself with SuiteAnalytics Workbook, see the following video and help topics:

[Getting Started with SuiteAnalytics Workbook](#)

- To learn how to set up Workbook:
 - Workbook is enabled by default. However, if you do not see the Analytics tab in the NetSuite navigation menu, see [Enabling SuiteAnalytics Workbook in Your NetSuite Account](#).
 - There should be at least one user with the Analytics Administrator in your account. To learn what this user is enabled to do in Workbook, see [The Analytics Administrator Permission](#).
 - If you experience long load times for your datasets or workbooks using your datasets, you can enable Cached Data in Datasets. For more information, see [Optimized Data Refresh](#).

- To learn about custom workbooks and datasets, and predefined workbook and dataset templates:
 - [Custom Workbooks and Datasets](#) contains information about creating your own workbooks and datasets. This section includes steps for defining, filtering, and pivoting your workbook source data, as well as procedures for how to link datasets and create custom formula fields.
 - Workbook supports multiple predefined workbook and dataset templates, some of which are in a beta state. For more information, see [Workbook and Dataset Templates](#).
- To learn about the analytics data source:
 - Workbook uses the new analytics data source which might require different fields, record types, joins, or formulas to replicate your existing saved searches. Review the [Analytics Data Source Overview](#) for more information, including guidelines for joining record types and steps for authoring sample workbooks.
 - Some record types have been created specifically for Workbook and exist only in the analytics data source. For more information, see [Analytical Record Types](#).
- To learn how to navigate the Workbook user interface:
 - Workbook is built using a new user interface. To learn about the elements within the interface, see [Navigating SuiteAnalytics Workbook](#).
 - There is new terminology associated with Workbook. For more information, see the [SuiteAnalytics Workbook Glossary](#).
- To learn how to create and run SuiteAnalytics Workbook queries using SuiteScript:
 - [Scripting with the N/query Module](#) explains the different module objects and how to use them when defining your queries.
 - To learn about linking datasets as part of your query definitions, see [Linking Datasets](#).
- Additional resources:
 - Complete the [SuiteAnalytics Workbook Tutorial](#) to walkthrough the creation of a sample sales order workbook.
 - For information about how to add workbook-based portlets to your NetSuite dashboards, see [Workbook-based Portlets](#).
 - There are some known limitations with the current iteration of Workbook. For more information, see [Known Limitations in SuiteAnalytics Workbook](#).
 - You can also visit the [SuiteAnalytics Workbook Community](#) to find answers and other information about Workbook.
- Workbook videos:
 -  [Data Model Basics](#): This video describes the different record types available in the analytics data source that is used by Workbook.
 -  [Joining Record Types in Workbook](#): This video explains the implications of joining record types in Workbook. It also provides some information about the SQL that is used to accomplish these joins in the background, and some suggestions to improve performance.
 -  [Separation of Dataset and Workbook in 2020.1](#): This video describes the change in NetSuite 2020.1 that separates datasets from workbooks.
 -  [Using Filters in SuiteAnalytics Workbook](#): This video describes the different filtering options available in Workbook, including: data grid filters, value-based and measure-based filters for pivot tables, chart filtering options, and workbook criteria filters.
 -  [Transaction Data Model in 2020.1](#): This video discusses the implications of joining the transaction, transaction line, and transaction accounting line record types in NetSuite 2020.1. The video also generally describes some of the changes between a transaction saved search and a

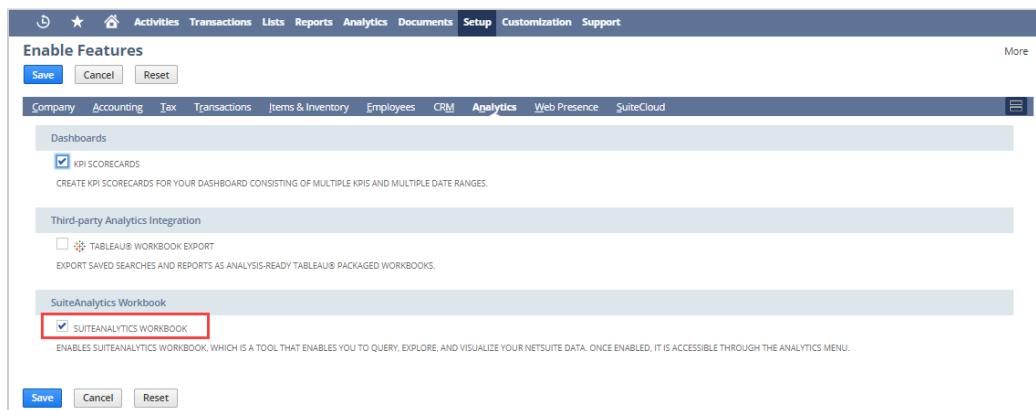
transaction workbook. You will also learn some guidelines for using the transaction record type as the root record for a dataset.

Enabling SuiteAnalytics Workbook in Your NetSuite Account

By default, the SuiteAnalytics Workbook feature is enabled in all NetSuite accounts. If you do not see the Analytics tab in the NetSuite navigation menu, complete the following steps to verify that the feature has been enabled in your account.

To verify that SuiteAnalytics Workbook is enabled in your account:

1. Log in to your NetSuite account as an administrator.
2. Go to Setup > Company > Enable Features, and click the **Analytics** subtab.
3. Ensure that the **SuiteAnalytics Workbook** box is checked.



4. Click **Save**.

With the feature enabled, all standard roles have access to Workbook except for the following roles:

- Customer Center
- Employee Center
- Vendor Center
- Partner Center
- Advanced Partner Center
- NetSuite Support Center
- NetSuite Support Center (Basic)
- Publisher Center

Because the Workbook feature is enabled for all NetSuite accounts, all users assigned standard roles have the SuiteAnalytics Workbook permission. However, users assigned custom roles might not have access to Workbook. To provide Workbook access to users assigned to a custom role, add the SuiteAnalytics Workbook permission to the role and set the access level to Edit on the Reports subtab of the Permissions tab.

To enable users to edit and monitor workbooks created by other users in your account, add the Analytics Administrator permission to the user's role on the Setup subtab of the Permissions tab. Alternatively, to

enable the Analytics Administrator permission for individual users, add the Analytics Administrator global permission to the employee record.

For more information about customizing roles, see the help topic [Customizing or Creating NetSuite Roles](#).

For more information about the Analytics Administrator permission, see [The Analytics Administrator Permission](#).

The Analytics Administrator Permission

The Analytics Administrator permission enables users to delete, share, and edit the workbooks and datasets created in your account. Users with this permission can also track changes to workbooks and datasets using the audit trail and execution log record types. Additionally, the Analytics Administrator permission enables users to define the audience for workbook and dataset templates.

The minimum access level for the permission is Full, and by default it is enabled for account administrators. The permission can be assigned to a role on the Setup subtab of the Permissions tab, or as a global permission on an employee record. Deleted workbooks and datasets are irretrievable however, so the permission should only be enabled for a small number of users in your account. Additionally, while users with the Analytics Administrator permission can delete workbooks and datasets created by other users, they can still only view the fields and records that they have access to based on their NetSuite permissions. Users with the Analytics Administrator permission are also not automatically given access to SuiteAnalytics Workbook and must be assigned to a role with the SuiteAnalytics Workbook permission to access the tool.

Editing Employee Workbooks and Datasets

Users with the Analytics Administrator permission have access to every workbook and dataset in your account through the Analytics Home page.

- To view workbooks and datasets created by other users, select Employee Workbooks or Employee Datasets in the drop down list of the respective subtab.
- To change the properties of a workbook or dataset without opening it, click the Edit icon next to name, description, or owner fields. You can also share or delete workbooks and datasets using the icons in the Actions column. Currently, notifications are not sent when workbooks are deleted, shared, or assigned to new owners.

Auditing Workbooks and Datasets



Important: As of 2020.1, the audit trail and execution log record types only contain data from release 2019.2 and earlier. You should not use these record types for auditing changes made to workbooks and datasets after your account was upgrading to 2020.1. If you want to query these record types using SuiteAnalytics Connect, use the NetSuite2.com data source and search for the usrauditlog, usrdsauditlog, and usrexecutionlong record type IDs. For more information, see the help topic [Connect Data Source](#).

To monitor the use of Workbook, users with Analytics Administrator permission have access to the audit trail and execution log record types. There is a version of each record type for workbooks and datasets, which you can differentiate using the record ID. The dataset audit trail and execution log record types both contain 'Ds' in the record ID. You can use these record types to create datasets that show you the data being accessed through Workbook and to track changes to the saved workbooks and datasets in your account.

Record Type	Description	Fields
Audit trail	<p>Displays changes made to each saved workbook or dataset in your account, including the date the change was saved and the user who saved it. Records for each saved workbook and dataset are stored indefinitely, or until the workbook or dataset is deleted.</p> <ul style="list-style-type: none"> ■ To view the most recent saved changes, join fields from the change audit change record type to the audit trail dataset. ■ To view the details of each saved change in a workbook or dataset, including the fields that were changed and the new and original values, join the new value and old value fields from the detail audit detail record type to the audit trail dataset. 	Date/Time, ID, User, Saved Workbook
Execution log	<p>Displays all workbooks and datasets that were created in the past 30 days, including those that were not saved. For each workbook or dataset listed, the execution log displays the following data:</p> <ul style="list-style-type: none"> ■ All fields used on the dataset ■ The base record type ■ The date the workbook or dataset was run ■ The name of the user who ran the workbook or dataset ■ If a workbook visualization or dataset was exported ■ Any formulas used ■ If applicable, the name of the associated saved workbook 	Base Record Type, Date/Time, Export, Expression, Fields, Formulas, ID, User Name, Saved Workbook

Defining the Audience for Workbook and Dataset Templates

Workbook offers many predefined workbook and dataset templates. For a detailed list including summaries of the content within each workbook or dataset, see [Workbook and Dataset Templates](#).

Some of these templates are in a beta or demo state and should be used for test purposes only.

To provide access to these templates, users with the Analytics Administrator permission can share most templates with specific users or roles directly from the Analytics Home page. If the Share icon is unavailable for a workbook or dataset, users with the Analytics Administrator permission must open the workbook or dataset and save and share a copy of it. For more information about sharing workbooks and datasets, see [Saving and Sharing Workbooks and Datasets](#).

Data Refresh in SuiteAnalytics Workbook

In SuiteAnalytics Workbook, by default the data presented in pivot tables and charts is cached after 60 minutes of inactivity, while datasets do not use cached data. However, you can enable the Cached Data in Datasets feature to use cached data also in your datasets.

With the Cached Data in Datasets feature enabled, you can load data more quickly also in your datasets, and optimize the data refresh in all your datasets and workbooks. See the following options:

- **Optimized Data Refresh** - With the Cached Data in Datasets feature enabled, you can choose to use cached data or real-time data in your datasets and workbooks, and change the data refresh option at any time. For more information, see [Optimized Data Refresh](#).

- **Standard Data Refresh** - With the standard option, cached data is available for pivot tables and charts only. To better understand the default caching process and how to see real-time data in your pivot tables and charts, see [Standard Data Refresh](#).

For information about the differences between the optimized data refresh and the standard data refresh, see [Comparison of Standard Data Refresh and Optimized Data Refresh](#).

Comparison of Standard Data Refresh and Optimized Data Refresh

The following tables provides a comparison of the data refresh options available for datasets and workbook visualizations when using the standard data refresh and the optimized data refresh.

Datasets and Workbook Visualizations	Standard Data Refresh		Optimized Data Refresh	
-	Cached Data in Datasets feature disabled (default option)		Cached Data in Datasets feature enabled	
-	Cached Data	Real-time Data	Cached Data	Real-time Data
Datasets and Table Views	Not available.	Always Available.	Available when set to Cached Response mode. Refresh frequency: adjusted to usage	Available when set to Real-time Response mode.
Pivot Tables and Charts	Always available. Refresh frequency: after 60 minutes of inactivity.	Available when clicking the refresh icon. In some scenarios, you need to clear the cached data first to see real-time data.	Available when set to Cached Response mode. Refresh frequency: after 60 minutes of inactivity.	Available when set to Real-time Response mode.

For a better understanding of the two data refresh options, see the following topics:

- [Standard Data Refresh](#)
- [Optimized Data Refresh](#)

Standard Data Refresh

With the standard data refresh, pivot tables and charts display cached data which is loaded from stored data to load data more quickly. Datasets and table views do not use cached data. Datasets display only real-time data which may take longer to load results.



Note: You are using the standard data refresh when the Cached Data in Datasets is not enabled. For more information about this feature, see [Optimized Data Refresh](#).

This caching process provides better performance by returning results faster. However, these results might not display the most current data. To find out when the data was last refreshed in your workbook, verify the time displayed in the **Last updated** field, in the upper-right corner of the Viewer. To better understand the default caching process and how to clear the cache manually, see the following topics:

- [Clearing the Cache for Pivot Tables and Charts](#)
- [Understanding Data Caching in Pivot Tables and Charts](#)

Understanding Data Caching in Pivot Tables and Charts

By default, the data is refreshed after 60 minutes of inactivity. To better understand when the data is refreshed, see the following scenarios:

- [Creating and modifying pivot tables and charts in an existing workbook](#)
- [Saving and re-opening a workbook before 60 minutes have elapsed](#)
- [Saving and re-opening a workbook after 60 minutes have elapsed](#)
- [Working with the Analytics Portlet](#)

Creating and modifying pivot tables and charts in an existing workbook

After you set the layout of a new pivot table or chart, click the **Refresh** icon to apply the changes and update the results. If you make changes to a pivot table directly from the Pivot Table Viewer, the results are automatically displayed.

When you create or modify your pivot tables and charts, SuiteAnalytics Workbook creates a backend query. This backend query definition retrieves the results for the defined pivot tables and charts, and enables you to see the results in the Viewer. If there are pivot tables and charts that share the same backend query definition, the data used for the refresh can vary:

- If the backend query was refreshed less than 60 minutes ago from either the pivot or chart tab, the data from the cache is used for the refresh.
- If the backend query was refreshed more than 60 minutes ago from either the pivot or chart tab, the most current data is used for the refresh.

Note that some changes in pivot tables and charts require loading real-time data. For example, adding a field to **Rows**, **Columns**, or **Measures**.

You can verify the time of the last refresh in the upper-right corner of the pivot table or chart viewer.

Saving and re-opening a workbook before 60 minutes have elapsed

When you re-open a saved workbook before 60 minutes have elapsed and open a pivot table or chart, by default the Viewer displays cached data. The time of the last refresh appears in the upper-right corner of the Viewer. To retrieve the most current data, click the **Refresh** icon.

Saving and re-opening a workbook after 60 minutes have elapsed

When you re-open a saved workbook after 60 minutes have elapsed, the cached data has expired. Consequently, when you open a pivot table or chart, the Viewer automatically displays the most current results.

Working with the Analytics Portlet

The pivot tables and charts in the Analytics portlet display the same data as their corresponding workbooks. However, you can retrieve the most current results by clicking the **Refresh** icon .

For more information about how to clear the cache manually, see [Clearing the Cache for Pivot Tables and Charts](#).

Clearing the Cache for Pivot Tables and Charts

The **Clear cache** option clears cached data from workbook pivot tables and charts so that you can retrieve the most current results. You can verify when the data was last updated by checking the date and time displayed in the upper-right corner of the pivot table or chart Viewer. The following steps describe how you can clear the cache for your open pivot tables and charts.

To clear the cache for pivot tables and charts:

1. On the Pivot or Chart tab, click the menu icon  in the upper right corner of the Viewer.
A popup message appears asking if you want to clear the cache.
2. Choose how you want to proceed:
 - **Clear cache** - Clears the cache for the entire workbook. The time and date in the upper-right corner of the Viewer is updated to reflect when the data was updated.
 - **Cancel** - The cache is not cleared. Data is updated following the default caching process.
3. If you clear the cache, a popup message appears asking you to refresh all pivot tables and charts and an icon is displayed next to each pivot table and chart in the workbook. There are three available icons:
 - **Warning icon**  - The displayed results are not up to date according to the date and time shown in the **Last updated** field. To retrieve the most current results, click the **Refresh** icon .
 - **Loading icon**  - The data is currently being updated according to the date and time shown in the **Last updated** field. After the refresh is completed, the icon changes to either the pivot or chart icon.
 - **Pivot and Chart icons**  - The data displayed is up to date according to the date and time shown in the **Last updated** field.
4. To retrieve the most current results in each open pivot table and chart, click the **Refresh** icon  in each tab.

For more information about the Pivot table and Chart tabs, see [Pivot Tab](#) and [Chart Tab](#).

To better understand the caching process of SuiteAnalytics Workbook, see [Understanding Data Caching in Pivot Tables and Charts](#).

Optimized Data Refresh

By default, only pivot tables and charts display cached data. If you also want to use cached data in your datasets and table views to load results more quickly, you can enable the Cached Data in Datasets feature. After the feature is enabled, you can choose between two options:

- **Real-time Response** - Datasets, workbooks, and Analytics portlets display the most current data available. When you use this option, loading your results may take longer.
- **Cached Response** - Datasets, workbooks, and Analytics portlets display cached data which is loaded from stored data to load results more quickly. The frequency to refresh cached data is adjusted to

your usage. The Cached Response mode monitors when you view your datasets, workbooks, or Analytics portlets. Then, the data refresh is scheduled based on this usage to ensure that the data is refreshed when you need to view it. The scheduled frequency can vary from 1 hour to 24 hours, depending on the usage in your account.

For information about how to enable the Cached Data in Datasets feature, see [Enabling the Cached Data in Datasets Feature](#).

For information about how to work with cached data or real-time data in SuiteAnalytics Workbook, see the following topics:

- [Using Cached Data in SuiteAnalytics Workbook](#)
- [Using Real-Time Data in SuiteAnalytics Workbook](#)

Enabling the Cached Data in Datasets Feature

To begin using cached data in datasets and table views, you must first enable the feature.

To enable Cached Data in Datasets in your account:

1. Go to Setup > Company > Setup Tasks > Enable Features.
2. Click the **Analytics** subtab.
3. Check the **Cached Data in Datasets** box.
4. Click **Save**.

After you have enabled the Cached Data in Datasets feature, you can choose to use Real-time Response or Cached Response mode in your datasets and workbooks. For information about the two options, see the following topics:

- [Using Cached Data in SuiteAnalytics Workbook](#)
- [Using Real-Time Data in SuiteAnalytics Workbook](#)

Using Cached Data in SuiteAnalytics Workbook

You can use Cached Response mode to load results more quickly in your datasets and workbooks. To enable the Cached Response mode, and understand the caching process and how you can monitor cached data in your datasets, see the following topics:

- [Enabling Cached Response Mode](#)
- [Cached Data in Connected Datasets and Workbooks](#)
- [Working with Cached Response Mode](#)
- [Monitoring Cached Data for Datasets](#)
- [Known Limitations in Cached Response Mode](#)

Enabling Cached Response Mode

To use cached data, you need to set the data refresh in your dataset or workbook to Cached Response mode. You can only set the data refresh to Cached Response mode for the datasets and workbooks that you can access. To use Cached Response mode, ensure that the Cached Data in Datasets feature is enabled in your account. For more information, see [Enabling the Cached Data in Datasets Feature](#).

To enable Cached Response mode:

1. Open your dataset or workbook.



Note: If you create a new workbook or dataset, you must save your dataset to change the data refresh to Cached Response mode.

2. On the upper-left corner, click the arrow next to **Real-time Response**.
3. In the Response Mode window, click **Cached**.
4. Click **Apply**.

The Cached Response mode is now enabled.

Cached Data in Connected Datasets and Workbooks

When you enable Cached Response mode in your datasets or workbooks, the mode also changes for the connected datasets and workbooks in some scenarios. For more information, see [Data Refresh Changes in Connected Datasets and Workbooks](#).

Working with Cached Response Mode

When you set the data refresh to the Cached Response mode, the results of your datasets and workbooks are loaded from stored data when you open them or make any changes. To see when your data was last updated, check the time in the upper-left corner of the Viewer, next to **Cached Response**.



Note: Some changes in datasets require loading real-time data, even if the Cached Response mode is enabled. Cached data is unavailable when you add or remove fields from your dataset, when you modify your dataset criteria filters, or when you change a formula that is used in your dataset. Save changes in your dataset to work with cached data again.

When you work with cached data, you can also get the most current data at any time. To refresh cached data and see the most up-to-date results, click the Refresh icon.

The frequency for refreshing cached data may be different for datasets and workbook visualizations:

Frequency to refresh cached data in datasets

The frequency to refresh cached data in your datasets is adjusted to your usage. The Cached Response mode monitors when you view your datasets, workbooks, or Analytics portlets. Then, the data refresh is scheduled based on this usage to ensure that the data is refreshed when you need to view it. The scheduled frequency can vary from 1 hour to 24 hours, depending on the usage in your account.

Frequency to refresh cached data in pivot tables and charts

The cached data in your pivot tables and charts is refreshed after 60 minutes of inactivity, as well as the standard data refresh option. For detailed information about cached data in pivot tables and charts, see [Understanding Data Caching in Pivot Tables and Charts](#).

Monitoring Cached Data for Datasets

To get an overview of all datasets using Cached Response mode, users with the Administrator role can access the Dataset Cache Management page. The Dataset Cache Management page lists all datasets using Cached Response mode, impacted workbooks, workbook owner, status, and other information. You can disable the Cached Response mode on any dataset from this page to change it to the Real-time Response mode. Only users with the Administrator role can access the Dataset Cache Management page.

To access the Dataset Cache Management Page

1. On the Analytics Home page, click the **Datasets** subtab.
2. In the Actions column, click the **More** icon.
3. Click **Dataset Cache Management**.



Note: This option does not disable the Cached Data in Datasets feature. The option only changes the data refresh to Real-time Response mode. The owners of workbooks and datasets can still change the response mode by opening the workbook or dataset and setting the data refresh to Cached Response mode. For more information, see [Enabling Cached Response Mode](#).

Known Limitations in Cached Response Mode

Character large object (CLOB) values are not available in cached response mode. When you use CLOB fields in your datasets and workbooks and try to work with cached response mode, you get a message that cached data cannot be enabled. To continue working with your datasets and workbooks, you can identify if you are working with CLOB fields and do one of the following:

- Remove CLOB fields – If you remove the fields from your dataset, you can work with cached response mode.
- Keep CLOB fields – If you keep the fields, you can work with real-time response mode only.

Identifying CLOB fields

There is no way to identify CLOB fields directly in your datasets or workbooks. However, you can use the Records Catalog to check field types.



Note: The Records Catalog is available for all users that have the Records Catalog permission assigned to their role. For more information, see the help topic [Records Catalog Overview](#).

To identify CLOB fields:

1. Go to Setup > Records Catalog.
2. Use the search field to find your record.
3. Click **SuiteScript and REST Query API**.
4. View the **Type** column on the right. All fields that appear with the **CLOBTEXT** type are CLOB fields.

For more information about the details available in the Records Catalog, see the help topic [Records Catalog Fields Available for SuiteScript and Rest Query API](#).

Using Real-Time Data in SuiteAnalytics Workbook

By default, all datasets, workbooks, and Analytics portlets use real-time data. This means that the most current data is displayed, but loading results may take longer. If your workbook or dataset displays cached data and you want them to display always real-time data, set the data refresh to Real-time Response mode.

Enabling Real-Time Response Mode

To enable Real-time Response mode:

1. Open your workbook or dataset.
2. On the upper-left corner, click the arrow next to **Cached Response**.
3. In the Response Mode window, click **Real-Time**.
4. Click **Apply**.

The Real-time Response mode is now enabled. When you enable Real-time Response mode in your datasets or workbooks, the mode also changes for the connected datasets and workbooks in some scenarios. For more information, see [Data Refresh Changes in Connected Datasets and Workbooks](#).

Users with the Administrator role can also change the Cached Response mode to Real-time Response mode for all datasets. For more information, see [Monitoring Cached Data for Datasets](#).

Working with Real-Time Response Mode

When you use Real-time Response mode, your datasets and workbooks always display the most updated data. Every time you open your workbooks and datasets or you make changes, the most current data is loaded and this may take longer. To optimize your resources, you can use cached data to work with your datasets and workbooks. For more information, see [Using Cached Data in SuiteAnalytics Workbook](#).

Data Refresh Changes in Connected Datasets and Workbooks

You can change the response mode to Cached Response or Real-time Response in several ways:

Changing the response mode from a dataset

When you change the response mode from the dataset, the mode changes for the workbook if your workbook uses one dataset only. If your workbook uses more than one dataset, you must change the response mode in all linked datasets first. See the following example with the same workbooks and datasets:

- Example a: **Workbook 1a** uses **Dataset 1a**.
- Example b: **Workbook 1b** uses **Dataset 1b** and **Dataset 2b**.

In the first example, when you enable the mode from **Dataset 1a**, the mode for **Workbook 1a** is also enabled. However, when you enable the mode from **Dataset 1b**, the mode for **Workbook 1b** does not change. You must enable the mode in both datasets first, **Dataset 1b** and **Dataset 2b**. Then, the mode is also enabled for **Workbook 1b**.

For more information about how to work with Cached Response mode, see [Working with Cached Response Mode](#).

Changing the response mode from a workbook

When you change the response mode from your workbook, the mode is also enabled for all linked datasets that your workbook uses. See the following examples:

- Example a: **Workbook 1a** uses **Dataset 1a**.
- Example b: **Workbook 1b** uses **Dataset 1b** and **Dataset 2b**.

In the first example, when you enable the mode from **Workbook 1a**, the mode is also enabled for **Dataset 1a**. In the second example, when you enable the mode for **Workbook 1b**, the mode also changes for both datasets, **Dataset 1b** and **Dataset 2b**.

Accessing and Sharing Workbooks and Datasets



Important: Workbooks created prior to 20.1 now exist as two separate workbook and dataset objects. Consequently, any users that you shared a workbook with prior to 20.1 have access to both the workbook and the associated dataset. However, because changes to a dataset are automatically propagated to any associated workbooks, recipients cannot edit a shared dataset unless they save their own versions. For more information, see [Custom Workbooks and Datasets](#).

When you first click the Analytics tab from the NetSuite navigation menu, you are presented with all the workbooks and datasets that you have access to on the Analytics Home page. This includes any workbooks or datasets that you own and any that have been shared with you. To preview the contents of a workbook or dataset, click the Details link to open the Details panel. If you have the Analytics Administrator permission, you can also open workbooks and datasets created by other users in your account, and access or share predefined workbook and dataset templates. For more information about the Analytics Administrator permission, see [The Analytics Administrator Permission](#).

The record types and fields displayed in each workbook or dataset are based on the features enabled in your account and the permissions assigned to the role you use to login to NetSuite. For example, to view fields and data from the invoice record type or to create a dataset based on the invoice record type, you must have the Invoice permission assigned to your role. Additionally, certain actions in SuiteAnalytics Workbook are controlled by your NetSuite permissions. For example, to export a dataset to a CSV file, you must have the Exports Lists permission at the Create level or higher.

For more information about specific record type and field access in Workbook, download the following worksheet: [NetSuitePermissionsUsage.xls](#). If you do not see a specific record type or field that you need to access, contact your system administrator. Also, keep in mind that you can only access certain workbooks and datasets such as the **Sales (Invoiced)** workbook, if they are shared with you by other users in your account. For more information, see [Saving and Sharing Workbooks and Datasets](#).

Saving and Sharing Workbooks and Datasets

You can share workbooks and datasets with individual users or groups of users on a role by role basis. To prevent data discrepancies in workbooks that use the same datasets, recipients of a shared workbook cannot edit the associated datasets. For example, if you share a workbook with users assigned to the Accountant role, all users assigned to that role can access and save their own versions of the workbook or connected datasets, but they cannot edit the original datasets. Additionally, if there are multiple datasets connected to a workbook, you must save each dataset before saving and sharing the workbook. This includes datasets that you create during the workbook authoring process but do not actually connect to a visualization. If you do not want to share datasets that are not connected to a workbook visualization, delete them before saving and sharing the workbook.

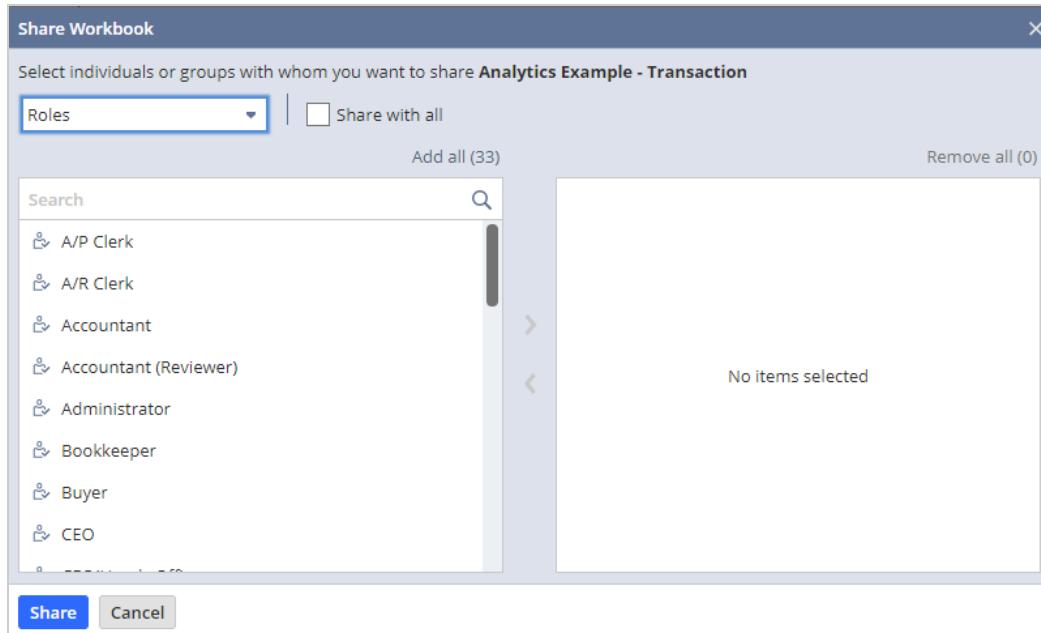
Anyone with access to Workbook can share a workbook or dataset, however recipients can only view the record types and fields that they have access to based on their permissions. Record types and fields that a recipient does not have access to are simply not displayed. If you have the Analytics Administrator permission, you can also share workbooks and datasets with every user in your account using the **Share with all** option. Keep in mind however, if you use **Share with all** in a workbook, the original author will not be able to connect additional datasets to it. Make sure you contact the original owner of a workbook before you share it with everyone in your account.

Shared workbooks include all workbook visualizations, including any selections made on the Table View, Pivot, or Chart tabs. If any of the visualizations are based on linked datasets, the link definition is shared as well. You cannot share a set of linked datasets without sharing the associated workbook. Shared datasets contain all the joined record types, fields, and criteria filters used to define the dataset. If you have the export lists permission and you only want to share a specific dataset, click the Export icon in the dataset to download a CSV file of the Data Grid.

To save and share a workbook or dataset:

1. Click **Share** from anywhere within the workbook or dataset.
2. Select the roles or users you want to share the workbook or dataset with, then click the right arrow to add them to the distribution list. If you have the Analytics Administrator permission, check **Share with All** to share the object with everyone in your account.

⚠ Important: If you use the **Share with all** option in a workbook, the original author will not be able to connect additional datasets to it.



3. Click **Save and share**.

The object is shared and appears on the Analytics Home page for the selected recipients. Currently, notifications are not sent when a workbook or dataset is shared with a user.

⚠ Important: You cannot save and share a workbook if it has visualizations based on unsaved datasets. Before you begin, save or delete any datasets connected to the workbook. Additionally, you cannot save and share workbooks and datasets that contain fields that are in a beta state. These fields are for test purposes only and are subject to change. Remove these fields before you attempt to save and share.

i Note: If you have the Analytics Administrator permission and you want to share workbook or dataset templates such as Open Sales Orders Lines or Number of Fulfillments, you must follow the steps in this procedure. These templates cannot be shared from the Analytics Home page.

Known Limitations in SuiteAnalytics Workbook

The following features and interactions in SuiteAnalytics Workbook have known limitations that are currently being addressed by our development teams:

- To mitigate the possibilities of a CSV injection, the following formatting is applied to all dataset and workbook results that are exported to CSV:

- All text values are enclosed in double quotation marks
- Values that begin with the characters -, +, =, @, tab, and EOL (end of line) have single apostrophes added in front of the value

This formatting also applies to workbook and dataset results that you export to CSV through SuiteScript and SuiteQL.

- The audit trail and execution log record types only contain data from 2019.2 and earlier. For more information, see [Auditing Workbooks and Datasets](#).
- On the Pivot tab, you cannot apply measure-based filters to fields that contain date or text values when the aggregation types for those fields are set to Min or Max.
For more information about measure-based filters, see [Workbook Visualization Filters](#).
- Some ad blocker plug-ins can prevent you from creating pivot tables or charts. To avoid errors with the Pivot or Chart tabs, deactivate these plug-ins and mark NetSuite as an exception. For more information about NetSuite browser support, see the help topic [Supported Browsers for NetSuite](#).
- If you define a criteria filter that is invalidated based on changes to the field used in the filter condition, you are currently unable to edit the filter.
- Users assigned to the employee center or partner center roles who have permission to access Workbook might not see the Analytics option in the NetSuite navigation menu.
- Filters that you create on the Dataset Builder based on custom duration values currently do not work as expected.
- When you query the analytics data source using SuiteScript, you currently cannot filter data using record types such as invoice.
- If you add multiple pivot tables or charts in a workbook, the user interface does not always display arrows so that you can switch between each table and chart.
- Certain ad blocking software can prevent data from loading in your workbook pivot tables.
- The search field on the Dataset Builder searches across record types that are a few joins away from the root record type. To find record types or fields that are further from the root record type, click **Show more results**.
- If you have multiple languages set up in your account, edits to workbook visualization texts are added as their own collections and not added to the existing workbook collection.
- Collection and string IDs on the Manage Translation page are not editable.
- You currently cannot manually disconnect a dataset from a workbook after it has been connected. For more information, see the [SuiteAnalytics Workbook FAQs](#).
- You currently cannot create Table Views based on linked datasets.
- If there are multiple fields in a dataset with the same ID but different join paths, using them to create a dataset link can cause issues. For more information, see [Link Datasets in a Workbook](#).
- Character large object (CLOB) data is not available in cached response mode. For more information, see [Known Limitations in Cached Response Mode](#).

Analytics Data Source Overview

Data Model Basics

The analytics data source is a collection of NetSuite data that is grouped according to record types and fields. The structure of the analytics data source enhances the capabilities of analyzing your NetSuite data which you can query using SuiteQL. This advanced query language lets you run complex SQL queries.

The analytics data source has been used in SuiteAnalytics Workbook since 2019.1. You can also retrieve data from the analytics data source using SuiteAnalytics Connect, SuiteScript, and SuiteTalk REST web services.

To help you understand the analytics data source, see the following topics:

- You can use the Records Catalog to see the fields and joins available for each record type. The Records Catalog displays also information such as the cardinality, field level help, and so on. Due to Records Catalog's reliance on the NetSuite role-based permissions, you can see only the record types for the data that you can access in the NetSuite user interface. The Records Catalog is supported for the SuiteScript Analytic API and the REST Query API only. Therefore, the record types available in the Records Catalog and the dataset user interface may differ slightly. For more information, see the help topic [SuiteScript 2.x Analytic APIs](#).

 **Note:** The Records Catalog is available for all users that have the Records Catalog permission assigned to their role. For more information, see the help topic [Records Catalog Overview](#).

- You can use SuiteQL to run queries against the analytics data source through SuiteAnalytics Connect, SuiteScript, and SuiteTalk REST web services. For more information, see [SuiteQL](#).
- In Workbook, you can further customize the record types and fields available in the analytics data source for use in your workbooks. For more information, see [The Analytics Data Source and SuiteAnalytics Workbook](#).
- Some record types have been created specifically for Workbook and exist only in the analytics data source. For more information, see [Analytical Record Types](#).
- To locate fields in the analytics data source using the Workbook user interface, you can browse and search the Records and Fields list on the Dataset tab. For more information, see [Locating Fields in Workbook](#).

The Analytics Data Source and SuiteAnalytics Workbook

For every NetSuite account, all data is stored using a single database. The data sources used to expose this data through Saved Searches and Reports however, are slightly unique. For some record types, these unique data sources have resulted in inconsistent field naming and data exposure between the two tools.

In SuiteAnalytics Workbook, fields and record types are exposed using a new analytics data source that is designed to display consistent data across the Workbook application. Fields created in support of new NetSuite features are exposed to Workbook using the analytics data source, as of 2019.1.

The location, names, and IDs of some fields and record types might be different in the analytics data source. Additionally, some record types and fields have not been ported to the analytics data source. This includes fields that contain calculated values for certain record types. Consequently, you might need

to use different record types and fields, or create new joins and custom formula fields to recreate your existing saved searches using Workbook.

To help you navigate the analytics data source, see the following topics:

- You can use the Records Catalog to see the fields and joins available for each record type. The Records Catalog displays also information such as the cardinality, field level help, and so on. Due to Records Catalog's reliance on the NetSuite role-based permissions, you can see only the record types for the data that you can access in the NetSuite user interface. The Records Catalog is supported for the SuiteScript Analytic API and the REST Query API only. Therefore, the record types available in the Records Catalog and the dataset user interface may differ slightly. For more information, see the help topic [SuiteScript 2.x Analytic APIs](#).

 **Note:** The Records Catalog is available for all users that have the Records Catalog permission assigned to their role. For more information, see the help topic [Records Catalog Overview](#).

- The analytics data source supports hundreds of record types and fields. For more information about the record types and fields that you can find in Workbook, see [Available Record Types](#).
- The location and names of some fields in the analytics data source might be different from their Search and Report counterparts. Additionally, some fields such as fields with calculated values have not yet been ported to the analytics data source. For more information about differences in the new data source, see [Analytics Data Source Changes](#).
- After you select a root record type for a custom dataset or open a saved or shared dataset, all related record types that you have access to are displayed in the Records list. For more information about using the Dataset Builder to select and join related record types, see [Defining a Dataset](#).
- Workbook supports multilevel joins, however there are certain functions and interactions that you should be cautious of when adding specific record types or fields to a dataset. For more information, see [Guidelines for Joining Record Types in SuiteAnalytics Workbook](#).
- You can link datasets in a workbook even if they are based on record types that do have a predefined relationship in the analytics data source. For more information, see <Linking Datasets in a Workbook>.

Available Record Types

SuiteAnalytics Workbook supports hundreds of record types available through the analytics data source. The analytics data source includes most of the record types used throughout NetSuite.

You can use the Records Catalog to see the fields and joins available for each record type. The Records Catalog displays also information such as the cardinality, field level help, and so on. Due to Records Catalog's reliance on the NetSuite role-based permissions, you can see only the record types for the data that you can access in the NetSuite user interface. The Records Catalog is supported for the SuiteScript Analytic API and the REST Query API only. Therefore, the record types available in the Records Catalog and the dataset user interface may differ slightly. For more information, see the help topic [SuiteScript 2.x Analytic APIs](#).

 **Note:** The Records Catalog is available for all users that have the Records Catalog permission assigned to their role. For more information, see the help topic [Records Catalog Overview](#).

To view each record type that you have access to in your account, you can also click New Dataset on the Analytics Home page. The record category and ID for each record type is displayed. Record types are divided into three categories:

- Standard – Record types that are included in each account of your NetSuite implementation and are available through all NetSuite data sources. For example, the transaction record type. Standard record

types consist of concrete and generic record types. For more information, see [Analytics Data Source Changes](#).

- Analytical – Record types that have been created specifically for use with Workbook. These record types are based on standard record types but contain a unique set of fields. For more information, see [Analytical Record Types](#).
- Custom – Record types created by users in your company that are unique to your NetSuite account.

Access to record types and fields in Workbook depends on the same permissions used for NetSuite saved searches. For example, to access sales order data through the transaction record type using Workbook, you must have the Sales Order permission assigned to your role. Analytical record types such as the sales (invoiced) record type are only accessible to users with the Analytics Administrator permission. Additionally, some record types are only available in Workbook if you have specific features enabled in your account. For information about permissions—related access to record types and fields in the analytics data source, download the following worksheet: [NetSuitePermissionsUsage.xls](#)

After you select a root record type for a dataset or open an existing dataset, the Records list on the Dataset Builder shows all the related record types that you have access to. If you cannot find a record type that you think you should have access to, contact your administrator.



Note: The search field in the Dataset Builder searches across record types that are a few joins away from the root record type. To search across record types that are further away, click **Show more results**.

Analytics Data Source Changes

The analytics data source is different than the data sources used by the NetSuite Saved Search and Report applications. In the analytics data source, concrete record types that have a corresponding entry form in NetSuite remain mostly unchanged and use the same field labels. For example, the employee record type in SuiteAnalytics Workbook contains mostly all the fields that are on the NetSuite employee form, with the same field labels that are used in saved searches and reports. Generic record types that do not have a single corresponding form in NetSuite, such as the transaction, entity, and item record types, use generic field labels. This is because generic record types are representative of multiple concrete record types. Additionally, some fields in the analytics data source are accessible through different record types than in saved searches and reports.

The design of the analytics data source means that the names of some record types and fields have changed or appear differently than on NetSuite forms, saved searches, and reports. Some fields, such as those with calculated values, have also not been ported to the data source and other fields are only accessible by joining specific record types. Furthermore, the analytics data source includes some record types and fields that have been created specifically for SuiteAnalytics Workbook and are not available through saved searches and reports. These record types and fields are in a beta state and should be used for test purposes only.

To help you navigate the analytics data source, the following resources are available in the Workbook user interface and the NetSuite Help Center:

- To view the record types and fields that you have access to, use the Records list in the Dataset Builder. By default, datasets display all joinable record types and fields based on the root record type selected and the permissions assigned to the role you use to log in to NetSuite. For example, if you select the customer record type for your dataset, the following joinable record types are available:

If you click a record type, additional related record types are displayed.

- To recreate your existing saved searches using Workbook, the changes to the analytics data source mean that you might have to use different record types and fields, create custom formula fields, or join multiple record types. Click this link to access a Microsoft Excel worksheet that compares the transaction record type used by the saved search data source to the transaction record type used in Workbook: [TransactionMapping.xlsx](#)
- For annotated steps on how to recreate some of your saved searches using Workbook, see the following topics. Each sample includes the fields, record types, and formulas required to recreate the search:
 - [Open Sales Orders Lines](#)
 - [Number of Fulfillments](#)
 - [Journal Entry to Approve](#)

Open Sales Orders Lines

This saved search displays the items allocated to open sales orders in your account, so that you can better monitor your inventory.

To perform this search using the Saved Search application, all required criteria and results fields are available through the transaction record type. To recreate this search using SuiteAnalytics Workbook however, some required fields are only available through the transaction line and transaction accounting line record types. You must therefore join these record types in your dataset to recreate the search. Additionally, some required fields might have different labels in Workbook. The following table lists key differences in fields between the saved search and the workbook:

Key Field Differences in Workbook

Field Name in Saved Search	Record Type in Workbook	Required Join in Workbook	Field Name in Workbook
Main Line	Transaction Line	Transaction > Transaction Line	Main Line
Tax Line	Transaction Line	Transaction > Transaction Line	Tax Line
Closed	Transaction Line	Transaction > Transaction Line	Closed
Account Type	Account	Transaction > Transaction Line > Transaction Accounting Line > Account	Type
Shipping Line	Transaction Account Line	Transaction > Transaction Line > Transaction Accounting Line	Shipping Cost Item Type



Warning: In this workbook you must add fields from the transaction line and transaction accounting line record types. When you add fields from either the transaction line or transaction accounting line record type to a transaction dataset, data duplication can occur. For more information, see [Joining Transaction Line and Transaction Accounting Line in a Dataset](#).

To recreate this search using Workbook:

Create a new dataset with transaction as the root record type. Then, define the dataset as follows:



Important: Users assigned to the Sales role do not have access to the account record type in Workbook, which is required to properly recreate this saved search. If you create and share this workbook with users assigned to the Sales role, make sure you give them access to the account record type first.

Root Record Type	Joined Record Type(s)	Custom Formula Field(s)	Data Grid	Criteria
Transaction	<ul style="list-style-type: none"> ■ Transaction Line □ Transaction Accounting Line <ul style="list-style-type: none"> - Account 	Create the following formula fields on the Dataset Builder:	Add the following fields to the grid from the	Set the following criteria in the Dataset Builder:

	<p>Important: Make sure you join the transaction accounting line record type from the transaction line record type. If you join both the transaction line and transaction accounting line record types directly from the transaction record, it can duplicate or otherwise skew your data. For more information, see Joining Transaction Line and Transaction Accounting Line in a Dataset.</p>	<ul style="list-style-type: none"> ■ Name: Unfulfilled Items Output Type: Float Definition: -{transactionlines.quantity}-NVL({transactionlines.quantitycommitted},0)-NVL({transactionlines.quantityshiprecv},0) Use this formula to replace null values in the Transaction Line Quantity and Quantity committed fields with 0. ■ Name: Positive Item Quantity Output Type: Float Definition: -{transactionlines.quantity} Use this formula to display your item quantities as positive values. <p>For more information about how to use formula fields in SuiteAnalytics Workbook, see Formula Fields.</p>	<ul style="list-style-type: none"> ■ transaction record type: <ul style="list-style-type: none"> ■ Transaction ■ Status ■ Posting Period ■ Entity ■ Add the following fields to the grid from the transaction line record type: <ul style="list-style-type: none"> ■ Item ■ Location ■ Quantity ■ Committed ■ Quantity Shipped/ Received/ Picked Up ■ Add the following custom formula fields to the grid: <ul style="list-style-type: none"> ■ Custom Formula Field 1 (Unfulfilled Items) ■ Custom Formula Field 2 (Positive Item Quantity) 	<ul style="list-style-type: none"> ■ [Custom Formula Field 2] greater than 0.00 ■ Type is Sales Order ■ Status any of Sales Order : Partially Fulfilled and Sales Order : Pending Fulfillment and Sales Order : Pending Billing/Partially Fulfilled ■ Transaction Line: Main Line is false ■ Transaction Line: Tax Line is false ■ Transaction Line: Item Type none of Shipping Cost Item ■ Account: Type is income
--	--	--	---	--

Open Sales Orders Lines Pivot Tables

Create a new workbook using the open sales order lines dataset you created, and set the following fields to the appropriate dimensions for each pivot table:

- Open Sales Orders by Date

Rows: Item
Columns: Date (Quarter)
Values: [Custom Formula Field 2] (Sum)
- Open Sales Orders by Location and Date

Rows: Location, Item
Columns: Date (Quarter)
Values: [Custom Formula Field 2] (Sum)
- Open Sales Orders by Customer and Date

Rows: Entity, Item
Columns: Date (Quarter)

Values: [Custom Formula Field 2] (Sum)

After you set each field to the appropriate table dimension, click the refresh  icon to generate the table.

You can also customize the table and add totals and grand totals, or rename each table. For more information, see [Pivot Table Customization](#).

Open Sales Orders Lines Charts

By recreating this saved search using Workbook, you can also chart your source data to visualize the information using different chart types.

Create the following charts by dragging the listed fields from the Dataset Panel to the corresponding section of the Layout panel, and select the chart type.

- Open Sales by Item

X-Axis: Item

Series: N/A

Values: Total Amount (Sum)

Chart type: Column chart

- Open Sales by Date

X-Axis: Date (Quarter)

Series: Item

Values: Total Amount (Sum)

Chart type: Line chart

After you set each field and the chart type, click the Refresh  icon to generate the chart. You can also filter values, add a title and subtitle, and rename each axis. For more information, see [Workbook Charts](#).

Number of Fulfillments



Note: Users with the Analytics Administrator permission can view a predefined version of this workbook from the Analytics Home Page. For more information, see [Workbook and Dataset Templates](#).

This saved search shows the number of items which have been fulfilled.

To perform this search using the Saved Search application, all required criteria and results fields are available through the transaction record type. To recreate this search using SuiteAnalytics Workbook however, some required fields are only available through the transaction line record type. You must therefore join this record type to your dataset to recreate the search. Additionally, some required fields might have different labels in Workbook. The following table lists key differences in fields between the saved search and the workbook:

Key Field Differences in Workbook

Field Name in Saved Search	Record Type in Workbook	Required Join in Workbook	Field Name in Workbook
Name	Transaction Line	Transaction > Transaction Line	Entity
Period	Transaction	—	Posting Period

Field Name in Saved Search	Record Type in Workbook	Required Join in Workbook	Field Name in Workbook
Amount	Transaction Line	Transaction > Transaction Line	Amount (Net)
Main Line	Transaction Line	Transaction > Transaction Line	Main Line

To recreate this search using Workbook:

Create a new dataset and select transaction as the root record type. Then, define the dataset as follows:

Root Record Type	Joined Record Type(s)	Custom Formula Field(s)	Data Grid	Criteria
Transaction	Transaction Line	—	<p>Add the following fields to the grid from the transaction record type:</p> <ul style="list-style-type: none"> ■ Transaction ■ Date ■ Due Date ■ Type ■ Total Amount (Transaction Currency) ■ Entity ■ Posting Period <p>Add the following field to the grid from the transaction line record type:</p> <ul style="list-style-type: none"> ■ Location 	<p>Set the following criteria on the Dataset Builder using fields from the transaction record type:</p> <ul style="list-style-type: none"> ■ Type is Item Fulfillment <p>Set the following criteria using fields from the transaction line record type:</p> <ul style="list-style-type: none"> ■ Main Line — Yes

Number of Fulfillments Pivot Tables

Create a new workbook using the Number of Fulfillments dataset you created, and set the following fields to the appropriate dimensions for each pivot table:

■ Number of Fulfillments Per Customer

Rows: Entity

Columns: Date (Year)

Measures: Transaction (Count)

After you set each field to the appropriate table dimension, click the Refresh icon  to generate the table.

You can also customize the table and add totals and grand totals, or rename each table. For more information, see [Pivot Table Customization](#).

Number of Fulfillments Chart

By recreating this saved search using Workbook, you can chart your source data to visualize the information using different chart types.

Create the following chart by dragging the listed fields from the Dataset Panel to the Layout panel, and select the chart type.

- **Number of Fulfillments per Customer**

X-Axis: Entity

Series: Date (Year)

Values: Transaction (Count)

Chart type: Column chart

After you set each field, click the Refresh  icon to generate the chart. You can also filter values, add a title and subtitle, and rename each axis. For more information, see [Workbook Charts](#).

Journal Entry to Approve



Important: This workbook is only useful if you have enabled the **Require Approvals on Journal Entries** permission in your NetSuite account. For more information, see the help topic [General Accounting Preferences](#).

This saved search shows the list of journal entries requiring approval. By recreating this saved search using SuiteAnalytics Workbook, you can use the drill down capability to open and approve each journal entry directly from the Data Grid.

To perform this search using the Saved Search application, all required criteria and results fields are available through the transaction record type. To recreate this search using Workbook however, some required fields are only available through the transaction line and transaction accounting line record types. You must therefore join these records in your dataset to recreate the search. Additionally, some required fields might have different labels in Workbook. The following table lists key differences in fields between the saved search and the workbook:

Key Field Differences in Workbook

Field Name in Saved Search	Record Type in Workbook	Required Join in Workbook	Field Name in Workbook
Period	Transaction	-	Posting Period
Tax Period	-	-	— This field has been deprecated in Workbook
Name	Transaction Line	Transaction > Transaction Line	Entity
Memo	Transaction Line	Transaction > Transaction Line	Memo
Created From	Transaction Line	Transaction > Transaction Line	Created From
Account	Transaction Accounting Line	Transaction > Transaction Line > Transaction Accounting Line	Account
Amount	Transaction Accounting Line	Transaction > Transaction Line > Transaction Accounting Line	Amount



Warning: In this workbook you must add fields from the transaction line and transaction accounting line record types. When you add fields from either the transaction line or transaction accounting line record type to a transaction dataset, data duplication can occur. For more information, see [Joining Transaction Line and Transaction Accounting Line in a Dataset](#).

To recreate this search using Workbook:

Create a new dataset and select transaction as the root record type. Then, define the dataset as follows:

Root Record Type	Joined Record Type(s)	Custom Formula Field(s)	Data Grid	Criteria
Transaction	<ul style="list-style-type: none"> ■ Transaction Line <ul style="list-style-type: none"> □ Transaction Accounting Line 	—	<p>Add the following fields to the grid from the transaction record type:</p> <ul style="list-style-type: none"> ■ Transaction ■ Date ■ Posting Period ■ Created By <p>Add the following fields from the Transaction Line record type:</p> <ul style="list-style-type: none"> ■ Entity ■ Memo ■ Created From ■ Location <p>Add the following fields from the Transaction Accounting Line record type:</p> <ul style="list-style-type: none"> ■ Account ■ Amount 	<p>Set the following criteria on the Dataset Builder:</p> <ul style="list-style-type: none"> ■ Type is Journal ■ Status is Journal: Pending Approval

Guidelines for Joining Record Types in SuiteAnalytics Workbook

The following video explains some basics for joining record types in SuiteAnalytics Workbook.



The following video describes how to join the transaction, transaction line, and transaction accounting line record types.



Workbook enables you to add fields from multiple record types to a single dataset. This includes record types that are more than one join away from the root record type of a dataset, enabling you to compile workbook source data from a more diverse set of record types.

By default, the Dataset Builder lists all joinable record types that you have access to. To complete a join, you add fields from the desired record type to the Data Grid or create a Criteria filter.

If joining record types is a new concept for you, there are a number of considerations you should make before you attempt to join different record types in a dataset. To best take advantage of this powerful new capability, see the following topics:

- The analytics data source contains a variety of record types with predefined relationships to one another. Depending on the type of relationship between each record type that you join in a dataset,

certain joins can cause unwanted duplication of your workbook source data. For more information, see [Data Duplication Based on Record Joins](#).

- The transaction and transaction line record types share a one-to-many relationship which can result in duplication of your workbook source data. Joining the transaction accounting line record type to a transaction dataset can also cause data duplication. For more information about using the transaction record type in a dataset, see [Joining Transaction Line and Transaction Accounting Line in a Dataset](#).
- Based on the relationship between each record type that you add to a dataset, the order in which you join record types can also impact your source data. For more information, see [Join Order in SuiteAnalytics Workbook](#).

Data Duplication Based on Record Joins

The analytics data source contains different record types with predefined relationships to one another. In the SuiteAnalytics Workbook user interface, you can see which record types are related to one another in the Records list on the Dataset Builder.

The screenshot shows the SuiteAnalytics Workbook's Dataset Builder interface. On the left, there's a sidebar with various record types listed under 'Transaction'. A search bar and filters are at the top. The main area shows a query builder with two conditions: 'Document Number/ID is 1008' and 'Entity is Phillip Van Hook'. Below the query is a table with one row, 'Sales Order #1008', showing details like Date (2/13/2002), Total Amount (\$6,982.85), Entity (Phillip Van Hook), and Document Number/ID (1008). At the bottom of the table, it says 'Sum: CAD 6,982.85 Average: CAD 6,982.85 Count: 1 Min: CAD 6,982.85 Max: CAD 6,982.85'.

Note: The record types listed on the Dataset Builder are based on the root record type of the dataset, the features enabled in your account, and the permissions assigned to the role you use to log in to NetSuite. If you do not see a record type that you think you should have access to, contact your NetSuite administrator.

There are two types of relationships that you should be cautious of when you join record types in a dataset: those with a one-to-many relationship with the source record type, and those with a many-to-many relationship with the source record type.

One-to-many relationships between record types are those in which the source record type can be associated to multiple iterations of the target record type. For example, consider the relationship between the transaction and transaction line record types. For each transaction record there can be multiple transaction line records. When records with a one-to-many relationship are joined in a dataset, the cardinality of the data from the source record type is duplicated for each instance of the target record type. This duplication can result in inaccurate data aggregations in your datasets and workbooks.

To better understand the implications of joining record types with a one-to-many relationship, consider the total amount field on the transaction record type.

In a transaction dataset, for each transaction record listed there is a single value for the total amount field.

The screenshot shows the Dynamics 365 Analytics workspace interface. At the top, there's a navigation bar with links for Activities, Transactions, Lists, Reports, Analytics, Documents, Setup, Customization, and Support. On the right side, there are buttons for 'New Workbook' and 'Give Feedback'. Below the navigation bar is a search bar with a 'Dataset' dropdown, 'Add Pivot +', and 'Add Chart +' buttons. The main area has a sidebar with a tree view of datasets, including 'Approval Status', 'Billing Address', 'Customer: Created By', 'Customer: Entity', 'Customer: Last Modified By', 'Employee: Created By', 'Employee: Entity', 'Employee: Last Modified By', 'Entity: Created By', 'Entity', 'Entity: Last Modified By', 'Fulfillment Type', 'Incomes', 'Job Entity', 'Linked Tracking Numbers T...', 'Memorized Transaction De...', 'Next Approver Employee', 'Nexus', 'Other Name Entity', 'Partner: Created By', 'Partner: Entity', 'Partner: Last Modified By', 'Partner', 'Payment Event', 'Payment Gateway Asynchron...', 'Payment Method', 'Posting Period Accounting ...', 'Promotions', and 'Sales Rep Employee'. The main content area displays a 'Criteria summary' for 'Document Number/ID AND Entity'. It shows a breakdown where 'Document Number/ID is 1008' and 'Entity is Philip Van Hook'. A 'Drop field to add a condition...' placeholder is present. Below this is a table with columns: TRANSACTION, DATE, TOTAL AMOUNT (TRANSACTION CURRENCY), ENTITY, and DOCUMENT NUMBER/ID. A single row is highlighted with a red border, representing a Sales Order #1008 from 2/13/2002 with a total amount of \$6,982.85, associated with Philip Van Hook and Document Number 1008. At the bottom, there's a summary: Sum: CAD 6,982.85, Average: CAD 6,982.85, Count: 1, Min: CAD 6,982.85, Max: CAD 6,982.85.

However, if you join the transaction line record type to the dataset by adding a transaction line field to the Data Grid, notice that the number of values listed for the total amount field is multiplied by the number of associated transaction line records.

Transaction

New Workbook Give Feedback

Dataset Add Pivot Add Chart +

Search

Criteria summary: Document Number/ID AND Entity Show details

AND

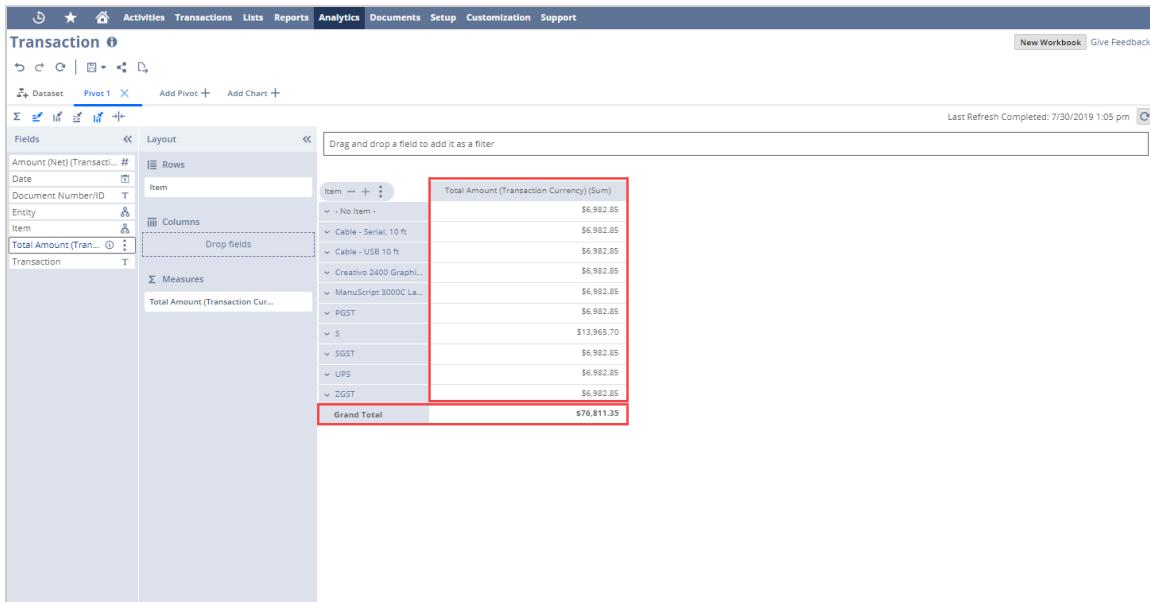
Drop field to add a condition.

New Group

TRANSACTION	DATE	TOTAL AMOUNT (TRANSACTION CURRENCY)	ENTITY	TRANSACTION LINE AMOUNT (NET) (TRANSACTION CURRENCY)	DOCUMENT NUMBER
Sales Order #1008	2/13/2002	\$6,982.85	Philip Van Hook		1008
Sales Order #1008	2/13/2002	\$6,982.85	Philip Van Hook		1008
Sales Order #1008	2/13/2002	\$6,982.85	Philip Van Hook		1008
Sales Order #1008	2/13/2002	\$6,982.85	Philip Van Hook		1008
Sales Order #1008	2/13/2002	\$6,982.85	Philip Van Hook		1008
Sales Order #1008	2/13/2002	\$6,982.85	Philip Van Hook	-\$5,999.00	1008
Sales Order #1008	2/13/2002	\$6,982.85	Philip Van Hook	-\$84.00	1008
Sales Order #1008	2/13/2002	\$6,982.85	Philip Van Hook	-\$419.93	1008
Sales Order #1008	2/13/2002	\$6,982.85	Philip Van Hook	-\$479.92	1008
Sales Order #1008	2/13/2002	\$6,982.85	Philip Van Hook	\$6,982.85	1008
Sales Order #1008	2/13/2002	\$6,982.85	Philip Van Hook		1008

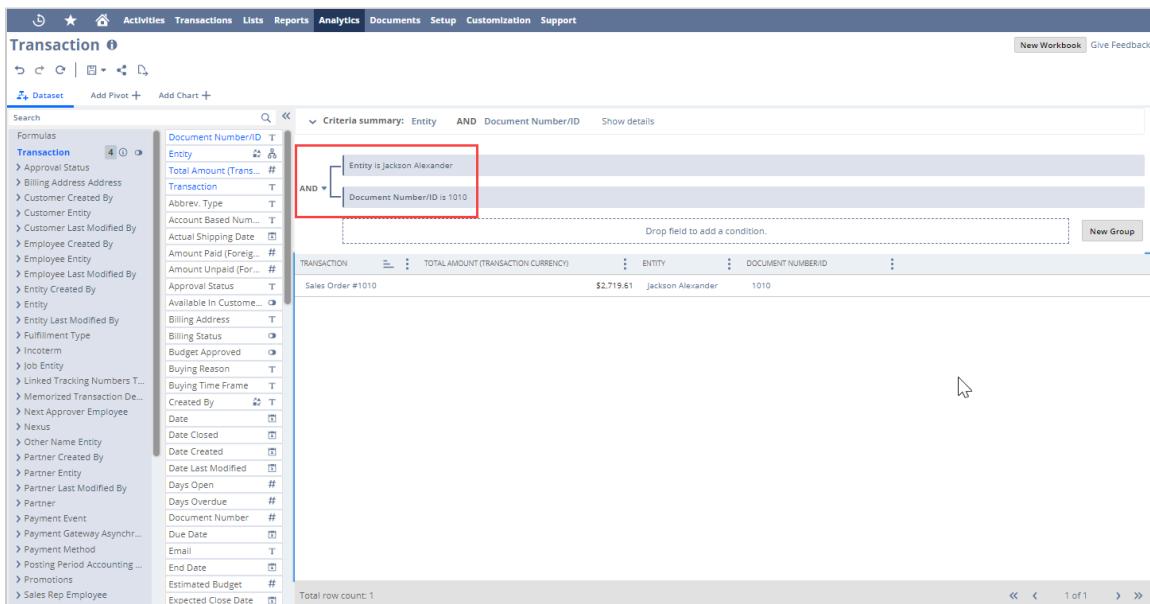
Sum: CAD 76,811.35 Average: CAD 6,982.85 Count: 11 Min: CAD 6,982.85 Max: CAD 6,982.85

The result is that aggregations based on the duplicated source record fields are inaccurate. This includes the summaries displayed at the bottom of the Data Grid, and the totals and grand totals that you set up in any pivot tables based on the dataset.



This data duplication can also occur when you create criteria filters using fields from record types that have a one-to-many relationship, even if you do not explicitly add fields from the joined record type to the Data Grid.

For example, assume you are working on a transaction dataset with only one transaction record that matches your selected criteria.



If you create an additional criteria filter based on the transaction line record type, the results are again multiplied based on the number of associated transaction line records.

TRANSACTION	TOTAL AMOUNT (TRANSACTION CURRENCY)	ENTITY	DOCUMENT NUMBER/ID
Sales Order #1010	\$2,719.61	Jackson Alexander	1010
Sales Order #1010	\$2,719.61	Jackson Alexander	1010
Sales Order #1010	\$2,719.61	Jackson Alexander	1010
Sales Order #1010	\$2,719.61	Jackson Alexander	1010
Sales Order #1010	\$2,719.61	Jackson Alexander	1010
Sales Order #1010	\$2,719.61	Jackson Alexander	1010
Sales Order #1010	\$2,719.61	Jackson Alexander	1010
Sales Order #1010	\$2,719.61	Jackson Alexander	1010
Sales Order #1010	\$2,719.61	Jackson Alexander	1010
Sales Order #1010	\$2,719.61	Jackson Alexander	1010
Sales Order #1010	\$2,719.61	Jackson Alexander	1010
Total row count: 12			

In record types with a many-to-many relationship, the same duplication can occur. Many-to-many relationships between record types exist when multiple iterations of the source record type are associated to multiple iterations of the target record type. For example, in a multi-subsidiary account, each account record can be assigned to multiple subsidiaries and each subsidiary record can be associated with multiple accounts. Consequently, the account and subsidiary record types share a many-to-many relationship, and joining these record types in a dataset can cause data duplication.

You should always be cautious when you create joins between record types that have a one-to-many or many-to-many relationship. If you do perform these types of joins, be aware of the impact they will have on your datasets and the aggregations throughout your workbooks.

Joining Transaction Line and Transaction Accounting Line in a Dataset

In the Saved Search application, by default a Transaction saved search could include both accounting and operational data. In SuiteAnalytics Workbook, to include accounting and operational data in a transaction workbook, you might need to join three record types in a single dataset: transaction, transaction line, and transaction accounting line. These record types exist in one-to-many and many-to-many relationships, which can cause data duplication. For more information, see [Data Duplication Based on Record Joins](#).

There additional implications you should be aware of when joining these record types in a dataset:

- If you need to join the transaction accounting line record type in a dataset, you must use the join path **Transaction> Transaction Line> Transaction Accounting Line**. This join path prevents increased data duplication which would occur if you used the join path **Transaction> Transaction Accounting Line**.
- If you have the Multi-Book Accounting feature enabled in your account and you join the transaction accounting line record type in a dataset, data duplication can be increased. This is because each transaction accounting line stores data for each accounting book in your NetSuite account. To limit this, consider creating a criteria filter using the accounting book field from the transaction accounting line record type, so that data from only one accounting book is included.
- The transaction accounting line record type stores monetary values in the base currency of the subsidiary assigned to the transaction. The amount fields in this record type can only be consolidated and are typically used for accounting purposes, such as calculating financial results for your entire company, including all subsidiaries.

The transaction and transaction line record types store monetary values in the currency of the transaction and can only be converted. These amount fields are typically used for operational purposes, such as analyzing total sales of a specific item. For more information about currency consolidation and conversion in Workbook, see [Currency in Datasets and Workbooks](#).

The following video demonstrates how data can be impacted by joining the transaction line and transaction accounting line record types in a dataset.



Join Order in SuiteAnalytics Workbook

As specified in the [Data Duplication Based on Record Joins](#) topic, data duplication can occur when you join record types with a one-to-many or many-to-many relationship in a dataset. We can also demonstrate this duplication by examining the order in which record types are added to a dataset.

Consider the relationship between the entity and transaction record types: for each transaction record, there can only be one associated entity or customer record. Each customer record however, can be associated with multiple transaction records. This means that summary information such as field counts can vary from dataset to dataset even if they contain the same fields and record types, depending on the order that the record types are joined in a dataset.

For example, assume you want to see a list of customers and their associated transactions. If you create a new dataset beginning with the entity record type, you can then join the transaction record type to view each associated transaction. If you select the entity field to view a count of the total number of distinct customers, it is displayed at the bottom of the Data Grid.

ENTRY ID	TYPE	NAME	TRANSACTIONS: TRANSACTION
-Accountant-	Vendor	-Accountant-	
-No Company-	Company	-No Company-	
-System-	-Internal-	-System-	
A Wolfe	Customer, Employee	A Wolfe	
Alice Stein	Contact	Stein Investments : Alice	
Allister Sullivan	Contact	Sullivan Distributors, Inc	
Amy Boughton	Customer	Amy Boughton Item Fulfillment #1001	
Amy Boughton	Customer	Amy Boughton Sales Order #1001	
Amy Boughton	Customer	Amy Boughton Cash Sale #1001	
Amy Nguyen	Employee	Amy Nguyen	
Angela A Hitchcock	Employee	Angela A Hitchcock	
Anita Jennings	Contact	Jennings Financial : Anita	
Anonymous Customer	Customer	Anonymous Customer	
BC Electric	Vendor	BC Electric Bill Payment #To Print	
BC Electric	Vendor	BC Electric Bill 3/10/2002	
BC Electric	Vendor	BC Electric Bill 1/10/2002	
BC Electric	Vendor	BC Electric Bill 2/10/2002	

Count: 229 Count (Distinct): 94

If you create your dataset beginning with the transaction record type and join the entity record type however, notice that the number of distinct customers decreases.

The screenshot shows the SuiteAnalytics Workbook interface. On the left, there's a sidebar with a tree view of datasets. A red box highlights the 'Customer' node under the 'Entity' category. The main area displays a table of transaction data with a red box around the 'ENTITY: CUSTOMER' column header. The table has columns for TRANSACTION, TYPE, TOTAL AMOUNT (TRANSACTION CURRENCY), and ENTITY. The ENTITY column lists names like Sandra Burns, Amy Boughton, etc.

TRANSACTION	TYPE	TOTAL AMOUNT (TRANSACTION CURRENCY)	ENTITY
Cash Refund #1001	Cash Refund	-\$45.95	Sandra Burns
Cash Sale #1001	Cash Sale	\$799.25	Amy Boughton
Cash Sale #1002	Cash Sale	\$2,910.55	Travis Gilbert
Cash Sale #1003	Cash Sale	\$96.43	Eric Schmidt
Cash Sale #1004	Cash Sale	\$1,009.53	Tony Matsuda
Cash Sale #1005	Cash Sale	\$1,930.73	Jackie Suh
Cash Sale #1006	Cash Sale	\$6,982.85	Philip Van Hook
Cash Sale #1007	Cash Sale	\$302.33	Tim Griffin
Cash Sale #1008	Cash Sale	\$2,879.25	Jackson Alexander
Cash Sale #1009	Cash Sale	\$2,064.25	Julie Frankel
Cash Sale #1010	Cash Sale	\$1,183.23	Monica Parker
Cash Sale #1011	Cash Sale	\$1,344.30	John Nguyen
Cash Sale #1012	Cash Sale	\$1,493.85	Ken Chua
Cash Sale #1013	Cash Sale	\$45.95	Sandra Burns
Check #2001	Check	-\$3,450.00	McGeever Property Man
Check #2002	Check	-\$228.85	Safety Net Security
Check #2003	Check	-\$159.90	Koka Office Supplies

This discrepancy is another implication of the one-to-many relationship between specific record types. To avoid confusion over these discrepancies, make sure you join related record types in the correct order according to the type of datasets and results you want to generate.

SuiteQL

SuiteQL is a query language based on the SQL-92 revision of the SQL database query language. It provides advanced query capabilities you can use to access your NetSuite records and data, and it supports querying the analytics data source. For more information, see [Analytics Data Source Overview](#).

SuiteQL is currently available using SuiteAnalytics Connect, the N/query module in SuiteScript, and SuiteTalk REST web services. For more information, see [Using SuiteQL](#).

Some benefits of using SuiteQL include the following:

- **Support for the analytics data source:** SuiteQL lets you query the analytics data source. This data source enhances the capabilities of querying your NetSuite data. The exposed data is consistent with SuiteAnalytics Workbook, which resolves previous data exposure inconsistencies in saved searches and reports. For more information, see [Analytics Data Source Overview](#).
- **Improved security:** SuiteQL enforces the same role-based access restrictions used in SuiteAnalytics Workbook. This approach means that SuiteQL lets you query the same data you can access in the SuiteAnalytics Workbook user interface, which helps to increase security. SuiteQL also includes a list of supported SQL functions and does not allow you to use unsupported SQL functions in your query, which prevents SQL injection.

SuiteQL supports the syntax for both SQL-92 and Oracle SQL. However, you cannot use both syntaxes in the same query.

To help you understand how to use SuiteQL, see the following help topics:

- [Using SuiteQL](#)
- [SuiteQL Join Types](#)
- [SuiteQL Syntax and Examples](#)

- SuiteQL Limitations and Exceptions
- SuiteQL Supported and Unsupported Functions
- SuiteQL Supported Built-in Functions

Using SuiteQL

SuiteQL lets you query your NetSuite data using advanced query capabilities. SuiteQL includes a list of supported SQL functions and does not allow you to use unsupported SQL functions in your query, which prevents SQL injection and other unauthorized access to data.

SuiteQL is currently available using SuiteAnalytics Connect and the N/query module in SuiteScript. You can also learn how to find record type names and field names to use in SuiteQL queries. See the following help topics:

- Using SuiteQL with the Connect Service
- Using SuiteQL with the N/query Module in SuiteScript
- Using SuiteQL with SuiteTalk REST Web Services
- Finding Record Type and Field Names

Using SuiteQL with the Connect Service

You can use SuiteQL to query your NetSuite data through the Connect Service. To do so, you must consider the following:

- Syntax Requirements
- Querying the Analytics Data Source
- Unsupported Features in SQL-92
- Supported and Unsupported Functions
- Supported Built-in Functions

Syntax Requirements

As you create queries using SuiteQL and the Connect Service, consider the following syntax requirements:

- For string concatenation, you cannot use the + operator. You should use the || operator instead. This restriction applies to both field and literal concatenation.
- You cannot use more than 1000 arguments in a single IN clause.
- You cannot use date literals. You must encapsulate dates using the to_date() function.
- You cannot use right outer joins. For example, the following Oracle SQL is not valid in SuiteQL:

```
1 | select a1.id from account a1, account a2 where a1.id (+) = a2.id
```

- SuiteQL supports the syntax for both ANSI and non-ANSI joins. However, you cannot use both syntax types in the same query.
- Do not use quotation marks for field names in subselections. See the following examples:
 - You should not use the following syntax with quotation marks:

```
1 | select * from ( select a.externalid "AccountId" from account a)
```

- You should use the following syntax without quotation marks:

```
1 | select * from ( select a.externalid AccountId from account a)
```

For more information, see the help topic [Query Language Compliance](#).

Querying the Analytics Data Source

To find all record types and fields that are currently available for the analytics data source schema, you can do the following:

- Use the OA_COLUMNS, OA_TABLES, and OA_FKEYS database tables. The database tables are only available through SuiteAnalytics Connect service and is not supported in SuiteQL with SuiteScript. To learn about the tables, see the help topic [SuiteAnalytics Connect System Tables](#).

 **Note:** The OA_FKEYS table does not provide information about foreign keys for NetSuite2.com data source. The table provides primary keys only.

- To find the names of record types and fields, see the help topic [Record Types and Fields](#).

The analytics data source is not accessible for some roles and permissions. For more information, see the help topic [Role and Permission Considerations for NetSuite2.com](#).

Unsupported Features in SQL-92

When using SQL-92 syntax, there are several features that are not supported, including embedded null values in CHAR fields, DEFAULT clauses for column values, and subqueries in SELECT lists.

For a full list of unsupported features, see [Non-Supported SQL-92 functionality](#).

Supported and Unsupported Functions

There are several functions that you can use when you run queries using SuiteQL. For a full list of supported and unsupported functions, see [SuiteQL Supported and Unsupported Functions](#).

Supported Built-in Functions

You can use built-in functions to perform certain operations in SuiteQL queries. For a full list of supported built-in functions, see [SuiteQL Supported Built-in Functions](#).

Using SuiteQL with the N/query Module in SuiteScript

You can use SuiteQL to query your NetSuite data using the N/query module in SuiteScript. The following object and methods are available in the N/query module to work with SuiteQL:

- The [query.SuiteQL](#) object represents a SuiteQL query. It includes properties that represent the query type, the SuiteQL string representation of the query, result columns to return from the query, and any additional parameters the query requires. You can use [SuiteQL.run\(\)](#) to run the query as a non-paged query or [SuiteQL.runPaged\(options\)](#) to run the query as a paged query.

- The `Query.toSuiteQL()` method converts an existing query (as a `query.Query` object) to its SuiteQL representation (as a `query.SuiteQL` object).
- The `query.runSuiteQL(options)` and `query.runSuiteQLPaged(options)` methods let you run an arbitrary SuiteQL query. You can specify this query as a SuiteQL string, `query.SuiteQL` object, or generic JavaScript Object containing a required `query` property and an optional `params` property.



Note: If the SuiteAnalytics Connect feature is enabled in your NetSuite account, there is no limit to the number of results these methods can return. If the SuiteAnalytics Connect feature is not enabled, these methods can return a maximum of 100,000 results across all pages in the result set. For more information about SuiteAnalytics Connect, see the help topic [SuiteAnalytics Connect](#).

Here is an example of how to load an existing query, convert it to SuiteQL, and run it:

```

1 var myLoadedQuery = query.load({
2   id: 'custworkbook237'
3 });
4
5 var mySuiteQLQuery = myLoadedQuery.toSuiteQL();
6 var myResultSet = mySuiteQLQuery.run();

```

For more information, see the help topic [SuiteQL in the N/query Module](#).

For a list of supported functions, see [SuiteQL Supported and Unsupported Functions](#) and [SuiteQL Supported Built-in Functions](#).

Using SuiteQL with SuiteTalk REST Web Services

To execute SuiteQL queries through REST web services, send a POST request to the `suiteql` resource, and specify the query in the request body after the `query` parameter `q`. The following example shows a SuiteQL query executed through REST web services.

```

1 > POST https://demo123.suitetalk.api.netsuite.com/services/rest/query/v1/suiteql
2 > Prefer: transient
3 .
4 . {
5 . . "q": "SELECT email, COUNT(*) as count FROM transaction GROUP BY email"
6 . }

```

For more information about executing SuiteQL queries through REST web services, see the help topic [Executing SuiteQL Queries Through REST Web Services](#).

Finding Record Type and Field Names

SuiteQL queries include references to record types and fields. For example, the following SuiteQL string queries for `entityid` field values that are included in employee records:

```
1 | SELECT employee.entityid AS entityidRAW FROM employee
```

To create a query using SuiteQL, you must know the names of the record types and fields that you want to use in your query. You can find these names in several ways:

- Using the [SuiteAnalytics Workbook UI](#)
- Using the [Records Catalog](#)

Using the SuiteAnalytics Workbook UI

You can use the SuiteAnalytics Workbook UI to find record type and field names. SuiteAnalytics Workbook uses the same data source that SuiteQL does (the analytics data source), so you can use the SuiteAnalytics Workbook UI to explore the available record types and fields before you create your query.

To find record type and field names using the SuiteAnalytics Workbook UI:

1. In your NetSuite account, click the **Analytics** tab in the NetSuite navigation menu.
2. On the **Workbooks** page, click **New Workbook**.
3. Search for the record type that you want to use as the root record of your query.

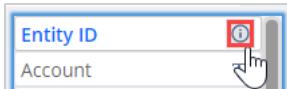
The name of each supported record type is listed in the Record ID column. You can use this page to find all supported record type names, but you cannot use this page to find field names. If you need to find field names, continue to step 4.

New Workbook: Select a Record Type			Give Feedback
Search for a record type that includes fields that you would like to show in your workbook.			
SEARCH RECORD TYPES			
employee	X		
RECORD TYPE	RECORD CATEGORY	RECORD ID	
Employee	Standard	employee	
Employee Directory	Standard	employeeList	
Employee Status	Standard	EmployeeStatus	
Employee-Subsidiary Relationship	Standard	employeeSubsidiaryRelationship	
Employee Type	Standard	EmployeeType	
RelationshipSelectEmployeeRecord	Standard	RelationshipSelectEmployeeRecord	

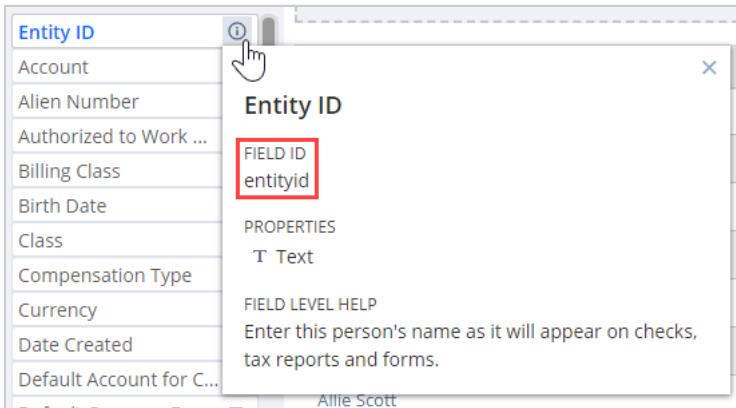
4. Click a record type. A workbook is created with the selected record type as the root record. The left column lists the record types that you can join with the root record. The middle column lists the fields that are available on the selected record.

Employee ⓘ			New Workbook Give Feedback
Dataset	+		
Formulas			
Employee	1	Entity ID	T
> Accrued Time		Account	T
> Address Book Address		Alien Number	T
> Billing Class		Authorized to Work ...	
> Bulk Merge		Billing Class	T
> Campaigns		Birth Date	
> Class		Class	
> Company Contributions		Compensation Type	T
> Currency		Currency	T
> Deductions		Date Created	
> Default Account for Corporate Card Ex...		Default Account for C...	T
> Default Billing Address Address		Default Currency For ...	T
> Default Currency For Expense Reports ...		Default Job Resource ...	T
> Default Job Resource Role Project Res...		Department	
> Default Shipping Address Address		Eligible for Commissi...	
> Department		Email	T
> Earnings		Employee Status	T

5. In the middle column, find the field that you want to use in your query. Hover over the field, and click the information button.



Information about the field is displayed. The field name is listed under Field ID. You can also see the type of data that the field contains (text, date, boolean, and so on).



If you want to find field names for fields that are on a different root record, repeat this procedure from step 2 and choose a new root record. To learn more about creating workbooks, see [SuiteAnalytics Workbook Tutorial](#).

Using the Records Catalog

If you are using the N/query module in SuiteScript, SuiteAnalytics Workbook, or SuiteAnalytics Connect, you can use the Records Catalog to see the fields and joins available for each record type. The Records Catalog displays also information such as the cardinality, field level help, and so on. Due to Records Catalog's reliance on the NetSuite role-based permissions, you can see only the record types for the data that you can access in the NetSuite user interface. The Records Catalog is supported for SuiteScript Analytic API and the REST Query API only. Therefore, the record types available in the Records Catalog and the dataset user interface may differ slightly.

Note: The Records Catalog is available for all users that have the Records Catalog permission assigned to their role. For more information, see the help topic [Records Catalog Overview](#).

To learn about the SuiteScript Analytic APIs, see the help topic [SuiteScript 2.x Analytic APIs](#).

SuiteQL Join Types

SuiteQL supports several SQL join types. You can use these join types to customize the results you receive from your SuiteQL queries. By default, when you join record types in SuiteAnalytics Workbook, the join performed is a left outer join. This join type is appropriate for many use cases, but in some situations, you may want to use a different join type to obtain a more customized result set.

For more information about joining record types in SuiteAnalytics Workbook, see [Guidelines for Joining Record Types in SuiteAnalytics Workbook](#). For more information about SQL joins in general, see [Join \(SQL\)](#).

SuiteQL supports the following SQL join types:

- **Cross Joins** — A cross join is used to produce all row combinations between two tables. The result is known as the Cartesian product of the rows in both tables.

- **Inner Joins** — An inner join is used to produce row combinations between two tables based on a common value. The results include only those rows that share the common value.
- **Outer Joins** — An outer join is also used to produce row combinations between two tables based on a common value. The results include all rows from one or both tables, depending on the type of outer join.

To demonstrate each join type, the examples in this topic use the following simplified database tables. These tables include some of the fields that appear on customer and employee records in NetSuite. You can use the Records Catalog to determine the fields you can use in SuiteQL queries. For more information, see the help topic [Records Catalog Overview](#).

customer table		
entityid	email	salesrep
Big Computers	bigcomputers@example.com	5
Fantastic Laptops	fantasticlaptops@example.com	3
PennyPack Systems	pennypacksystems@example.com	NULL
Tall Manufacturing	tallmanufacturing@example.com	1
Vision Corporation	visioncorporation@example.com	1

employee table	
id	entityid
1	Allie Anderson
2	Billy Brown
3	Carol Connors
4	David Davis
5	Eugene Evans

Cross Joins

A cross join is the Cartesian product of the rows in two tables. It combines each row in the first table with each row in the second table. This type of join is computationally expensive, so you should use this type of join rarely. You can filter the results using the WHERE keyword, which produces an inner join.

An explicit cross join uses the CROSS JOIN keywords. Here is an example of an explicit cross join:

```

1 | SELECT *
2 | FROM customer CROSS JOIN employee

```

This join is equivalent to the following implicit cross join, which does not use the CROSS JOIN keywords:

```

1 | SELECT *
2 | FROM customer, employee

```

When using SuiteQL with SuiteAnalytics Connect, the CROSS JOIN keywords are not supported. To perform a cross join, you can use the implicit notation. You can also use a full outer join to simulate a cross join, as follows:

```

1 | SELECT *
2 | FROM customer
3 | FULL OUTER JOIN employee ON 1=1

```

Using the example database tables, these joins produce 25 result rows:

customer.entityid	customer.email	customer.salesrep	employee.id	employee.entityid
Big Computers	bigcomputers@example.com	5	1	Allie Anderson
Big Computers	bigcomputers@example.com	5	2	Billy Brown
Big Computers	bigcomputers@example.com	5	3	Carol Connors
Big Computers	bigcomputers@example.com	5	4	David Davis
Big Computers	bigcomputers@example.com	5	5	Eugene Evans
Fantastic Laptops	fantasticlaptops@example.com	3	1	Allie Anderson
Fantastic Laptops	fantasticlaptops@example.com	3	2	Billy Brown
...

The first row in the first table is combined with each row in the second table, the second row in the first table is combined with each row in the second table, and so on. Additional result rows have been omitted from this table, but the same pattern continues for the remaining result rows.

Inner Joins

An inner join combines rows in the first table with rows in the second table based on a common value. The results of an inner join include only those rows that share the common column (or field) value.

An explicit inner join uses the `INNER JOIN` and `ON` keywords. Here is an example of an explicit inner join:

```

1 | SELECT customer.entityid, customer.email, employee.entityid
2 | FROM customer
3 | INNER JOIN employee ON customer.salesrep = employee.id

```

This explicit join is equivalent to the following implicit inner join, which uses the `WHERE` keyword instead of the `INNER JOIN` and `ON` keywords:

```

1 | SELECT customer.entityid, customer.email, employee.entityid
2 | FROM customer, employee
3 | WHERE customer.salesrep = employee.id

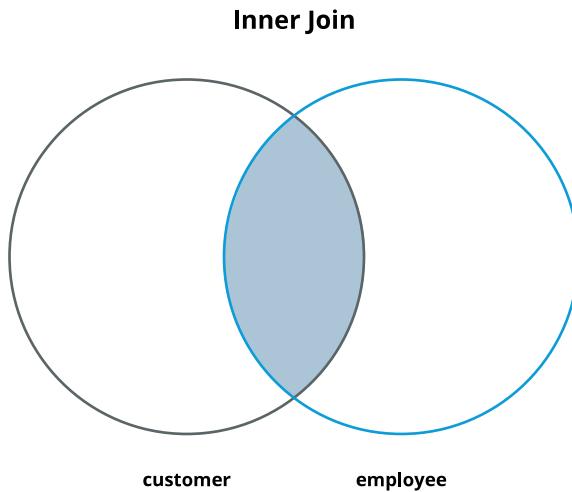
```

Using the example database tables, these joins produce four result rows:

customer.entityid	customer.email	employee.entityid
Big Computers	bigcomputers@example.com	Eugene Evans
Fantastic Laptops	fantasticlaptops@example.com	Carol Connors
Tall Manufacturing	tallmanufacturing@example.com	Allie Anderson
Vision Corporation	visioncorporation@example.com	Allie Anderson

In this example inner join, a result row is included only if the value of the `salesrep` field in the `customer` table matches the value of the `id` field in the `employee` table.

The following diagram illustrates an inner join:



Outer Joins

Similar to an inner join, an outer join combines rows in the first table with rows in the second table based on a common value. However, an outer join includes all of the rows in one or both tables in the results, depending on the type of outer join being performed.

SuiteQL supports the following outer join types:

- [Left Outer Joins](#)
- [Right Outer Joins](#)
- [Full Outer Joins](#)

Left Outer Joins

A left outer join combines all rows in the first (left) table with rows in the second (right) table based on a common value. The results of this join always contain every row in the first table, even if the join condition does not find any matching row in the second table.

This join is the default join type when you join record types in SuiteAnalytics Workbook. For more information, see [Guidelines for Joining Record Types in SuiteAnalytics Workbook](#).

An explicit left outer join uses the LEFT OUTER JOIN and ON keywords. Here is an example of an explicit left outer join:

```

1 | SELECT customer.entityid, customer.email, employee.entityid
2 | FROM customer
3 | LEFT OUTER JOIN employee ON customer.salesrep = employee.id

```

The OUTER keyword is optional, so this explicit join is equivalent to the following:

```

1 | SELECT customer.entityid, customer.email, employee.entityid
2 | FROM customer
3 | LEFT JOIN employee ON customer.salesrep = employee.id

```

This join is also equivalent to the following implicit left outer join, which uses the (+) syntax:

```

1 | SELECT customer.entityid, customer.email, employee.entityid
2 | FROM customer, employee
3 | WHERE customer.salesrep = employee.id(+)

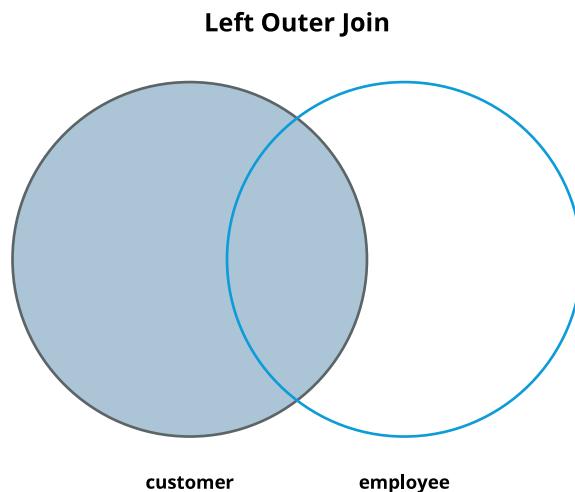
```

Using the example database tables, these joins produce five result rows:

customer.entityid	customer.email	employee.entityid
Big Computers	bigcomputers@example.com	Eugene Evans
Fantastic Laptops	fantasticlaptops@example.com	Carol Connors
PennyPack Systems	pennypacksystems@example.com	NULL
Tall Manufacturing	tallmanufacturing@example.com	Allie Anderson
Vision Corporation	visioncorporation@example.com	Allie Anderson

In this example left outer join, a row for PennyPack Systems is included in the results even though there is no matching value for that row in the employee table.

The following diagram illustrates a left outer join:



Right Outer Joins

A right outer join combines rows in the first (left) table with all rows in the second (right) table based on a common value. This join is similar to a left outer join, but the table order is reversed. The results of this join always contain every row in the second table, even if the join condition does not find any matching row in the first table.

An explicit right outer join uses the `RIGHT OUTER JOIN` and `ON` keywords. Here is an example of an explicit right outer join:

```

1 | SELECT customer.entityid, customer.email, employee.entityid
2 | FROM customer
3 | RIGHT OUTER JOIN employee ON customer.salesrep = employee.id

```

The `OUTER` keyword is optional, so this explicit join is equivalent to the following:

```

1 | SELECT customer.entityid, customer.email, employee.entityid
2 | FROM customer
3 | RIGHT JOIN employee ON customer.salesrep = employee.id

```

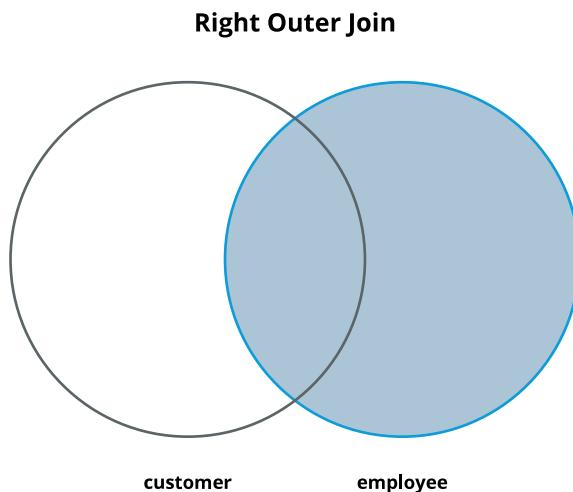
There is no implicit version of a right outer join in SuiteQL.

Using the example database tables, these joins produce six result rows:

customer.entityid	customer.email	employee.entityid
Tall Manufacturing	tallmanufacturing@example.com	Allie Anderson
Vision Corporation	visioncorporation@example.com	Allie Anderson
NULL	NULL	Billy Brown
Fantastic Laptops	fantasticlaptops@example.com	Carol Connors
NULL	NULL	David Davis
Big Computers	bigcomputers@example.com	Eugene Evans

In this example right outer join, rows for Billy Brown and David Davis are included in the results even though there are no matching values for those rows in the customer table.

The following diagram illustrates a right outer join:



Full Outer Joins

A full outer join combines the effects of a left outer join and a right outer join. A full outer join combines all rows in the first (left) table with all rows in the second (right) table based on a common value.

The results of this join always contain every row in the first table and every row in the second table.

Depending on the relationship between the tables, the results may include the same table row more than once.

An explicit full outer join uses the FULL OUTER JOIN and ON keywords. Here is an example of an explicit full outer join:

```

1 | SELECT customer.entityid, customer.email, employee.entityid

```

```

2 | FROM customer
3 | FULL OUTER JOIN employee ON customer.salesrep = employee.id

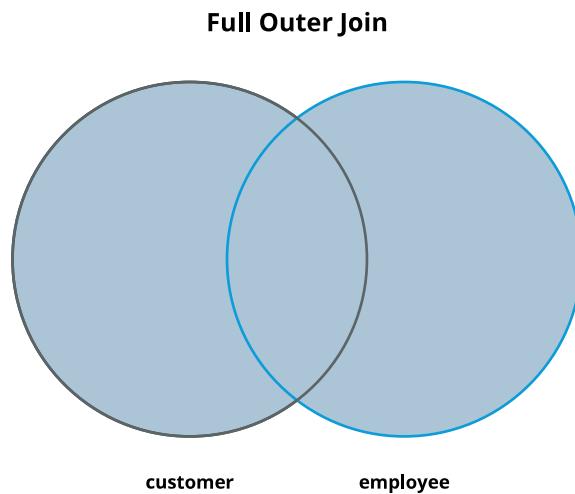
```

There is no implicit version of a full outer join in SuiteQL.

Using the example database tables, this join produces seven result rows:

customer.entityid	customer.email	employee.entityid
Big Computers	bigcomputers@example.com	Eugene Evans
Fantastic Laptops	fantasticlaptops@example.com	Carol Connors
PennyPack Systems	pennypacksystems@example.com	NULL
Tall Manufacturing	tallmanufacturing@example.com	Allie Anderson
Vision Corporation	visioncorporation@example.com	Allie Anderson
NULL	NULL	Billy Brown
NULL	NULL	David Davis

The following diagram illustrates a full outer join:



SuiteQL Syntax and Examples

SuiteQL supports the syntax for both SQL-92 and Oracle SQL, but you cannot use both syntaxes in the same query. To learn about these syntaxes, see the following links:

- [SQL-92 Language Reference](#)
- [Oracle Database SQL Language Reference](#)

The following sections show you how to create queries using SuiteQL and the analytics data source.

Simple Queries

This section demonstrates how to perform simple queries using SuiteQL.

Field Values from Records

This string queries for entityid, firstname, and lastname field values in all employee records. By default, field values use the RAW field context. For more information, see the help topic [query.FieldContext](#).

```
1 | SELECT employee.entityid AS entityidRAW, employee.firstname AS firstnameRAW, employee.lastname AS lastnameRAW FROM employee
```

Field Values with Conditions

This string queries for entityid field values in employee records where the expenselimit of the employee is greater than or equal to 5000.

```
1 | SELECT employee.entityid AS entityidRAW FROM employee WHERE employee.expenselimit >= 5000
```

This string queries for entityid field values in employee records where the expenselimit of the employee is greater than or equal to 5000 or the employee is a sales representative.

```
1 | SELECT employee.entityid AS entityidRAW FROM employee WHERE employee.expenselimit >= 5000 OR employee.issalesrep = 'T'
```

Joined Records

This string queries for paytype and payrate field values in employee records joined to employee earning records where the employee is the same on both records:

```
1 | SELECT paytype, payrate FROM employeeEarning earning, employee WHERE earning.employee = employee.id
```

Advanced Queries

The following strings demonstrate advanced queries using SuiteQL.

```
1 | /* set operations */
2 | SELECT * FROM transaction UNION SELECT * FROM transaction
3 |
4 | /* top n */
5 | SELECT TOP 10 * FROM transaction
6 | SELECT TOP 1 id FROM transaction UNION SELECT TOP 1 id FROM transaction
7 |
8 | /* select list */
9 | SELECT DISTINCT email FROM transaction
10 | SELECT (SELECT MAX(1) FROM transaction) AS one FROM transaction
11 | SELECT COUNT(DISTINCT 1+id) FROM transaction
12 | SELECT COALESCE(email, id, 'some value', 1 + 1) FROM transaction
13 |
14 | /* from */
15 | SELECT * FROM transaction, transactionLine
16 | SELECT * FROM (SELECT * FROM transaction) a INNER JOIN ((SELECT * FROM transaction UNION SELECT * FROM transaction) b INNER JOIN
17 | (SELECT * FROM transaction) c ON 1=1) ON 1=1
18 | SELECT * FROM (SELECT id, COUNT(*) cnt FROM transactionLine GROUP BY id) WHERE cnt > 2
19 |
20 | /* where */
21 | SELECT * FROM transaction t WHERE id IN (SELECT id FROM transaction WHERE id = t.id UNION SELECT -1 FROM transaction)
22 | SELECT * FROM transaction WHERE 1 = (SELECT MAX(1) FROM transaction)
23 | SELECT * FROM transaction WHERE EXISTS(SELECT 1 FROM transaction)
24 | SELECT * FROM transaction WHERE id IN ((SELECT MAX(1) FROM transaction), 2+1)
25 |
26 | /* having */
27 | SELECT email, COUNT(*), MAX(create_date) FROM transaction GROUP BY email HAVING COUNT(*) > 2
```

SuiteQL Limitations and Exceptions

The following are known limitations and exceptions when using SuiteQL:

- You can create SuiteQL queries using the syntax for either SQL-92 or Oracle SQL, but you cannot use both syntaxes in the same query. To learn about SuiteQL syntax and grammar, see [SuiteQL Syntax and Examples](#).
- The casing of record type names and field names in SuiteQL query results may not always be consistent and can change. For example, the current name of the account record type is account, but in query results, this name could appear as Account or ACCOUNT. These name changes may occur after a NetSuite release or scheduled e-fix.

```
1 | SELECT * FROM account
```

```
1 | SELECT * FROM Account
```

```
1 | SELECT * FROM ACCOUNT
```

When you work with query results, make sure that you do not depend on name casing. You can convert these names to uppercase or lowercase to ensure consistent casing.

When you run SuiteQL queries using the N/query module, the query results may not reflect the casing used to build the query. For example, you can use aliases (the AS keyword) in your query string. However, the query results do not reflect the casing of the aliases you specify, and the aliases appear in lowercase. Consider the following:

```
1 | query.runSuiteQL({
2 |   query: 'SELECT id AS myID FROM transaction'
3 | }).asMappedResults();
```

This method runs the specified SuiteQL query string and returns the query results as mapped results. In the array of mapped results, the key appears as myid instead of myID. For more information, see the help topics [query.runSuiteQL\(options\)](#) and [ResultSet.asMappedResults\(\)](#).

When you run constructed queries (those you create using the [query.create\(options\)](#) method in the N/query module), you do not need to consider potential casing changes. For constructed queries, you should specify the query type using values from the [query.Type](#) enum. If you specify the query type as a string directly and do not use values from this enum, the query type is processed in a case-insensitive way, and your query will still run correctly.

- `BUILTIN.HIERARCHY(PARENT, 'LEVEL')` can sometimes produce inconsistent results if the depth of the parent field is directly connected to the base order. The level value will be taken from $n-1$ instead of n , where n represents the depth of the parent. For more information, see [SuiteQL Supported Built-in Functions](#).
- To mitigate the possibilities of a CSV injection, the following formatting is applied to all dataset and workbook results that are exported to CSV using SuiteScript and SuiteQL:
 - All text values are enclosed in double quotation marks
 - Values that begin with the characters -, +, =, @, tab, and EOL (end of line) have single apostrophes added in front of the value

SuiteQL Supported and Unsupported Functions

The following sections list the supported and unsupported functions when using SuiteQL.

- [Supported Functions](#)

- [Unsupported Functions](#)

Supported Functions

The following table lists and describes functions you can use when you run queries using SuiteQL.

Supported Function	Description
ABS	returns the absolute value of n
ACOS	returns the arc cosine of n
ADD_MONTHS	returns the date date plus integer months
APPROX_COUNT_DISTINCT	returns the approximate number of rows that contain distinct values of expr
ASCII	returns the decimal representation in the database character set of the first character of char
ASCIIISTR	takes as its argument a string, or an expression that resolves to a string, in any character set and returns an ASCII version of the string in the database character set
ASIN	returns the arc sine of n
ATAN	returns the arc tangent of n
ATAN2	returns the arc tangent of n1 and n2
AVG	returns the average value of expr
BFILENAME	returns a BFILE locator that is associated with a physical LOB binary file on the server file system
BITAND	computes an AND operation on the bits of expr1 and expr2
CEIL	returns smallest integer greater than or equal to n
CHARTOROWID	converts a value from CHAR, VARCHAR2, NCHAR, or NVARCHAR2 datatype to ROWIDdatatype
CHR	returns the character having the binary equivalent to n as a VARCHAR2 value
COALESCE	returns the first non-null expr in the expression list
COMPOSE	takes as its argument a string, or an expression that resolves to a string, in any datatype, and returns a Unicode string in its fully normalized form in the same character set as the input
CONCAT	concatenates char1 and char2 into one string
CORR	returns the coefficient of correlation of a set of number pairs
CORR_K	calculates the Kendall's tau-b correlation coefficient
CORR_S	calculates the Spearman's rho correlation coefficient
COS	returns the cosine of n (an angle expressed in radians)
COSH	returns the hyperbolic cosine of n
COUNT	returns the number of rows returned by the query
COVAR_POP	returns the population covariance of a set of number pairs

Supported Function	Description
COVAR_SAMP	returns the sample covariance of a set of number pairs
CURRENT_DATE	returns the current date in the session time zone, in a value in the Gregorian calendar of datatype DATE
CURRENT_TIMESTAMP	returns the current date and time in the session time zone
DECODE	compares expr to each search value one by one. If expr is equal to a search, then the Oracle database returns the corresponding result
DECOMPOSE	takes as its argument a string in any datatype and returns a Unicode string after decomposition in the same character set as the input
DENSE_RANK	computes the rank of a row in an ordered group of rows and returns the rank as a NUMBER
EMPTY_BLOB	returns an empty LOB locator
EMPTY_CLOB	returns an empty LOB locator
EXP	returns e raised to the nth power
FLOOR	returns largest integer equal to or less than n
FROM_TZ	converts a timestamp value and a time zone to a TIMESTAMP WITH TIME ZONE value
GREATEST	returns the greatest of a list of one or more expressions
INITCAP	returns char, with the first letter of each word in uppercase, all other letters in lowercase
INSTR	searches string for substring
LAST_DAY	returns the date of the last day of the month that contains date
LEAST	returns the least of the list of exprs
LENGTH	returns the length of char
LENGTH2	returns the length of the specified string, using UCS2 code points
LENGTH4	returns the length of the specified string, using UCS4 code points
LENGTHB	returns the length of the specified string, using bytes instead of characters
LENGTHC	returns the length of the specified string, using Unicode complete characters
LN	returns the natural logarithm of n, where n is greater than 0
LOCALTIMESTAMP	returns the current date and time in the session time zone in a value of datatype TIMESTAMP
LOG	computes the logarithm of an expression
LOWER	returns char, with all letters in lowercase
LPAD	returns expr1, left-padded to length n characters with the sequence of characters in expr2
LTRIM	removes from the left end of char all of the characters contained in set
MAX	returns the maximum value of expr

Supported Function	Description
MEDIAN	is an inverse distribution function that assumes a continuous distribution model. It takes a numeric or datetime value and returns the middle value or an interpolated value that would be the middle value once the values are sorted
MIN	returns the minimum value of expr
MOD	returns the remainder of n2 divided by n1. Returns n2 if n1 is 0
MONTHS_BETWEEN	returns the number of months between date1 and date2
NANVL	useful only for floating-point numbers of type BINARY_FLOAT or BINARY_DOUBLE. This function is useful for mapping NaN values to NULL
NEW_TIME	returns the date and time in time zone timezone2 when date and time in time zone timezone1 are date
NEXT_DAY	returns the date of the first weekday named by char that is later than the date
NLSSORT	returns the string of bytes used to sort char
NLS_INITCAP	returns char, with the first letter of each word in uppercase, all other letters in lowercase
NLS_LOWER	returns char, with all letters in lowercase
NLS_UPPER	returns char, with all letters in uppercase
NULLIF	compares expr1 and expr2. If they are equal, then the function returns null. If they are not equal, then the function returns expr1
NVL	lets you replace null (returned as a blank) with a string in the results of a query
NVL2	lets you determine the value returned by a query based on whether a specified expression is null or not null
ORA_HASH	computes a hash value for a given expression
POWER	returns n2 raised to the n1 power
RANK	calculates the rank of a value in a set of values
REGEXP_INSTR	extends the functionality of the INSTR function by letting you search a string for a regular expression pattern
REGEXP_REPLACE	extends the functionality of the REPLACE function by letting you search a string for a regular expression pattern
REGEXP_SUBSTR	extends the functionality of the SUBSTR function by letting you search a string for a regular expression pattern
REMAINDER	returns the remainder of n2 divided by n1
REPLACE	returns char with every occurrence of search_string replaced with replacement_string
ROUND	returns n rounded to integer places to the right of the decimal point
ROW_NUMBER	analytic function that assigns a unique number to each row to which it is applied
RPAD	returns expr1, right-padded to length n characters with expr2, replicated as many times as necessary
RTRIM	removes from the right end of char all of the characters that appear in set

Supported Function	Description
SIGN	returns the sign of n
SIN	returns the sine of n (an angle expressed in radians)
SINH	returns the hyperbolic sine of n
SOUNDEX	returns a character string containing the phonetic representation of char
SQRT	returns the square root of n
SUBSTR	returns a portion of char, beginning at character position, substring_length characters long
SUM	returns the sum of values of expr. You can use it as an aggregate or analytic function
SYS_EXTRACT_UTC	extracts the UTC from a datetime value with time zone offset or time zone region name
TAN	returns the tangent of n (an angle expressed in radians)
TANH	returns the hyperbolic tangent of n
TO_BINARY_DOUBLE	returns a double-precision floating-point number
TO_BINARY_FLOAT	returns a single-precision floating-point number
TO_CHAR	(number) converts n to a value of VARCHAR2 datatype
TO_CLOB	converts NCLOB values in a LOB column or other character strings to CLOB values
TO_DATE	converts char of CHAR, VARCHAR2, NCHAR, or NVARCHAR2 datatype to a value of DATE datatype
TO_MULTI_BYTE	TO_MULTI_BYTE returns char with all of its single-byte characters converted to their corresponding multibyte characters
TO_NCHAR	converts a character string, CHAR, VARCHAR2, CLOB, or NCLOB value to the national character set
TO_NCLOB	converts CLOB values in a LOB column or other character strings to NCLOB values
TO_NUMBER	converts expr to a value of NUMBER datatype
TO_SINGLE_BYTE	returns char with all of its multibyte characters converted to their corresponding single-byte characters
TO_TIMESTAMP	converts char of CHAR, VARCHAR2, NCHAR, or NVARCHAR2 datatype to a value of TIMESTAMP datatype
TO_TIMESTAMP_TZ	converts char of CHAR, VARCHAR2, NCHAR, or NVARCHAR2 datatype to a value of TIMESTAMP WITH TIME ZONE datatype
TRANSLATE	returns expr with all occurrences of each character in from_string replaced by its corresponding character in to_string
TRUNC	(number) returns n1 truncated to n2 decimal places
TZ_OFFSET	returns the time zone offset corresponding to the argument based on the date the statement is executed
UNISTR	takes as its argument a text literal or an expression that resolves to character data and returns it in the national character set

Supported Function	Description
UPPER	returns char, with all letters in uppercase
VSIZE	returns the number of bytes in the internal representation of expr
WIDTH_BUCKET	lets you construct equiwidth histograms, in which the histogram range is divided into intervals that have identical size

For a list of supported built-in functions, see [SuiteQL Supported Built-in Functions](#).

Unsupported Functions

The following table provides a list of functions that are not supported when you run queries using SuiteQL. It also provides an alternative function you can use for each unsupported function, if available.

Unsupported Function	Alternative Function (If Available)
BIT_LENGTH	—
CEILING	CEIL
CHAR	—
CHARINDEX	INSTR
CHAR_LENGTH	LENGTH
CHARACTER_LENGTH	LENGTH
CONVERT	—
COT	—
DATEDIFF	—
LCASE	LOWER
LEFT	—
LOCATE	INSTR
POSITION	—
REPEAT	—
RIGHT	—
SUBSTRING	SUBSTR
UCASE	UPPER

SuiteQL Supported Built-in Functions

You can use built-in functions to perform certain operations in SuiteQL queries. These functions extend the capabilities that are provided by the SQL-92 specification. For example, you can use the CONSOLIDATE built-in function to convert a currency amount stored in a field to a target currency.

To use a built-in function, you must add BUILTIN. before the function name. For example, to call the CONSOLIDATE built-in function, you must use BUILTIN.CONSOLIDATE in your SuiteQL query.

The following table describes the built-in functions that are supported in SuiteQL and provides examples of each function. Some parameters are optional and are documented as such in the table, but all other parameters are required.



Note: Some parameters include descriptions that apply to the N/query module only.

Function	Description	Parameters	SuiteQL Examples
CF	Sets the field usage context to CRITERIA	<ul style="list-style-type: none"> ■ Field name 	<pre>1 BUILTIN.CF(sales.item)</pre>
CONSOLIDATE	Converts a currency amount stored in a field to a target currency	<ul style="list-style-type: none"> ■ Original amount field name ■ View type ■ Consolidation rate type ■ Subsidiary rate type ■ Target subsidiary ■ Period ■ Book <p>To use the default value for any parameter (except original amount field name and view type), you can specify 'DEFAULT' as the parameter value.</p>	<pre>1 BUILTIN.CONSOLIDATE(sales.amount, 'INCOME', 'DEFAULT', 'DEFAULT', 1, NUMBER, 'DEFAULT')</pre>
CURRENCY	Returns the currency code	<ul style="list-style-type: none"> ■ Amount <p>This parameter value can be any of the following types:</p> <ul style="list-style-type: none"> ■ amount ■ SUM(amount) ■ BUILTIN.CONsolidate ■ SUM(BUILTIN.CONsolidate) ■ BUILTIN.CURRENCY_CONVERT ■ SUM(BUILTIN.CURRENCY_CONVERT) 	<pre>1 BUILTIN.CURRENCY(sales.amount)</pre> <pre>1 BUILTIN.CURRENCY(2 BUILTIN.CONSOLIDATE(3 sales.amount, 'INCOME', 4 'DEFAULT', 'DEFAULT', 1, 5 304, 'DEFAULT' 6) 7)</pre>
CURRENCY_CONVERT	Converts a currency amount stored in a field to a target currency using the exchange rate that was in effect on a specific date	<ul style="list-style-type: none"> ■ Original amount field name ■ Target currency (optional; the default value is the default currency of the subsidiary) ■ Date of the exchange rate to use (optional; the default value is today's date) 	<pre>1 BUILTIN.CURRENCY_CONVERT(sales.amount)</pre> <pre>1 BUILTIN.CURRENCY_CONVERT(sales.amount, 1)</pre> <pre>1 BUILTIN.CURRENCY_CONVERT(sales.amount, 1, 2 TO_DATE('2019-01-30', 3 'YYYY-MM-DD'))</pre>
DF	Returns the display value of a field from the target record type without having to join the target record type explicitly	<ul style="list-style-type: none"> ■ Field name <p>This field name represents the relationship between the field and the source record type (for example, sales.item).</p>	<pre>1 BUILTIN.DF(sales.item)</pre>
HIERARCHY	Returns the full hierarchical path to a value	<ul style="list-style-type: none"> ■ Hierarchical field ■ Expansion type <p>This parameter accepts the following values (including the single quotation marks):</p>	<pre>1 BUILTIN.HIERARCHY(parent, 'IDENTIFIER_SEPARATED')</pre>

Function	Description	Parameters	SuiteQL Examples
		<ul style="list-style-type: none"> □ 'DISPLAY' □ 'DISPLAY_JOINED' □ 'DISPLAY_SEPARATED' □ 'IDENTIFIER' □ 'IDENTIFIER_SEPARATED' □ 'LEVEL' □ 'SELF_DISPLAY' □ 'SELF_IDENTIFIER' <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> i Note: Hierarchy level can sometimes return varying results. For more information, see SuiteQL Limitations and Exceptions. </div>	
NAMED_GROUP	Returns filter options for workgroups for some record types (department, class, location, subsidiary, and entity)	<ul style="list-style-type: none"> ■ Record type ■ Workgroup <p>This parameter accepts the following value (including the single quotation marks):</p> <ul style="list-style-type: none"> □ 'me' 	<pre>1 BUILTIN.NAMED_GROUP('employee', 'me')</pre>
PERIOD	Returns the contents of the IN predicate for a relative date range as a subselection	<ul style="list-style-type: none"> ■ Range ID <p>This parameter accepts the same property values listed in the query.RelativeDateRange enum in the N/query module (for example, 'LFY' to represent a range starting or ending last fiscal year).</p> <ul style="list-style-type: none"> ■ Range type <p>This parameter accepts the following values (including the single quotation marks):</p> <ul style="list-style-type: none"> □ 'START' □ 'END' <ul style="list-style-type: none"> ■ Adjustment <p>This parameter accepts the following values (including the single quotation marks):</p> <ul style="list-style-type: none"> □ 'NOT_LAST' □ 'ALL' <ul style="list-style-type: none"> ■ Operator <p>This parameter accepts the following values (including the single quotation marks):</p> <ul style="list-style-type: none"> □ 'BETWEEN' □ 'NOT BETWEEN' □ '<' □ '<=' □ '>' □ '>=' 	<pre>1 BUILTIN.PERIOD('LFY', 'END', 'NOT_LAST', '>')</pre> <pre>1 BUILTIN.PERIOD('LFY', 'START', 'ALL', 'BETWEEN')</pre>

Function	Description	Parameters	SuiteQL Examples
RELATIVE_RANGES	Returns dynamic calendar ranges for the filter options that apply to relative date fields (such as Last Fiscal Year and Current Week)	<ul style="list-style-type: none"> ■ Range ID This parameter accepts the same property values listed in the query.RelativeDateRange enum in the N/query module (for example, 'LFYTD' to represent a range starting or ending last fiscal year to date). ■ Range type This parameter accepts the following values (including the single quotation marks): <ul style="list-style-type: none"> □ 'START' □ 'END' ■ Date flag (optional) This parameter accepts the following values (including the single quotation marks): <ul style="list-style-type: none"> □ 'DATE' □ 'DATETIME_AS_DATE' □ 'DATE_FROM_GMT_TO_USER_TZ' □ 'DATE_TO_USER_TZ' □ 'ODBC_DATE_TO_GMT' □ 'ODBC_TIMESTAMP_TO_GMT' □ 'ODBC_TIMESTAMP_TO_GMT_AS_DATE' □ 'TIMESTAMP_WITH_TIMEZONE' □ 'TIMESTAMP_WITH_TIMEZONE_AS_DATE' 	<pre>1 BUILTIN.RELATIVE_RANGES('LFYTD', 'START')</pre> <pre>1 BUILTIN.RELATIVE_RANGES('LFYTD', 'END', 'DATETIME_AS_DATE')</pre>

Custom Workbooks and Datasets

In SuiteAnalytics Workbook, you analyze your company data using two distinct objects: a dataset and a workbook.

Datasets are the basis for all workbooks and workbook visualizations in your account. In a dataset, you combine record type fields and criteria filters to create a query. The results of this query act as the source data for your workbook visualizations. A single dataset can be used in multiple workbooks and workbook visualizations, so to prevent data discrepancies they can only be edited or deleted by dataset owners or users with the Analytics Administrator permission. Additionally, changes that you make to a dataset are automatically propagated to the workbook visualizations that the dataset is used in. This reduces the need for multiple dataset objects in your account and enables less experienced users to create complex workbooks.

Workbooks are where you analyze the results of your dataset queries using different visualizations, such as tables, pivot tables, and charts. All workbook visualizations are based on a dataset or a set of linked datasets, and you can use different datasets for each visualization in a workbook. You can access the dataset connected to a workbook visualization from the Dataset Panel. Moreover, when you create a new workbook visualization, you can base it on a new custom dataset or any of the existing datasets that you have access to in your account. You can also link two datasets and use them as the basis for your workbook visualizations. This enables you to compare metrics from two datasets in a single visualization.

To better understand the functionality and relationship between datasets and workbooks, see the following table:

Functionality	Dataset	Workbook
Add fields and join record types?	<p>Yes</p> <p>You define the source data for your workbook visualizations by adding fields and joining record types in a dataset query.</p>	<p>—</p> <p>If you want to add fields or join record types to a workbook visualization, you must edit the underlying dataset.</p>
Create visualizations?	<p>—</p> <p>The dataset is only where you query your business data. To analyze the results, create a workbook visualization based on the dataset.</p>	<p>Yes</p> <p>The workbook is where you analyze your dataset query results using different visualizations such as tables, pivot tables, and charts. Each visualization in a workbook can be based on a different dataset.</p>
Automatically propagate edits to other workbook visualizations?	<p>Yes</p> <p>A single dataset can be used in multiple workbooks and workbook visualizations. To prevent data discrepancies, edits that you make to a dataset are automatically propagated to workbook visualizations that are based on that dataset.</p>	<p>—</p> <p>Workbooks and workbook visualizations in your account exist independently of one another. If you edit a workbook visualization, your changes apply only to the visualization you are in.</p>
Currency conversion or consolidation?	<p>Yes, but only with custom formula fields.</p> <p>Prior to 20.1, currency conversion or consolidation was applicable from the Data Grid in the Dataset Builder. As of 20.1, you can only apply currency conversion and consolidation to a field from within a workbook visualization unless you create a custom formula field. For more information, see Currency Conversion Using Custom Formula Fields.</p>	<p>Yes</p> <p>You can convert and consolidate field values to a specific currency directly from the user interface in any workbook visualization. For more information, see Currency Conversion from the User Interface.</p>

Sharing?	Yes You can share a dataset with other users in your account as its own object, or as part of a shared workbook. Recipients of a shared workbook cannot edit the underlying datasets but can edit and save their own versions of the datasets using different names. For more information, see Accessing and Sharing Workbooks and Datasets .	Yes Shared workbooks include all workbook visualizations and any underlying datasets.
Translatable text?	Yes You can translate the name, description, and fields of a dataset. For more information, see the help topic Manage Translations .	Yes You can translate the name, description, and fields of the workbook and visualizations. For more information, see the help topic Manage Translations .
Conditional formatting?	— You can sort and filter dataset results but you cannot apply conditional formatting to them. Use workbook table views and pivot tables to highlight results.	Yes You can highlight or add icons to your table view and pivot table results based on rules that you define. For more information, see Conditional Formatting .
Dataset linking?	—	Yes. Link definitions exist as part of the workbook that they are created in. This means you can use linked datasets as the basis for multiple visualizations, but only in the workbook where the link was defined. For more information, see Dataset Linking in SuiteAnalytics Workbook .

- For information about how to author a dataset, including steps for how to join record types and fields or create custom formula fields, see [Defining a Dataset](#).
- After you define your dataset, you can filter the results by creating criteria filters. Criteria filters can be based on existing values in the dataset results, or custom values that you define. For more information, see [Dataset Criteria Filters](#).
- For information about creating a workbook based on a dataset, see [Creating a Workbook](#). This section includes steps for setting up and customizing the different visualizations that you can include in a workbook, and how to access the underlying datasets.
- To learn about the difference between formula fields and calculated measures, see [Formula Fields and Calculated Measures](#).
- You can link two datasets as part of your workbooks, enabling you to analyze metrics from both datasets in either a pivot table or chart. For more information, see [Link Datasets in a Workbook](#).

Defining a Dataset

Datasets are the basis for all workbook visualizations in your account. You can create a new dataset from the Analytics Home page or from the Dataset Panel within a workbook visualization.

In a dataset, you combine the fields of a root record type and any joined related record types to create a query. The record types and fields that you have access to are based on the features enabled in your account and the permissions assigned to the role you use to log into NetSuite. Related record types that you can join to a dataset are also based on predefined common keys in the analytics data source. If

standard record types do not contain the values you need for your analysis, you can also create custom formula fields as part of your dataset.

In the Dataset Builder, the results of your query are displayed in a tabular format in the Data Grid on the right. When you first create a new dataset, the Data Grid is automatically populated with fields based on the root record type. Joinable record types and fields that you can add to your query appear in the Records and Fields lists on the left. Only fields that you add to the Data Grid can be used to build workbook visualizations such as tables, pivot tables, and charts.

- To view a record type's fields, click the name of the record type to refresh the Fields list.
- To view additional related record types, click the arrow next to any record type in the Records list.
- To join a record type to a dataset, add fields from the record type to the Data Grid or use the field to create a criteria filter. For more information about dataset criteria filters, see [Dataset Criteria Filters](#).
- In workbook visualizations based on linked datasets, to view the fields within each dataset click the dataset name in the Dataset Panel.

You can join any number of record types in a dataset, however more than ten joined record types can negatively impact performance.

For more information about joining record types in a dataset, see [Guidelines for Joining Record Types in SuiteAnalytics Workbook](#).

For information about access to specific record types and fields in NetSuite, download the following worksheet: [NetSuitePermissionsUsage.xls](#).

For information about editing a dataset that is used in a workbook visualization, see [Editing a Dataset](#).

To define a new dataset:

1. Click **New Dataset** on the Dataset subtab of the Analytics Home page or on the **Select a dataset to create a new workbook** page.

Alternatively, click **Connect Dataset** in the Dataset Panel of a workbook visualization, then click **New Dataset** on the **Select a dataset** page.

2. Select a root record type for the dataset. The record type you choose will determine the record types and fields that you can join to the dataset.
3. Add fields from the root record type to the Data Grid. You have three options:
 - Drag the desired fields from the Fields list to the Data Grid.
 - Double-click the desired field names in the Fields list.
 - Type the name of the desired fields in the search bar above the Fields list, then drag or double-click them to add them to the Data Grid.
4. Add fields from related record types to the Data Grid.
 - a. Click the arrow next to any record type in the Records list to view additional related record types.
 - b. Click the desired record type name to update the Fields list.
 - c. Double-click or drag the desired fields to the Data Grid.



Note: In Workbook, certain record types have multiple variants. For example, there are entity job, entity customer, and entity vendor record types accessible in a transaction dataset. Make sure you select the correct record type depending on the type of data you want to query.

5. (Optional) Click **Formulas** to view, create, and add custom formula fields to the Data Grid.

For step-by-step instructions on how to create a custom formula field, see [Formula Fields](#).

6. Remove any fields you do not want to include in the dataset.
 - a. Click the Field Menu icon  in the column you want to remove from the Data Grid.
 - b. Select **Remove Column** from the dropdown list.



Important: If you create a workbook visualization based on the current dataset, removing a field from the Data Grid also removes it from any other workbook visualizations that are based on the dataset. This can cause workbook failures or other unwanted results. Exercise caution when you remove a field from datasets that are used in multiple workbook visualizations.

7. Click **Save** to save the dataset. Alternatively, if you are creating the dataset from within a new workbook, click **Save & Close** to save the dataset and access the workbook.

The Save Dataset As window appears.

8. Enter a name and description for the dataset in the fields provided, then click **Save**.
9. (Optional) To preview your dataset in a new workbook visualization before saving your changes, click **Create New Workbook** or **Apply to workbook**. The workbook will appear.



Note: You must save the dataset before you can save the connected workbook. Unsaved datasets are denoted by a warning icon  in the Dataset Panel of a workbook. For more information, see [Editing a Dataset](#).

10. (Optional) To export your dataset results to a CSV file, click the Export icon .



Important: To mitigate the possibility of a CSV injection, dataset results exported to CSV may have additional formatting applied. For more information, see [Known Limitations in SuiteAnalytics Workbook](#).

Editing a Dataset

Only a dataset owner or users with the Analytics Administrator permission can save changes to an existing dataset. This prevents data discrepancies among workbook visualizations that share the same dataset. Users with whom a dataset is shared can edit the dataset but must save it under a new name. Additionally, all users with access to a dataset can edit it and preview their changes in a workbook to see how they will affect their workbook visualizations. This enables you to create a single dataset for multiple users with different use cases.



Note: In workbooks connected to linked datasets, use caution when removing fields from a dataset. If you delete a field that was used to define the link, it can impact your results and prevent you from saving.



Note: If you have multiple languages in your account and you edit and save default text in a dataset, a new translation collection is added to the Manage Translations page. For more information, see the help topic [Manage Translations](#).

To edit a dataset that you own:

1. Click the dataset or workbook name on the Analytics Home page. To quickly filter the available options, select **My Workbooks** or **My Datasets** in the dropdown list of either the Workbooks or Datasets subtabs.

2. Edit the dataset. For information, see [Defining a Dataset](#).
3. Click **Save**. Alternatively, if you accessed the dataset through a connected workbook, click **Save & Close** to save your changes and open the workbook.
4. (Optional) Click **Create New Workbook** or **Apply to workbook** to preview your changes in a workbook visualization before saving the dataset.

The workbook will appear with a warning icon  next to the dataset name, denoting an unsaved dataset.

5. If you are satisfied with your changes, click the arrow next to the dataset name and select **Save Changes**.

If you are not satisfied, click the arrow and select **Open Dataset** to continue editing the dataset, or **Discard Changes**.



Important: Remember that changes you save to your datasets are automatically propagated to any workbook visualizations based on the dataset, including visualizations created by other users in other workbooks. This can drastically change the associated workbook visualizations so if you are not sure about the impact of your changes, consider saving the dataset using a different name.

To edit a dataset that has been shared with you:

1. Click the dataset or workbook name on the Analytics Home page. To quickly filter the available options, select **Shared with me** in the dropdown list of either the Workbooks or Datasets subtabs.

2. Edit the dataset. For more information, see [Defining a Dataset](#).

3. Click **Save As**.

The Save Dataset As window appears.

4. Enter a name and description for the dataset in the fields provided, then click **Save**.

5. (Optional) Click **Create New Workbook** or **Apply to workbook** to preview your changes in a workbook.

The workbook will appear with a warning icon  next to the dataset name, denoting an unsaved dataset.

6. If you are satisfied with your changes, click the arrow next to the dataset name and select **Open Dataset** to save it using a new name.

Formula Fields

In SuiteAnalytics Workbook, you can create custom formula fields to calculate values that are not available through standard record fields. You can add formula fields to your datasets on the Data Grid and use them to define criteria filters. You can also use them in any workbook visualizations that are based on the dataset. If you try to link two datasets but there are no fields with matching data types, you can also use formulas to cast the field values to the appropriate types.

The values of custom formula fields are updated each time you refresh the dataset or the connected workbook visualizations. Currently, you cannot create custom formula fields based on existing formula fields.

To create a custom formula field, you combine NetSuite fields and SQL formula functions in a formula definition. You can also customize some fields by using formula context values such as #converted. When you define a custom formula field, you must also select an output type for the formula, depending on the fields, functions, and context values you include in your definition. Workbook supports the following formula output types:

- Boolean- Returns True, False, or NULL values
- Date- Returns day, month, and year values
- Datetime- Returns day, month, year, hour, minute, and second values
- Float- Returns values that include a decimal, such as currency values
- Integer- Returns values that are in whole numbers only, such count values
- String- Returns fixed string character values, such as names
- Clobtext- Returns long unicode character strings, such as item descriptions
- Percent- Returns numeric values as percentages
- Duration- Returns hour and minute values, such as hours worked

Knowledge of SQL will help you to fully leverage the supported SQL functions when you are defining a new formula. To read a description of each SQL function currently supported in the formula builder, click the Functions subtab of the Formula Field window. From the Functions subtab, you can also add a sample formula definition for each of the listed functions. Use this feature if you are not sure about which function to use in your formula definitions.

For more information about SQL formula functions, see the help topic [SQL Expressions](#).

For more information about how to use formula fields, see [Creating Formula Fields in SuiteAnalytics Workbook](#).

For sample formulas you can create, see [Sample Formulas](#).

In your workbooks, you can also create calculated measures that you can use in your pivot tables and charts. However, using formula fields or calculated measures has an impact on the results that you obtain. To understand the difference between formula fields and calculated measures, see [Formula Fields and Calculated Measures](#).

Creating Formula Fields in SuiteAnalytics Workbook

You can customize formula fields by using SQL functions and context values. After you create formula fields, you can use them in your datasets and workbook visualizations. When you are working with custom formula fields, consider the following:

- [Creating a Custom Formula Field in SuiteAnalytics Workbook](#)
- [Creating Criteria Filters Using Formula Fields](#)
- [Using Context Values in Formula Fields](#)
- [Adding Formula Fields to a Workbook Visualization](#)

Creating a Custom Formula Field in SuiteAnalytics Workbook

In the following example, you create a formula field that converts date values to a different format. Remember that the Formula Field window includes only fields from the root record type on the dataset and any fields from related record types that have been added to the Data Grid.



Important: To perform arithmetic calculations or view totals for fields that contain values in multiple currencies, you must first convert or consolidate the values to a specific currency. In the Dataset Builder, you can only do this using custom formula fields. For more information, see [Currency Conversion Using Custom Formula Fields](#).

To create a custom formula field in SuiteAnalytics Workbook:

1. Click **Formulas** above the Records list on the Dataset Builder.
2. In the Fields list, click **New Formula**.

The Formula Field window appears.

FORMULA FIELD NAME* **50**

CHOOSE OUTPUT TYPE* **INTEGER**

FORMULA*

1 Type in your formula

Need help with defining the formula? **Validate**

Fields **Functions**

CHOOSE SOURCE

Show all available fields

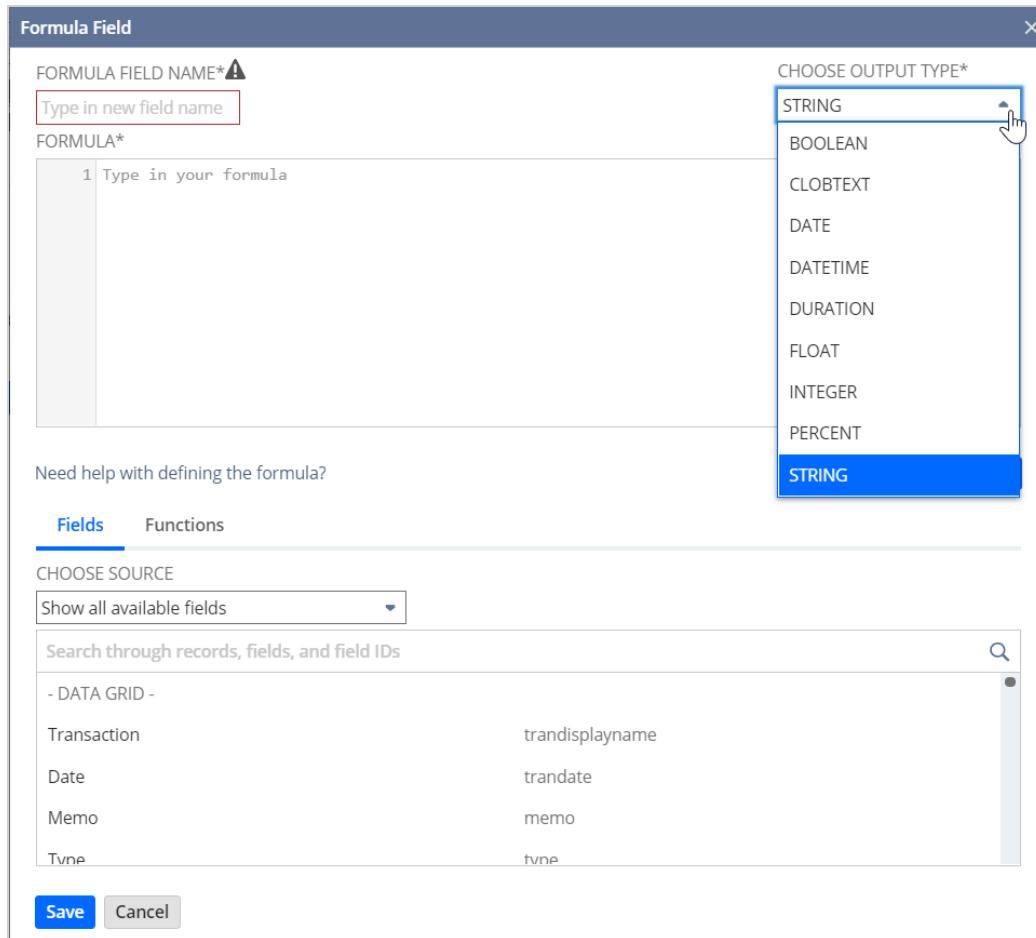
Search through records, fields, and field IDs

- DATA GRID -

Transaction	trdisplayname
Date	trandate
Memo	memo
Tvinc	tvinc

Save **Cancel**

3. Enter a Formula Field Name.
For this example, enter **Formatted Date**.
4. Select an Output Type for the formula field values.
For the purposes of this procedure, select **String**.



Note: Each output type only works with certain fields and formula functions. If you select an incompatible output type for the formula, the formula is invalidated and the Formula Builder highlights the error in the definition. To view details about the validation error and for information about how to correct the definition, click the error icon .

5. In the Formula field, enter the field IDs and SQL formula functions to use in the formula expression. Alternatively, double-click the desired field IDs or formula functions from the Functions and Fields subtabs to add them to the expression.

For the purposes of this procedure, select the TO_CHAR function and replace the content in the brackets with **{trandate}, 'Month DD, YYYY'**.

FORMULA FIELD NAME*

Formatted Date

CHOOSE OUTPUT TYPE*

STRING

FORMULA*

```
1 TO_CHAR({trandate}, 'Month DD, YYYY')
```

Need help with defining the formula? [Validate](#)

[Fields](#) [Functions](#)

CHOOSE SOURCE

Show all available fields

Search through records, fields, and field IDs

- DATA GRID -

Transaction	trandisplayname
Date	trandate
Memo	memo
Type	type

[Save](#) [Cancel](#)

- To validate the formula, click **Validate**.



Note: If there are errors with the formula, click the error icon to view the details.

- Click **Apply** to add the formula field to the dataset.

The field appears in the Field lists on the Dataset Builder when you click **Formulas**. If you add the field to the Data Grid, it is also available for use in any workbook visualizations that are based on the dataset.

Creating Criteria Filters Using Formula Fields

In the following example, you use the Criteria Builder to create a filter using a formula field.

To create criteria filters using formula fields:

- Create a valid formula field on the Dataset Builder.
- Double-click or drag the field from the Fields list or the Data Grid to the Criteria Builder.
- Set the desired filter conditions in the Filter window. For details, see [Filter Types](#).

Using Context Values in Formula Fields

You can use different context values in some fields to define how values are displayed. For example, you can specify that a field displays results by showing the name of the field values instead of the ID.

The following example shows a formula definition where *type* is the field and *display* is the context value that defines how results are displayed. Context values are placed after the number sign # within the same parentheses containing the field name in a formula definition.

Example: {type#display}

Before you start working with context values, consider the following:

- [Supported Context Values](#)
- [Editing Context Values](#)

Supported Context Values

SuiteAnalytics Workbook supports the following context values:

Context Value	Description
#currency_consolidated	<p>Displays consolidated currency amounts in the currency of the lowest level subsidiary that is a common parent to all the subsidiaries that you have access to in your account. The exchange rate used for the consolidation is the rate that is in effect on the date when the query or workbook visualization is run. For more information, see Currency in Datasets and Workbooks.</p> <p>Formula example:</p> <ul style="list-style-type: none"> ■ ID of the field used for this example: amount ■ Formula output type: FLOAT ■ SQL function required: TO_NUMBER() ■ Formula definition: TO_NUMBER({amount#currency_consolidated})
#converted	<p>Displays converted currency amounts in the base currency of your account. The exchange rate used for the conversion is the rate that is in effect on the date when the query or workbook visualization is run. For more information, see Currency in Datasets and Workbooks.</p> <p>Formula example:</p> <ul style="list-style-type: none"> ■ ID of the field used for this example: amount ■ Formula output type: FLOAT ■ SQL function required: TO_NUMBER() ■ Formula definition: TO_NUMBER({amount#converted})
#converted[1]	<p>Displays the currency amounts in the currency ID specified in the square brackets [1]. The exchange rate used for the conversion is the rate that is in effect on the date when the query or workbook visualization is run. For more information, see Currency in Datasets and Workbooks.</p> <p>Note: The currency ID 1 corresponds to the base currency.</p> <p>Formula example:</p> <ul style="list-style-type: none"> ■ ID of the field used this example: amount ■ Formula output type: FLOAT ■ SQL function required: TO_NUMBER() ■ Formula definition: TO_NUMBER({amount#converted[1]})
#converted[1;yyyy-mm-dd]	<p>Displays the currency amounts in the currency specified in the square brackets [1], using the exchange rate that is in effect on a specific date. You must enter the date using the YYYY-MM-DD format. By default, the currency ID is set to 1. For more information, see Currency in Datasets and Workbooks.</p>

Context Value	Description
	<p>Note: The currency ID 1 corresponds to the base currency.</p> <p>Formula example:</p> <ul style="list-style-type: none"> ■ ID of the field used for this example: amount ■ Formula output type: FLOAT ■ SQL function required: TO_NUMBER() ■ Formula definition: TO_NUMBER({amount#converted[1;2020-03-10]})
#display	<p>Displays more user-friendly field values. For example, for the type field on the transaction record type, the transaction type is displayed, such as bill, cash refund, and inventory adjustment.</p> <p>Formula example:</p> <ul style="list-style-type: none"> ■ ID of the field used for this example: type ■ Formula output type: INTEGER ■ SQL function required: none ■ Formula definition: {type#display}
#raw	<p>Displays raw field values (such as internal IDs). For example, for the type field on the transaction record type, using the raw value displays the transaction type ID.</p> <p>Note: The ID is a unique identifier that cannot be changed. If the name of a field changes, you can identify the field by getting the ID instead of the name. For example, the result is CashRfnd instead of cash refund.</p>
	<p>Formula example:</p> <ul style="list-style-type: none"> ■ ID of the field used for this example: type ■ Formula output type: INTEGER ■ SQL function required: none ■ Formula definition: {type#raw}
#hierarchy	<p>Displays more user-friendly field values for hierarchical fields. This context value is similar to #display but applies to hierarchical fields.</p>
	<p>Formula example:</p> <ul style="list-style-type: none"> ■ ID of the field used for this example: subsidiary ■ Formula output type: STRING ■ SQL function required: none ■ Formula definition: {subsidiary#hierarchy}
#hierarchy_identifier	<p>Displays more user-friendly field values for hierarchical fields. This value is similar to #display but applies to hierarchical fields</p> <p>Formula example:</p> <ul style="list-style-type: none"> ■ ID of the field used for this example: subsidiary ■ Formula output type: FLOAT ■ SQL function required: TO_NUMBER() ■ Formula definition: TO_NUMBER({subsidiary#hierarchy_identifier})

Editing Context Values

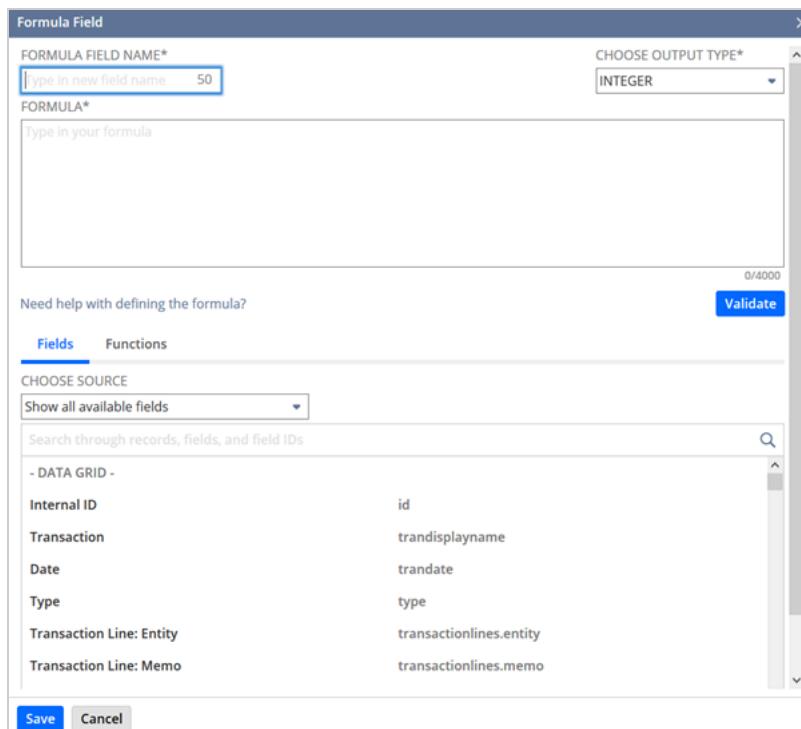
In the following example, you can change the context value of the type field. By default, the type field displays results by showing the name. If you change the context value, the field displays results by showing the ID.



Note: This example is applicable for any record type that includes the type field. The following example is the transaction record type, available in the Transaction Detail dataset template.

To edit the context value in a formula

1. Click **Formulas** above the Records list on the Dataset Builder.
2. In the Fields list, click **New Formula**.



The Formula Field window appears.

3. Enter a Formula Field Name.
For example, enter **Type (ID)**.
4. Select an Output Type for the formula field values.
For the purposes of this procedure, select **String**.
5. Double-click the desired field from the Fields subtabs to add them to the FORMULA text box.
For the purposes of this procedure, double-click Type.
The formula appears in the **Formula** field. In this example, `{type#display}`.
6. In the **Formula** field, replace the content after the number sign with the value that you want to add.
For the purposes of this procedure, remove `#display` and enter `#raw`.
7. To validate the formula, click **Validate**.



Note: If there are errors with the formula, click the error icon (warning) to view the details.

8. Click **Apply** to add the formula field to the dataset.

The field appears in the Field lists on the Dataset Builder when you click **Formulas**. If you add the field to the Data Grid, it is also available for use in any workbook visualizations based on the dataset. When you use this field in workbook visualizations, the type field displays the ID of the values.

Adding Formula Fields to a Workbook Visualization

In the following example, you add a formula field to a workbook and then place it on a table, pivot, or chart of the workbook.

To add formula fields to a workbook visualization:

1. Create a valid formula field on the Dataset Builder.
2. Add the field to the Data Grid so that it is included in any workbook visualizations based on the dataset.
3. On the Table, Pivot, or Chart tabs of the workbook, drag the field from the Dataset Panel to the desired section of the Layout panel.

Sample Formulas

The SuiteAnalytics Workbook Formula Builder enables you to create custom formula fields using any of the formula functions listed in the [SuiteAnalyticsWorkbookFormulaFunctions.xls](#) spreadsheet. You can add sample formulas for any of these functions directly from the Functions subtab in the Formula Builder using the **Add the example to the formula** link. See the following sections for additional sample formulas that you can create using some of the available functions:

- Calculating Duration Values with TO_NUMBER and TO_NCHAR
- Combining CONCAT and other Functions to Calculate String Values
- Casting Field Values using TO_NUMBER and TO_NCHAR
- Currency Consolidation and Conversion Using Custom Formula Fields
- Conditional Evaluations Using CASE WHEN
- Calculating Amounts for Relative Date Ranges
- Calculating Specific Dates

For more sample formulas, see the help topic [Search Formula Examples and Tips](#).

For information about using custom formula fields for currency conversion or consolidation, see [Currency Conversion Using Custom Formula Fields](#).

Calculating Duration Values with TO_NUMBER and TO_NCHAR

Duration values are incompatible with the TO_NUMBER function unless they are first converted to a variable format using the TO_NCHAR function. Consequently, if you want to create a custom formula field that displays duration values as a number, you must use the following formula definition:

1 | TO_NUMBER(TO_NCHAR({field_ID}))

For example, assume you want to see the amount of time that has elapsed for each case in your account, in hours.

CREATION DATE/TIME	TIME ELAPSED	PRIORITY	ASSIGNED TO	STATUS	SUBJECT
6/4/2010 12:33 pm	83649.00	Medium	Cameron Murdock	Re-Opened	Access to Mana
6/16/2010 5:27 pm	1774.00	Medium	Edwin Goldwasser	Closed	Additional Trai
8/5/2010 12:55 pm	578.00	Medium	Aubrey Pober	Closed	Auctions going
11/10/2010 11:24 am	81369.00	Medium	German Support Rep	Not Started	Auftrag nicht er
9/3/2010 12:37 pm	81752.00	Medium	suite analytics	Escalated	Auto Update
11/10/2010 11:29 am	81369.00	Medium	German Support Rep	Escalated	Can't addieren
7/20/2010 11:31 am	82834.00	High	suite analytics	Escalated	Can't import pr
6/17/2010 6:25 am	1186.00	Medium	Cameron Murdock	Closed	Changing Your
6/18/2010 1:50 pm	1653.00	Medium	suite analytics	Closed	Config Problem
6/29/2010 11:26 am	1513.00	High	suite analytics	Closed	Data Configura
6/17/2010 6:21 am	83367.00	Medium	Edwin Goldwasser	Escalated	Enhancement t

In this case, use the following formula definition with an output type of Float:

1 | TO_NUMBER(TO_NCHAR({timeelapsed}))

Formula Field

FORMULA FIELD NAME*
Time Elapsed (Hours)

CHOOSE OUTPUT TYPE*
FLOAT

FORMULA*
TO_NUMBER(TO_NCHAR({timeelapsed}))

Need help with defining the formula? [Validate](#)

Fields Functions

CHOOSE SOURCE
Show all available fields

Search through records, fields, and field IDs

- DATA GRID -

Creation Date/Time	datecreated
Time Elapsed	timeelapsed
Priority	priority
Assigned To	assigned
Status	status
Subject	title
Company	company
Type	category

[Save](#) [Cancel](#)

	CREATION DATE/TIME	TIME ELAPSED	TIME ELAPSED (HOURS)	PRIORITY	ASSIGNED TO	S1
1	6/4/2010 12:33 pm	83663.00	83,663.00	Medium	Cameron Murdock	R
2	6/6/2010 5:27 pm	1774:00	1,774.00	Medium	Edwin Goldwasser	C
3	8/5/2010 12:55 pm	578:00	578.00		Aubrey Pober	C
4	11/10/2010 11:24 am	81383:00	81,383.00	Medium	German Support Rep	N
5	9/3/2010 12:37 pm	81766:00	81,766.00	Medium	suite analytics	E
6	11/10/2010 11:29 am	81383:00	81,383.00	Medium	German Support Rep	E
7	7/20/2010 11:31 am	82848:00	82,848.00	High	suite analytics	E
8	6/17/2010 6:25 am	1186:00	1,186.00	Medium	Cameron Murdock	C
9	6/8/2010 1:50 pm	1653:00	1,653.00		suite analytics	C
10	6/29/2010 11:26 am	1513:00	1,513.00	High	suite analytics	C
11	6/17/2010 6:21 am	83381:00	83,381.00	Medium	Edwin Goldwasser	E

Combining CONCAT and other Functions to Calculate String Values

You can combine the CONCAT function with other functions to calculate complex string values. For example, if you want to view the duration values for all the cases in your account in days and hours, you must normalize the values with TO_NUMBER and TO_NCHAR, then concatenate them using the following definition. The output type for this formula definition is STRING:

```

1 | CONCAT(CONCAT('Days: ',TO_NCHAR(FLOOR(TO_NUMBER(TO_NCHAR({timeelapsed}/24)))),),
2 | CONCAT(' Hours: ',TO_NCHAR(MOD(TO_NUMBER(TO_NCHAR({timeelapsed})),24))))

```

CREATION DATE/TIME	TIME ELAPSED	TIME ELAPSED (DAYS AND HOURS)	PRIORITY	ASSIGNED TO
6/4/2010 12:33 pm	83664.00	Days: 3486 Hours: 0	Medium	Cameron Murdock
6/6/2010 5:27 pm	1774.00	Days: 73 Hours: 22	Medium	Edwin Goldwasser
8/5/2010 12:55 pm	578.00	Days: 24 Hours: 2		Aubrey Poer
11/10/2010 11:24 am	81384.00	Days: 3391 Hours: 0	Medium	German Support Rep
9/3/2010 1:37 pm	81767.00	Days: 3406 Hours: 23	Medium	suite analytics
11/10/2010 11:29 am	81384.00	Days: 3391 Hours: 0	Medium	German Support Rep
7/20/2010 11:31 am	82849.00	Days: 3452 Hours: 1	High	suite analytics
6/17/2010 6:25 am	1186.00	Days: 49 Hours: 10	Medium	Cameron Murdock
6/18/2010 1:50 pm	1653.00	Days: 68 Hours: 21		suite analytics
6/29/2010 11:26 am	1513.00	Days: 63 Hours: 1	High	suite analytics
6/17/2010 6:21 am	83382.00	Days: 3474 Hours: 6	Medium	Edwin Goldwasser

Try combining different formula functions with the CONCAT function to calculate different string values.

Casting Field Values using TO_NUMBER and TO_NCHAR

Fields in NetSuite store values in different data types, such as STRING, INTEGER, and FLOAT.

- String- text values, such as a customer name
- Integer- positive or negative whole numbers that do not contain a decimal, such as counts
- Float- numeric values that can contain a decimal, such as amounts

Some fields need to be cast to a different data type before you can use them in a custom formula field, or before you can perform arithmetic operations with them. For example, if item price was stored as an INTEGER value and item cost was stored as a FLOAT value, you could use the following formula definition to calculate the difference between the two:

```
1 | TO_NUMBER({item<pricing.unitprice}) - TO_NUMBER({cost})
```

Formula Field

FORMULA FIELD NAME*
Possible Profit

CHOOSE OUTPUT TYPE*
FLOAT

FORMULA*
TO_NUMBER({item<pricing_unitprice}) - TO_NUMBER({cost})

Need help with defining the formula? [Validate](#)

[Fields](#) [Functions](#)

CHOOSE SOURCE
Show all available fields

Search through records, fields, and field IDs

- DATA GRID -

Item Name/Number	itemid
- ITEM -	
Item Name/Number	itemid
[Missing Label:hashOfLinkedBoms]	hashoflinkedboms
Allocation Price	vsoeprice
Allocation Type	vsoesopgroup
Alternate Source Item	alternatedemandsourceitem
Amortization Period	amortizationperiod

[Save](#) [Cancel](#)

Item Dataset

Search for records and fields...

Formulas

- Item
- > Amortization Template
- > Asset Account Account
- Bed Size
- Bed Size 3 Bed Sizes
- > Billing Schedule
- > Class
- Color
- Colors
- Consumption Units Units
- Correlated Items
- > Customer Return Variance A...
- > Default WT Code Withholding...
- > Deferred Expense Account
- > Department
- > Dropship Expense Account A...
- > Exchange Rate Variance Acc...
- > Expense/COGS Account Acco...
- > Field [Asset/Sales Tax Item] ...
- > Field [Inventory Item] on [Pr...

New Formula

Drop fields here to add criteria.

ITEM NAME/NUMBER	POSSIBLE PROFIT
Armless Dining Chair	18.00
Armless Dining Chair	7.80
Total row count: 675	

Use the formula definitions in the following table when you want to cast fields to a different data type:

Data Type	To cast to STRING	To cast to INTEGER	To cast to FLOAT
INTEGER	TO_CHAR({field_ID})	—	Automatic casting
FLOAT	TO_CHAR({field_ID})	CEIL({field_ID}) FLOOR({field_ID}) ROUND({field_ID})	—

		TRUNC({field_ID})	
BOOLEAN	Automatic casting, returns 'T' or 'F'	CASE field WHEN 'T' THEN 1 ELSE 0 END	CASE field WHEN 'T' THEN 1 ELSE 0 END
DATE	TO_CHAR({field_ID})	—	—
DATETIME	TO_CHAR({field_ID})	—	—
PERCENT	TO_CHAR({field_ID}) Returns values between 0 and 1	—	TO_NUMBER({field_ID})
DURATION	TO_CHAR({field_ID}) Returns FLOAT values, not in the hours:minutes format	—	TO_NUMBER({field_ID})
STRING	—	TO_NUMBER({field_ID}) Not preferred	TO_NUMBER({field_ID})
CURRENCY	TO_CHAR({field_ID}) Returns values without the currency symbol	—	TO_NUMBER({field_ID})
CURRENCY_HIGH_PRECISION	TO_CHAR({field_ID}) Returns values without the currency symbol	—	TO_NUMBER({field_ID})

Currency Consolidation and Conversion Using Custom Formula Fields

You can consolidate or convert values that are in multiple currencies using the context values #currency_consolidated or #converted. In the Dataset Builder, this is required if you want to view totals or perform arithmetic operations using amount fields that have values in multiple currencies. In workbook visualizations, all fields with values in multiple currencies have consolidation and conversion options available directly from the user interface. For more information, see [Currency in Datasets and Workbooks](#).

In Workbook, currency consolidation is only applicable to amount fields from the transaction accounting line record type. Currency conversion is only applicable to amount fields from the transaction and transaction line record types.

For example, to consolidate values from the amount net field on the transaction accounting line record type, use the following definition:

```
1 | TO_NUMBER({transactionlines.accountingimpact.netamount#currency_consolidated})
```

To convert the values from the amount paid field on the transaction record type, use this definition:

```
1 | TO_NUMBER({foreignamountpaid#converted})
```

With the context value #consolidated, there are also additional options to specify the exact currency code and exchange rate date to use for the conversion. For more information, see [Currency Conversion Using Custom Formula Fields](#).

The function TO_NUMBER is always required when you use #currency_consolidated or #converted. This is because the current version of the Formula Builder does not support the CURRENCY output type and

TO_NUMBER casts the results to the FLOAT output type. For more information about using TO_NUMBER to cast currency values, see [Casting Field Values using TO_NUMBER and TO_NCHAR](#).

Conditional Evaluations Using CASE WHEN

You can perform conditional evaluations by creating a formula field using the CASE WHEN function. In these formulas, values that meet the conditions you set in the WHEN statement produce the results you define in the THEN statement. Values that do not meet the conditions in the WHEN statement produce the results that you define in the ELSE statement.

For example, if you want to determine the season that a transaction took place, you can use the following formula definition. The output type for this formula is STRING:

```

1 CASE
2 WHEN EXTRACT(Month FROM {trandate})= 12 THEN 'winter'
3 WHEN EXTRACT(Month FROM {trandate})= 6 THEN 'summer'
4 ELSE 'it was fall or spring' END

```

Formula Field

FORMULA FIELD NAME*
Season

CHOOSE OUTPUT TYPE*
STRING

FORMULA* ✓

```

1 CASE
2 WHEN EXTRACT(Month FROM {trandate})= 12 THEN 'winter'
3 WHEN EXTRACT(Month FROM {trandate})= 6 THEN 'summer'
4 ELSE 'it was fall or spring'
5 END

```

Need help with defining the formula? [Validate](#)

Fields Functions

Search functions

- MOST RECENTLY USED -
- ALL -
- ABS**
- ACOS
- ADD_MONTHS
- ASCII
- ASCIISTR

ABS(n)

Example
ABS(-2.57)
[Add the example to the formula](#)

The ABS function calculates the absolute value of an expression. Because the absolute value of a real number is its numeric value without regard to its sign, this function always returns a positive value. For example, 3 is the absolute value of both 3 and -3.

Save **Cancel**

TRANSACTION	DATE	SEASON	MEMO	TYPE	TOTAL AMOUNT (TRANSACTION CURRENCY)
Check #2074	12/5/2004	winter	WE00001	Check	-\$3
Check #2028	12/15/2002	winter	451-9935	Check	-\$1
Payment #12	12/8/2002	winter		Payment	\$
Check #2053	12/15/2003	winter	451-9935	Check	-\$1
Check #2050	12/5/2003	winter	WE00001	Check	-\$3
Check #2027	12/5/2002	winter	WE00001	Check	-\$3
Check #2077	12/15/2004	winter	451-9935	Check	-\$1
Check #2065	6/15/2004	summer	451-9935	Check	-\$1
Check #2062	6/5/2004	summer	WE00001	Check	-\$3
Check #2041	6/15/2003	summer	451-9935	Check	-\$1

In Workbook, you can nest different CASE WHEN statements within the same formula definition.

Calculating Amounts for Relative Date Ranges

You can create formula fields to calculate amounts for relative date ranges. This includes transaction sales amounts, quantities, and other numerical values. Depending on the date range, you can use the following formulas. The output type for each of these formulas is FLOAT:



Important: If you are working with values in multiple currencies, you must consolidate or convert the field values to a single currency as part of your formula definition. For more information, see [Currency Consolidation and Conversion Using Custom Formula Fields](#).



Note: The following samples are valid for formula fields in datasets based on the transaction record type. If you working on a dataset based on the sales (ordered) or sales (invoiced) analytical record types, replace {foreigntotal} with {amountnet}. If you want to calculate other values like quantity sold, replace {foreigntotal} with the appropriate field ID.

- Amount Last Year to Date
- Amount This Year to Date
- Amount This Year
- Amount Last Year
- Amount This Quarter to Date
- Amount Last Quarter
- Amount This Month to Date
- Amount Last Month to Date

After you create your formulas, you can use them to create a calculated measure to analyze period over period variances. For more information, see [Year over Year Growth](#).

Amount Last Year to Date

Use this formula to calculate transaction amounts from all of last year until today's date:

```

1 CASE
2 WHEN (ADD_MONTHS({trandate}, -12) <= CURRENT_DATE) AND (EXTRACT(YEAR FROM {trandate}) + 1 = EXTRACT(YEAR FROM CURRENT_DATE))
3 THEN TO_NUMBER({foreigntotal})

```

```
4 | END
```

Amount This Year to Date

Use this formula to calculate transaction amounts for transactions created from the beginning of this year to today's date:

```
1 | CASE
2 |   WHEN {trandate} <= CURRENT_DATE AND EXTRACT(YEAR FROM {trandate}) = EXTRACT(YEAR FROM CURRENT_DATE)
3 |   THEN TO_NUMBER({foreigntotal})
4 | END
```

Amount This Year

Use this formula to calculate the total amounts for transactions created this year:

```
1 | CASE
2 |   WHEN EXTRACT(YEAR FROM {trandate}) = EXTRACT(YEAR FROM CURRENT_DATE)
3 |   THEN {foreigntotal}
4 | END
```

Amount Last Year

Use this formula to calculate the total amounts for transactions created last year:

```
1 | CASE
2 |   WHEN EXTRACT(YEAR FROM {trandate}) + 1 = EXTRACT(YEAR FROM CURRENT_DATE)
3 |   THEN TO_NUMBER({foreigntotal})
4 | END
```

Amount This Quarter to Date

Use this formula to calculate amounts for transactions created during this quarter:

```
1 | CASE
2 |   WHEN TO_CHAR({trandate}, 'Q') = TO_CHAR(CURRENT_DATE, 'Q') AND EXTRACT(YEAR FROM {trandate}) = EXTRACT(YEAR FROM CURRENT_DATE)
3 |   THEN TO_NUMBER({foreigntotal})
4 | END
```

Amount Last Quarter

Use this formula to calculate transaction amounts from all of last quarter

```
1 | CASE
2 |   WHEN TO_NUMBER(TO_CHAR(CURRENT_DATE, 'Q')) > 1 AND TO_NUMBER(TO_CHAR({trandate}, 'Q')) = TO_NUMBER(TO_CHAR(CURRENT_DATE, 'Q'))-1 AND
3 |     EXTRACT(YEAR FROM {trandate}) = EXTRACT(YEAR FROM CURRENT_DATE)
4 |   THEN TO_NUMBER({foreigntotal})
5 |
6 |   WHEN TO_NUMBER(TO_CHAR(CURRENT_DATE, 'Q')) = 1 AND TO_NUMBER(TO_CHAR({trandate}, 'Q')) = 4 AND EXTRACT(YEAR FROM {trandate})+1 =
7 |     EXTRACT(YEAR FROM CURRENT_DATE)
8 |   THEN TO_NUMBER({foreigntotal})
9 | END
```

Amount This Month to Date

Use this formula to calculate amounts for transactions created from the beginning of this month to today's date:

```

1 CASE
2 WHEN {trandate} <= CURRENT_DATE AND EXTRACT(Month FROM {trandate}) = EXTRACT(Month FROM CURRENT_DATE) AND EXTRACT(YEAR FROM
3 {trandate}) = EXTRACT(YEAR FROM CURRENT_DATE)
4 THEN TO_NUMBER({foreigntotal})
END

```

Amount Last Month to Date

Use this formula to calculate transaction amounts from all of last month until today's date:

```

1 CASE
2 WHEN (ADD_MONTHS({trandate}, 1) <= CURRENT_DATE) AND EXTRACT(Month FROM ADD_MONTHS({trandate}, 1)) = EXTRACT(Month FROM CURREN
3 T_DATE) AND EXTRACT(YEAR FROM ADD_MONTHS({trandate}, 1)) = EXTRACT(YEAR FROM CURRENT_DATE)
4 THEN TO_NUMBER({foreigntotal})
END

```

Calculating Specific Dates

Use the following formula definitions to calculate specific dates. The output type for each of these formulas is DATE:

- First Day of This Month
- Last Day of This Month
- First Day of Last Month
- Last Day of Last Month

First Day of This Month

Use this formula to calculate the first day of the month:

```
1 | TRUNC(LAST_DAY(CURRENT_DATE)-1, 'MONTH')
```

Last Day of This Month

Use this formula to calculate the last day of the month:

```
1 | LAST_DAY(CURRENT_DATE)
```

First Day of Last Month

Use this formula to calculate the first day of the last month:

```
1 | TRUNC(LAST_DAY(ADD_MONTHS(CURRENT_DATE,-1)), 'MONTH')
```

Last Day of Last Month

Use this formula to calculate the last day of the last month:

```
1 | LAST_DAY(ADD_MONTHS(CURRENT_DATE,-1))
```

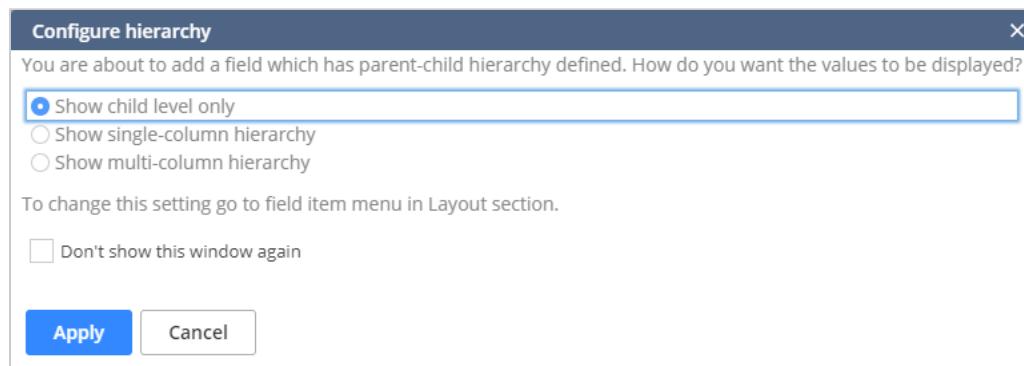
Hierarchical Fields

SuiteAnalytics Workbook supports hierarchical fields. Hierarchical fields behave similarly to standard record fields, except that you can choose to display the field values as hierarchies or child values. You can also create filter conditions at the parent or child levels of the hierarchy.

Display Options for Hierarchical Fields

When you add a hierarchical field to a pivot table or chart, you are prompted to select a display option for the field values.

You have the following options:

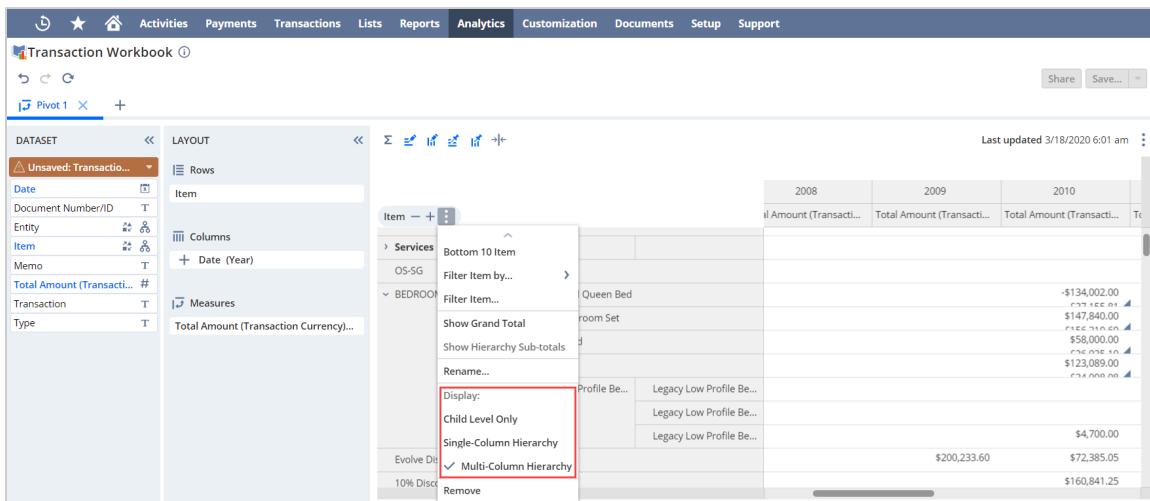


- Select **Show child level only** to display only the child-level values for the field.
- Select **Show single-column hierarchy** to display the full hierarchy values for the field in a single column or chart measurement.
- Select **Show multi-column hierarchy** to display each level of the hierarchy in a separate table column or chart measurement. If you select this option and enable total or grand totals for the pivot table, totals are calculated for each hierarchy level.

For example, in the following table the Item field is displayed as a multi-column hierarchy. Consequently, the table calculates totals for each bedroom item and for the bedroom parent level.

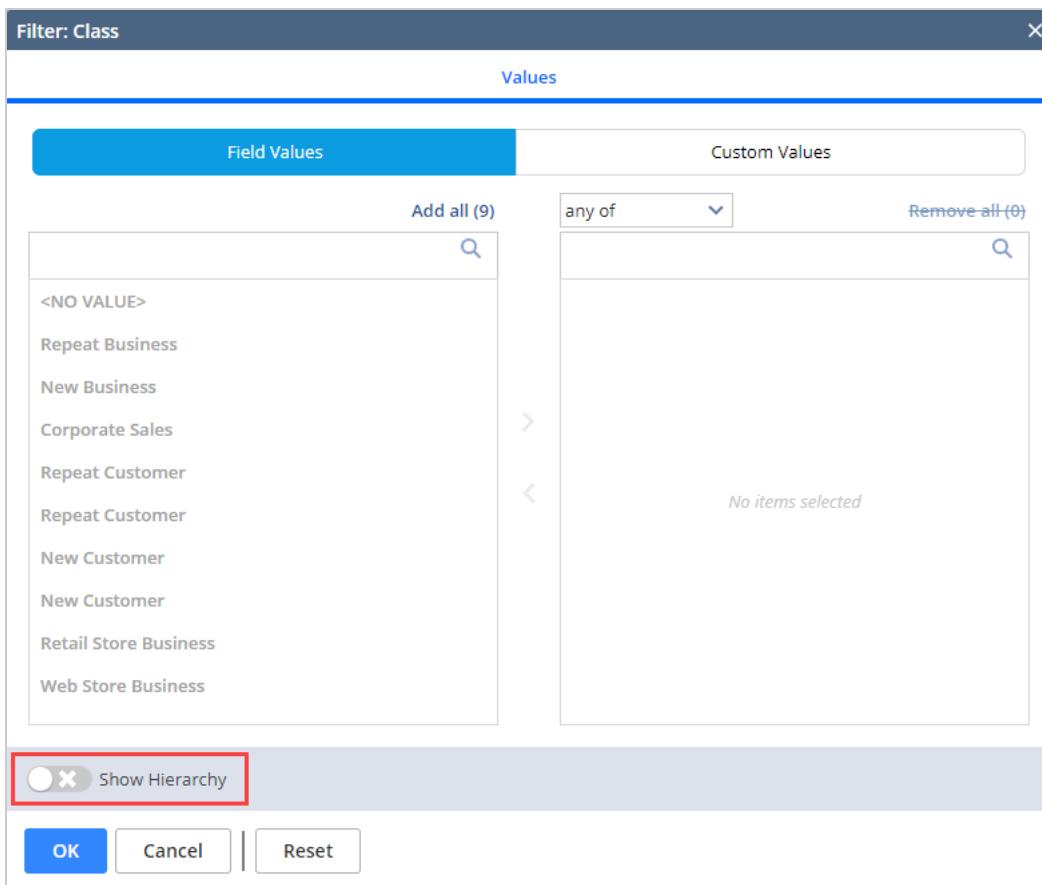
The screenshot shows a Pivot Table in the "Pivot 1" tab of a Transaction Workbook. The dataset includes fields like Date, Entity, Item, Memo, Total Amount (Transaction...), Transaction, and Type. The layout panel shows the Item field is being used as a column. The pivot table itself has columns for 2008, 2009, and 2010, and rows for Item, Services, OS-SG, BEDROOM, and various bed models. A red box highlights the BEDROOM row, showing detailed data for Handcrafted Queen Bed, Regashi Bedroom Set, Platform Bed, and Yao Bed, along with discounts and totals for each item and the parent category.

To change the hierarchy display for a field after adding it to a pivot table or chart, click the Field Menu icon in the Layout panel and select a new display option.



Filter Conditions for Hierarchical Fields

When you create a filter condition based on a hierarchical field, there is an additional **Show Hierarchy** option displayed in the Filter window.



When this option is enabled, the Filter window displays only the parent values for the selected field. If you create a filter condition at the parent level, only records with fields that have the selected parent are included in your pivot table or chart. For example, assume your dataset has two records with a class

of New Customer but only one of those records has a parent class of Web Store Business. If you create a filter condition to exclude records that have a parent value of Web Store Business, one of the New Customer records is removed from your workbook visualization.

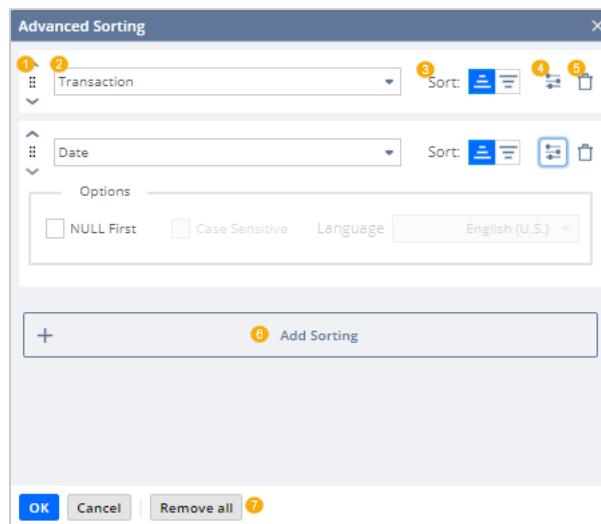
Advanced Sorting Options

You can sort the values that are displayed in the Data Grid using the Advanced Sorting window. Sorting options that you apply to the Data Grid do not affect any workbook visualizations that are based on the dataset and should only be used to improve the readability of the values presented in the grid. To filter the values included in your dataset, you must create criteria filters using the Criteria Builder, located above the Data Grid. For more information, see [Dataset Criteria Filters](#).

To open the Advanced Sorting window, click the Field Menu icon  next to a column in the Data Grid and select **Add Sort...** from the menu.

In the advanced sorting window, you can set up sorting conditions for any field in the Data Grid. After you set up each condition, you can select the order in which each condition should be considered.

The elements of the Advanced Sorting window are identified in the following image:



1	Up/ Down Arrows — Click the arrows to change the order of the sorting conditions. Alternatively, click and drag the entire row to the desired location.
2	Field list — Select the field you want to create a sorting condition for.
3	Sort — Sort the result set in ascending  or descending  order, based on the values of the selected field.
4	Options — Click this icon to view the following options specific to the selected sorting condition: <ul style="list-style-type: none"> ■ NULL First — If checked, blank values for the selected field are listed first in the corresponding column in the Data Grid. ■ Case Sensitive — If checked, values beginning with capital letters are listed first in the corresponding column in the Data Grid. ■ Language — Sort the result set based on the alphabetic sequence of the selected language.
5	Delete — Remove the selected sorting condition.
6	Add Sorting — Add another sorting condition, for a different field.

- 7 **OK** — Apply the selected sorting conditions to the Data Grid.
Cancel — Close the Advanced Sorting window without making any changes to the existing sorting conditions.
Remove All — Remove all of the sorting conditions currently listed in the Advanced Sorting window.

Joining Record Types in a Dataset

SuiteAnalytics Workbook enables you to join multiple record types in the same dataset. This includes record types that are more than one join away from the root record type selected for the dataset, enabling you to compile data from more diverse record types.

By default, the Dataset Builder lists all joinable record types that you have access to. To complete a join, simply add fields from the desired record type to the Data Grid or create a criteria filter using the Criteria Builder. If you are not sure which record type to join to your dataset, look for the field you want to add using the Records and Fields list search function. For more information, see [Locating Fields in Workbook](#) and [Dataset Builder](#).

If joining record types is a new concept for you, there are several things you should consider before you begin, such as the effect of one—to—many relationships between record types, and the direction and order of your joins. This is especially true if you are working on a transaction dataset and want to join the transaction line and transaction accounting line record types.

For more information, see [Guidelines for Joining Record Types in SuiteAnalytics Workbook](#).

Locating Fields in Workbook

If you are not sure which record type you want to join to a dataset, you can browse or search the Records and Fields list in the Dataset Builder. To get an overview of the Dataset Builder, see [Dataset Builder](#).

The record types and fields that you have access to are based on your permissions and the role you use to log in to NetSuite. If you do not see a record type or field that you think you should have access to, contact your administrator.

By default, record types and fields are listed in alphabetical order except if you are in a transaction dataset. In a transaction dataset, the transaction line and transaction accounting line record types are listed at the top of the Records list. This is because these are the most commonly joined record types in a transaction dataset.

You can search for fields using their labels or internal IDs. Searches are performed across record types that are up to three joins away from the root record type of the dataset. If you want to search across more record types, click **Show more results**.

If you are working in an existing dataset and you do not know which record type a field belongs to, point to the field in the Data Grid or in the criteria filter, then click the field menu icon and select **Show Location in Field List**. The associated record type is highlighted in the Records List.

Linking Datasets

You can link two datasets if they each have at least one field that shares common data, such as date. You can then use the linked datasets to build a single workbook visualization to compare data from both.

Link definitions exist as part of the workbook they are created in and cannot be shared on their own, however you can share each linked dataset individually. If a user shares a workbook with you that is based on a set of linked datasets, you can use them to create new visualizations within that workbook. For more information, see [Dataset Linking in SuiteAnalytics Workbook](#).

Dataset Criteria Filters

On the Dataset Builder above the Data Grid, the Criteria Builder enables you to filter your dataset results. Unlike filters that you create within workbook visualizations, criteria filters remove entire records and fields from a dataset which impacts all workbook visualizations that are based on the dataset. For example, if you create a criteria filter that removes invoice records from a dataset, workbook visualizations based on that dataset will no longer include invoice data. Use caution when applying new criteria filters to a dataset that is used in multiple workbook visualizations.

Criteria filters are joined by AND/OR logic within the Criteria Builder and can be grouped so that multiple filter conditions are evaluated as one item within the list of filters. You can edit or delete criteria filters as necessary, or move filters so that they are evaluated in a different order. Any fields that you use in a criteria filter are highlighted in the Fields list, however they do not appear at the top of the Fields list like fields that are added to the Data Grid. If you are not sure which record type a field in the Criteria Builder belongs to, click the Field menu icon next to a criteria filter and click **Show Location in Field List**.

When you create a new criteria filter, the results in the Data Grid are automatically updated. To update the results in a chart or pivot table that is based on the dataset however, you must manually refresh it.



Important: If you create a criteria filter based on a joined record type that has a one-to-many or many-to-many relationship with the root record type of the dataset, it can cause data duplication. For more information, see [Data Duplication Based on Record Joins](#).

If you are trying to filter results from a pivot table or chart that is based on linked datasets, you must apply the filter to the matching field in each dataset. If you do not, the results might include blank rows. For more information, see [Create Visualizations Based on Linked Datasets](#).

To create a dataset criteria filter:

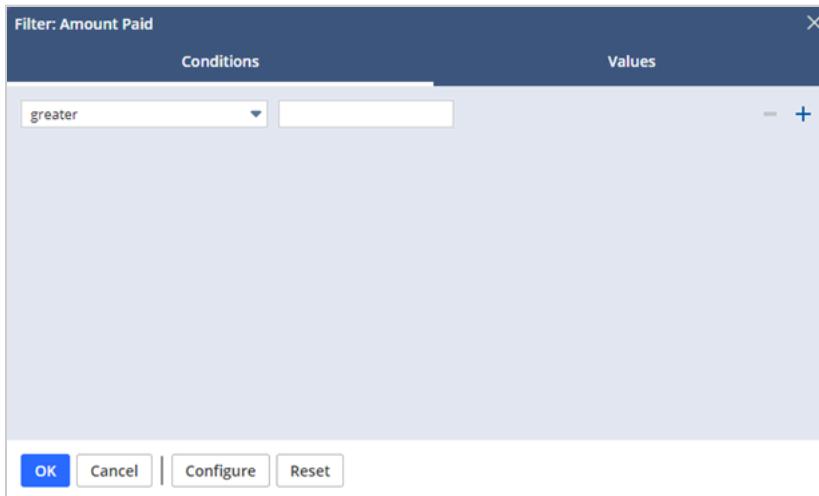
1. On the Dataset Builder, drag a field from the Fields list or the Data Grid to the Criteria Builder. The Filter window appears.
2. In the Filter window, select the filter conditions you want to apply to the field. Up to four types of filters are available depending on the field you select. For more information, see [Filter Types](#).
3. (Optional) If you want to group your criteria filters, click **New Group**. Grouped criteria filters are considered one item in the list of criteria filters and are evaluated in the order that they appear in the Criteria Builder. For more information, see [Grouping Filters](#).
4. Repeat steps 1–3 for each filter or filter group you want to define. By default, criteria filters are added using an AND operator. To change the relationship between filters and filter groups, click **AND** and select **OR** from the popup window.
5. (Optional) If you accessed the dataset from within a workbook, click **Apply to workbook** to preview your changes in the workbook.
6. Click Save to save your dataset changes. If you do not own the dataset, click **Save As** to save it under a different name.

Filter Types

When you add a field from the Fields list or the Data Grid to the Criteria Builder, the Filter window appears. Use the Filter window to set up specific filter conditions for a field. The types of filters you can set up depend on the values of the field, with each filter type represented by a different tab in the Filter window.



Important: The Filter window is also available from the Pivot tab within a workbook, however filter conditions set up from the Pivot tab only impact the pivot table and do not affect the underlying dataset or other visualizations in the workbook. Only filters that you set up from the Criteria Builder above the Data Grid can impact a dataset.



In each tab of the Filter window, you can define the values and expressions for the filter. You can only create one type of filter per field at a time however, unless you use a group of filters. For more information, see [Grouping Filters](#). The available values and expressions in each tab of the Filter window depend on the type of field you select.

- Values: Create a filter condition based on existing values or dates in the source data, or custom values
- Ranges/ Date Ranges: Create a filter condition based on a range of values or dates available in the source data
- Relative Conditions/ Relative Dates: Create a filter condition relative to the existing values or dates in the source data
- Conditions/ Specific Dates: Create a filter condition using custom values or dates and an expression



Note: For hierarchical fields, filter conditions can be set at the parent or child level. For more information, see [Hierarchical Fields](#).

If you are in a NetSuite OneWorld account and create a criteria filter based on a field that contains values in multiple currencies, a Consolidate Field box is also displayed in the Filter window. Check the Consolidate Field box to the apply the filter based on the consolidated values of the field. For more information, see [Currency in Datasets and Workbooks](#).

Grouping Filters

In the Criteria Builder, you can click **New Group** to create a group of criteria filters.

Grouped criteria filters are the same as parenthetical expressions in the existing Saved Search application. Use them to define a string of filter conditions connected by AND or OR operators, to be evaluated as one item within the list of criteria filters.

For example, if you set up the following criteria filters, SuiteAnalytics Workbook evaluates the entire set of grouped filter conditions (Sales Rep is A Wolfe-admin OR Status is Open) before the Start Date filter, and after the Date filter.

The screenshot shows the SuiteAnalytics Workbook interface. On the left is a sidebar with a tree view of fields under 'Transaction'. The main area displays a table titled 'Criteria summary' with the following content:

DATE	MEMO	ENTITY	DOCUMENT NUMBER/ID	TRANSACTION	TYPE	TOTAL AMOUNT (TRANSACT)
11/6/2010	Paid upfront	Gilchrist Pty Ltd	0117	Invoice #0117	Invoice	
8/15/2010	USD Coding	2001		Sales Order #2001	Sales Order	
6/30/2010	Motiva Mobility	2005		Sales Order #2005	Sales Order	
7/15/2010	Fibre Technology	2006		Sales Order #2006	Sales Order	
10/22/2010	Venture Capital Inc.	2007		Sales Order #2007	Sales Order	
1/1/2010	Jupiter Technology	2014		Sales Order #2014	Sales Order	
11/6/2010	Paid upfront	Gilchrist Pty Ltd	2075	Sales Order #2075	Sales Order	

Total row count: 7

You can also create a group of filters within another group. If you do, SuiteAnalytics Workbook evaluates the conditions in order from inside-to-outside.

For example, in the following group of filters, (Status is Sales Order: Closed AND Sales Rep is Franco Africa) is considered before (Status is Bill: Open OR Sales Rep is <NO VALUE>). Moreover, because the entire group is placed before the Start Date filter, the grouped filters are evaluated first.

The screenshot shows the SuiteAnalytics Workbook interface. On the left is a sidebar with a tree view of fields under 'Transaction'. The main area displays a table titled 'Criteria summary' with the following content:

DATE	MEMO	SALES REP	ENTITY	DOCUMENT NUMBER/ID	TRANSACTION	TYPE

Total row count: 0

Creating a Workbook

A workbook is where you analyze the results of a dataset query by creating workbook visualizations such as tables, pivot tables, and charts.

All workbook visualizations are based on a dataset, however a single dataset can be used in multiple workbooks and workbook visualizations. Additionally, you can use a different dataset for each visualization in a workbook. Although you can access the underlying dataset from within a workbook visualization, the dataset exists as its own object, with its own creator and access rights. Moreover, changes to a dataset are automatically propagated to any connected workbook visualizations. This prevents data discrepancies but also means that your workbooks can change unexpectedly if you do not own the underlying dataset.

You can only create workbook visualizations using fields that have been added to the underlying dataset. If you do not see a field that you want to include in a workbook visualization, you must add it to the

dataset or contact the dataset owner so that they can add it for you. For more information about defining a dataset, see [Defining a Dataset](#).

In workbooks connected to linked datasets, the link definition exists as part of the workbook. You can use the linked datasets as the basis for multiple visualizations, but only in the workbook where the link was defined. You cannot share the link definition on its own.

The types of workbook visualizations that you can create include table views, pivot tables, and charts:

- Table Views enable you to display and analyze your dataset results in a simplified table. You cannot use linked datasets to build a table view. For more information, see [Workbook Table Views](#).

 **Note:** By default, all workbook templates include a Table View based on the underlying dataset. For more information about workbook and dataset templates, see [Workbook and Dataset Templates](#).

- Pivot tables enable you to pivot your dataset results so that you can analyze different subsets of the data, such as the number of unfulfilled orders per customer in your account. With linked datasets, you can also use pivot tables to analyze metrics at two different levels of aggregation. For more information, see [Workbook Pivot Tables](#).
- Charts enable you to create different visualizations of your dataset results so that they can be analyzed at a glance. You can also build charts using linked datasets. For more information, see [Workbook Charts](#).

After you create your workbook visualizations, you can rearrange the order in which the workbook tabs are displayed as needed.

 **Note:** If you have multiple languages in your account and you edit and save default text in a workbook or workbook visualization, a new translation collection is added to the Manage Translations page. For more information, see the help topic [Manage Translations](#).

To create a new workbook or workbook visualization based on new dataset:

1. From the Analytics Home page on the Workbooks subtab, click **New Workbook**.
2. On the **Select a dataset to create a new workbook** page, click **New Dataset**.
3. Select a record type for the dataset.
The Dataset Builder appears with preselected fields in the Data Grid.
4. Select the fields and criteria filters you want to include in the dataset. For more information, see [Defining a Dataset](#) and [Dataset Criteria Filters](#).
5. When you are satisfied with your results, click **Apply to workbook**.
6. Select a workbook visualization to begin.
For more information, see [Workbook Table Views](#), [Workbook Pivot Tables](#), or [Workbook Charts](#).
8. Click **Save...**

The Save Workbook As window appears.



Important: You cannot save a workbook if it has visualizations that are based on unsaved datasets. If you have unsaved datasets connected to your workbook, click the menu icon next to the dataset name in the Dataset Panel and select **Save changes as** or **Discard changes**. Alternatively, if you want to continue editing the dataset, click **Open Dataset**.

9. Enter a name and description for the workbook in the fields provided, then click **Save**.

10. (Optional) Click **Share** to share the workbook with other users in your account.
The Share Workbook window appears.
11. Select the roles or user IDs that you want to share the workbook with, then click **Share**.
For more information about sharing workbooks, see [Accessing and Sharing Workbooks and Datasets](#).
12. (Optional) To change the order in which the workbook visualizations are displayed, drag the workbook tab to the desired position.

To create a new workbook or workbook visualization based on an existing dataset:

1. From the Analytics Home page on the Workbooks subtab, click **New Workbook**.
Alternatively, if you are already in an existing dataset, click **Create New Workbook** and proceed to step 3.
2. Select a dataset on the **Select a dataset to create a new workbook** page.
3. Select a workbook visualization to begin.



Important: Keep in mind that you can only create workbook visualizations using fields that have been added to the selected dataset. If you do not own the selected dataset and you want to add fields or criteria to it, you must save the dataset using a different name. Additionally, if you proceed without saving your own version of the dataset, the dataset creator can make changes to the dataset which will impact your visualization. Use caution when creating workbook visualizations based on datasets that you do not own.

4. Edit the layout and formatting for each workbook visualization that you add.
For more information, see [Workbook Table Views](#), [Workbook Pivot Tables](#), or [Workbook Charts](#).
5. Click **Save...**
The Save Workbook As window appears.
6. Enter a name and description for the workbook in the fields provided, then click **Save**.
7. (Optional) click **Share** to share the workbook with other users in your account.
The Share Workbook window appears.
8. Select the roles or user IDs that you want to share the workbook with, then click **Share**.
For more information about sharing workbooks, see [Accessing and Sharing Workbooks and Datasets](#).
9. (Optional) To change the order in which the workbook visualizations are displayed, drag the workbook tab to the desired position.

To create a new workbook visualization within an existing workbook:

1. From the Analytics Home page on the Workbooks subtab, click the name of the workbook you want to add the visualization to.
2. Click the Add icon
3. Select the visualization type you want to add to the workbook.
By default, the visualization is based on the dataset currently selected in the Dataset Panel. If you want to base it on a different dataset, click **Connect Dataset**.
4. Edit the layout and format for the visualization.
For more information, see [Workbook Table Views](#), [Workbook Pivot Tables](#), or [Workbook Charts](#).
5. (Optional) To change the order in which the workbook visualizations are displayed, drag the workbook tab to the desired position.

Workbook Table Views

Workbook table views are where you can explore your dataset query results without altering the source data of any workbook visualizations that are based on the selected dataset. Using the table view does not require complex customization and enables you to view your data without setting up a layout or defining custom formula fields. You cannot use linked datasets to build a table view.

You can only create table views using the fields selected in the underlying dataset, as displayed in the Dataset Panel. Each field in the Dataset Panel includes an icon that identifies the type of value contained within the field. For more information about each field type, open the underlying dataset. By default, all workbook templates include a Table View based on the underlying dataset. For more information about workbook and dataset templates, see [Workbook and Dataset Templates](#).

To create a table view, you drag fields from the Dataset Panel to the Table Viewer. After you add fields to the Table Viewer, you can filter and sort them as needed. When you add a new field to the Table Viewer or apply new sorting or filtering options, the results are automatically updated. You can further customize your table view by renaming the columns or applying conditional formatting to the results. You can create as many table views as you like using different datasets in the same workbook. Additionally, you can export the data presented in your table view to a CSV file, and add your table views to the Analytics portlet on any NetSuite dashboard.



Important: Sorting, filtering, conditional formatting, and other customization options that you apply to the table view do not affect the underlying dataset or any other workbook visualizations. To change the associated dataset, click the dataset name in the Dataset Panel. If you do not own the underlying dataset, contact the dataset owner.

To create a table view:

1. Click the add icon from anywhere within the workbook and select **Table**.



Important: By default, the table is based on the most recently connected dataset in the workbook. To change the dataset that the table is based on, expand the dataset selector in the Dataset Panel and select a new dataset. Alternatively, if the dataset you want is not already used in the workbook you are in, click **Connect Dataset** to select from any of the datasets you have access to in your account, or create a new dataset.

2. On the Table tab, drag fields from the Dataset Panel to the Table Viewer.

Fields added to the Table Viewer are highlighted in blue on the Dataset Panel.



Note: If you add hierarchical fields to the table, you are prompted to select a display type for the field values. For more information, see [Hierarchical Fields](#).

3. Customize the appearance of your table:

- a. Change the order of the columns by dragging them to the desired position or clicking the Field Menu icon and selecting one of the move options.
- b. Rename the columns by clicking the Field Menu icon in the column header that you want to rename, then selecting **Rename**.



Note: The change applies to the column name of the selected table view only. The name does not change in the Dataset Panel, the underlying dataset, or any other workbook visualizations that are based on the selected dataset.

- c. Remove unwanted columns by clicking the Field Menu icon in the column header that you want to remove, then selecting **Remove Column**.

4. Sort the data presented in the table:

- a. Click the Field Menu icon next to the field you want to sort.
- b. Select **Ascending** or **Descending** to apply one of the default sorting options, or select **Add Sort...** to apply advanced sorting.



Note: The advanced sorting options available for a table view are the same as those that you can apply to a dataset. For more information, see [Advanced Sorting Options](#).

5. Filter the data presented in the table:
 - a. Click the Field Menu icon next to the field that you want to filter and select **Filter...**
The Filter window appears.
 - b. Select the conditions or values for the filter, then click **Apply**. For more information about the available filtering options, see [Value-based Filters](#).
The Table Viewer is automatically updated.
6. Apply conditional formatting to the data presented in the table.



Important: Conditional formatting does not differentiate between values in multiple currencies. To accurately highlight values in multiple currencies, consider converting the values to a single currency first and then applying conditional formatting. For more information, see [Currency in Datasets and Workbooks](#).

- a. Click the Field Menu icon in the column header that you want to highlight and point to **Conditional Formatting**.
 -
-
- Note:** Conditional formatting is currently not available for fields containing DATE values.
- b. Click **Manage Conditional Formatting**, then select the operators, values, and colors or icons for the rule. You can click the add icon **+** to create multiple rules and apply different colors or icons to the same measure or column. For more information, see [Conditional Formatting](#).
 -
-
- Note:** To apply conditional formatting to percentage values, use decimal format when defining the values for the rule. For example, rather than **greater or equal to 20%**, the rule should be defined as **greater or equal to 0.2**.
- c. Click **Apply**.
 - d. Repeat steps A-C for each column that you want to highlight.

7. (Optional) Click the Export icon to save a CSV file of your completed table.



Note: To export your table view, you must have the Export Lists permission.

Table View Portlets

You can see your table view data on your home dashboard by adding the Analytics portlet. The portlet enables you to quickly see the workbook data on your dashboard and provides quick access to your workbook. Visualizing the data right on your dashboard enables you to spend more time learning from it and less time gathering it.

You can also choose how you want to visualize your table view portlet by entering the number of result rows you want to view at one time. To identify the Analytics portlet on your dashboard, you

can also enter a title. By default, the title displays the name of the workbook and the pivot table. Any conditional formatting you apply to your table views is also displayed in your table view portlets. For more information, see [Conditional Formatting](#).

In your table view portlets, you can do the following:

- View the column summary – When you select a column in your table view, you can also view the summary of the results in the table footer. The information shown depends on the field type, and can display information such as the count, count (distinct), average, and so on.
- Sort and filter your results – You can use the same sorting and filtering options that are included in your workbook table view.

The screenshot shows a table view portlet titled "My Sales (Invoiced) Table View Portlet". The table has columns: DATE, TRANSACTION, TRANSACTION TYPE, AMOUNT (NET), QUANTITY, and CLASS. The first column, DATE, is highlighted with a red border. The table contains 10 rows of transaction data. At the bottom of the table, there is a summary bar with the text "Summary of all data: Count: 53 Count (Distinct): 24". Below the table, there are navigation icons for a left arrow, a right arrow, and a double-right arrow, followed by the text "1 of 2".

DATE	TRANSACTION	TRANSACTION TYPE	AMOUNT (NET)	QUANTITY	CLASS
1/4/2002	Cash Sale #1001	Cash Sale	\$695.00	1	Repeat Customer
1/10/2002	Cash Sale #1002	Cash Sale	\$2,299.00	1	Repeat Customer
1/10/2002	Cash Sale #1002	Cash Sale	\$169.95	1	Repeat Customer
1/14/2002	Cash Sale #1003	Cash Sale	\$12.95	1	New Customer
1/14/2002	Cash Sale #1003	Cash Sale	\$59.95	1	New Customer
1/14/2002	Cash Sale #1003	Cash Sale	\$10.95	1	New Customer
1/18/2002	Invoice #1001	Invoice	\$71.92	2	New Customer
1/18/2002	Invoice #1001	Invoice	\$269.96	1	New Customer
1/18/2002	Invoice #1001	Invoice	\$1,798.20	2	New Customer
1/18/2002	Invoice #1001	Invoice	\$404.96	1	New Customer

Summary of all data: Count: 53 Count (Distinct): 24

As with charts and pivot tables, you can also access your workbooks through table view portlets.

For more information about how to add and set up the Analytics portlet, see the help topic [Workbooks](#).

Workbook Pivot Tables

The Pivot tab is where you pivot your dataset query results to analyze different subsets of data. Within each table you can define multiple fields for each dimension, add measures and create calculated measures, or create filters unique to the table to customize your results. You can also change the appearance of your table by formatting the numeric values that are presented, renaming your table rows and columns, adding totals and grand totals, or applying sorting options and conditional formatting. Additionally, you can add your pivot tables to the Analytics portlet on any NetSuite dashboard and export your pivot table to a CSV file. If your table is connected to linked datasets, you can also compare data at different levels of aggregation based on the fields you select. For more information, see [Dataset Linking in SuiteAnalytics Workbook](#).

Note: The following formatting options are not exported to the CSV file: percentages, currency symbols, totals, and layout settings such as conditional formatting and compact mode.

You can only create a pivot table based on the fields that are included in the underlying datasets. On the Pivot tab, these fields appear in the Dataset Panel on the left. In pivot tables connected to linked datasets,

a link icon next to a field denotes that it is a common key in the link definition. For more information about common keys, see [Common Keys](#).

The Pivot Table Viewer on the right displays your completed table based on the fields you define as rows, columns, and measures in the Layout Panel. Each time you change or update your layout, you must click the Refresh icon  to update the table. You must also refresh your table if changes are made to the underlying datasets.

To create a pivot table:

1. Click the add icon  from anywhere within the workbook and select **Pivot**.



Important: By default, the pivot table is based on the most recently connected dataset in the workbook. To change the dataset that the table is based on, expand the dataset selector in the Dataset Panel and select a new dataset. Alternatively, if the dataset you want is not already used in the workbook you are in, click **Connect Dataset** to select from any of the datasets you have access to in your account, or create a new dataset.

2. On the Pivot tab, drag the desired fields from the Dataset Panel to the Rows, Columns, or Measures tabs in the Layout panel. Alternatively, drag the fields from the Dataset Panel directly to the Pivot Table Viewer.
3. (Optional) In the Layout Panel, click **Create Calculated Measures** to create new measures from the fields available in the Dataset Panel. Calculated measures are represented by the calculator icon . For more information, see [Calculated Measures](#).



Note: If you add hierarchical fields to the table, you are prompted to select a display type for the field values. Depending on where you add the field and the display type you select, you can also add additional subtotals to the pivot table for each level in the hierarchy. For more information, see [Hierarchical Fields](#).

4. (Optional) If you want to build your pivot table using two datasets, click the Dataset Menu icon and select **Create Dataset Link**. Depending on the common keys you select and where you place them in the table, your results will be aggregated. For more information, see [Create Visualizations Based on Linked Datasets](#).
5. Select the summary type and format options for any date or numerical fields you add to the pivot table.

1. Click the Field Menu icon  next to the field you want to format in the Layout panel.
2. Select a summary type from the popup window.

The summary options vary depending on the type of field you select.

3. (Optional) Select **Currency...** to view the currency consolidation or conversion options for any fields with values in multiple currencies.

For more information, see [Currency Conversion from the User Interface](#).

4. (Optional) Click **Format...** to customize the numeric values for a field.

For more information about numeric formatting options, see [Customizing Numeric Values](#).

6. Add totals and grand totals to the pivot table.

1. Click the Totaling icon .
2. In the Totaling window, select where you want the totals or grand totals for each applicable field to appear. If there are multiple fields that can be totalled in the rows or columns, check the **Set Individually** box to select where the totals for each field will appear on the pivot table.

3. Click **OK**.
7. Click the Refresh icon  to generate the pivot table.
8. Filter the data displayed in the pivot table.



Note: Filter conditions created on the Pivot tab only impact the data displayed in the pivot table. No changes are made to the dataset that the workbook is based on.

1. Click the Field Menu icon  next to the field you want to create a filter for. Depending on whether the field has been defined as a column, row, or measure, the following options are available. If you click the Field menu icon from the Fields List or the Layout Panel, it can also impact the available options:
 - Top 10: display only the top 10 rows or columns based on the measures defined for the table.
 - Bottom 10: display only the bottom 10 rows or columns based on the measures defined for the table.
 - Filter [Field Name] by...: enables you to define a custom measure-based filter for the selected row or column.
 - Filter [Field Name]: enables you to define a custom value-based filter based on specific values within the table results.
 - Add as Filter...: enables you to define a custom value-based filter based on specific values within the table results.
 - Filter Date: enables you to define a custom date filter based on date ranges that you choose.
2. The results in the table are updated automatically.

For more information, see [Workbook Visualization Filters](#).



Important: If you build your pivot table using linked datasets, you must apply filters to the matching field in each dataset. If you do not, your results might include cells that you wanted to filter. For more information, see [Create Visualizations Based on Linked Datasets](#).

9. Apply conditional formatting to the pivot table measures.



Important: If the measure field you select has values in multiple currencies, you must convert or consolidate the values before you can apply conditional formatting. For more information, see [Currency in Datasets and Workbooks](#).

- a. Click the Field Menu icon next to the measure field you want to highlight in the Layout Panel and point to **Conditional Formatting**.
- b. Click **Manage Conditional Formatting**, then select the operators, values, and colors or icons for the rule. You can click the add icon  to create multiple rules and apply different colors or icons to the same measure or column. For more information, see [Conditional Formatting](#).



Note: To apply conditional formatting to percentage values, use decimal format when defining the values for the rule. For example, rather than **greater or equal to 20%**, the rule should be defined as **greater or equal to 0.2**.

- c. Check **Apply to subtotals and grand totals** if you want to apply your rules to the measure subtotals and grand totals.
- d. Click **Apply**.

- e. Repeat steps A-D for each measure you want to highlight.
10. (Optional) Click the Export icon to save a CSV file of your pivot table.

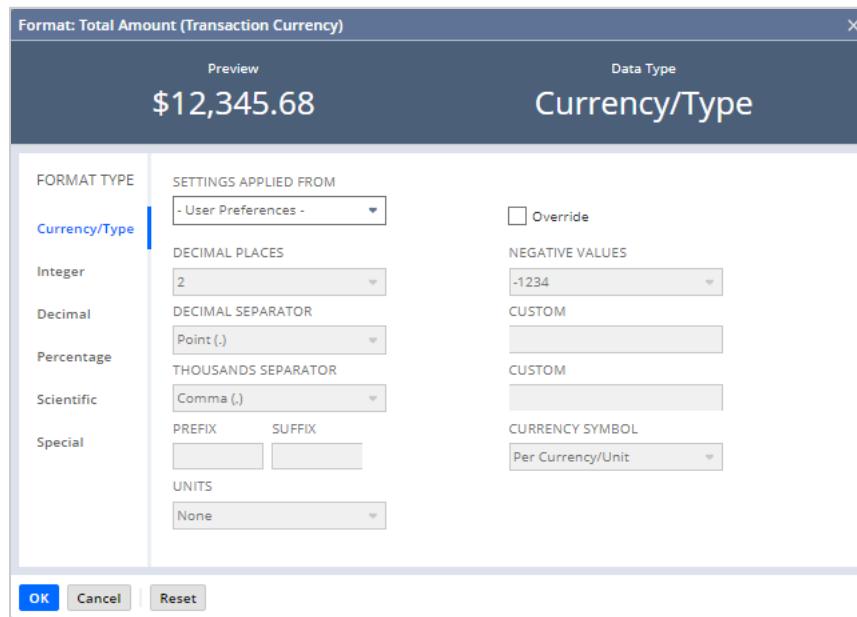
Pivot Table Customization

SuiteAnalytics Workbook enables you to customize many aspects of your pivot tables. Click the following links for more information:

- [Customizing Numeric Values](#)
- [Grouping Pivot Table Fields](#)
- [Compact and Expanded Mode](#)
- [Additional Pivot Table Customization Options](#)

Customizing Numeric Values

You can customize the numeric values displayed in your pivot table using the Format window. To access the Format window, click the Field Menu icon  next to the desired field in the Fields list or Layout panel and select **Format...** from the list.



The following options are available. To change the default settings, check the **Override** box. After you make a selection, the Preview field displays your changes. To apply your changes to the table, click **OK** and refresh the table. To define your own syntax, click the Special tab and enter the syntax in the field provided.

- **Settings Applied From:** Enables you to apply numeric formatting based on a specific language. To define each setting manually, select **Custom**. To use the numeric formatting set up in your NetSuite personal preferences, select **User Preferences**. For more information about personal preferences, see the help topic [Setting Personal Preferences](#).
- **Decimal Places:** The number of decimal places to include for each value.
 - Select Per Currency/Unit to use the decimal places of the currency or unit of measure for each value.
 - Select Unbound to place no limits on the number of decimal places used for each value.

- **Negative Values:** The format to use for negative values.
- **Decimal Separator:** The punctuation to indicate the decimal place. Select **Custom** to define your own punctuation.
- **Thousands Separator:** The punctuation to separate groups of thousands. Select **Custom** to define your own punctuation.
- **Prefix:** Enables you to define a prefix for the values in the field.
- **Suffix:** Enables you to define a suffix for the values in the field.
- **Currency Symbol:** Enables you to define where the currency symbol appears for dollar values in the field.
- **Units:** Enables you to abbreviate the values of the field by thousands, millions, or billions.

Custom Number Formatting

You can customize number formatting in pivot tables in the **special** tab. Define your own syntax, or choose from the provided Predefined Values.

The following pivot table displays numbers without custom formatting in the first column. In the second column, the numbers are custom formatted with the syntax: [0000ff]"gain "00.00;[ff0000] (#,#00.00)" loss";[ff00ff]"nothing "0.00" zero"

Entity	Amount (Sum)	Amount (Sum)
D&G Consulting	\$1,898.00	gain 01.90k
Best Computer Services	\$0.00	nothing 0.00 zero
Eala Land Holdings	-\$338,972.26	(338,972.26) loss
Clemens Fritz	-\$10,330.68	(10,330.68) loss
Waste Management Sys...	-\$3,730,682.45	(3,730,682.45) loss
Code Development	\$155,520.00	gain 155.52k
Denmark 5	\$0.00	nothing 0.00 zero

The available symbols for custom numbering syntax are:

Symbol	Name	Application
#	Number Sign	Displays significant digits
0	Zero	Displays non-significant zeros
""	Double Quotation Marks	Displays text enclosed in the quotation marks. Text can be placed before or after numbers (positive, negative, or zero).
+	Plus Sign	Typically used to display positive numbers
()	Parentheses	Typically used to display negative numbers
-	Minus Sign	Typically used to display negative numbers
,	Comma	Thousands separator, or scale a number by a 1000
.	Period	Decimal separator
;	Semicolon	Separates sections in the syntax. When you create custom number formats, you can specify up to three sections of syntax for positive numbers, negative numbers, and zero values, in that order.
[]	Brackets	Hexadecimal number (without #) in square brackets defines the color of the custom format section. For example, [0000ff] is blue.

Grouping Pivot Table Fields

Grouping fields in the pivot table Layout panel changes the granularity of the data presented in the table. By defining multiple fields as rows or columns, you can display subsets of data in the table.

For example, in the following transaction table, only the sales rep field has been defined as a row. The pivot table therefore only displays the total transactions for each sales representative.

The screenshot shows the SuiteAnalytics Workbook interface with a pivot table titled 'Pivot 1'. In the 'LAYOUT' panel, under 'Rows', 'Sales Rep' is selected and highlighted with a red box. The main table displays data grouped by Sales Rep, with columns for Date (Year), Total Amount (Transaction), and Total Amount (Transaction Currency). The data shows various sales representatives and their total transaction amounts across different years.

Date (Year)	Total Amount (Transaction)	Total Amount (Transaction Currency)
2007	\$100.00	\$5,600.00
2008	\$1,440.00	\$157,125.00
2009	\$9,488,058.28	\$398,140.00
2010	\$5,443,295.23	\$450,000.00
2011	\$696,416.00	\$92,120.00
2012	\$107,000.00	PHP83,160.00
	\$2,751.28	
	\$139,933.00	
		-PHP112.00
		-\$1,002.80
		-\$370.33
		\$822,673.75
		\$726,680.00
		\$59,404.13

If you also define the entity field as a row however, the table displays the total transactions per sales representative and customer.

The screenshot shows the SuiteAnalytics Workbook interface with a pivot table titled 'Pivot 1'. In the 'LAYOUT' panel, under 'Rows', both 'Sales Rep' and 'Entity' are selected and highlighted with red boxes. The main table displays data grouped by Sales Rep and Entity, with columns for Date (Year), Total Amount (Transaction), and Total Amount (Transaction Currency). The data shows various sales representatives and entities, with their total transaction amounts across different years.

Date (Year)	Total Amount (Transaction)	Total Amount (Transaction Currency)
2007	\$694,150.00	
2008	\$700.00	
2009	\$263,390.00	
2010	\$4,775.00	
2011	\$5,000.00	
	\$29,337.38	
	\$29,070.00	
	\$14,320.00	
	\$4,512.00	\$3,120.00
	\$300,000.00	
	\$1,200.00	\$3,600.00
	\$75,050.00	

Try grouping different fields in your pivot table layouts to analyze different subsets of data.

Compact and Expanded Mode

By default, if you group fields or include multicolumn hierarchical fields in your pivot table, the table expands horizontally which can make it difficult to view your data.

To display the table data vertically, click the Compact Mode icon .

Additional Pivot Table Customization Options

To further customize the appearance of your pivot tables, SuiteAnalytics Workbook offers the following options:

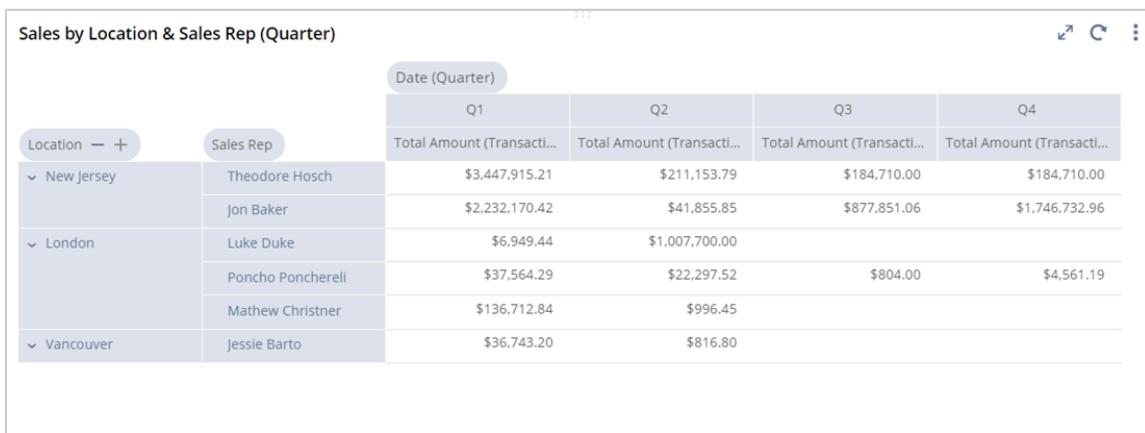
- To freeze column or row headers so that they are always visible as you scroll through your table, click the Freeze Column  or Freeze Row  icons.
- To enable row or column highlighting when you point to a specific part of the table, click the Highlight Column  or Highlight Row  icons.
- To rename a column or row, click the Field Menu icon  and select Rename... from the menu.
- To resize a column or row, drag one of the column or row boundaries until it is the size you want.
- To change the width of a column based on its values, double-click the column header.
- To expand or minimize the number of rows displayed in a column, click the  icons in the column header.
- To expand or minimize the number of values displayed for a row, click the  icon in the row header.

Pivot-based Portlets

You can see your pivot-based workbook data on your home dashboard by adding the Analytics portlet. The portlet enables you to quickly see the workbook data on your dashboard and provides quick access to your workbook. Visualizing the data right on your dashboard enables you to spend more time learning from it and less time gathering it.

You can also choose how you want to visualize your pivot-based workbook by entering the number of visible rows you want to view at one time. To identify the Analytics portlet on your dashboard, you can also enter a title. By default, the title displays the name of the workbook and the pivot table. Any conditional formatting you apply to your pivot table is also displayed in your pivot table portlets.

For more information about how to add and set up the Analytics portlet, see the help topic [Workbooks](#).



		Date (Quarter)			
		Q1	Q2	Q3	Q4
Location	Sales Rep	Total Amount (Transacti...)	Total Amount (Transacti...)	Total Amount (Transacti...)	Total Amount (Transacti...)
▼ New Jersey	Theodore Hosch	\$3,447,915.21	\$211,153.79	\$184,710.00	\$184,710.00
	Jon Baker	\$2,232,170.42	\$41,855.85	\$877,851.06	\$1,746,732.96
▼ London	Luke Duke	\$6,949.44	\$1,007,700.00		
	Poncho Ponchereli	\$37,564.29	\$22,297.52	\$804.00	\$4,561.19
	Mathew Christner	\$136,712.84	\$996.45		
▼ Vancouver	Jessie Barto	\$36,743.20	\$816.80		

Workbook Charts

The Chart tab is where you create visualizations of your dataset query results. There are multiple chart types you can create, all of which you can add to the Analytics portlet on any NetSuite dashboard. You can also print or export your charts to an SVG file. Additionally, you can customize the appearance of your chart by adding a title and subtitle, or by renaming each axis. Filtering capabilities are also supported, enabling you to display only the most pertinent values in each chart. If you want to compare metrics from two datasets in the same chart, you can also build it using linked datasets. For more information, see [Link Datasets in a Workbook](#).

The Chart tab displays any fields that are included in the underlying dataset in the Dataset Panel on the left. In charts connected to linked datasets, a link icon next to a field denotes that it is a common key in the link definition. For more information about common keys, see [Common Keys](#).

Completed charts are displayed in the Chart Viewer on the right, based on the fields you define as the x-axis, series, and measures in the Layout Panel. Each time you change or update your layout, you must click the Refresh icon  to update the chart. You must also refresh the chart if you make changes to the underlying dataset.

To create a chart:

1. Click the add icon  from anywhere within the workbook and select **Chart**.
-  **Important:** By default, the chart is based on the most recently connected dataset in the workbook. To change the dataset that the chart is based on, expand the dataset selector in the Dataset Panel and select a new dataset. Alternatively, if the dataset you want is not already used in the workbook you are in, click **Connect Dataset** to select from any of the datasets you have access to in your account, or create a new dataset.
2. Drag the desired fields from the Dataset Panel to the X-Axis, Series, or Measures tabs in the Layout panel.
 3. (Optional) In the Layout Panel, click **Create Calculated Measures** to create new measures from the fields available in the Dataset Panel. Calculated measures are represented by the calculator icon . For more information, see [Calculated Measures](#).
 4. (Optional) If you want to build your chart using two datasets, click the Dataset Menu icon and select **Create Dataset Link**. Depending on the common keys you select and where you place them in the layout, your results will be aggregated. For more information, see [Create Visualizations Based on Linked Datasets](#).
 5. Select the summary type and format options for any date or numerical fields you add to the chart.
 - a. Click the Field Menu icon next to the field you want to format in the Layout panel.
 - b. Select a summary type from the popup window.
 - c. (Optional) Click **Format...** to customize the numeric values for a field.
For more information about numeric formatting options, see [Customizing Numeric Values](#).
 - d. (Optional) Select **Currency...** to view the currency consolidation or conversion options for any fields with values in multiple currencies.
For more information, see [Currency Conversion from the User Interface](#).
 6. In the Layout panel, select the type of chart you want to produce from the popup window.
For information about each chart type, see the [Chart Types](#) topic.
 7. (Optional) Click the Properties subtab in the Layout panel to add a title and subtitle to the chart, or to rename each axis.
 8. Click the Refresh icon  to generate your chart.
 9. (Optional) Filter the data displayed in the chart:



Note: Filters applied on the Chart tab affect only the data displayed in the chart. No changes are made to the underlying dataset or any other workbook visualizations.

- a. Click the Field Menu icon  next to the field you want to create a filter for. Depending on whether the field has been defined as the X-Axis, Series, or Measure, or if you click the Field menu icon from the Fields List or the Layout Panel, the following options are available:
 - Top 10: display only the top 10 results based on the measures defined for the chart.
 - Bottom 10: display only the bottom 10 results based on the measures defined for the chart.

- Filter [Field Name] by...: enables you to define a custom measure-based filter for the selected chart.
 - Filter [Field Name]: enables you to define a custom value-based filter based on specific values within the chart results.
 - Add as Filter...: enables you to define a custom value-based filter based on specific values within the chart results.
- b. The results in the table are updated automatically.

For more information, see [Workbook Visualization Filters](#).



Important: If you build your chart using linked datasets, you must apply filters to the matching field in each dataset. If you do not, your results might include values that you wanted to filter. For more information, see [Create Visualizations Based on Linked Datasets](#).

Chart Types

There are multiple chart types you can use to visualize your data. To apply a chart type from either the Chart tab or the Analytics portlet, click the chart type dropdown and select an option from the popup window. Changing the chart type from the Analytics portlet does not affect the chart type selected for the corresponding workbook. Chart types let you compare values, and show trends. When you select a chart type, keep in mind the data included in the chart and the type of information you want to display. The following chart type options are available:

- Column Chart – Enables you to compare values for different categories or compare value changes over a period of time for a single category. For example, sales by period of time (weekly, monthly, quarterly, or yearly). You can also use column charts to display negative values and to show an overview of the highest and lowest categories.
- Bar Chart – Enables you to compare values offering the same capabilities of the column chart. You can use this chart type when the data labels are long as it improves readability. For example, when using regions or sales representatives as dimensions.
- Area Chart – Enables you to display trends and accumulative value changes over time, such as item stock, number of employees, or savings account.
- Line Chart – Enables you to display trends for a higher number of data points. For example, when showing more than 20 data points.
- Stacked Column Chart – Enables you to show the composition of the data, by displaying the proportion of each category to a whole. For example, you can use it to see the percentage of the base price and discount price by customer.
- Stacked Bar Chart – Enables you to display data in a bar chart emphasizing the composition, instead of comparison.
- Stacked Area Chart – Enables you to display data in an area chart showing how each category contributes to the total over a period of time.

To see some examples, see the following charts:

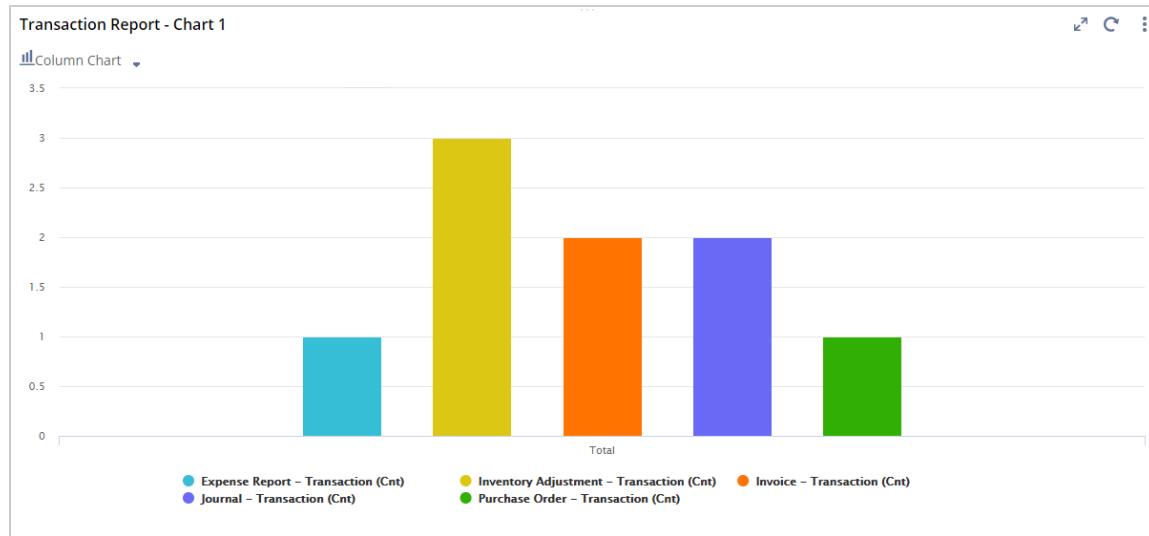
- [Open Sales Orders Lines Charts](#)
- [Number of Fulfillments Chart](#)

Chart-based Portlets

You can see your chart data on your home dashboard by adding the Analytics portlet. The portlet enables you to quickly see the chart on your dashboard and provides quick access to your workbook. Visualizing the data on your dashboard enables you to spend more time learning from it and less time gathering it.

You can also choose how you want to visualize your chart by setting layout options such as the portlet size and chart type. To identify the Analytics portlet on your dashboard, you can also enter a title. By default, the title displays the name of the workbook and the chart.

For more information about how to add and set up the Analytics portlet, see the help topic [Workbooks](#).



Workbook Visualization Filters

After you set up a table view, pivot table, or chart, you can filter the data that is displayed using several filter types: value-based, condition-based, measure-based, or date filters.

In table views and pivot tables, value-based filters are applied to every row and column and removes those that do not contain the selected value. Similarly, in charts, any columns, bars, or lines that do not contain the selected value are removed from the results.

In pivot tables, measure-based filters are applied to the row and column totals and remove any columns or rows that do not match the selected filter criteria. In charts, any columns, bars, or lines with totals that do not meet the selected criteria are removed. Measure-based filters are available in measure fields, also known as fields that require an aggregate function such as sum, count, average, and so on. Therefore, measure-based filters apply to charts and pivot tables only.

In all workbook visualizations, you can also create filters based on specific dates and conditions. In date fields, you can filter results by adding specific dates or select dynamic dates that update when the workbook is run. In numeric fields, you can add condition-based filters to show results that match the numeric range that you define.

There is no limit on how many value-based filters you can apply to a workbook visualization, however you can only apply one measure-based or date-based filter. Additionally, if you apply both value-based and measure-based filters, all value-based filters are applied first.

Filters that you apply to a workbook visualization only refine the results displayed in the selected visualization and do not affect the dataset that the visualization is based on. To filter the dataset, you must create criteria filters from the Dataset Builder. Additionally, if you are in a NetSuite account that has values in multiple currencies or a NetSuite OneWorld account with multiple subsidiaries, you must convert or consolidate these values before you can use them in a measure-based filter condition.

For more information about filtering a dataset, see [Dataset Criteria Filters](#).



Important: If you are working in a visualization based on linked datasets, you must apply dataset criteria filters to the matching field in each dataset. If you do not, your results might include values that you wanted to filter. For more information, see [Create Visualizations Based on Linked Datasets](#).

For more information converting or consolidating values in a workbook visualization, see [Currency in Datasets and Workbooks](#).

To learn how to use the workbook visualization filters, see the following:

- [Available Filters for Aggregate and Non-Aggregate Fields](#)
- [Value-based Filters](#)
- [Condition-based Filters](#)
- [Measure-based Filters](#)
- [Date Filters](#)

Available Filters for Aggregate and Non-Aggregate Fields

When you work with filters, each field displays different options depending on the type of field that you select (numeric field, date field, and so on) and how you use the available fields in your workbook visualizations. In charts and pivot tables, you can use some fields in two different ways:

- Use fields as aggregate fields – If you drag fields to the **Measure** section, the visualization displays results as an aggregate field. Therefore, results are grouped together to display a single summary result for several value entries.
- Use fields as non-aggregate fields – If you drag fields to the **Rows** or **Columns** sections (for pivot tables) or the **X-Axis** and **Color Stack** sections (for charts), each aggregate field displays a result for each value entry.

Using fields as aggregate or non-aggregate values has an impact on how your workbook visualizations display results. For example, if you want to see the sales amount broken by sales rep, your workbook visualization can display results in different ways. By using the Amount (Net) field as an aggregate field, also known as a measure field, the visualization displays a single result for all the sales transactions of a particular sales rep, based on the aggregate function that you select. For example, if you select the sum function, the result returns a single value for all the sales transactions. In this case, the field displays 10,000 USD. If you use the Amount (Net) field as a non-aggregate field, the visualization displays a result for each sales transaction of the same sales rep. In this case, the visualization displays the following field results: transaction A (9,000 USD), transaction B (600 USD), and transaction C (400 USD).

Value-based Filters

Value-based filters enable you to filter the data displayed in your workbook visualizations using specific values from any field that has been added to the Layout panel or Table Viewer. In pivot tables, value-based filters are applied to every row and column in the table and removes those that do not contain the selected value. In charts, any columns, bars, or lines that do not contain the selected value are removed from the results.

To apply a value-based filter to a workbook visualization:

1. Click the Field Menu icon next to any field in the Layout panel or Viewer and select **Filter [Field Name]...**
The Filter window appears.
2. In the Filter window, select the field values and operator to use in the filter condition.

3. (Optional) In the search box, you can enter the value that you are searching.

4. Click **OK**.

The results are updated and the selected filter criteria is displayed above the Viewer.

Condition-based Filters

You can use condition-based filters in numeric fields to show results that match a specific numeric range. To do so, you can select the operator that you want to apply (greater than, equal to, and so on), and then enter the numeric value. For example, you can see the opportunity amounts broken by sales representative, and show only results with a won probability above 80%. To see these results, in your workbook visualization you can add a condition-based filter to the probability (%) field, and set the condition values to "greater" and "80%".

Note: This option is available for percentage, integer and float fields that do not return currency values. If you want to apply this option to currency fields, customize your field using a TO_NUMBER formula function and changing the output type of the field. For more information about how to customize a currency field, see [Sample Formulas](#).

To apply a condition-based filter to a workbook visualization:

1. Click the Field Menu icon  next to a field label and select **Filter [Field Name]...**

The Filter window appears.

2. In the Filter window, select the operator to use in the filter condition, and then enter the numeric value.

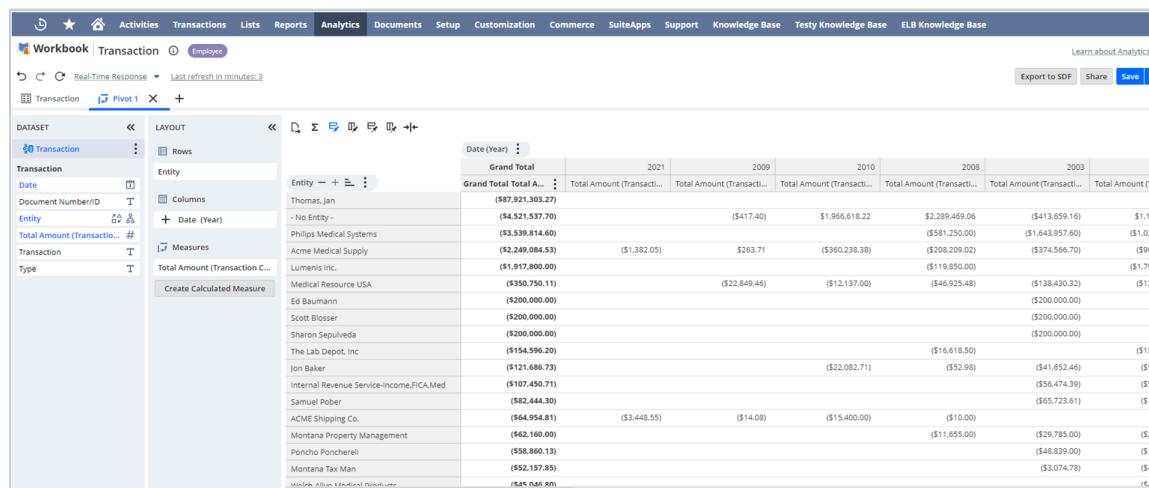
3. Click OK.

The results are updated and the selected filter criteria is displayed above the Viewer.

Measure-based Filters

In pivot tables, measure-based filters enable you to filter out entire columns or rows using filter conditions that are applied to the totals displayed in the table, based on the measures defined for the table. In charts, any columns, bars, or lines with total values that do not meet the selected criteria are removed from the chart.

For example, assume you have set up the following pivot table and you only want to see customers with over thirty thousand dollars in total transactions:



Dataset		Layout		Date (Year)					
Transaction		Rows		Grand Total					
Entity		Entity		Grand Total Total A...					
Date	Entity	Entity	Entity	Grand Total Total A...	2021	2009	2010	2008	2003
Document Number/ID	Entity	Entity	Entity	Grand Total Total A...	(\\$87,921,303.27)				
Total Amount (Transaction...)	Entity	Entity	Entity	(\\$4,521,537.70)	(\\$417,40)	\$1,966,618.22	\$2,289,469.06	(\\$413,659.16)	\$1,1
Type	Entity	Entity	Entity	(\\$3,539,814.60)		(\\$581,250.00)	(\\$1,643,957.60)	(\\$1,62	
	Entity	Entity	Entity	(\\$2,249,084.53)	(\\$1,382.05)	\$263.71	(\\$360,238.38)	(\\$208,209.02)	(\\$374,566.70)
	Entity	Entity	Entity	(\\$1,917,800.00)				(\\$119,850.00)	(\\$1,76
	Entity	Entity	Entity	(\\$530,750.11)	(\\$22,649.46)	(\\$12,137.00)	(\\$46,925.48)	(\\$138,430.32)	(\\$1,
	Entity	Entity	Entity	(\\$200,000.00)				(\\$200,000.00)	
	Entity	Entity	Entity	(\\$200,000.00)				(\\$200,000.00)	
	Entity	Entity	Entity	(\\$200,000.00)				(\\$200,000.00)	
	Entity	Entity	Entity	(\\$154,596.20)				(\\$16,618.50)	(\\$1,
	Entity	Entity	Entity	(\\$121,686.73)				(\\$22,082.71)	(\\$52,98)
	Entity	Entity	Entity	(\\$107,450.71)					(\\$56,474,39)
	Entity	Entity	Entity	(\\$82,444.30)					(\\$65,723.61)
	Entity	Entity	Entity	(\\$64,954.81)	(\\$3,448.55)	(\\$14,08)	(\\$15,400.00)	(\\$10,00)	
	Entity	Entity	Entity	(\\$62,160.00)				(\\$11,655.00)	(\\$29,785.00)
	Entity	Entity	Entity	(\\$58,860.13)					(\\$48,839.00)
	Entity	Entity	Entity	(\\$52,157.85)					(\\$3,074.78)
	Entity	Entity	Entity	(\\$4,160.00)					(\\$4,

To do so, you can apply a measure-based filter on the entity field of **Entity by Total Amount (Transaction Currency) (Sum) greater 30,000.00**.

The screenshot shows a SuiteAnalytics Workbook titled "Transaction". In the Layout panel, under the "ROWS" section, there is a dropdown menu with the option "Entity by Total Amount (Transaction Currency) (Sum) greater 30,000.00" highlighted with a red box. The main area displays a Pivot table with the following data:

	Grand Total	2021	2009	2010	2008	2003
Grand Total Total A...	\$30,042.16					
Andrew Goldwasser	\$31,091.97					
Justin Garcia	\$31,261.34					
Garrett Garrett, Esq.	\$31,732.72		\$230.00			
Aaron Rosewall-Godley Scherba and Salinas Medical ...	\$31,978.07	\$361.80	(\$15,505.55)	(\$123.45)		
Abdullah Bhupathiraju	\$33,446.76					
Andrew Kuykendall	\$35,436.53					
Billings Dental Clinic	\$35,676.00					
Twin Bridges Family Medicine	\$39,312.00					
Joshua Woodward	\$42,784.30					
Chris Lipscomb	\$50,000.00					
31 Test Company	\$65,159.46					
Lou Liang, MD	\$69,590.00					
59 Gerbera Unlimited	\$73,940.00					
Abe Simpson Abe Si... - Unassigned -	\$80,078.55					
Grand Total Total A...	\$71,687.50		\$8,047.05			

You can also apply a measure-based filter to a dimension to display a certain top or bottom of your choice, for example, **Entity by Total Amount (Transaction Currency) (Sum) TOP 10**.

The screenshot shows a SuiteAnalytics Workbook titled "Transaction". In the Layout panel, under the "ROWS" section, there is a dropdown menu with the option "Entity by Total Amount (Transaction Currency) (Sum) TOP 10" highlighted with a red box. The main area displays a Pivot table with the following data:

	Grand Total	2021	2009	2010	2008	2003
Grand Total Total A...	\$161,862.51	\$67.50				
Craig Kooser, DDS	\$323,027.95		\$157,422.97	\$164,116.93	\$1,339.30	
Abe Lincoln	\$360,000.00					
40 Test Cart	\$769,804.82					
Julie Kelly	\$1,428,151.76		\$11,248.52	\$210,120.06	\$8,079.53	
AAKASH CHEMICAL - Unassigned -	\$1,511,765.00					
Craig Erickson	\$1,647,123.26					
Cesar Petras	\$2,425,750.00					
Cameron Sardella	\$2,446,820.00					
Clayton Przybyla	\$3,255,991.76					
Grand Total Total A...	\$2,492,923.34					

In both cases, a filter is applied to the Entity rows based on their total amount (transaction currency) totals.

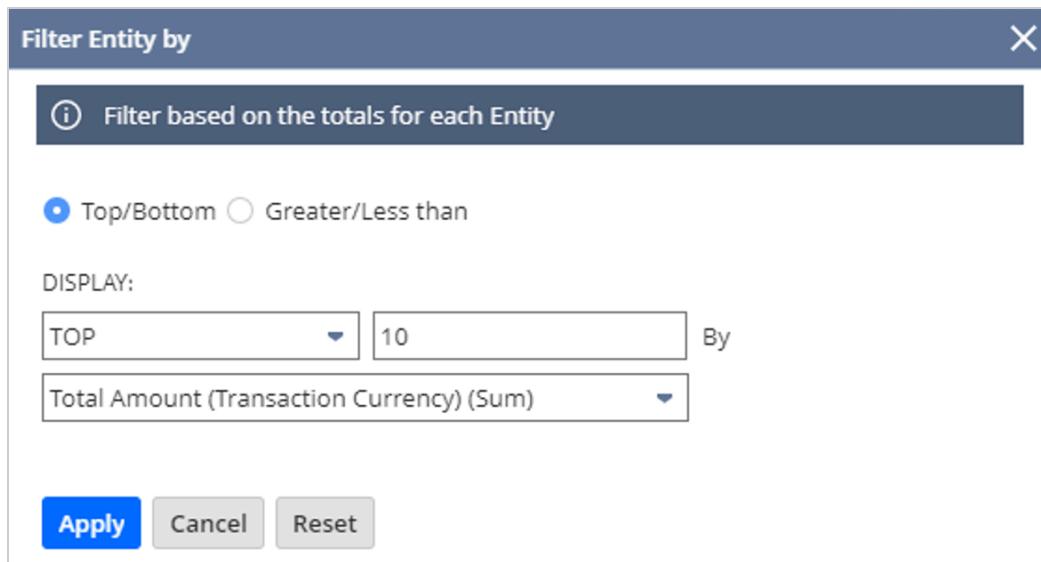
Note: If you are in a NetSuite account that has values in multiple currencies or a NetSuite OneWorld account with multiple subsidiaries, you must convert or consolidate these values before you can use them in a measure-based filter condition. For more information about currency conversion in SuiteAnalytics Workbook, see [Currency in Datasets and Workbooks](#).

Measure-based filters are available in measure fields, also known as aggregate fields, and apply to charts and pivot tables only. You can apply measure-based filters from a dimension or a measure field.

To apply a measure-based filter to your pivot table or chart from a dimension field:

1. Click the Field Menu icon next to a dimension field in the Layout panel or Viewer.
2. Select **Filter Top/Bottom**.

The Filter window appears.



3. In the Filter window, select **Top/Bottom** (default option) or **Greater/Less than**, then select the measure and set the conditions for the filter.
4. Click **Apply**.

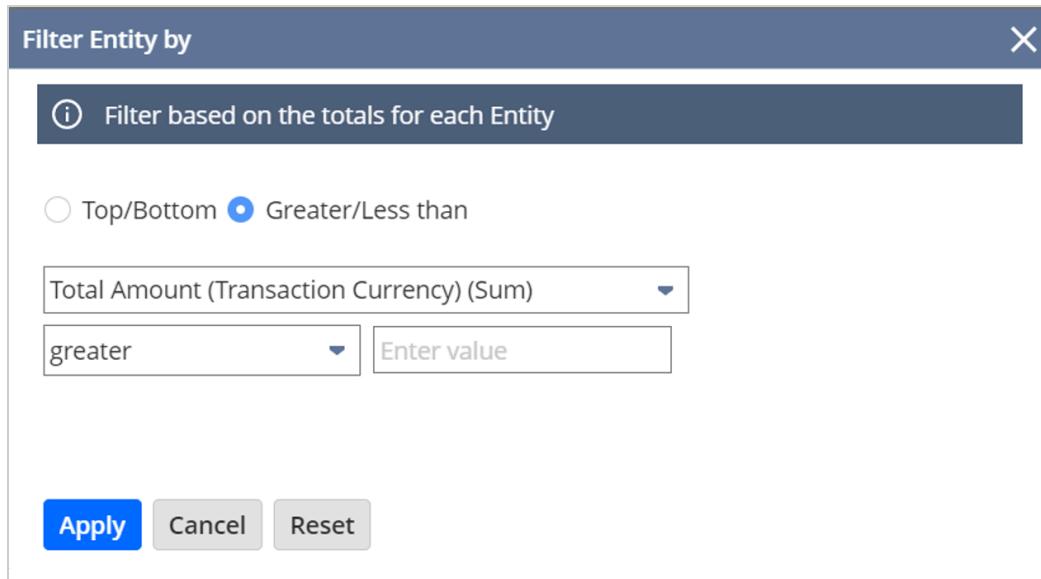
The table or chart automatically updates the results.

To apply a measure-based filter to your pivot table or chart from a measure field:

1. Click the Field Menu icon next to a measure field in the Layout panel. In pivot tables, you can also access the filter by clicking the Field Menu icon next to the **Grand Total** field in the Viewer, or from any measure column if there are no dimension fields displayed in the columns section.
2. Select **Filter by** and click the desired dimension.

Date (Year)	Grand Total	2021
Grand Total Total A...	(\$67,921,303.27)	Total Amount (Transact...
Lumenis Inc.	(\$350,750.11)	
Medical Resource USA	(\$200,000.00)	
Ed Baumann	(\$200,000.00)	
Scott Blosser	(\$200,000.00)	
Sharon Sepulveda	(\$154,596.20)	
The Lab Depot, Inc.	(\$121,686.73)	
Jon Baker	(\$107,450.71)	
Internal Revenue Service-Income:FICA,Med	(\$82,444.30)	
Samuel Pober	(\$64,954.81)	(\$3,448.55)
ACME Shipping Co.	(\$62,160.00)	
Montana Property Management	(\$58,860.13)	
Poncho Poncherelli	(\$52,157.85)	
Montana Tax Man	(\$4,040.00)	
Montana Allianz Financial Services		

The filter window appears.



3. In the Filter window, select **Greater/Less than** (default option) or **Top/Bottom**, then select the measure and set the conditions for the filter.
4. Click **Apply**.

The table or chart automatically updates the results.

Date Filters

Date filters enable you to filter your results based on a specific date, custom date, or dynamic date range that you choose. For example, you can select a dynamic date such as **today**, that update when the workbook is run. You can only apply one date filter per workbook visualization.

Note: Date filters are available in date fields and apply to all workbook visualizations (charts, pivot tables, and table views).

To apply a date filter to a workbook visualization:

1. Click the Field Menu icon  in the Dataset Panel or Layout Panel next to a date field and select **Filter Date> Advanced...**
2. In the Filter Date (Advanced...) window, use the **Specific Dates** tab to enter a custom date or the **Date Ranges** tab to select a dynamic date. Use the Values tab to select specific dates from the underlying dataset query results.
3. Click **Apply**.

The workbook visualization automatically updates the results.

The results are updated and the selected filter criteria is displayed above the Viewer.

Calculated Measures

A calculated measure is a measure that you can create in your pivot tables and charts using arithmetic operations with base measures. For example, you can use the following definition to calculate profit percentages:

$\% \text{ Profit} = \text{Est. Gross Profit (Sum)} / \text{Amount (Sum)}$

Unlike formula fields which are built at the dataset level to evaluate results on a row by row basis, calculated measures are built at the workbook level using aggregated results. For example, if you create a formula field that calculates sales amounts, each row in the dataset will show a single value based on the formula definition. If you then use that formula field as a measure in a pivot table or chart, the results are simply an aggregate of the formula field results. By contrast, calculated measures use the aggregated results of base measures combined with arithmetic operations to display results such as year over year variances and profit percentages. These types of results are not achievable using formula fields alone. For more information about the difference between formula fields and calculated measures, see [Formula Fields and Calculated Measures](#).

You can create as many calculated measures as you need using the fields available in the connected dataset, including custom formula fields. In visualizations based on linked datasets, you can even combine fields from both datasets to create a calculated measure. However, you can only use a calculated measure in the pivot tables or charts where you create them.

To create your calculated measures, see the following:

- [Working with Calculated Measures](#)
- [Supported Calculations](#)
- [Examples of Calculated Measures](#)

Working with Calculated Measures

Before you create a calculated measure, you should consider the fields and operators you want to include in the calculation. To learn about the type of fields and basic operators that you can use in your calculations, see [Supported Calculations](#).

You can define your calculated measure using fields from the dataset connected to the workbook visualization you are in. In visualizations based on linked datasets, this includes any combination of fields from either dataset. If the fields you want to use to build your measure are not available, edit the connected datasets or contact the dataset owner if you are not the original author. For more information, see [Editing a Dataset](#).

After you create a calculated measure, you can use it to sort and filter your results. Calculated measures support all sorting and filtering options available for base measures. You can also change the format of a calculated measure's numeric values, add totals and grand totals, and apply conditional formatting. For more information about these options, see [Workbook Visualization Filters](#), [Pivot Table Customization](#), and [Conditional Formatting](#).

To open the calculated measure editor, click Create Calculated Measure in the Layout panel of any pivot table or chart. You can create as many calculated measures as you need, however you can only use them in the workbook visualization where you created them. If you want to use the same calculated measure in other visualizations, copy and paste the measure definition.

To learn how to work with calculated measures, see the following:

- [Creating Calculated Measures](#)
- [Calculated Measure Editor](#)

Creating Calculated Measures

When you create a calculated measure, you define the fields and basic operators of your calculation. All calculated measures must contain a name and a valid definition. As you create your calculated measure,

a message at the bottom of the editor indicates if the definition is valid. If the definition is not valid, an error icon appears next to the line that contains validation errors. Hover over the error icon to see how to correct the definition.

After you create your measure, you can edit and delete it from the Layout panel of the visualization.

To create a calculated measure:

1. On a pivot table or chart, click **Create Calculated Measure** from the Layout panel.

The calculated measure editor appears and the Definition field shows instructions on how to define your calculations.

2. In the Name field, enter the name of your calculated measure.
3. Add fields to the editor:
 - Drag the desired fields from the Dataset Panel to the Definition field. In visualizations based on linked datasets, you can switch between the fields within each dataset by clicking the dataset name.
 - In the Definition field, start typing the field names.

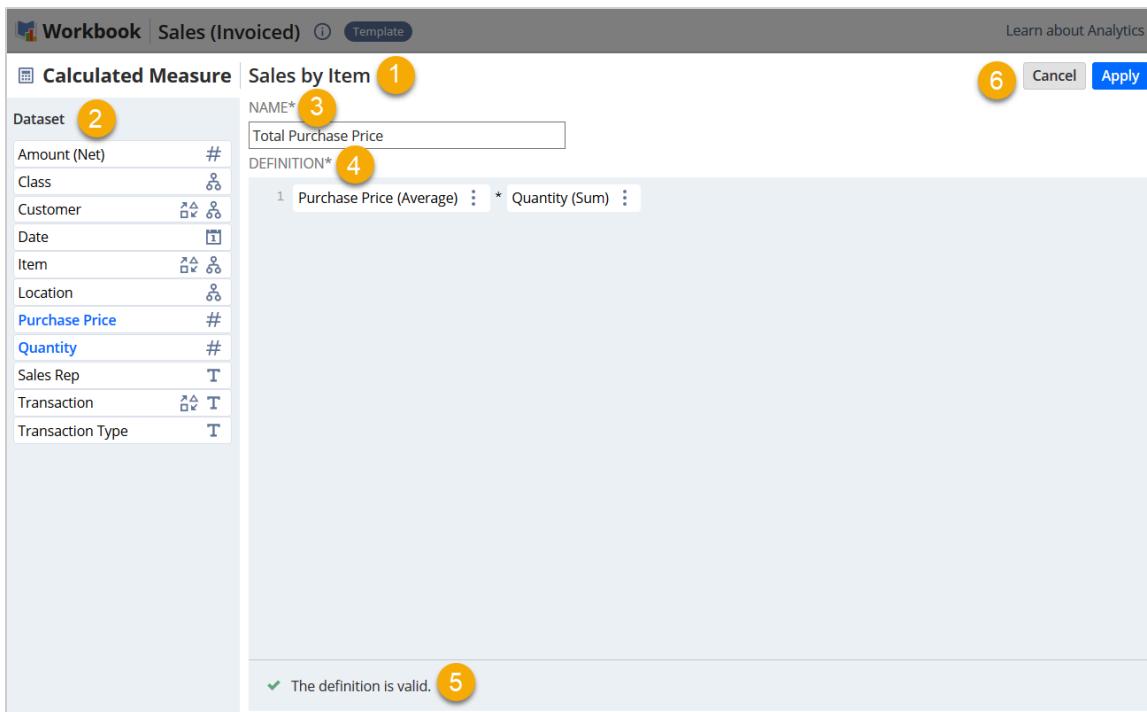
Fields that you use in your calculated measure are highlighted in blue in the Dataset panel.

4. Add calculation operators:
 - In the Definition field, type a basic operator such as plus (+), minus (-), multiplied by (*), divided by (/), or parenthesis ().
 - Press Ctrl + Space to display all options and select the desired operator.
 5. (Optional) To make changes to one of the fields used in the definition, click the Field Menu icon . The following options appear:
 - If you want to change the aggregate function of the field, select a different function from the list.
 - If you are working with fields that have values in multiple currencies, select **Currency...** to view the consolidation and conversion options. For more information, see [Currency Conversion in Workbook](#).
 6. (Optional) To remove a field from the definition, place the cursor next to the field and press Backspace, or click the Field Menu icon and select **Remove**.
 7. Click **Apply** to add the calculated measure to your pivot table or chart.
- Your calculated measure appears in the list of measures in the Layout panel with a Calculator icon .
8. On your pivot table or chart, click the Refresh icon  to see the results of the calculated measure in the viewer.

Calculated Measure Editor

The Calculated Measure editor is where you define new calculated measures.

The elements of the editor are identified in the image below:



1	Workbook Visualization Name - Displays the name of the visualization where you are creating the calculated measure.
2	Dataset Panel - Contains all of the fields included in the connected dataset. In visualizations based on linked datasets, both datasets are shown, enabling you to switch between them. To view additional information about a field, point to the field and click the Information icon on the right.
3	Calculated Measure Name - Type the name of your calculated measure.
4	Definition Editor - Create your calculations with the fields available in the Dataset Panel and basic operators. To create your calculations, you can do the following: <ul style="list-style-type: none"> ■ Type a field name in the editor or drag a field from the Dataset panel to the editor. ■ Type a basic operator such as plus (+), minus (-), multiplied by (*), divided by (/), and parenthesis (). ■ (Optional) To change the aggregate function of the selected field, click the Field Menu icon next to the field name. ■ (Optional) To display all options, press Ctrl + Space and select the fields and operators that you want to add.
5	Validation Message - Displays a message that the definition is valid or that the definition contains errors. If the definition is not valid, an error icon appears next to the line that contains validation errors. Hover over the error icon to see how to correct the definition.
6	Editor Menu - You can do the following: <ul style="list-style-type: none"> ■ Click Apply to leave the editor and see the in your workbook visualization. You can only apply the calculated measure if the name and the definition are valid. ■ Click Cancel to go back to your workbook visualization without applying the changes that you made in the editor.

Supported Calculations

In your calculations, you can use different basic operators depending on the type of data within each field you use in the measure definition: numeric, single currency, multiple currency, or duration.

To determine which measures and basic operators you can use when you define your calculated measures, see the supported calculations for each data type.

- Numeric values
- Single currency values
- Multiple currency values
- Duration values

Numeric values

In calculated measures, numeric values are those that return percentage, integer, and float values. They do not return currency results.

This table shows the type of measures and basic operators that you can use with calculated measures that return numeric values.

Data Type	Plus (+) Minus (-)	Multiplied by (*)	Divided by (/)
Single currency values	✓	✓	—
Multiple currency values	—	✓	—
Duration values	—	✓	—
Numeric values	✓	✓	✓

i Note: If the calculation that you want to use is not supported for multiple currency values but is supported for single currency values, consider consolidating or converting the selected field to a single currency field. For more information, see [Currency Consolidation in Workbook](#).

Returned format for percentage values

When you combine percentage values with integer or float values in your calculation, the result is displayed as a decimal value.

For example, if a field with the percentage value **200%** is divided by the integer value **10**, the operation calculates **2.0** divided by **10**. Therefore, the result returns the decimal value **0.2**.

$$0.2 = 2.0 / 10$$

i Note: The result that you obtain is not a percentage value. The decimal value **0.2** corresponds to the percentage value **20%**. If you want to see results as percentage values in your pivot tables and charts, you can change them to percentages. For more information see, [Pivot Table Customization](#).

Single currency values

Single currency values are those that return values in a single currency, such as dollar amounts.

This table shows the type of measures and basic operators that you can use with calculated measures that return single currency values.

Data Type	Plus (+) Minus (-)	Multiplied by (*)	Divided by (/)
Numeric values	✓	✓	✓
Multiple currency values	✓	—	—

Data Type	Plus (+) Minus (-)	Multiplied by (*)	Divided by (/)
Duration values	—	—	—
Single currency values	✓	—	✓

Multiple currency values

Multiple currency values are those that return values in multiple currencies, such as dollars and British pounds.

This table shows the type of measures and basic operators that you can use with measures that return multiple currency values.

Data Type	Plus (+) Minus (-)	Multiplied by (*)	Divided by (/)
Numeric values	—	✓	✓
Single currency values	✓	—	—
Duration values	—	—	—
Multiple currency values	✓	—	—

i Note: If the calculation that you want to use is not supported for multiple currency values but is supported for single currency values, consider consolidating or converting the selected field to a single currency field. For more information, see [Currency Consolidation in Workbook](#).

Returned values for multiple currency calculations

The results that you obtain depend on how the measure definition is built:

- Calculations with numeric values** - When multiple currency values are multiplied by numeric values, the calculation returns a result for each currency. For example, if a field with the numeric value **100** is multiplied by a field with the multiple currency values **10 USD** and **20 EUR**, the calculation returns a result for each currency: **1000 USD** and **2000 EUR**.

$1000 \text{ USD} = 100 * 10 \text{ USD}$

$2000 \text{ EUR} = 100 * 20 \text{ EUR}$

i Note: This also applies to multiple currency values divided by numeric values.

- Calculations with single currency values** - When you use multiple currency values and single currency values in a measure definition, the calculation only returns a result for the values that share the same currency. For example, a field with the multiple currency values **100 USD** and **200 EUR** plus a field with the single currency value **20 EUR** returns **100 USD** and **220 EUR**.

$100 \text{ USD} = 100 \text{ USD}$

$220 \text{ EUR} = 200 \text{ EUR} + 20 \text{ EUR}$

i Note: This also applies when you calculate multiple currency values minus single currency values.

Duration values

Duration values are those that return values in units of time such as hours, minutes, and seconds.

This table shows the type of measures and basic operators that you can use with calculated measures that return duration values.

Data Type	Plus (+) Minus (-)	Multiplied by (*)	Divided by (/)
Numeric values	—	✓	✓
Single currency values	—	—	—
Multiple currency values	—	—	—
Duration values	✓	—	✓

Examples of Calculated Measures

The following examples show how you can use calculated measures in your pivot tables and charts.

Year over Year Growth

You can create a calculated measure to compare the year over year difference for your sales reps' performance. The following steps show you how to create a pivot table with a calculated measure that returns this difference.

Step	Description
Step 1: Create two formula fields using the Sales (Ordered) dataset	<p>To calculate the year over year variance, you need to split the sales amount into two separate fields in the dataset. You can create the two following formula fields:</p> <ul style="list-style-type: none"> ■ Amount LYTD - The Amount Last Year to Date field returns the sales amounts from all of last year up until today's date. ■ Amount TYTD - The Amount This Year to Date field returns the sales amounts for transactions created from the beginning of this year to today's date. <p>To copy and paste the formulas, see Calculating Amounts for Relative Date Ranges.</p>
Step 2: Create your pivot table	<p>Set the following fields to the corresponding section of the Layout panel:</p> <ul style="list-style-type: none"> ■ Rows: Sales Rep ■ Columns: None ■ Measures: Amount TYTD (Sum) and Amount LYTD (Sum) <p>For information about how to create pivot tables, see Workbook Pivot Tables.</p>
Step 3: Create your calculated measure	<p>Use the following definition:</p> $(Amount TYTD (Sum) - Amount LYTD (Sum)) / Amount LYTD (Sum)$ <p>For information about how to create calculated measures, see Working with Calculated Measures.</p>
Step 4: Format the results	<p>You can also customize your pivot table results, add totals and grand totals, or set the decimal values to percentage values. For more information, see Pivot Table Customization.</p>

After you create the calculated measure, the pivot table displays the difference over time.

The screenshot shows a Power BI Workbook titled "Sales YoY". The ribbon at the top includes "Employee" and "Shared with me" buttons. Below the ribbon, tabs for "Year Over Year Variance", "YoY", "Chart 1", "Pivot 2", "Tab 1", and "+" are visible. The main area is divided into two sections: "DATASET" and "LAYOUT".

DATASET:

- Sales YoY**:
 - Amount
 - Amount (Net)
 - Amount LYTD*
 - Amount TYTD*
 - Class
 - Customer
 - Date
 - Department
 - Est. Gross Profit
 - Item
 - Location
 - Quantity
 - Sales Channel
 - Sales Rep
 - Transaction Type

LAYOUT:

- Rows:** Sales Rep
- Columns:** Drop fields
- Measures:** Amount TYTD* (Sum), Amount LYTD* (Sum), YoY Variance %

A table is displayed with the following data:

Sales Rep	Amount TYTD* (Sum)	Amount LYTD* (Sum)	YoY Variance %
Susan Miller	51,520.00	31,540.00	63.35 %
Jasmin Coke	68,130.00	50,434.00	35.09 %
Bill Clark	26,752.00	18,216.00	46.86 %
Timothy Little	13,604.00	37,760.00	-63.97 %
Brent Raines	298,940.00	87,229.00	242.71 %
Alex Fabre	41,240.00	56,760.00	-27.34 %
Evaline Mann	30,619.00	21,016.00	45.69 %
Bob Sales	100,130.00	60,424.00	65.71 %
Grand Total	630,935.00	363,379.00	73.63 %

For more information about comparing sales amounts over time using calculated measures, see the following video:



Conditional Formatting

Conditional formatting enables you to highlight your workbook visualization results based on rules that you define. Currently, you can only apply conditional formatting to table views and pivot tables using fields that contain NUMBER or STRING values. This includes standard, custom, custom formula, and calculated measure fields. Fields with DATE values are not supported.

Conditional formatting does not differentiate between values in multiple currencies. For example, if you define a conditional formatting rule to highlight results greater than 10 dollars, results with values greater than 10 euros are also highlighted. To accurately highlight results with values in multiple currencies, you should convert or consolidate the values before you apply conditional formatting. For more information, see [Currency in Datasets and Workbooks](#).

To apply conditional formatting to fields that contain percentage values, you must use decimal values in the rule definition. For example, if you want highlight results greater than 20%, use the value 0.2 in your rule definition.

The current iteration of conditional formatting does not support filtering results by color or icon. If you apply conditional formatting to a field containing numeric values however, you can sort the highlighted results in ascending or descending order to simulate filtering.

Conditional formatting that you apply a table view or pivot table is automatically applied to any Analytics portlets based on that visualization. To turn off conditional formatting in the portlet, you must edit the visualization in the associated workbook. For more information, see the help topic [Opening Workbooks from the Analytics Portlet](#).

- Conditional Formatting in Table Views
 - Conditional Formatting in Pivot Tables

Conditional Formatting in Table Views

In table views, you can highlight any column that you add to the table if it contains NUMBER or STRING values.

To apply conditional formatting to a table view:

1. Add the field you want to highlight to the table.
2. Click the Field Menu icon in the column header and point to **Conditional Formatting**.
3. Click **Manage Conditional Formatting**.
4. Select the operators, values, and color or icon for the rule. You can click the add icon  to create multiple rules and apply different colors or icons to the same column.
5. Click **Apply**.
6. Repeat steps 1–5 for each column you want to highlight.

Conditional Formatting in Pivot Tables

In pivot tables, you can highlight measures, subtotals, and grand totals. Highlighting is only applied to individual cells. Currently, you cannot highlight entire rows.

To apply conditional formatting to a pivot table:

1. Add the field you want to highlight to the measures section of the Layout panel.
2. Click the Field Menu icon next to the field in the Layout panel and point to **Conditional Formatting**.
3. Click **Manage Conditional Formatting**.
4. Select the operators, values, and color or icon for the rule. You can click the add icon  to create multiple rules and apply different colors or icons to the same measure.
5. Check **Apply to subtotals and grand totals** if you want to apply your rules to the measure subtotals and grand totals.
6. Click **Apply**.
7. Repeat steps 1–6 for each measure you want to highlight.

Dataset Linking in SuiteAnalytics Workbook

Watch the following help video for an introduction to dataset linking in SuiteAnalytics Workbook:



[Working with Linked Datasets in SuiteAnalytics Workbook](#)

You can link two datasets if they each have at least one field that shares common data, such as date. You can do this even if the datasets are based on record types that do not have predefined relationships in the analytics data source.

For example, assume you want to compare sales rep quotas to their actual sales. In this case, you would create two datasets: one based on the sales (invoiced) record type and the other based on the quota

record type. These record types have fields with common data such as date and department, however you cannot join them in a dataset because they do not have predefined relationships in the analytics data source. More importantly, even if you could join them, the one-to-many relationship of sales reps to transactions would produce unwanted data duplication of the results.

This happens because there is no data aggregation in a dataset and all possible results are displayed. This is intentional, since datasets are meant to be used as source data for visualizations such as pivot tables and charts where you can define the aggregation. For more information, see [Data Duplication Based on Record Joins](#).

By contrast, with linked datasets aggregation occurs immediately before your results are generated based on the fields you assign as common keys and how you use them to build your visualization. For example, in the case of sales quotas versus actual sales, you can create a dataset link using the sales rep, accounting period, and posting period fields. Because quotas are created on a monthly basis, if you then use the accounting period field from the quota dataset as a dimension for your pivot table, the results are aggregated at the month level before they are rendered in the visualization.

For more information about linking datasets in SuiteAnalytics Workbook, see the following topics:

- Data aggregation is not the only difference between linking datasets in a workbook and joining record types in a dataset. To learn more, see [Joining Record Types Versus Linking Datasets](#).
- To link datasets in a workbook, you need to define common keys between each dataset as part of a link definition. Common keys need to contain the same values and be of the same data type, however you can use matching fields with different data types if you cast their values using a custom formula field. To learn more about creating links and defining common keys, see [Link Datasets in a Workbook](#).
- After you link two datasets in a workbook, you can use them to build pivot tables and charts. Table views are currently not supported. For more information, see [Create Visualizations Based on Linked Datasets](#).

Joining Record Types Versus Linking Datasets

In SuiteAnalytics Workbook, joins within a dataset are realized through SQL queries that are executed in the background. Specifically, a left outer join is executed for each record type you add to a dataset. This means that the results include all the rows from the primary record type, plus any rows that match the join condition from the joined record type. No aggregation is performed, and some results may be overstated or understated depending on the relationship between the record types and the order that you join them in. For more information, see [Guidelines for Joining Record Types in SuiteAnalytics Workbook](#).

By contrast, linked datasets behave more like an SQL union. In a union, two queries are executed to produce one set of results, with the union being defined by a set of fields that share the same data type within each query. In Workbook, this means that each dataset you link is executed as a query in the background, with the results being aggregated and displayed based on the fields you define as common keys when you create the link.

To summarize, joins in a dataset enable you to combine and analyze fields from multiple record types in a single set of results. Linked datasets enable you to analyze and aggregate the results of two queries as though they were executed as one query. For more information about linking datasets and common keys, see [Link Datasets in a Workbook](#).

To learn more about the differences between joining record types in a dataset and linking datasets in a workbook, see the following table:

	Joining Record Types	Linking Dataset
--	----------------------	-----------------

SQL execution	Left outer join. Results include all primary record fields, plus matching joined record fields.	Union. The results from two datasets are displayed as one visualization, with only distinct values included in the results and aggregation based on the fields you define as common keys when you create the link. For more information, see Common Keys .
Number of data sources	Unlimited. You can join any number of record types in a dataset, however too many joins can result in performance issues.	Two datasets.
Data duplication	Data duplication is possible depending on the relationship between the record types you join, and the order in which you join them. For more information, see Data Duplication Based on Record Joins .	No. Results only contain distinct values from each dataset, so there is no data duplication.
Aggregation	No. This is intentional, since datasets are meant to be used as raw data for your visualizations.	Yes. Aggregation occurs automatically based on the fields you define as common keys and where you place them in the visualization.
Sharing	Yes. After you join record types in a dataset, you can save and share it with other users in your account.	Yes, but only as part of the workbook they are linked in.
Requirements	<ul style="list-style-type: none"> ■ Access to record types ■ Predefined relationship between record types in the analytics data source. For more information, see <Analytics Data Source>. 	<ul style="list-style-type: none"> ■ Access to datasets and underlying record types. ■ At least two datasets connected to the workbook. ■ At least one common key. For more information, see Common Keys.

Link Datasets in a Workbook

To link datasets in a workbook, you must first connect at least two datasets to a workbook and define at least one set of common keys between each dataset. Common keys are fields that contain the same data within each dataset, in the same data type. For example, the sales rep field in the sales (invoiced) and transaction record types both contain sales representative names in a text format and could be used as common keys when defining a link.

When you create a link, preferred common keys are automatically defined for fields with matching names and data types. If you want to define your own common keys, you can use any fields from either dataset if they are of the same data type: either date, text, numeric, or polymorphic. If the fields you want to use are not the same data type, you can cast their values using a custom formula field with functions such as TO_NCHAR. For more information, see [Common Keys](#).

To prevent linking errors, when you are defining your common keys, after you select a field from the first dataset, any fields with a different data type in the second dataset become unavailable. Additionally, after you create a link, you cannot unlink the datasets if they have been used anywhere within the workbook. Linked datasets can also only be shared as part of the workbook they were linked in. You cannot share them on their own, as a separate object.



Important: If two fields in a dataset share the same ID but have different join paths, you may experience issues if you try to define them as common keys. To prevent errors, try deleting one of the fields from the dataset before creating the link.

To link datasets in a workbook:

1. In the Dataset Panel of a workbook, click **Connect Dataset**.
2. Select a dataset from the list or click **New Dataset** if you want to define a new one. For more information see [Defining a Dataset](#).
3. After you have at least two datasets connected to the workbook, click the Dataset Menu icon next to a dataset in the Dataset Panel and select **Create Dataset Link**.
4. On the Dataset Linking page, double click or drag and drop the fields you want to define as common keys to the Common Keys area of the page.



Important: Any fields you plan on using as dimensions in your pivot table or chart need to be defined as common keys for your results to render properly. For more information, see [Common Keys](#).

5. Click **Link Datasets**.

After you link your datasets, you can use them to build a pivot table or chart in a workbook. For more information, see [Create Visualizations Based on Linked Datasets](#).

You can also build calculated measures using fields from either of the linked datasets. For more information, see [Calculated Measures](#).

Common Keys

Common keys are fields that contain the same data in each dataset, that you use to define a dataset link. To properly display and aggregate results in visualizations based on linked datasets, any fields that you plan to use as a dimension must also be defined as a common key. If you do not do this, when you add the field as a dimension, your results may include blank cells. This happens because without a matching key field, as opposed to displaying only matching results from both datasets, all possible values from one dataset are included in the results.

For example, assume you are in a workbook with linked budget and sales datasets. The posting period and accounting period fields, and department fields from each dataset are defined as common keys to create the link. There is also an item field in each dataset, but it is not defined as a common key.

The screenshot shows the 'Dataset Linking' dialog box. It has three main sections: 'Dataset 1' (Budget) and 'Dataset 2' (Sales), both with dropdown menus and search bars; and a 'Common Keys' section. In the 'Common Keys' section, three fields are selected: 'Accounting Period', 'Department', and 'Posting Period'. Each selected field has a 'Drop fields' button to its right. At the top right of the dialog are buttons for 'About Dataset Linking', 'Link Datasets', and 'Cancel'.

When you add the posting period field to the table rows then refresh the table, the results are generated as though an SQL union was executed in the background. That is, each dataset is executed as a query in the background, and the results for both the matching posting period and accounting period fields values are displayed. Because posting periods are defined at the monthly level, the results are also aggregated at the month level before they are rendered.

The screenshot shows the 'DATASET' pane on the left with the 'Sales' and 'Budget' datasets selected. The 'LAYOUT' pane on the right shows a pivot table structure. The 'Rows' section contains the 'Posting Period' field, which is highlighted with a red box. The 'Columns' section contains 'Sales (Sum)' and 'Sales budget (Sum)'. The main area displays a table with three rows: Jan 2021 (\$77,353.00, 75,000.00), Feb 2021 (\$80,109.00, 75,000.00), and Mar 2021 (\$83,022.00, 75,000.00).

The same occurs when you add the department field from either dataset to the pivot table, with matching results being displayed.

This screenshot shows the 'DATASET' pane with the 'Sales' and 'Budget' datasets selected. In the 'LAYOUT' pane, the 'Posting Period' field is in the 'Rows' section, and the 'Department' field is in the 'Columns' section. The main area shows a pivot table with three rows for Jan, Feb, and Mar 2021. For each month, there are two columns: 'Marketing' and 'Sales'. Each column has 'Sales (Sum)', 'Sales budget (Sum)', and 'Variance' sub-fields.

If you add the item field from the sales dataset however, the results will include blank cells. This happens because the union does not recognize the matching item field in the budget dataset, since it was not defined as a common key.

This screenshot shows the 'DATASET' pane with the 'Sales' and 'Budget' datasets selected. The 'LAYOUT' pane shows 'Posting Period' in the 'Rows' section and 'Item' in the 'Columns' section. The main area displays a pivot table with three rows for Jan, Feb, and Mar 2021. For each month, there is a single column labeled 'Item'. Under 'Item', the first row is 'Compaq Evo n800c'. The 'Sales (Sum)' and 'Sales budget (Sum)' cells are filled with values, but the 'Variance' cell is empty. The 'Grand Total' row shows the sum of all items.

To prevent this, when you create a new link Workbook automatically defines common keys using fields with matching names and data types. If the fields you want to define as common keys do not have matching data types, you can also create a custom formula field using functions such as TO_NCHAR to cast the field values to a different data type. For more information, see [Sample Formulas](#).

After you define your common keys, use caution when you edit the associated datasets. If you delete a field from a dataset after it is defined as a common key, it can cause issues in your workbooks and prevent you from saving. Additionally, if you create a filter in one dataset based on a field that is defined as a common key, make sure you apply the same filter to its matching key field in the other dataset. If

you do not, only results from one dataset will be filtered. Additionally, any calculated measures you have created will only be rendered at the total and grand total levels.

Create Visualizations Based on Linked Datasets

You can use linked datasets to create pivot tables and charts within a workbook. Table views do not support data aggregation, so you cannot build them using linked datasets.

As with visualizations based on individual datasets, you build visualizations based on linked datasets by assigning fields as dimensions and measures in the Layout panel. Any fields defined as common keys as part of the link definition are denoted by a chain link icon in the Dataset panel. To properly render and aggregate your results, any fields you use as a dimension should also be defined as a common key. If they are not, your results might include blank cells. For more information, see [Common Keys](#).

Additionally, if you apply a filter to a field that has been defined as a common key, make sure you apply the same filter to its matching key field. If you do not, only results from one dataset will be filtered.

To create a visualization based on linked datasets:

1. Click the **Add** icon from anywhere within the workbook.
2. Select the type of visualization you want to create.



Note: Table Views do not support linked datasets.

3. Drag fields from the Dataset Panel to the Layout Panel. To switch between datasets, click the dataset name



Important: Any fields you assign as dimensions should also be defined as common keys in the link definition. This dictates how your results are aggregated, and prevents empty cells. For more information, see [Common Keys](#).

4. (Optional) Create a calculated measure using fields from either dataset. For more information, see [Calculated Measures](#).
5. (Optional) Filter your results. For more information, see [Workbook Visualization Filters](#).



Important: If you want to filter your results using a dataset filter and you use a field that has been defined as a common key, make sure you apply the same filter to the matching key field in the other dataset. If you do not, only results from one dataset will be filtered. Additionally, any calculated measures you have created will only be rendered at the total and grand total levels.

6. Click the **Refresh** icon to generate the results.

Currency in Datasets and Workbooks

If you are in a NetSuite account with transactions in multiple currencies, or a NetSuite OneWorld account with multiple subsidiaries, amount fields in your datasets and workbooks might display data in multiple currencies. To perform arithmetic operations and view totals for these fields, you must consolidate or convert the field values to a single currency. This is particularly important if you are working with the transaction, transaction line, or transaction accounting line record types.

- In transaction accounting line, monetary values in amount fields are stored in the base currency of the subsidiary assigned to the transaction. The amount fields in this record type can only be consolidated

and are typically used for accounting purposes, such as calculating financial results for your entire company, including all subsidiaries. In Workbook, currency consolidation consolidates the values of a field to the currency of the lowest level subsidiary that is a common parent to all the subsidiaries that you have access to in your account. This is identical to how currency consolidation behaves in NetSuite saved searches and reports. By default, when you include amount fields from the transaction accounting line record type in workbook table views, pivot tables, and charts, currency consolidation is enabled.

- In transaction and transaction line, monetary values in amount fields are stored in the currency of the transaction. These values can only be converted and are typically used for operational use cases, such as analyzing total sales of a specific item. Currency conversion is an additional tool available in Workbook, which enables you to translate the values of a field from one currency to another currency that you have set up in your account.

You can apply currency consolidation or conversion to applicable fields directly from the user interface in any workbook visualization. To consolidate or convert field values in a dataset, you must use a custom formula field.

For more information about amount fields in each transaction record type, see the video  [Transaction Data Model in 2020.1](#).

For more information about currency consolidation and conversion in Workbook, see the following topics:

- [Currency Consolidation in Workbook](#)
- [Currency Conversion in Workbook](#)

Currency Consolidation in Workbook

Currency consolidation in Workbook consolidates the values of a field to the currency of the lowest level subsidiary that is a common parent to all the subsidiaries that you have access to in your account. The rate type used for the consolidation is based on the general rate type set up for each account in your chart of accounts: either Historical, Average, or Current. This is the same as the Per-Account option used for exchange rate types in transaction saved searches. For more information, see the help topic [Consolidated Exchange Rate Types for Transaction Searches](#).

 **Note:** In Workbook, if **Current** is selected for an account, you cannot choose the exact period or date to be used for the exchange rate. Instead, the rate is based on the date you run the workbook visualization that the field being consolidated is in. For more information, see the help topic [Consolidated Exchange Rates](#).

In Workbook, you can only apply currency consolidation to amount fields from the transaction accounting line record type. Like with saved searches and reports, you can also only apply consolidation if you are in NetSuite OneWorld account with multiple subsidiaries. To ensure that the correct rates are applied during consolidation, make sure that your chart of accounts and consolidated exchange rates are set up correctly. You should also verify that your currency exchange rates are set up correctly since they are used in some consolidated exchange rate calculations. For more information about currency exchange rates, see the [Currency Exchange Rates List Page](#).

Currency Consolidation Using Custom Formula Fields

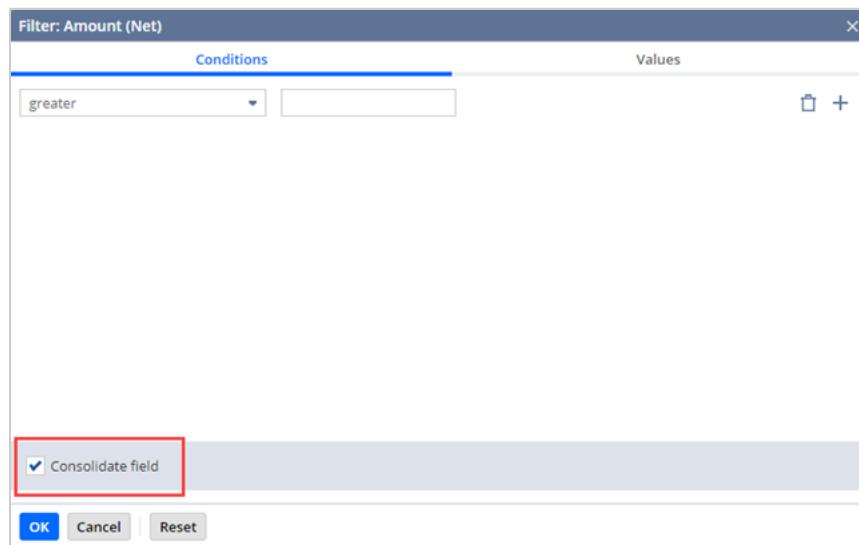
To apply currency consolidation in a dataset, you must create a custom formula field using the context value `#currency_consolidated`. For example, to consolidate the values of the amount net field from the transaction accounting line record type, use the following formula definition:

```
1 | TO_NUMBER({transactionlines.accountingimpact.netamount#currency_consolidated})
```

Currency Consolidation from the User Interface

In workbook visualizations with amount fields from the transaction accounting line record type, currency consolidation is enabled by default. If you deactivate it, click the Field Menu icon  next to a transaction accounting line field in the Layout Panel or column header, then point to **Currency** and click **Apply Consolidation**. If you are creating a calculated measure that includes fields with values in multiple currencies, you can also apply consolidation as part of the measure definition. For more information, see [Calculated Measures](#).

In the Dataset Builder, the Filter window also includes a Consolidate field box for when you create criteria filters based on transaction accounting line amount fields.



When you check the Consolidate field box, the values of the field are consolidated before the selected filter conditions are applied. The rate type used for the consolidation is based on the general rate type set up for each account in your chart of accounts: either Historical, Average, or Current.



Important: The Consolidate field box is only available for fields from the transaction accounting line record type. To create criteria filters based on converted values from other record types, you must first define a custom formula field that applies currency conversion to the field, then create your filter conditions based on the custom formula field. For more information, [Currency Conversion Using Custom Formula Fields](#).

Currency Conversion in Workbook

Currency conversion in Workbook enables you to convert the values of a field to any currency that you have set up in your account. To apply currency conversion in a dataset or workbook, you must have the Multiple Currencies feature enabled in your account. Additionally, you should make sure that your currency records and currency exchange rates are set up correctly to ensure correct conversion results. For more information, see the help topic [Currency Exchange Rates](#).

Currency Conversion Using Custom Formula Fields

In the Dataset Builder, you can create custom formula fields using the context value #converted to convert field values that are in multiple currencies.

For example, to display the remaining balances on the invoices in your account regardless of the original currency used on each invoice, you can create a custom formula field using the following definition:

```
1 | TO_NUMBER({foreignamountpaid#converted}) - TO_NUMBER({foreignamountunpaid#converted})
```

The function TO_NUMBER is required in the definition because the current version of the Formula Builder does not support the CURRENCY output type, and TO_NUMBER casts these values to the FLOAT output type. In single currency accounts for example, you could use the following formula definition without the #converted context value:

```
1 | TO_NUMBER({foreignamountpaid}) - TO_NUMBER({foreignamountunpaid})
```

Depending on your formula definition, you can convert the values of a field to the base currency of your account or a specific currency that you choose. You can also specify the exchange rate based on the date you enter in the formula definition.

Use the following sample formulas to convert the values of the total amount (transaction currency) field. The output type for these formulas is FLOAT:



Note: To convert values from other fields, replace foreigntotal with a different field ID.

```
1 | TO_NUMBER({foreigntotal#converted})
```

- Displays the currency amounts in the base currency of your account. The exchange rate used for the conversion is the rate that is in effect on the date when the query or workbook visualization is run.

```
1 | TO_NUMBER({foreigntotal#converted[1]})
```

- Displays the currency amounts in the currency ID specified in the square brackets [1]. The exchange rate used for the conversion is the rate that is in effect on the date when the query or workbook visualization is run.

```
1 | TO_NUMBER({foreigntotal#converted[1;2017-01-30]})
```

- Displays the currency amounts in the currency specified in the square brackets [1], using the exchange rate that is in effect on a specific date. You must enter the date using the YYYY-MM-DD format.

Currency Conversion from the User Interface

In a workbook visualization, you can convert currency values directly from the user interface for amount fields from the transaction and transaction line record types. These options are not available from within a dataset.

To apply conversion from within a workbook visualization, click the Field Menu icon next to a transaction or transaction line field in the Layout Panel or column header, then select **Currency**. The following options are presented:

- **Display Original:** displays each amount in the original currency used for the transaction. In table views and pivot tables, this is the default selection for transaction and transaction line fields.
- **Apply Conversion:** converts each amount using the currency and date selected on the Convert To window. By default, the Convert To window is set to the current date and the base currency assigned to your account or subsidiary. In charts, this is the default selection for transaction and transaction line fields.

- **Conversion Options...**: opens the Convert To window. In the Convert To window, you select the currency and date to use when the Apply Conversion option is selected. The rates used in the conversion are drawn from the table of currency exchange rates set up in your account. For more information, see the help topic [Currency Exchange Rates List Page](#).

Note: If you check the **Anchor Relative to Today** box, the current date will be used for the conversion each time the workbook is opened.

If you are creating a calculated measure that includes fields with values in multiple currencies, you can also apply conversion as part of the measure definition. For more information, see [Calculated Measures](#).

Formula Fields and Calculated Measures

Unlike formula fields which are built at the dataset level to evaluate results on a row by row basis, calculated measures are built at the workbook level using aggregated results. For example, if you create a formula field that calculates sales amounts, each row in the dataset will show a single value based on the formula definition. If you then use that formula field as a measure in a pivot table or chart, the results are simply an aggregate of the formula field results. By contrast, calculated measures use the aggregated results of base measures combined with arithmetic operations to display results such as year over year variances and profit percentages. These types of results are not achievable using formula fields alone.

The following examples describe why you obtain a different result in each case:

- [Creating a formula field and using it as a measure](#)
- [Creating a calculated measure](#)

Creating a formula field and using it as a measure

In your dataset, you can use the following definition to create a formula field that returns the profit percentage:

% Profit Formula = {estgrossprofit}/{amount}

The formula field column returns a percentage value for each entry. For example, the % Profit (formula) column displays a value for each of the three entries of the Sales Order 100537: 18%, 100%, and 100%.

DATE	EST. GROSS PROFIT	AMOUNT	% PROFIT (FORMULA)	TRANSACTION	TRANSACTION ...	QUANTITY
13.5.2020	648.40	\$3,602.00	18.0011 %	Sales Order #SORD100537	Sales Order	1
13.5.2020	4,500.00	\$4,500.00	100.0000 %	Sales Order #SORD100537	Sales Order	450
13.5.2020	294.00	\$294.00	100.0000 %	Sales Order #SORD100537	Sales Order	7
9.6.2020	179.00	\$895.00	20.0000 %	Sales Order #SORD100538	Sales Order	1
9.6.2020	2.34	\$11.69	20.0171 %	Sales Order #SORD100538	Sales Order	1
21.10.2020	4.13	\$18.75	22.0267 %	Sales Order #SORD100539	Sales Order	1
21.10.2020	4.13	\$18.75	22.0267 %	Sales Order #SORD100539	Sales Order	1
21.10.2020	4.13	\$18.75	22.0267 %	Sales Order #SORD100539	Sales Order	1
21.10.2020	22.92	\$81.40	28.1572 %	Sales Order #SORD100540	Sales Order	1
21.10.2020	330.00	\$330.00	100.0000 %	Sales Order #SORD100540	Sales Order	30
21.10.2020	51.52	\$110.00	46.8364 %	Sales Order #SORD100540	Sales Order	1
11.1.2021	296.51	\$500.00	59.3020 %	Sales Order #SORD100541	Sales Order	5
19.1.2021	5.00	\$10.20	49.0196 %	Sales Order #SORD100542	Sales Order	1
19.1.2021	205.35	\$1,026.75	20.0000 %	Sales Order #SORD100542	Sales Order	1

If you use this formula field as a measure in a pivot table, the returned value is 72.67%, which is not entirely accurate.

$$(18\% + 100\% + 100\%) / 3 = 72.67\%$$

Transaction	Est. Gross Profit (Sum)	Amount (Sum)	% Profit (formula) (Avg)
Sales Order #SORD100537	5,442.40	\$8,396.00	72.67 %
Sales Order #SORD100538	181.34	\$906.69	20.01 %
Sales Order #SORD100539	12.39	\$56.25	22.03 %
Sales Order #SORD100540	404.44	\$521.40	58.33 %
Sales Order #SORD100541	296.51	\$500.00	59.30 %
Sales Order #SORD100542	210.35	\$1,036.95	34.51 %

Creating a calculated measure

In your pivot table, you can use the following definition to create a calculated measure:

$$\% \text{ Profit} (\text{calculated measure}) = \text{Est. Gross Profit} (\text{Sum}) / \text{Amount} (\text{Sum})$$

The calculated measure column returns a percentage value based on the aggregated values calculated from the Est. Gross Profit (Sum) and the Amount (Sum).

$$(648.40 + 4,500 + 294) / (3,602 + 4,500 + 294) = 64.82$$

The % Profit (calculated measure) in the following pivot table returns 64.82% and the %Profit (formula) (Avg) column returns 72.67%.

Transaction	Est. Gross Profit (Sum)	Amount (Sum)	% Profit (formula) (Avg)	% Profit (calculated measure)
Sales Order #SORD100537	5,442.40	\$8,396.00	72.67 %	64.82 %
Sales Order #SORD100538	181.34	\$906.69	20.01 %	20.00 %
Sales Order #SORD100539	12.39	\$56.25	22.03 %	22.03 %
Sales Order #SORD100540	404.44	\$521.40	58.33 %	77.57 %
Sales Order #SORD100541	296.51	\$500.00	59.30 %	59.30 %
Sales Order #SORD100542	210.35	\$1,036.95	34.51 %	20.29 %

In this example, the formula field measure is showing only an average of the percentages displayed on the dataset (that is $(18\% + 100\% + 100\%) / 3 = 72.67\%$). With the calculated measure however, the amount is calculated by aggregating the values of the Est. Gross Profit and Amount fields first and then performing the division (that is $648.40 + 4500 + 294 = 5442.4$, $3602 + 4500 + 294 = 8396$, so $5442.4 / 8396 = 64.82\%$). Therefore, the calculated measure returns the expected value.

For more information about formula fields, see [Formula Fields](#).

For more information about calculated measures, see [Calculated Measures](#).

Workbook and Dataset Templates

SuiteAnalytics Workbook currently offers over 20 predefined workbook and dataset templates.

In a dataset template, all of the required fields, record types, and criteria filters are predefined so that you can create a workbook without having to first define a complex, custom dataset. Conversely, workbook templates include predefined workbook visualizations such as table views, pivot tables, and charts, so that you can analyze and visualize the results of the accompanying dataset templates. In a workbook template you can also add additional visualizations based the dataset template already connected to the workbook, or any datasets that you have access to in your account.

By default, only users with the Analytics Administrator permission can access workbook and dataset templates. If you do not have the Analytics Administrator permission, you can access these templates only if a user with the Analytics Administrator permission shares them with you. You must also have access to the underlying dataset template to open a workbook template. Additionally, some templates are feature dependent. If you do not see one of the listed templates in your account, contact your account administrator.

If you have the Analytics Administrator permission, you can access the templates from the Analytics Home page, from either the Workbooks or Datasets subtabs.

For more information about the Analytics Administrator permission, see [The Analytics Administrator Permission](#).

To see the dataset templates currently available in SuiteAnalytics Workbook, see [Dataset Templates](#).

To see the workbook templates currently available in SuiteAnalytics Workbook, see [Workbook Templates](#).

The sales (ordered), sales (invoiced), and manufacturing workbook and dataset templates are based on custom analytical record types designed specifically for SuiteAnalytics Workbook. These record types combine data from the transaction, transaction line, and transaction accounting line record types.

To see the analytical record types currently available in SuiteAnalytics Workbook, see [Analytical Record Types](#)

All other workbook and dataset templates that are in a beta state are also for test purposes only. Do not use these templates in your production accounts unless they are first reviewed by a NetSuite administrator.

Dataset Templates



Important: By default, only users with the Analytics Administrator permission can access the workbook and dataset templates in your account. If you do not have the Analytics Administrator permission, contact a user with the Analytics Administrator permission about sharing the templates with you.

SuiteAnalytics Workbook offers over 20 dataset templates. Each template includes preselected fields, joined record types, and criteria filters so that you can quickly create a custom workbook or your own version of the dataset. Additionally, each dataset template has an accompanying, predefined workbook template. For information about workbook templates, see [Workbook Templates](#).

The following dataset templates are currently available in SuiteAnalytics Workbook. If you have the Analytics Adminstrator permission, access the templates by selecting Templates on the Datasets subtab of the Analytics Home page:



Important: Some templates are feature dependent. If you do not see one of the listed templates in your account, contact your account administrator.

- Actual Time Dataset
- Customers by Item Dataset
- Employee Bonus History and Current Compensation Dataset
- Employee Tax Jurisdictions Dataset
- Estimated Revenue Dataset
- Intelligent Item Recommendations Dataset
- Items by Customer Dataset
- Manufacturing Dataset
- Merchandise Hierarchy Dataset Template
- Recurring Revenue Dataset
- My Transactions Dataset
- Performance Management — Goals Dataset
- Performance Management — Reviews Dataset
- Performance Management – Reviews Scheduler Dataset Template
- Purchases (Billed) Dataset
- Purchases (Ordered) Dataset
- Reviews Scheduler by Relative Date Dataset Template
- Sales (Ordered) Dataset
- Sales (Invoiced) Dataset
- SuiteCommerce Top Searches Dataset Template
- SuiteCommerce Top Searches with No Results Dataset
- Supply Chain Dataset
- System Notes v2 Dataset Template
- Time-Off Analysis Dataset
- Transaction Detail Dataset
- Warehouse Inventory: Counts & Adjustments Dataset
- Warehouse Inventory: Inbound Dataset
- Warehouse Inventory: Outbound Dataset
- Web Store Orders Dataset
- Workplace Tax Jurisdictions Dataset

Workbook Templates



Important: By default, only users with the Analytics Administrator permission can access the workbook and dataset templates in your account. If you do not have the Analytics Administrator permission, contact a user with the Analytics Administrator permission about sharing the templates with you.

SuiteAnalytics Workbook offers over 20 workbook templates, based on the available dataset templates in your account. Each template includes a different set of workbook visualizations for you to analyze your

data, including a Table View containing all of the fields from the underlying dataset. Use these templates to quickly analyze and visualize different parts of your company data. To access the underlying dataset template, click the dataset name in the Dataset Panel from any of the workbook visualizations and select **Open Dataset**. Some templates might also be based on a set of linked datasets. For more information, see [Dataset Linking in SuiteAnalytics Workbook](#).

The following workbook templates are currently available in SuiteAnalytics Workbook. If you have the Analytics Administrator permission, access the templates by selecting Templates on the Workbooks subtab of the Analytics Home page:



Important: Some templates are feature dependent. If you do not see one of the listed templates in your account, contact your account administrator.

- Actual Time Workbook
- Customers by Item Workbook
- Employee Tax Jurisdictions Workbook
- Intelligent Item Recommendations Workbook
- Items by Customer Workbook
- Manufacturing Workbook
- Merchandise Hierarchy Workbook Template
- Recurring Revenue Workbook
- My Transactions Workbook
- Performance Management — Goals and Reviews Workbook
- Purchases (Billed) Workbook
- Purchases (Ordered) Workbook
- Sales (Ordered) Workbook
- Sales (Invoiced) Workbook
- SuiteCommerce Top Searches Workbook
- SuiteCommerce Top Searches with No Results Workbook Template
- Supply Chain Workbook
- System Notes v2 Workbook Template
- Time-Off Analysis Workbook
- Transaction Detail Workbook
- Warehouse Inventory: Counts & Adjustments Workbook
- Warehouse Inventory: Inbound Workbook
- Warehouse Inventory: Outbound Workbook
- Web Store Orders Workbook
- Workplace Tax Jurisdictions Workbook

Analytical Record Types

Analytical record types are custom record types designed specifically for SuiteAnalytics Workbook and exist only in the [The Analytics Data Source and SuiteAnalytics Workbook](#). These record types are based on standard record types but contain a unique set of fields and predefined criteria. They combine data from the transaction, transaction line, and transaction accounting line record types. Currently, the sales

(ordered), sales (invoiced), and manufacturing workbook and dataset templates are based on analytical record types.

The following analytical record types are currently available in SuiteAnalytics Workbook:

- Manufacturing Component Analytical Record Type
- Manufacturing Transaction Analytical Record Type
- Sales (Invoiced) Analytical Record Type
- Sales (Ordered) Analytical Record Type

Workbook-based Portlets

With the Analytics portlet, SuiteAnalytics Workbook enables you to view your workbook data on your dashboards. The Analytics portlet lets you add charts, pivot tables, and table views from workbooks listed in the My Workbooks and Shared with Me sections of the Workbook Listing Page. You can add up to 10 Analytics portlets to your dashboards, each with a unique workbook visualization. SuiteAnalytics Workbook also supports the translation of Analytics portlet names through the Manage Translation.

For more information about each of the portlet types, see the following topics:

- [Chart-based Portlets](#)
- [Pivot-based Portlets](#)
- [Table View Portlets](#)

For more information on how to add and set up an Analytics portlet, see the help topics [Adding an Analytics Portlet](#) and [Setting Up the Analytics Portlet](#).

For more information about translatable Analytics portlet names, see the help topic [Adding Translatable Analytics Portlet Names](#).

SuiteAnalytics Workbook Tutorial

Use this tutorial to walk through the authoring process for SuiteAnalytics Workbook.

In this tutorial, you create a sample workbook with visualizations based on two different datasets. One dataset is based on the sales (ordered) record type, and the other is based on the item record type. By combining these datasets in a workbook, you should be able to quickly verify that you have enough inventory to fulfill all of the orders in your account.

The following table describes the steps to create the datasets, then the associated workbook visualizations. Each step builds on the previous step, so you must complete them in order:

Step	Description
Step One: Select a Root Record Type	Create your first dataset by choosing a root record type.
Step Two: Add Fields and Join Record Types	Choose the fields and record types that you want to include in your dataset.
Step Three: Filter Your Dataset	Apply criteria filters to refine your dataset query results.
Step Four: Create a Workbook Based on Your Dataset	Continue your analysis by creating a workbook based on your dataset.
Step Five: Pivot Your Dataset Query Results	Create a pivot table using your dataset query results.
Step Six: Connect a Second Dataset to Your Workbook	Create and join an additional dataset to your workbook.
Step Seven: Set up a Table View	Create a table view using the query results from your second dataset.



Note: This tutorial assumes that you have the appropriate permissions and features enabled in your account to view sales order and item data. If you are not sure about which permissions you need to view sales order records or fields, download the following worksheet: [NetSuitePermissionsUsage.xls](#). If you do not have a specific permission enabled, contact your NetSuite administrator.

Select a Root Record Type

The following steps show you how to select a root record type for your initial dataset. The available record types are based on the features enabled in your account and the permissions assigned to the role you use to log in to NetSuite. If you do not see the sales (ordered) record type, contact your administrator.

To select a root record type:

1. Click the **Analytics** tab in the NetSuite navigation menu.
2. On the Analytics Home page, click the Datasets subtab and select **New Dataset**.
3. Select a record type from the list.

The Dataset Builder opens with preselected fields on the Data Grid based on the record type you choose.

For the purposes of this tutorial, select the sales (ordered) record type.

Continue to Step Two: [Add Fields and Join Record Types](#).

Add Fields and Join Record Types

If you have not already done so, complete Step One: [Select a Root Record Type](#).

The following steps show you how to add fields and join record types to your dataset. Only fields that are included on the Data Grid can be used to build workbook visualizations such as tables, pivot tables, and charts.

By default, the Data Grid displays preselected fields based on the root record type selected for the dataset. Fields that you add to the grid appear highlighted at the top of the Fields list.

To add fields and join record types:

1. Add fields from the root record type to the Data Grid. You have three options:
 - Drag the fields from the Fields list to the Data Grid.
 - Double-click the fields in the Fields list.
 - Type the name of the fields in the search bar at the top of the Fields list, then drag or double-click them to add them to the Data Grid.

For the purposes of this tutorial, you do not need to add any additional fields from the sales (ordered) record type.

2. Join related record types to the dataset by adding their fields to the Data Grid.
 - a. By default, the Records list on the Dataset Builder shows all related record types that you have access to in your account.
Click the arrow next to any record type in the Records list to view additional related record types.
 - b. Click the desired record type name to update the Fields list.
 - c. Double-click or drag the desired fields to the Data Grid.



Note: Depending on the relationship between record types that you join in a dataset, data duplication might occur. For more information, see [Guidelines for Joining Record Types in SuiteAnalytics Workbook](#).

For the purposes of this tutorial, add the status field from the transaction record type.

3. Create and add formula fields to the Data Grid.
 - a. Click the **Formulas** link above the Records list.
 - b. Click **New Formula** in the Fields list.
 - c. In the Formula Field window, enter the formula field name.
 - d. Select an output type for the formula field values.



Note: Each output type only works with certain fields and formula functions. If you select an incompatible output type for the formula, the formula is invalidated. For more information, see [Formula Fields](#).

- e. In the Formula field, enter the field IDs and SQL formula functions to use in the formula expression. Alternatively, double-click the desired field IDs or formula functions from the Functions and Fields subtabs to add them to the expression.



Note: The Fields subtab only includes fields from the root record type in the dataset and any related record types that you have accessed on the Dataset Builder. If you want to include fields from a related record type in your formula expression, you must either access the record type on the Dataset Builder or manually enter the field ID into the expression.

- f. To validate the formula, click **Validate**.
- g. If there are no errors, click **Apply** to add the formula field to the dataset.

For the purposes of this tutorial, no formula fields are required.

4. Remove any fields you do not want to include in the dataset.
 - a. Click the Field Menu icon  in the column you want to remove from the Data Grid.
 - b. Select **Remove Column** from the dropdown list.

For the purposes of this tutorial, remove the memo and partner fields.



Warning: Any changes you make to a dataset are automatically propagated to workbook visualizations that are based on the dataset. Removing a field from the Data Grid can therefore cause failures in the associated workbook visualizations. Exercise caution when removing a field from the grid.

Continue to Step Three: [Filter Your Dataset](#).

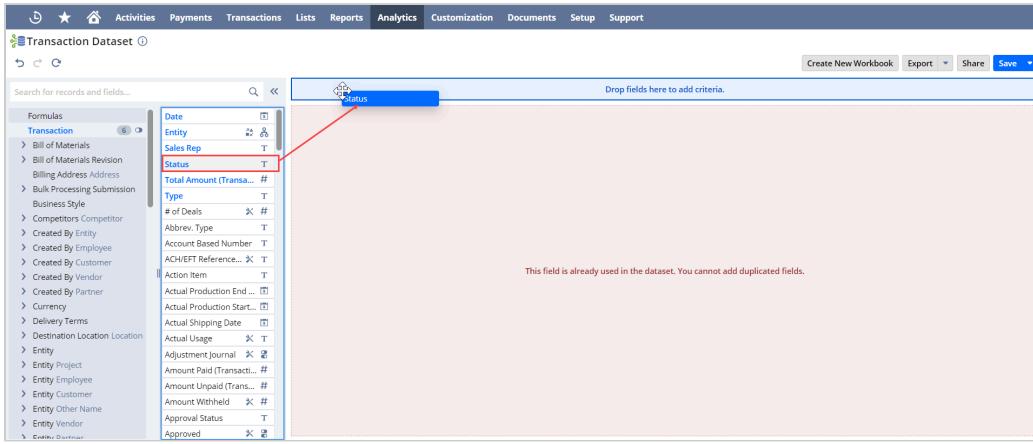
Filter Your Dataset

If you have not already done so, complete Step Two: [Add Fields and Join Record Types](#).

The following steps show you how to filter your dataset query results. Criteria filters that you apply to the dataset are propagated to any new or existing workbook visualizations that you create based on the dataset. For example, if you create a criteria filter to exclude any sales orders created in the last week, data from those sales orders are not presented in any previously associated workbook visualizations, or new visualizations that you create using this dataset.

To filter your dataset:

1. On the Dataset Builder, drag the desired record field or formula field from the Fields list to the Criteria Builder above the Data Grid.
(Optional) If the field you want to use is already in the Data Grid, drag the field header from the grid to the Criteria Builder.



The Filter window appears.

2. In the Filter window, select the filter conditions you want to apply to the field. Up to four options are available for filtering the data, depending on the type of field that is selected:
 - Values: Existing values or dates from the source data, or custom values
 - Ranges/Date Ranges: Range of values or dates available in the source data
 - Relative Conditions/Relative Dates: Conditions relative to the existing values in the source data
 - Conditions/Specific Dates: Specific value or date and an expression

For the purposes of this tutorial, no filter conditions are required.



Note: If you want to group filter conditions, click **New Group**. Grouped filters are considered one item in the list of conditions and are evaluated in the order that they appear in the Criteria Builder. For more information, see the help topic [Grouping Filters](#).

3. Repeat steps 1-2 for each filter you want to define.

By default, filters are added using an AND operator. To change the relationship between filters and filter groups, click **AND** and select **OR** from the popup window.

The screenshot shows the 'Transaction Dataset' in the SuiteAnalytics Workbook. The 'Analytics' tab is selected. The Data Grid displays a list of transactions. Above the grid, a filter summary is shown: AND > Status is Bill : Paid In Full. The grid columns include DATE, ENTITY, TYPE, TOTAL AMOUNT (TRANSACTION CURRENCY), SALES REP, and STATUS. The data shows various transactions with their details.

DATE	ENTITY	TYPE	TOTAL AMOUNT (TRANSACTION CURRENCY)	SALES REP	STATUS
11/6/2010	Equity Office	Bill	\$200.00		Bill : Paid In Full
11/1/2010	Cebuana Furnitures	Bill	-PHP5,600.00		Bill : Paid In Full
10/3/2010	License Manufacturing	Bill	\$80,000.00		Bill : Paid In Full
11/1/2010	Cebuana Furnitures	Bill	-PHP5,560.00		Bill : Paid In Full
9/15/2010	Charles Schwab	Bill	\$24,000.00		Bill : Paid In Full
1/1/2011	Equity Office	Bill	-\$8,000.00		Bill : Paid In Full
1/10/2011	Cebuana Furnitures	Bill	-PHP1,110.00		Bill : Paid In Full
1/14/2011	Cebuana Furnitures	Bill	-PHP2,582.70		Bill : Paid In Full
1/14/2011	Cebuana Furnitures	Bill	-PHP3,330.00		Bill : Paid In Full
1/16/2011	Cebuana Furnitures	Bill	-PHP11,100.00		Bill : Paid In Full
1/16/2011	Cebuana Furnitures	Bill	PHP12,210.00		Bill : Paid In Full

4. After you define each filter, the Data Grid is updated with only the data that matches your selected criteria and a criteria summary is displayed above the grid.



Important: If you want to convert field values that are in multiple currencies, you must do so from within a workbook visualization or using a custom formula field with the TO_NUMBER function. For more information, see [Currency in Datasets and Workbooks](#).

5. Click **Save**.
- The Save Dataset As window appears.
6. Enter a name and description for the dataset in the fields provided.
7. Click **Save**.

Continue to Step Four: [Create a Workbook Based on Your Dataset](#).

Create a Workbook Based on Your Dataset

If you have not already done so, complete Step Three: [Filter Your Dataset](#).

The following steps show you how to create a workbook based on your completed dataset.



Note: If you want to create additional workbooks based on the dataset you saved in Step Three, go to the Analytics Home page and click New Workbook on the Workbooks subtab. Then, select the dataset you saved to open a new workbook.

If you want to edit the dataset from within your workbook, click the dataset name in the Dataset Panel in any workbook visualization.

To create a workbook based on your dataset:

1. On the Dataset Builder, click **Create New Workbook**.
2. A workbook will open.
By default, the workbook is named after the root record type on the dataset.
3. Select a workbook visualization you want to create: either a Table View, Pivot table, or Chart.
For the purposes of this procedure, select **Pivot**.

Continue to Step Five: [Pivot Your Dataset Query Results](#).

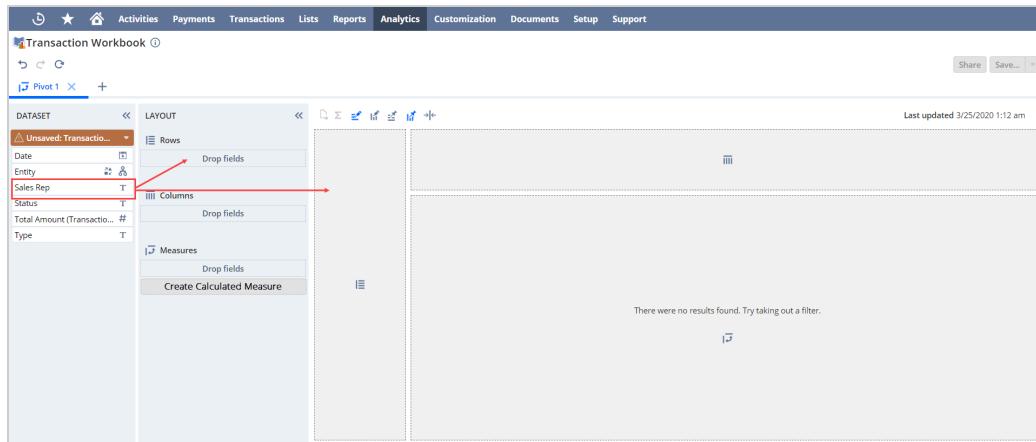
Pivot Your Dataset Query Results

If you have not already done so, complete Step Four: [Create a Workbook Based on Your Dataset](#).

The following steps show you how to pivot your dataset query results to create a pivot table.

To pivot your source data:

1. On the Pivot tab, drag the desired fields from the Fields list to the Rows, Columns, or Measures tabs in the Layout panel. Alternatively, drag the fields from the Fields list directly to the Pivot Table Viewer.
(Optional) In the Layout Panel, click **Create Calculated Measures** to define custom measures from the fields available in the Dataset Panel. Calculated measures are represented by the calculator icon. For more information, see [Calculated Measures](#).

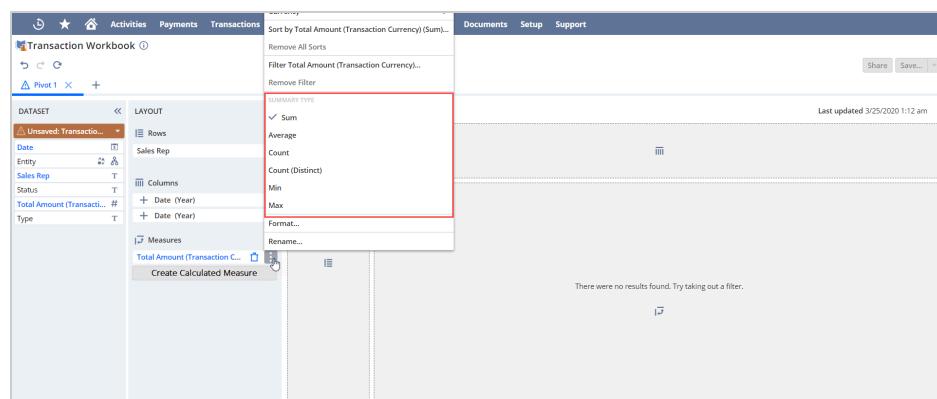


For the purposes of this tutorial, add the item and status fields to the pivot table Rows, add the amount (net) and quantity fields to the pivot table Measures.



Note: If you add hierarchical fields to the table, you are prompted to select a display type for the field values. Depending on where you add the field and the display type you select, you can also add additional subtotals to the pivot table for each level in the hierarchy. For more information, see [Hierarchical Fields](#).

2. Select the summary type and format options for any date or numerical fields you add to the pivot table.
 1. Click the Field Menu icon next to the field you want to format in the Layout panel.
 2. Select a summary type from the popup window.



3. (Optional) Select **Currency...** to view the currency consolidation or conversion options for any fields with values in multiple currencies.

For more information, see [Currency in Datasets and Workbooks](#).



Important: If you have fields with values in multiple currencies, you must convert the values to a single currency to perform arithmetic operations or other types of numeric manipulation.

4. (Optional) Click **Format...** to customize the numeric values for a field.

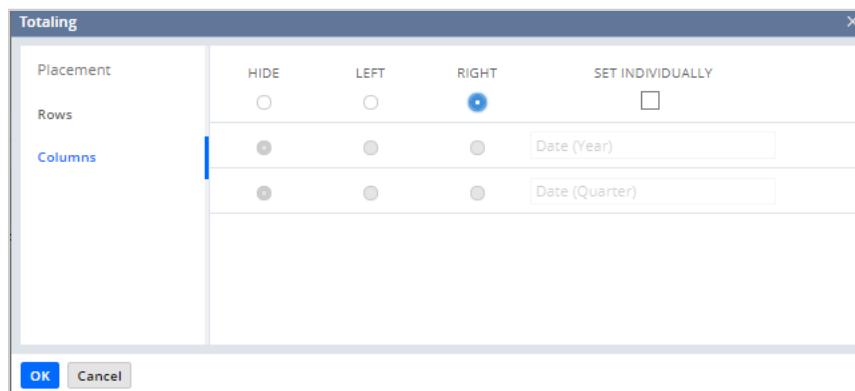
For more information about numeric formatting options, see [Customizing Numeric Values](#).

For the purposes of this tutorial, select the following summary types. No numeric formatting is required:

- Amount (Net) (Sum)
- Quantity (Sum)

3. Add totals and grand totals to the pivot table.

1. Click the Totaling icon Σ .
2. In the Totaling window, select where you want the totals or grand totals for each applicable field to appear. If there are multiple fields that can be totalled in the rows or columns, check the **Set Individually** box to select where the totals for each field will appear on the pivot table.



3. Click **OK**.

For the purposes of this tutorial, no totals are required.

4. Click the Refresh icon to generate the pivot table.

Status	Amount (Net) (Sum)	Quantity (Sum)
Creative 2400	\$5,398.20	2
Crusher Game Pad	\$12.95	1
EZ DVD-RW Drive	\$449.95	1
Impressivo 1000	\$9,990.00	10
Cable - Cat. 5, 10 ft	\$10.95	1
CD-R	\$49.30	5
Manuscript 1000 Las...	\$399.95	1
Inertio 1600	\$799.90	2
Creative 2000	\$1,349.10	1
	\$1,499.00	1
	\$2,299.00	1
	\$2,299.00	1

5. (Optional) Filter the data displayed in the pivot table.



Note: Filter conditions created on the Pivot tab only impact the data displayed in the pivot table. No changes are made to the underlying dataset or other workbook visualizations.

- a. Click the Field Menu icon next to the field you want to create a filter for. Depending on whether the field has been defined as a column, row, or measure, or if you click the Field menu icon from the Fields List or the Layout Panel, the following options are available:

- Top 10: display only the top 10 rows or columns based on the measures defined for the table.
 - Bottom 10: display only the bottom 10 rows or columns based on the measures defined for the table.
 - Filter [Field Name] by...: enables you to define a custom measure-based filter for the selected row or column.
 - Filter [Field Name]: enables you to define a custom value-based filter based on specific values within the table results.
 - Add as Filter...: enables you to define a custom value-based filter based on specific values within the table results.
- b. The results in the table are updated automatically.
- For more information, see [Workbook Visualization Filters](#).
- For the purposes of this tutorial, no pivot table filters are required.
6. (Optional) Apply conditional formatting to the pivot table measures to highlight your results.



Important: If the measure field you select has values in multiple currencies, you must convert or consolidate the values before you can apply conditional formatting. For more information, see [Currency in Datasets and Workbooks](#).

- a. Click the Field Menu icon next to the measure field you want to highlight in the Layout Panel and point to **Conditional Formatting**.
- b. Click **Manage Conditional Formatting**, then select the operators, values, and colors or icons for the rule. You can click the + add icon to create multiple rules and apply different colors or icons to the same measure or column. For more information, see [Conditional Formatting](#).



Note: To apply conditional formatting to percentage values, use decimal format when defining the values for the rule. For example, rather than **greater or equal to 20%**, the rule should be defined as **greater or equal to 0.2**.

- c. Check **Apply to subtotals and grand totals** if you want to apply your rules to the measure subtotals and grand totals.
- d. Click **Apply**.
- e. Repeat steps A-D for each measure you want to highlight.

For the purposes of this tutorial, no conditional formatting is required.

7. (Optional) Click the Export icon to save a CSV file of your pivot table.

Continue to Step Six: [Connect a Second Dataset to Your Workbook](#).

Connect a Second Dataset to Your Workbook

If you have not already done so, complete Step Five: [Pivot Your Dataset Query Results](#).

The following steps show you how to create and add an additional dataset from within your workbook.

To connect a second dataset to your workbook:

1. Click the add + icon to open a new workbook visualization tab.

2. Click **Connect Dataset**.
3. On the Select a dataset page, click **New Dataset**.
4. Select a record type for the dataset.
For the purposes of this tutorial, select the item record type.
5. In the Dataset Builder, add the quantity on hand and quantity back ordered fields from the item record type to the Data Grid.
6. Click **Save**.
7. In the Save Dataset As window, enter a name and description for the dataset, then click **Save**.
8. Click **Apply to workbook**.

Continue to Step Seven: [Set up a Table View](#).

Set up a Table View

If you have not already done so, complete Step Six: [Connect a Second Dataset to Your Workbook](#).

The following steps show you how to create a table view using the query results from your second dataset.

To create a table view:

1. Select **Table**.
2. Double click the names of the fields in the order that you want them to appear in the table.
For the purposes of this tutorial, add the fields in the following order: item name, quantity on hand, quantity ordered.
3. Click the **Save**.
4. In the Save Workbook As window, enter a name and description for the workbook in the fields provided and click **Save**.

All selections made in each workbook visualization are saved.



Important: You must save any datasets created as part of the workbook authoring process before you can save the workbook. For more information, see [Saving and Sharing Workbooks and Datasets](#)

5. (Optional) Share the workbook with other users in your account.
 - a. Click **Share**.
 - b. In the Share Workbook window, select the users or roles you want to share the workbook with.
 - c. Click **Share**.

For more information about sharing workbooks, see [Accessing and Sharing Workbooks and Datasets](#).
6. (Optional) If you want to make edits to any of the datasets connected to the workbook, click the dataset name in the Dataset Panel from within any workbook visualization.
For more information, see [Editing a Dataset](#).

Navigating SuiteAnalytics Workbook

The SuiteAnalytics Workbook user interface includes an entry point, a Dataset Builder, and three workbook visualization tabs. Click the following links to learn more about each area of the interface:

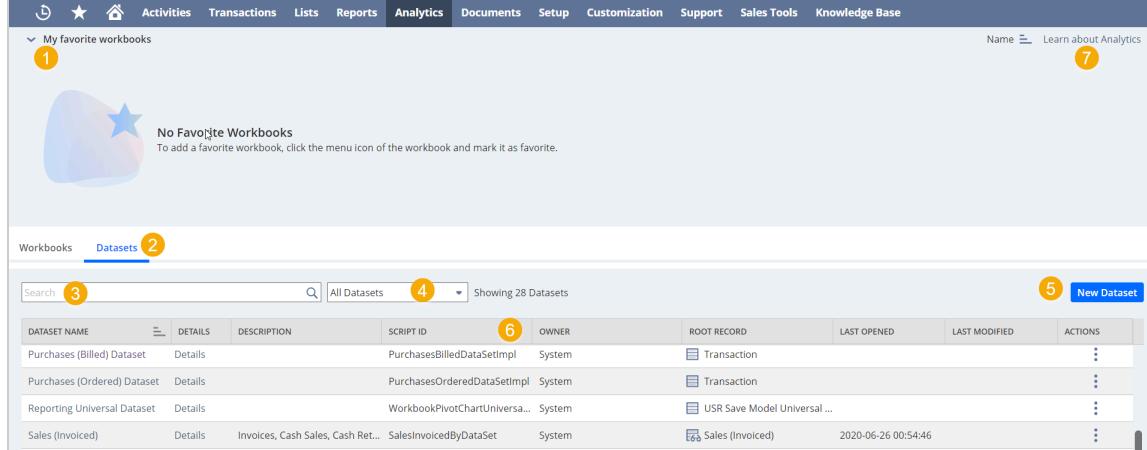
- [Analytics Home Page](#)
- [Dataset Builder](#)
- [Table Tab](#)
- [Pivot Tab](#)
- [Chart Tab](#)

Analytics Home Page

The Analytics Home page appears when you click the Analytics tab in the NetSuite navigation menu. The page includes Workbook and Dataset subtabs, enabling you to open and view any of the workbooks and datasets that you have access to. If you have the Analytics Administrator permission, you can also access predefined workbook and dataset templates and any of the workbooks or datasets created by other users in your account. For more information about the Analytics Administrator permission, see [The Analytics Administrator Permission](#).

The elements of the Analytics Home page are identified in the image below:

Note: The options available on this page vary depending on whether you are in the Workbooks or Datasets subtab.



- 1 **Favorite Workbooks** — Includes any workbooks you have marked as a favorite. Click the icon in the upper right corner to sort the workbooks by name, in ascending or descending order.
- 2 **Workbooks/Datasets subtabs** — Switch between viewing the workbooks and datasets in your account.
- 3 **Search** — Search for saved workbooks or datasets using a field name, user name, criteria filter, or any other workbook or dataset element. Searches are only executed across the selected list (4).

For example, to search workbooks that you are the owner of, make sure that My Workbooks is selected in the drop down list of the Workbooks subtab.

- 4 **Workbook/Dataset lists** — Specify the workbooks or datasets you want to view. You can only view templates and objects created by other users if you have the Analytics Administrator permission:

	<ul style="list-style-type: none"> ■ All Workbooks/ Datasets- Every workbook or dataset that you have access to in your account. ■ My Workbooks/ Datasets- Workbooks or datasets that you have created. ■ Shared with me- Workbooks and datasets that other users have created and shared with you. ■ Employee Workbooks/ Datasets- Workbooks and datasets created by other users in your account. ■ Templates- Predefined workbook and dataset templates. For more information, see Workbook and Dataset Templates.
5	<p>New Workbook/Dataset — On the Workbook subtab, click New Workbook to begin authoring a workbook using an existing dataset or a new dataset that you create. On the Dataset subtab, click New Dataset to select a record type and begin authoring a new dataset. For more information, see Custom Workbooks and Datasets.</p>
6	<p>Workbook/Dataset Viewer — All workbooks and datasets that you have access to are presented in the viewer. On the Workbook subtab, click the icons next to the New Workbook button to switch between a list view and tile view of the workbooks in your account.</p> <ul style="list-style-type: none"> ■ List view – Displays the following information: workbook name, details, description, owner, and date and time when the workbook was last opened and modified. ■ Tile view – Displays the following information: workbook name, description, and workbook type. <p>Within the Workbook/Dataset Viewer, you can perform the following actions:</p> <ul style="list-style-type: none"> ■ To view additional details about a dataset including a summary of the fields and criteria filters used, or the record types that are joined within it, click Details. From the Details Panel, you can also create a new workbook based on the dataset. ■ To view additional details about a workbook including a list of each visualization and the datasets they are based on, click Details. From the Details Panel, you can also access each dataset that the visualizations are based on. ■ To delete or share a workbook or dataset, click the menu icon and select Share or Delete. For datasets, you also have the option to Create Workbook. ■ To mark a workbook as favorite, do the following: <ul style="list-style-type: none"> □ From the tile view, click the menu icon and select Mark as Favorite. □ From the list view, click the Star icon next to the workbook name.
7	<p>Learn about Analytics — Click Learn about Analytics to open a panel that lists all SuiteAnalytics Workbook videos as well as a description. To view a video, click the thumbnail and it will open in a new window.</p>

Dataset Builder

The Dataset Builder appears after you select a root record type for a new dataset, or if you click the dataset name in the Dataset Panel of a workbook. On this tab you combine record types, fields, and criteria filters to create queries. The results of these queries are the basis for all workbook visualizations in your account.

On the Dataset Builder, your query results are displayed in a tabular format in the Data Grid on the right. Record types, fields, custom formula fields, and configuration options appear in the Records and Fields lists on the left. Only fields that are included in the Data Grid can be used to build workbook visualizations such as pivot tables and charts.

Above the Data Grid is the Criteria Builder where you create criteria filters to refine the results displayed in the grid. Criteria filters impact not only your dataset query results but also the data presented in workbook visualizations that are based on the dataset.

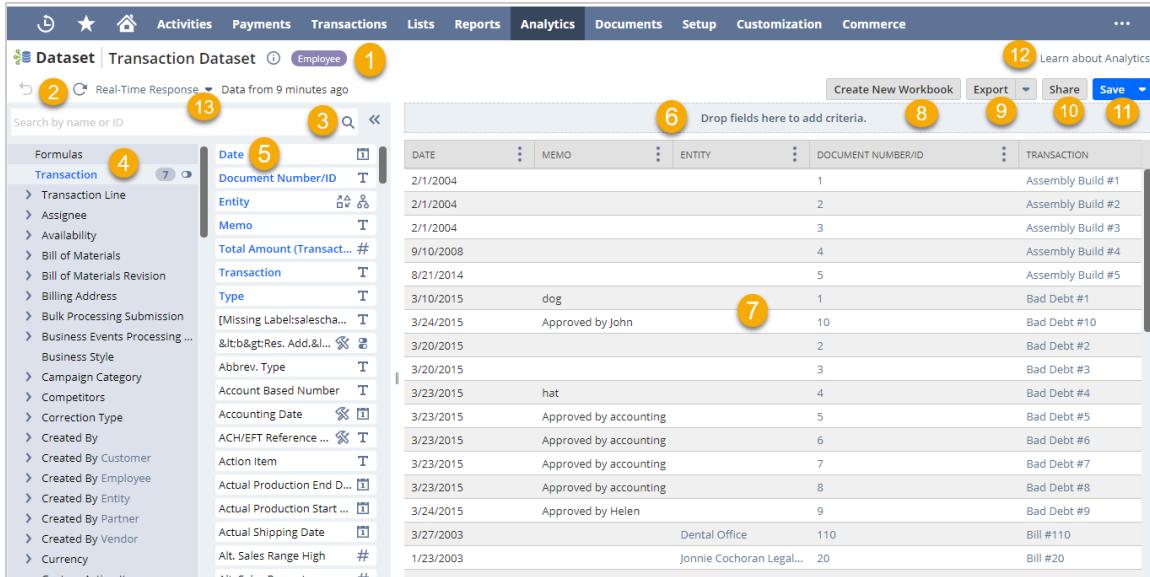
To build your dataset, drag fields from the Fields list to the Data Grid. To create a filter condition, drag fields from the Fields list or the Data Grid to the Criteria Builder. On the grid itself, you can sort and filter the values that are presented. After you create your query, you have multiple options to apply or preview the results in a workbook.

- For information about how to use the Dataset Builder to create a query, see [Defining a Dataset](#).

- For information about creating criteria filters to refine your query results, see [Dataset Criteria Filters](#).
- For information about creating a workbook based on your dataset query results, see [Creating a Workbook](#).

The elements of the Dataset Builder are identified in the image below:

Note: The appearance of the Dataset Builder varies slightly if you access it from within a workbook or on its own. See the following table for more information.



1	Dataset Information — Click to view or edit the dataset name, description, or internal ID.
2	Undo, Redo, and Refresh — Undo or redo your latest edits, and refresh the data in the Data Grid.
3	Search — Search for record types and fields using keywords, field labels, or internal field IDs. The search is performed against the root record type and record types that are three joins away from the root record type. To search against other record types, click Show more results .
4	Records list — Lists all the related record types that you have access to, based on the root record type selected for the dataset. Any record type with a number listed beside it indicates how many fields from that record type are in the Data Grid. You can perform the following actions from the Records list:
	<p>Note: By default, record types and fields are listed in alphabetical order except if you are in a transaction dataset. In a transaction dataset, the transaction line and transaction accounting line record types are listed at the top of the Records list.</p> <ul style="list-style-type: none"> To join a record type to the dataset, add a field from the record type to the Data Grid or use it in a criteria filter. For more information, see Guidelines for Joining Record Types in SuiteAnalytics Workbook. To create and view custom formula fields, click Formulas. For more information about formula fields, see Formula Fields. To view additional information about a record type, point to the record type name and click the Information icon. To view the fields of a record type and update the Fields list, click the record type name. To view additional related record types, click the Arrow icon next to any record type listed. To limit the number of record types displayed in the list and view only those with fields that have been added to the Data Grid, click the toggle icon.

5	<p>Fields list — The Fields list displays all the fields that you have access to for the record type you currently have selected in the Records list. If you click Formulas in the Records list, the Fields list displays any custom formula fields created for the dataset. At the top of the Fields list, any fields from the selected record type that have been added to the Data Grid are highlighted and listed in alphabetical order. Fields that are used in a criteria filter are highlighted but are not included at the top of the list. Next to each field, the following icons are used to provide additional information:</p> <ul style="list-style-type: none"> ■ : This icon denotes a custom field. For more information about custom fields, see the help topic Custom Fields. ■ : This icon denotes a polymorphic field. Polymorphic fields contain values that exist on multiple record types. For example, the customer field on the sales (ordered) record type, which contains the same data as the Name/ID field in the Customer record. ■ : This icon denotes a hierarchical field. Hierarchical fields have a defined parent-child relationship in NetSuite. These fields have multiple display options and can be used to create criteria filters based on the parent or child values in each record. For more information, see Hierarchical Fields. ■ : These icons denote the type of values contained within the field. Fields can include either text, numerical, date, or boolean data. <p>Using the Fields list, you can perform the following actions:</p> <ul style="list-style-type: none"> ■ To collapse the Records and Fields lists and expand the Data Grid, click the double arrows . ■ To view additional information about a field, point to the field and click the  icon. ■ To add a field to your dataset, double-click or drag the field from the Fields list to the Data Grid. ■ To create a criteria filter, drag a field from the Fields list to the Criteria Builder above the Data Grid. ■ (Optional) If the field you want to use is already in the Data Grid, drag the field header from the grid to the Criteria Builder. <div style="border: 1px solid #f0e68c; padding: 10px; margin-top: 10px;"> <p> Important: When you add fields from related record types to the Data Grid or use fields from related record types to create criteria filters, the record type is joined to the dataset. Depending on the relationship between the related record type and the root record type of the dataset, certain joins can have unexpected results such as data duplication. If joining record types is a new concept for you, see Guidelines for Joining Record Types in SuiteAnalytics Workbook for more information.</p> </div>
6	<p>Criteria Builder — The Criteria Builder displays the criteria filters you have created for the dataset in the order that they are applied to the Data Grid, as well as the operators that connect them. For example, the filters "Cleared is true AND Status any of Open, Paid In Full" means the values of the cleared field are evaluated first, followed by the values of the status field.</p> <p>Criteria filters refine your dataset query results and are automatically propagated to any workbook visualizations that are based on the dataset. For example, if you create a criteria filter to remove all invoices from your dataset, invoice data is also removed from any connected workbook visualizations.</p> <p>You can add filters to the Criteria Builder by dragging fields from the Fields list to the builder, then setting the specific filter conditions in the Filter window. Alternatively, you can drag a column header from the Data Grid to the Criteria Builder. For more information, see Dataset Criteria Filters. Within the Criteria Builder, you can perform the following actions:</p> <ul style="list-style-type: none"> ■ To edit a filter, click the name of the filter. ■ To delete a filter, point to the filter and click the Delete  icon. ■ To change the order of the filters, point to the filter you want to move and click the arrows. ■ To create a group of filters, click New Group. For more information about grouping filters, see Grouping Filters. ■ To change the operator between individual or grouped filters, click AND or OR and select a different operator from the popup window. ■ To see where a field is located in the Fields list, point to the associated filter, click the Menu icon  and select Show Location in Field List.

	<ul style="list-style-type: none"> ■ To reset your criteria filters, click Reset Criteria. ■ To hide the Criteria Builder and expand the Data Grid, click the arrow next to the criteria summary.
7	<p>Data Grid — Your dataset query results are displayed in the Data Grid. By default, the grid displays preselected fields based on the root record type selected for the dataset. You can add fields to the grid by double-clicking them on the Fields list, or dragging them to the grid. The number of rows displayed on each page of the grid is based on the Number of Rows in List Segments setting in your NetSuite user preferences, to a maximum of 500 rows per page. Only fields that are included on the grid can be used in workbook visualizations that are based on the dataset. On the grid itself, you can perform the following actions:</p> <ul style="list-style-type: none"> ■ To sort the values displayed in the grid, click the Field Menu icon  in the column header and select Sort Ascending, Sort Descending, or Edit Sort... to apply advanced sorting to the grid. For more information, see Advanced Sorting Options. ■ To filter the values that are displayed in the grid, click the Field Menu icon  next to the field you want to apply the filter to and select Filter... from the dropdown list. The filter window appears with up to four types of filters you can set depending on the values of the field. For more information about the Filter window, see Filter Types. <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> Note: Filter conditions and sorting options set up on the Data Grid only impact the values that are displayed in the grid itself and do not affect field values in any connected workbook visualizations. If you want to change your dataset query results and any associated visualizations, you must set up criteria filters using the Criteria Builder above the Data Grid. For more information, see Dataset Criteria Filters.</p> </div> <ul style="list-style-type: none"> ■ To move a column, drag the column header to the desired location. Alternatively, click the Field Menu icon  next to the column you want to move and select a move option from the list. ■ To remove a column from the grid, click the Field Menu icon  next to the column you want to delete and select Remove Column. ■ To rename a column, click the Field Menu icon  next to the field column and select Rename.... ■ To change the width of a column, drag the right boundary of the column header until it is as wide as you want. ■ To view the records associated to any links displayed in the grid, click the link. The corresponding NetSuite record opens in a new browser tab. ■ To view different pages, click the arrows at the bottom of the grid. ■ To view a summary of the values for any column in the grid, click the column header. The summary is displayed at the bottom of the grid. ■ To see where a field is located in the Records and Fields lists, click the Field Menu icon  and select Show Location in Field List.
8	<p>Create New Workbook/ Apply to Workbook — Click Create New Workbook to create a workbook based on the current dataset. If you accessed the dataset from within an existing workbook, click Apply to Workbook to apply your dataset changes to the workbook visualiations before saving them. This feature enables you to preview the impact that your dataset changes will have before saving the connected workbook.</p>
9	<p>Export — Export the data presented in the Data Grid to a CSV file. To use this feature, you must have the Export Lists permission.</p>
10	<p>Share — Share your dataset with individual users or roles. For more information about sharing, see Accessing and Sharing Workbooks and Datasets</p>
11	<p>Save — Save your dataset, including any joined record types, custom formula fields, or criteria filters. If you do not own the dataset you are working in, you must save a copy of the dataset using the Save As function.</p>
12	<p>Learn about Analytics — Click Learn about Analytics to open a panel that lists all SuiteAnalytics Workbook videos as well as a description. To view a video, click the thumbnail and it will open in a new window.</p>
13	<p>Real-Time / Cached Response — If you are experiencing long load times for your datasets or workbooks using your datasets, you can enable Cached Data in Datasets. For more information, see Optimized Data Refresh.</p>

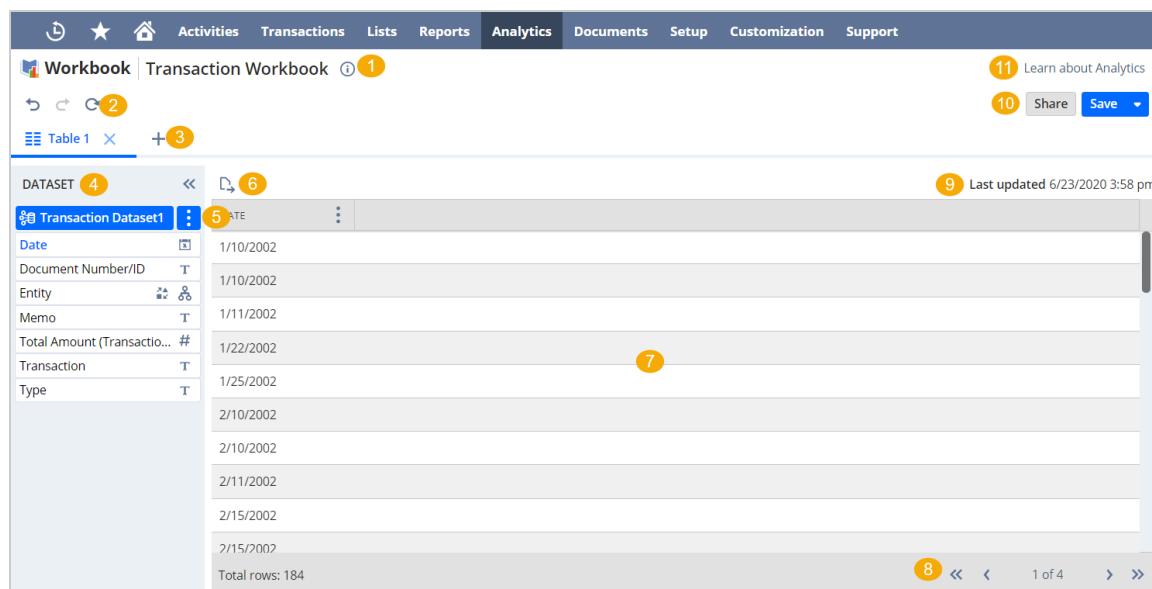
Table Tab

The Table tab is where you can explore your dataset query results without altering the source data of any associated workbook visualizations and without having to set a layout. Any fields that are included in the underlying dataset appear in the Dataset Panel on the left. To build your table, drag fields from the Dataset Panel to the Viewer.

You can create multiple table views in a single workbook using the dataset query results.

For information about how to explore your data in a table view, see [Workbook Table Views](#).

The elements of the Table View tab are identified in the image below:



1	Workbook Information — Click to view or edit the workbook name, description, internal ID, or to add a translatable portlet name to your Analytics portlet. This information is available from anywhere within the workbook. For more information about translatable Analytics portlet names, see the help topic Adding Translatable Analytics Portlet Names .
2	Undo, Redo, and Refresh — Undo or redo your latest edits, and refresh the data in the workbook visualization you are currently in. The undo and redo options apply to any action within the workbook and not just the current tab you are in. For more information about data refreshing, see Data Refresh in SuiteAnalytics Workbook .
3	Add — Click the add icon to add a new workbook visualization.
4	Dataset Panel — The leftmost area of any workbook visualization, the Dataset Panel contains all of the fields included in the selected dataset. To build a workbook visualization, drag fields from the Dataset Panel to the Layout Panel or the Table Viewer. You can also access and edit the underlying dataset, or switch to a different dataset using the Dataset Selector at the top of the panel. <ul style="list-style-type: none"> ■ To view or edit a dataset, click the dataset name in the Dataset Selector. ■ To view additional information about a field, point to the field and click the icon. ■ To collapse the Dataset Panel and expand the Viewer, click the double arrows .
5	Dataset Selector — This list enables you to access and edit the selected dataset, select a different dataset, or connect a dataset that is not already used in another workbook visualization.

	<ul style="list-style-type: none"> ■ To view or edit a dataset, click the dataset name in the Dataset Selector. Alternatively, click the menu icon and select Open Dataset. ■ To connect or create a new dataset, click the arrow and select Connect Dataset. ■ To save or discard changes to the selected dataset, click the menu icon and select Save Changes or Discard Changes. You must save or discard changes to any datasets connected to the workbook before you can save the workbook.
6	<p>Export — Click this icon to export a CSV file of the table.</p> <div style="border: 1px solid #fca; padding: 5px; margin-top: 10px;">  Important: To mitigate the possibility of a CSV injection, table results exported to CSV may have additional formatting applied. For more information, see Known Limitations in SuiteAnalytics Workbook. </div>
7	<p>Table Viewer — The Table Viewer displays your table automatically after you add fields to it. In the Table Viewer itself, you can customize the appearance of the columns, and sort and filter your data.</p> <ul style="list-style-type: none"> ■ To customize the appearance of your table, you can perform the following actions: <ul style="list-style-type: none"> □ To rename a field, click the Field Menu icon  and select Rename... <div style="background-color: #e0f2ff; border: 1px solid #0070C0; padding: 10px; margin-top: 10px;">  Note: The change applies only to the column name of the selected table view . </div> <ul style="list-style-type: none"> □ To change the order of the columns, drag them to the desired position or click the Field Menu icon  and select one of the move options. □ To remove unwanted columns, click the Field Menu icon  in the column header that you want to remove, and then select Remove Column. ■ To filter and sort your data, click the Field Menu icon  next to the field you want and then perform the following actions as needed: <ul style="list-style-type: none"> □ To apply the default sorting options to the table, select Ascending or Descending. □ To apply advanced sorting options, select Add Sort.... For more information, see Advanced Sorting Options. □ To filter your data, select Filter... and then define the filtering conditions. For more information, see Value-based Filters. ■ To apply conditional formatting to your data, click the Field Menu icon  in the column header of the field you want to highlight, then point to Conditional Formatting and select Manage Conditional Formatting. For more information, see Conditional Formatting
8	<p>Navigation Panel — Displays the number of rows shown in your table view and lets you navigate through different pages.</p> <p>To view different pages, click the arrows at the bottom of the viewer.</p>
9	<p>The displayed data refresh details are different depending on whether the Cached Data in Datasets feature is enabled or not:</p> <ul style="list-style-type: none"> ■ Cached Response / Real-time Response Data from [time] — Displayed when the Cached Data in Datasets feature is enabled. Displays if the workbook visualization is showing real-time data or cached data, and the time when the data was updated. ■ Last Updated — Displayed when the Cached Data in Datasets feature is disabled. Displays the time when the data in the current workbook visualization was last updated. To retrieve the latest results, click the Refresh icon. <p>For more information, see Data Refresh in SuiteAnalytics Workbook.</p>
10	<p>Workbook Menu — This menu is available from within all workbook visualizations.</p> <ul style="list-style-type: none"> ■ Click Share to share the workbook and any connected datasets with other users in your account. <p>For more information, see Accessing and Sharing Workbooks and Datasets</p>

	<ul style="list-style-type: none"> Click Save to save your workbook, including all selections made in each visualization. You can only save your workbook if the connected datasets have also been saved. If you do not own the connected datasets, save copies using different names.
11	Learn about Analytics — Click Learn about Analytics to open a panel that lists all SuiteAnalytics Workbook videos as well as a description. To view a video, click the thumbnail and it will open in a new window.

Pivot Tab

The Pivot tab is where you pivot your dataset query results. Any fields that are included in the underlying dataset appear in the Dataset Panel on the left. To build your pivot table, drag fields from the Dataset Panel to the Layout panel, then click the Refresh icon . The generated pivot table is displayed in the Pivot Table Viewer on the right.

You can create multiple pivot tables using the same dataset query results.

For information about how to use the Pivot tab to pivot your dataset query results, see [Creating a Workbook](#).

The elements of the Pivot tab are identified in the image below:

Date	Total Amount (Transaction Currency)
Cash Sale #1004	\$1,009.53
Sales Order #1006	\$1,009.53
Cash Sale #1010	\$1,183.23
Cash Sale #1011	\$1,344.30
Cash Sale #1012	\$1,493.85
Deposit #1	\$1,905.21
Cash Sale #1005	\$1,930.73
Sales Order #1007	\$1,930.73

1	Workbook Information — Click to view or edit the workbook name, description, internal ID, or to add a translatable portlet name to your Analytics portlet. This information is available from anywhere within the workbook. For more information about translatable Analytics portlet names, see the help topic Adding Translatable Analytics Portlet Names .
2	Undo, Redo, and Refresh — Undo or redo your latest edits, and refresh the data in the workbook visualization you are currently in. The undo and redo options apply to any action within the workbook and not just the current tab you are in. For more information about data refreshing, see Data Refresh in SuiteAnalytics Workbook .
3	Add — Click the add icon to add a new workbook visualization.
4	Dataset Panel — The leftmost area of any workbook visualization, the Dataset Panel contains all of the fields included in the selected dataset. In workbook visualizations based on linked datasets, the panel shows both dataset names.

	<p>To build a workbook visualization, drag fields from the Dataset Panel to the Layout Panel or the Table Viewer. You can also access and edit the underlying dataset, or switch to a different dataset using the Dataset Selector at the top of the panel. In workbook visualizations based on linked datasets, fields used to define the link are denoted by a chain link icon. If you delete fields used to define the link, it can cause errors within the workbook. For more information, see Dataset Linking in SuiteAnalytics Workbook.</p> <ul style="list-style-type: none"> ■ To view or edit a dataset, click the dataset name in the Dataset Selector. ■ To view additional information about a field, point to the field and click the  icon. ■ To create a value-based filter, click the Field Menu icon  and select Filter [Field Name] from the list. You can also create measure-based filters from the Layout panel or the Viewer. For more information, see Workbook Visualization Filters. ■ To collapse the Dataset Panel and expand the Viewer, click the double arrows .
5	<p>Dataset Selector — This list enables you to access and edit the datasets connected to the visualization, select a different dataset, or connect a dataset that is not already used in another workbook visualization. In visualizations connected to linked datasets, it also enables you to edit the dataset link.</p> <ul style="list-style-type: none"> ■ To view or edit a dataset, click the dataset name in the Dataset Selector. Alternatively, click the menu icon and select Open Dataset. ■ To connect or create a new dataset, click the arrow and select Connect Dataset. ■ To edit the dataset link, click the dataset menu icon and select Edit Dataset Link. ■ To save or discard changes to the selected dataset, click the menu icon and select Save Changes or Discard Changes. You must save or discard changes to any datasets connected to the workbook before you can save the workbook.
6	<p>Layout panel — The Layout panel displays the layout for your pivot table. For date and numeric fields, the summary type is also displayed next to the field. Every time you make changes to your pivot table layout, you must click the Refresh icon  for the changes to take effect.</p> <ul style="list-style-type: none"> ■ To add fields to the layout, drag them from the Dataset Panel to the desired section of the pivot table. You can define multiple fields for the same table component. For more information, see Grouping Pivot Table Fields. ■ To add a calculated measure, click Create Calculated Measures and define your calculations. Calculated measures are represented by the calculator  icon. For more information, see Calculated Measures. ■ If you are in a NetSuite account with multiple currencies or subsidiaries and your workbook contains fields with values in multiple currencies, click the Field Menu icon  next to an applicable field and select Currency... to apply currency conversion or consolidation. You must convert or consolidate values that are in multiple currencies to perform arithmetic operations and other types of numeric manipulation. For more information, see Currency in Datasets and Workbooks. ■ To create a value-based filter for the table, click the Field Menu icon  and select Filter [Field name].... You also create measure-based filters for the table if you select the Top 10 [Field name], Bottom 10 [Field name], or Filter [Field name] by options. For more information, see Workbook Visualization Filters. ■ To rename a field, click the Field Menu icon  and select Rename.... ■ To format the numeric values that are displayed in the table, click the Field Menu icon  and select Format.... For more information, see Customizing Numeric Values. ■ To change the summary type for dates and numeric values displayed in the table, click the Field Menu icon  next to an applicable field and select the preferred summary type. ■ To apply conditional formatting to your pivot table measures, click the Field Menu icon  next to the measure field you want to highlight, then point to Conditional Formatting and select Manage Conditional Formatting. For more information, see Conditional Formatting ■ To collapse the Layout panel and expand the Pivot Table Viewer, click the double arrows . ■ To add a grand total for a field, click the Field Menu icon  and select Show Grand Total.
7	<p>Pivot Table menu — This menu contains additional formatting options for the pivot table. The following options are available:</p>

	<ul style="list-style-type: none"> ■ Export icon  — Export a CSV file of the pivot table currently displayed in the viewer. ■ Important:  To mitigate the possibility of a CSV injection, table results exported to CSV may have additional formatting applied. For more information, see Known Limitations in SuiteAnalytics Workbook. ■ Totaling icon  — Add totals and grand totals to your pivot table for any applicable fields. Totals for columns can be added to the right or left of the table. Totals for rows can be added to the top or bottom. ■ Highlight Rows icon  — When enabled, rows are highlighted when you point to them. ■ Highlight Columns icon  — When enabled, columns are highlighted when you point to them. ■ Row Lock icon  — When enabled, the top row is locked so that it remains visible while scrolling through the table. ■ Column Lock icon  — When enabled, the first column is locked so that it remains visible while scrolling through the table. ■ Compact Mode icon  — If you define multiple fields as rows or add a multi-column hierarchy to your table, turn on compact mode to condense the fields into a single column.
8	<p>Pivot Table filter summary — Displays a summary of the filter conditions applied to the pivot table. Filter conditions created on the Pivot tab only affect the pivot table and do not change the underlying dataset.</p>
9	<p>Pivot Table Viewer — The Pivot Table Viewer displays your generated pivot table after updating the layout and clicking the Refresh  icon. As an alternative to dragging fields to the Layout panel to construct your table, you can add them directly to the Pivot Table Viewer.</p> <p>In the Pivot Table Viewer itself, you can perform the following actions:</p> <ul style="list-style-type: none"> ■ To resize a column or row, drag one of the column or row boundaries until it is the size you want. ■ To change the width of a column based on its values, double-click the column header. ■ To add a total for a column or row, click the Field Menu icon  and select Show Grand Total. ■ To create a measure-based filter for the table, click the Field Menu icon  and select Top 10 [Field name], Bottom 10 [Field name], or Filter [Field name]. ■ To create a value-based filter for the table, click the Field Menu icon  and select Filter [Field Name].... For more information, see Workbook Visualization Filters. ■ To rename a field, click the Field Menu icon  and select Rename... from the list. ■ To apply sorting to the table, click the Field Menu icon  and select Sort A to Z, Sort Z to A, or Sort [Field name...].
10	<p>The displayed data refresh details are different depending on whether the Cached Data in Datasets feature is enabled or not:</p> <ul style="list-style-type: none"> ■ Cached Response / Real-time Response Data from [time] — Displayed when the Cached Data in Datasets feature is enabled. Displays if the workbook visualization is showing real-time data or cached data, and the time when the data was updated. ■ Last Updated — Displayed when the Cached Data in Datasets feature is disabled. Displays the time when the data in the current workbook visualization was last updated. To retrieve the latest results, click the Refresh icon. <p>For more information, see Data Refresh in SuiteAnalytics Workbook.</p>
11	<p>Workbook Menu — This menu is available from within all workbook visualizations.</p> <ul style="list-style-type: none"> ■ Click Share to share the workbook and any connected datasets with other users in your account. For more information, see Accessing and Sharing Workbooks and Datasets ■ Click Save to save your workbook, including all selections made in each visualization. You can only save your workbook if the connected datasets have also been saved. If you do not own the connected datasets, save copies using different names.

- 12 **Learn about Analytics** — Click **Learn about Analytics** to open a panel that lists all SuiteAnalytics Workbook videos as well as a description. To view a video, click the thumbnail and it will open in a new window.

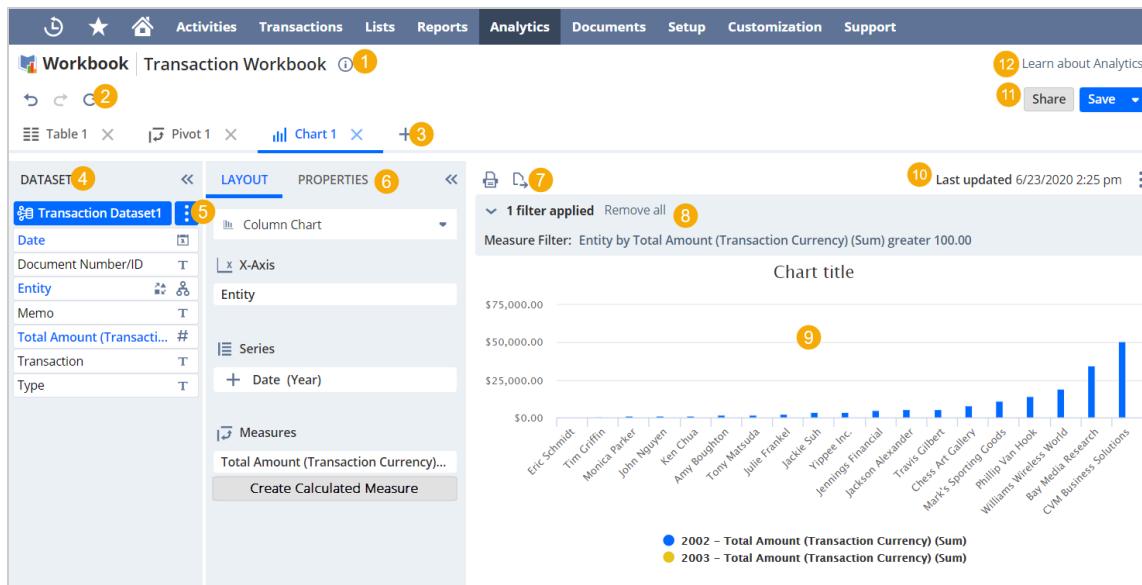
Chart Tab

The Chart tab is where you create visualizations of your dataset query results. Any fields that are included in the underlying dataset appear in the Dataset Panel on the left. To build your chart, drag fields from the Dataset Panel to the Layout panel, then click the Refresh icon . The generated chart is displayed in the viewer on the right.

You can create multiple charts using the same dataset query results.

For information about how to use the Chart tab to create a chart with your dataset query results, see [Workbook Charts](#).

The elements of the Chart tab are identified in the image below:



- | | |
|---|---|
| 1 | Workbook Information — Click to view or edit the workbook name, description, internal ID, or to add a translatable portlet name to your Analytics portlets. This information is available from anywhere within the workbook.

For more information about translatable Analytics portlet names, see the help topic Adding Translatable Analytics Portlet Names . |
| 2 | Undo, Redo, and Refresh — Undo or redo your latest edits, and refresh the data in the workbook visualization you are currently in. The undo and redo options apply to any action within the workbook and not just the current tab you are in. For more information about data refreshing, see Data Refresh in SuiteAnalytics Workbook . |
| 3 | Add — Click the add icon to add a new workbook visualization. |
| 4 | Dataset Panel — The leftmost area of any workbook visualization, the Dataset Panel contains all of the fields included in the selected dataset. In workbook visualizations based on linked datasets, the panel shows both dataset names.

To build a workbook visualization, drag fields from the Dataset Panel to the Layout Panel or the Table Viewer. You can also access and edit the underlying dataset, or switch to a different dataset using the Dataset Selector at the top of the panel. In workbook visualizations based on linked datasets, fields used to define the link are |

	<p>denoted by a chain link icon. If you delete fields used to define the link, it can cause errors within the workbook. For more information, see Dataset Linking in SuiteAnalytics Workbook.</p> <ul style="list-style-type: none"> ■ To view or edit a dataset, click the dataset name in the Dataset Selector. ■ To view additional information about a field, point to the field and click the  icon. ■ To create a value-based filter, click the Field Menu icon  and select Filter [Field Name] from the list. You can also create measure-based filters from the Layout panel or the Viewer. For more information, see Workbook Visualization Filters. ■ To collapse the Dataset Panel and expand the Viewer, click the double arrows .
5	<p>Dataset Selector — This list enables you to access and edit the datasets connected to the visualization, select a different dataset, or connect a dataset that is not already used in another workbook visualization.</p> <ul style="list-style-type: none"> ■ To view or edit a dataset, click the dataset name in the Dataset Selector. Alternatively, click the menu icon and select Open Dataset. ■ To connect or create a new dataset, click the arrow and select Connect Dataset. ■ To edit the dataset link, click the dataset menu icon and select Edit Dataset Link. ■ To save or discard changes to the selected dataset, click the menu icon and select Save Changes or Discard Changes. You must save or discard changes to any datasets connected to the workbook before you can save the workbook.
6	<p>Layout panel — The Layout panel displays the layout for your chart. For date and numeric fields, the summary type is also displayed next to the field. Every time you make changes to your chart layout, you must click the Refresh icon  for the changes to take effect.</p> <ul style="list-style-type: none"> ■ To select a chart type, click the chart type link and select an option from the menu. The following options are available: <ul style="list-style-type: none"> □ Column Chart □ Bar Chart □ Area Chart □ Line Chart □ Stacked Column Chart □ Stacked Bar Chart □ Stacked Area Chart For more information, see Chart Types. ■ To display the measures in the chart as a percentage of the totals for each record, click Show percentage. You can show percentages for stacked charts only. ■ To add fields to the layout, drag them from the Dataset Panel to the desired section of the chart. As with pivot tables, you can also group multiple fields in the same chart component to change the granularity of information displayed in the chart. For more information, see Grouping Pivot Table Fields. ■ To add a calculated measure, click Create Calculated Measures and define your calculations. Calculated measures are represented by the calculator  icon. For more information, see Calculated Measures. ■ If you are in a NetSuite account with multiple currencies or multiple subsidiaries and your workbook contains fields with values in multiple currencies, click the Field Menu icon  next to an applicable field and select Currency... to apply currency conversion or consolidation. You must convert values that are in multiple currencies to perform arithmetic operations and other types of numeric manipulation. For more information, see Currency in Datasets and Workbooks. ■ To create a value-based filter for the chart, click the Field Menu icon  and select Filter [Field name].... You can also create measure-based filters for the chart if you select the Top 10 [Field name], Bottom 10 [Field name], or Filter [Field name] by options. For more information, see Workbook Visualization Filters. ■ To rename a field, click the Field Menu icon  and select Rename.... ■ To format the numeric values that are displayed in the chart, click the Field Menu icon  and select Format.... For more information, see Customizing Numeric Values.

	<ul style="list-style-type: none"> ■ To change the summary type for dates and numeric values displayed in the table, click the Field Menu icon  next to an applicable field and select the preferred summary type. ■ To change the summary type of a field, click the Field Menu icon  and select the preferred summary type. ■ To create a title and subtitle for the chart or to add labels for the X and Y axes, click the Properties subtab and complete the appropriate fields. ■ To collapse the Layout panel and expand the Chart Viewer, click the double arrows .
7	<p>Chart menu — This menu contains additional options for the chart displayed in the viewer. The following options are available:</p> <ul style="list-style-type: none"> ■ Print  — Print the chart currently displayed in the viewer. ■ Export  — Export an SVG file of the chart currently displayed in the viewer.
8	<p>Chart filter summary — Displays a summary of the filter conditions applied to the chart. Filter conditions created on the Chart tab only affect the data displayed in the chart and do not change the underlying dataset.</p>
9	<p>Chart Viewer — The Chart Viewer displays your generated chart after updating the layout and clicking the Refresh  icon.</p> <p>In the Chart Viewer itself, you can perform the following actions:</p> <ul style="list-style-type: none"> ■ To view the exact amounts for a specific measure, point to the desired bar or data point in the chart. ■ To hide specific data points in the chart, click the corresponding color in the legend below the chart.
10	<p>The displayed data refresh details are different depending on whether the Cached Data in Datasets feature is enabled or not:</p> <ul style="list-style-type: none"> ■ Cached Response / Real-time Response Data from [time] — Displayed when the Cached Data in Datasets feature is enabled. Displays if the workbook visualization is showing real-time data or cached data, and the time when the data was updated. ■ Last Updated — Displayed when the Cached Data in Datasets feature is disabled. Displays the time when the data in the current workbook visualization was last updated. To retrieve the latest results, click the Refresh icon. <p>For more information, see Data Refresh in SuiteAnalytics Workbook.</p>
11	<p>Workbook Menu — This menu is available from within all workbook visualizations.</p> <ul style="list-style-type: none"> ■ Click Share to share the workbook and any connected datasets with other users in your account. For more information, see Accessing and Sharing Workbooks and Datasets ■ Click Save to save your workbook, including all selections made in each visualization. You can only save your workbook if the connected datasets have also been saved. If you do not own the connected datasets, save copies using different names.
12	<p>Learn about Analytics — Click Learn about Analytics to open a panel that lists all SuiteAnalytics Workbook videos as well as a description. To view a video, click the thumbnail and it will open in a new window.</p>

SuiteAnalytics Workbook Glossary

This glossary contains descriptions of the elements used for workbook and dataset authoring.

Analytical Record Type: Record types created specifically for Workbook that are made up of fields from various record types and predefined criteria. For more information, see [Analytical Record Types](#).

Analytics data source: The data source used by SuiteAnalytics Workbook. Record types and fields in this data source might have different locations or labels than the data sources used for saved searches and reports. For more information, see [Analytics Data Source Overview](#).

Calculated measure: A measure that you can create in your pivot table or chart using arithmetic operations with the measures available in the connected dataset. For more information, see [Calculated Measures](#).

Charts: A workbook visualization that enables you to visualize your dataset query results using predefined chart and graph types, such as line graphs and bar charts. For more information, see [Workbook Charts](#).

Common keys: Common keys are fields that contain the same data in two different datasets. You setup common keys when you want to link two datasets in a workbook. When you select the **Create Dataset Link** option within a workbook, NetSuite automatically assigns fields with matching names and data types as common keys. For more information, see <Common Keys>.

Conditional formatting: Enables you to highlight results for fields containing NUMBER and STRING values in your workbook table views and pivot tables. For more information, see [Conditional Formatting](#)

Criteria Builder: The area above the Data Grid in the Dataset Builder. You use this area to create criteria filters, which change the dataset query results and the values presented in any workbooks that are based on the dataset.

Criteria filters: Filter conditions that are applied to a dataset using the Criteria Builder. Criteria filters are automatically propagated to any workbooks and workbook visualizations that are based on the dataset, when the dataset is saved. For more information, see [Dataset Criteria Filters](#).

Data Grid: The Data Grid is where you add fields to your dataset. You can only build workbook visualizations such as table views, pivot tables, and charts using fields that have been added to the underlying dataset of a workbook, and therefore the Data Grid.

Dataset: Datasets are the basis for all workbooks and workbook visualizations in your account. In a dataset, you combine record type fields and criteria filters to create a query. The results of this query act as the source data for the workbook visualizations you create. A single dataset can be used in multiple workbook visualizations, so to prevent data discrepancies, they can only be edited or deleted by dataset owners or users with the Analytics Administrator permission. Additionally, you can use multiple different datasets in the same workbook, enabling you to analyze different metrics. For more information, see [Defining a Dataset](#).

Dataset Builder: The area of the user interface where you build your datasets by creating a query. The Dataset Builder contains the Records and Fields lists, the Data Grid, and the Criteria Builder. For more information, see [Dataset Builder](#).

Dataset Panel: The leftmost area of any workbook visualization, containing all of the fields of the selected dataset. You can access and edit the underlying dataset, switch to a different dataset, or create a new dataset from this panel if the selected one does not contain the fields you need. Use the fields in this panel to build your table views, pivot tables, and charts.

Filters: Filter conditions applied only to a specific workbook visualization, such as a table view, pivot table, or chart. These filters do not affect the underlying dataset or any other visualizations within the workbook. For example, if you apply a filter to a pivot table, it does not change the data presented in the workbook's charts. To filter the values presented throughout the workbook or in the underlying dataset, you need to create a criteria filter.

Join: The action of combining fields from multiple record types to define a dataset. To complete a join, select a record type from the Records list in the Dataset Builder, then add one of the record type fields to the Data Grid or use it to build a criteria filter. There are certain behaviors you should be aware of when you join record types in a dataset, such as unwanted data duplication. For more information, see [Guidelines for Joining Record Types in SuiteAnalytics Workbook](#).

Join path: Some record types are only accessible through a specific join path. For example, to join the transaction accounting line record type in a dataset, you must first access the transaction line record type through the transaction record type. In this scenario, the join path is transaction> transaction line> transaction accounting line.

Linked datasets: After you connect a dataset to a workbook, you can use it as the basis for any visualizations you create. You can also link two connected datasets, so that you can compare data from each within the same visualization. For more information, see [Dataset Linking in SuiteAnalytics Workbook](#).

Pivot Table: A workbook visualization that enables you to pivot your dataset query results by defining measures and dimensions, so that you can analyze different subsets of data. For more information, see [Workbook Pivot Tables](#).

Records and Fields lists: The leftmost area of the Dataset Builder. These lists display any of the record types and fields that you can join in a dataset, based on the selected root record type on the dataset, your NetSuite permissions, and the features enabled in your account.

Records Catalog: A cross-channel, contextual tool that provides all records available for use in NetSuite. Within the Records Catalog, you can find all available records, their fields, sublists, IDs, APIs, joins, and search filter fields for those records.

Table View: A workbook visualization that enables you to view your dataset query results in a simple table. For more information, see [Workbook Table Views](#).

Workbook: Workbooks are where you analyze the results of your dataset queries using different visualizations, such as table views, pivot tables, and charts. All workbook visualizations are based on a dataset, and you can use different datasets for each visualization in a single workbook. This enables you to analyze different metrics within a single workbook. You can access the underlying dataset from the Dataset Panel in any workbook visualization. When you author a workbook visualization you can either define a new custom dataset, or use any of the existing datasets that you have access to in your account. For more information, see [Creating a Workbook](#).

SuiteAnalytics Workbook FAQs

See the questions and answers below for information about SuiteAnalytics Workbook.

How can I use condition-based filters in currency fields?

Condition-based filters are available for percentage, integer and float fields that do not return currency values. If you want to apply this option to currency fields, customize your field using a TO_NUMBER formula function and changing the output type of the field.

For more information about how to customize a currency field, see [Sample Formulas](#).

How can I use relative dates in Workbook formulas?

The SuiteAnalytics Workbook Formula Builder enables you to create custom formula fields using relative dates.

For more information about Workbook formulas and examples of formula definitions with relative dates, see [Calculating Amounts for Relative Date Ranges](#) and [Sample Formulas](#).

How do I remove a dataset from a workbook after it's been connected?

You currently cannot manually disconnect a dataset after you connect it to a workbook. After you save and close a workbook however, any datasets not used in a visualization are automatically removed.