

# Data Source: Populate from a data table

Last Modified on 07/08/2025 1:56 pm EDT

After you have **created a data table**, you can design indicators that pull results directly from the data table. This article covers how to:

- **Define a data table indicator**
- Configure one of four calculation types:
  - **Count all**
  - **Count unique sets**, including:
    - Count first
    - Count unique per date
    - Count unique per reporting period
    - Deduplicate across dates
  - **Sum**
  - **Average**
- **Apply filters** to exclude certain rows of the data table from indicator calculations

## Define a data table indicator

Begin on the indicator's definition page. For data source, select **populate from a data table** and choose the table that contains information for this indicator.

The screenshot shows the DevResults interface for an indicator named '# trainees' (EG.001). The 'Definition' tab is active, and the 'Data Source' section is expanded. Under 'Data Source', the option 'Populate from a data table' is selected with a radio button. An orange arrow points to this option. Below, the 'Data Table' dropdown menu is open, showing a list of tables: 'Select a data table...', 'Organizations', 'Training Table' (highlighted in blue), 'Farmers', and 'Crop Sales'.

Next, select the calculation. You can either **count all** rows, **count unique sets** of columns, or get a **sum** or **average** of a column of numeric data. Additional configuration options will depend on which calculation type you use. This example shows the **count all** configuration.

After selecting the calculation, identify which geography column and which date column should be associated with the indicator results. You must choose these even if your table only has one geography column and one date column. (For example, you would probably choose to report your indicator results by "date of training" rather than "date of birth" for a beneficiary.)

**Data Source**

**Data Source** ☐ Enter indicator results directly  
☒ Populate from a data table  
☐ Calculate from a formula

**Data Table** Training Table   
[View Table Data](#)

**Calculation** Count All

**Geography Column** Select a geography column...   
Headquarters of partner organization  
Hometown of Trainee  
Location of Training

**Date Column** Select a date column...   
Data will be associated with dates from this column.  
 No date column selected

Finally, you can choose to exclude rows from your calculation by applying one or more filters. See the [filter section](#) at the bottom for more information.

The rest of the indicator's definition works like any other indicator with three caveats:

- Any disaggregations assigned to the indicator must be included in the data table.
- The geographic disaggregation must either be the same as the geography column selected in the data source section *or* a geographic disaggregation that's less specific. For example, if the data source geography column is "district", the indicator could be reported per country, but not per location.
- If the indicator is reported per activity, there must be an activity column in the data table.

Note: any rows missing data in a column relevant to the indicator definition (such as the date, geography, or a disaggregation) will be ignored when indicator results are calculated.

[back to top](#)

## Data table indicator calculation: Count all

The **Count All** calculation counts all rows in your data table. (The same caveats apply for any indicator: rows of data are excluded from the calculation if they don't have complete information for all columns relevant to the indicator definition, or if filters are applied).

To create a count-all indicator, set the data table **Calculation** to **Count All**.

Data Source

☐ Enter indicator results directly
 ☒ Populate from a data table
 ☐ Calculate from a formula

Data Table
 

Training Table

View Table Data

Calculation
 

Count All

Geography Column
 

Location

Data will be associated with places from this column.

Date Column
 

Date

Data will be associated with dates from this column.

Filters
 

Add a new filter...

The result for this indicator mapping will be a count of all the rows in the data table.

Have a look at this Training Table:

Example Organization  
DevResults

Quick search...

Help

Data

Program Info

Tools

Administration

Bookmarks

Home

Results

Data Tables

Training Table

Training Table

Data

Design

Search

6649 rows selected

Add new row

View Notes

Delete

<input checked="" type="checkbox"/>	Key Value	Date	Last Day of RP	Activity	Location	Trainee ID	Sex	Age	Training Topic	Training Type	
<input checked="" type="checkbox"/>		12.609	29 Jun 2020	30 Jun 2020	Improving Health Outcomes through Training	Tiguélipo (Lola, Nzérékoré)	F-684999	Male	20-24 (Young Adult)	Agriculture	Adult learning
<input checked="" type="checkbox"/>		12.608	29 Jun 2020	30 Jun 2020	Capacity Development in Rural Areas	Songbo (Kissidougou, Faranah)	V-526771	Female	25-64 (Adult)	Business	Adult learning
<input checked="" type="checkbox"/>		12.607	29 Jun 2020	30 Jun 2020	Capacity Development in Rural Areas	Mangadian (Siguiri, Kankan)	U-252571	Female	25-64 (Adult)	Agriculture	Gender sensitivity
<input checked="" type="checkbox"/>		12.606	29 Jun 2020	30 Jun 2020	Capacity Development in Rural Areas	Fadia (Dinguiraye, Faranah)	J-423203	Female	25-64 (Adult)	Civil society	Adult learning
<input checked="" type="checkbox"/>		12.605	29 Jun 2020	30 Jun 2020	Capacity Development in Rural Areas	École Primaire de Lafanyi (Coyah, Kindia)	D-248301	Female	25-64 (Adult)	Education	Gender sensitivity
<input checked="" type="checkbox"/>		12.604	29 Jun 2020	30 Jun 2020	Capacity Development in Rural Areas	École Primaire de Kotouba Dgninta (Siguiri, Kankan)	C-822907	Female	20-24 (Young Adult)	Education	Gender sensitivity
<input checked="" type="checkbox"/>		12.603	29 Jun 2020	30 Jun 2020	Capacity Development in Rural Areas	Dougna (Kankan, Kankan)	U-550317	Female	15-19 (Youth)	Technology	Advocacy
<input checked="" type="checkbox"/>		12.602	29 Jun 2020	30 Jun 2020	Capacity Development in Rural Areas	Centre de Santé de Konianfara (Siguiri, Kankan)	D-229769	Female	15-19 (Youth)	Health	Gender sensitivity
<input checked="" type="checkbox"/>		12.601	29 Jun 2020	30 Jun 2020	Access to School Lunches	Telininkoro (Siguiri, Kankan)	R-315570	Female	25-64 (Adult)	Civil society	Other
<input checked="" type="checkbox"/>		12.600	29 Jun 2020	30 Jun 2020	Access to School Lunches	Komatiguia (Siguiri, Kankan)	P-587390	Male	20-24 (Young Adult)	Technology	Other
<input checked="" type="checkbox"/>		12.599	29 Jun 2020	30 Jun 2020	Access to School Lunches	Fadia (Dinguiraye, Faranah)	F-104283	Female	25-64 (Adult)	Civil society	Advocacy
<input checked="" type="checkbox"/>		12.598	29 Jun 2020	30 Jun 2020	Access to School Lunches	Din (Lola, Nzérékoré)	R-498241	Female	20-24 (Young Adult)	Technology	Financial
<input checked="" type="checkbox"/>		12.597	26 Jun 2020	30 Jun 2020	Capable Local Governance	Ossokourouma (Kissidougou, Faranah)	C-745573	Female	15-19 (Youth)	Technology	Financial
<input checked="" type="checkbox"/>		12.596	26 Jun 2020	30 Jun 2020	Capable Local Governance	École Primaire de Voumou (Nzérékoré, Nzérékoré)	J-435293	Female	25-64 (Adult)	Education	Advocacy
<input checked="" type="checkbox"/>		12.595	26 Jun 2020	30 Jun 2020	Capable Local Governance	Centre de Santé de Youkhokhori (Dubreka, Kindia)	J-584199	Female	25-64 (Adult)	Health	Other

DevResults

Created by John Sapp on Feb 2016

Delete this data table

Done

There are 6649 rows, so **Count All** would give the result of 6649 (if all the relevant columns are filled out).

[back to top](#)

## Data table indicator calculation: Count unique sets

The **Count Unique Sets** calculation lets you avoid certain kinds of double counting when generating results from a data table. For example, if you train someone multiple times but only want to count them once, **Count Unique Sets** can not only deduplicate these results, but adjust how and when they are counted.

### How Does 'Count Unique Sets' Work?

**Count Unique Sets** looks at the combination of fields selected as **Count Unique Columns** and then counts each identical set only once within the time period defined by the selected **Count Type**. For example, if you have a training

table and select *First Name* and *Last Name* as your **Count Unique Columns**, you can use the **Count Type** dropdown to count training participants four different ways:

- **Count first:** Each set will only be counted for the earliest date in the selected **Date Column** and the reporting period that contains it. Only one training for each person will be counted, and all subsequent trainings for the same person will be ignored.
- **Count unique per date:** The selected **Date Column** will be automatically added to the **Count Unique Column** list, then sets that share a date will each be counted once. Multiple trainings for the same person on the same date will be deduplicated.
- **Count unique per reporting period:** The selected **Date Column** will be converted to the appropriate Reporting Period and automatically added to the **Count Unique Columns** list, then sets that share a reporting period will each be counted once. Multiple trainings for the same person in the same reporting period will be deduplicated, but multiple trainings for the same person across *different* reporting periods will each be counted.
- **Deduplicate across dates:** Each set will be deduplicated and associated with the latest date from the selected **Date Column**. Multiple trainings for the same person will be deduplicated across all time, but the result will exist entirely within a single recent reporting period (compare to count first which distributes the data across reporting periods based on first participation.)

For more detailed examples of various types of counts, see [Count Unique Sets Examples](#). Basic configuration of a Count Unique Set indicator is demonstrated below.

**Note:** If you use *count unique per reporting period*, you no longer need to include a "Last day of reporting period" or similar column in your data table. This "Last day..." column is a clever way to take different actual event dates and convert them to a single date which represents the reporting period itself; there is now an explicit count type that does this conversion for you.

## Configure a Count Unique Sets Indicator

To create a count-unique-sets indicator, set the data table **Calculation** to **Count Unique Sets**.

The screenshot shows the 'Data Source' configuration panel. It includes several fields: 'Data Source' with radio buttons for 'Enter indicator results directly', 'Populate from a data table' (selected), and 'Calculate from a formula'; 'Data Table' set to 'Training Table' with a 'View Table Data' link; 'Calculation' set to 'Count All' with a dropdown menu open showing 'Count All', 'Count Unique Sets' (highlighted with a blue bar and an orange arrow), 'Sum', and 'Average'; 'Geography Column' set to 'Date'; 'Date Column' set to 'Date'; and 'Filters' set to 'Add a new filter...'.

Once you select **Count Unique Sets** in the calculation, this might add some **Count Unique Column(s)** for you:

- By default, if the indicator is set for **Results are reported separately for each activity** the **Activity** column will be added and not removable from the **Count Unique Columns**. To remove Activity here, you will need to uncheck the appropriate box in the Disaggregations section.
- If any disaggregations are assigned to the indicator, these will also be automatically added to the **Count Unique Columns**. To remove the disaggregation from the **Count Unique Columns**, remove it from the **Disaggregations** section of the indicator.

- Geography columns (if present) will *not* be automatically added, but may be necessary depending on your indicator definition and geographic disaggregation. Generally speaking, you will likely need to add a geography column to the **Count Unique Columns** list *unless* you are collecting top-level results only (e.g. national or global level results only, depending on your site's largest geographic extent).

**Add columns** that uniquely identify your rows, but ignore any date columns for now. For example, the definition below will take the Activity + Trainee First Name + Trainee Last Name to determine the unique sets, deduplicating any records with the same name and same activity. This will *not* deduplicate records that have the same name but *different* activities. If this definition of uniqueness isn't specific enough (for instance, if multiple attendees had the same name), additional fields could be added to distinguish them, such as Location.

**Data Source**

**Data Source** ☐ Enter indicator results directly  
☒ Populate from a data table  
☐ Calculate from a formula

**Data Table** Training Table ▼  
[View Table Data](#)

**Calculation** Count Unique Sets ▼ using

**COUNT UNIQUE COLUMN(S)**

Column	Type
Activity	Activity
Trainee First Name	Other Text
Trainee Last Name	Other Text

**Add column...** ▼

- Add column...
- Location (Geography: Location)
- Certification Exam Taken? (Yes/No)
- Certification Exam Passed? (Yes/No)
- Train the Trainers Training Completed? (Yes/No)
- Certified Trainer Passed? (Yes/No)
- Number Days Attended (Number: Whole)

**Date Column** ▼

**Count Type** ▼

**Filters** Add a new filter... ▼

Select a **Date Column** from the available date formatted fields (if multiple). If your table only has one, it may already be selected for you. The **Date Column** you choose is used to associate your data with reporting periods for that indicator.

Then select a **Count Type**. This choice determines *how* the date column will be used to associate date with reporting periods. [See above](#) for definitions of each count type.

**Data Source**

**Data Source**
☐ Enter indicator results directly
 ☒ Populate from a data table
 ☐ Calculate from a formula

**Data Table**

Training Table

View Table Data

**Calculation**

Count Unique Sets

using

**COUNT UNIQUE COLUMN(S)**

	Column	Type
	Activity	Activity
	Trainee First Name	Other Text
	Trainee Last Name	Other Text
	Add column...	

**Date Column**

Date

**Count Type**

Count first

Count first
 

Count unique per date

Count unique per reporting period

**Filters**

Deduplicate across dates

**Note:** If your indicator is set to report at any **Geographic Disaggregation** below your top level, you will need to include a Geography column from your data table in your **Count Unique Columns** for it to calculate properly. If you don't, you'll see a warning like this:

**Disaggregation**

**Disaggregations**

	Disaggregation	Categorie	Data column	Disable
+ Add a disaggregation...				
<input checked="" type="radio"/> Results are cross-disaggregated <input type="radio"/> Results are parallel disaggregated				

Note: this setting will be ignored because there are no disaggregations

**Geographic Disaggregation**

Results are reported by:

The current geographic disaggregation is more specific than the selected count unique columns. Change your geographic disaggregation or select an appropriate geography column in your count unique columns list.

☒ Location
 ☐ Prefecture
 ☐ Region
 ☐ Guinea

To fix this error, add the appropriate geography column to your **Count Unique Columns** list.

Data Source

Data Source

☐ Enter indicator results directly
☒ Populate from a data table
☐ Calculate from a formula

Data Table

Training Table

View Table Data

Calculation

Count Unique Sets

using

COUNT UNIQUE COLUMN(S)

Column	Type
Activity	Activity
Location of Training	Geography: Location
Trainee First Name	Other Text
Trainee Last Name	Other Text
Add column...	

### Troubleshooting Tips for Count Unique Columns

- If you use the count type *deduplicate across dates*, the calculation results are assigned to the most recent date in the data table, without any time series or historical data.
- If you don't count unique per geography, the calculation results are assigned to the largest geographic division in the system, such as the whole country or the whole world.
- If the indicator is designated to be reported per activity, then the calculation generates separate unique counts for each activity.
- You can still add filters as usual so that the indicator calculation ignores any rows of data that do not meet the filter criteria.
- Rows of data are excluded from the calculation if they don't have complete information for all columns relevant to the indicator definition.

[back to top](#)

## Data table indicator calculation: Sum

The **Sum** calculation gives a total of a numeric column in your data table.

To create a sum indicator, set the data table **Calculation** to **Sum** and select the column of data to be totaled. The dropdown only includes columns defined with numeric formats.

Data Source

Data Source

☐ Enter indicator results directly
☒ Populate from a data table
☐ Calculate from a formula

Data Table

Training Table

View Table Data

Calculation

Sum

of

Number of Days Attended

### What result do I get from Sum?

The result for this indicator mapping will be a sum of the values in the **Number Days Attended** column. In reports, the results can be subdivided by reporting period and geographic place, plus by activity and by any other disaggregations

if relevant.

Rows of data are excluded from the calculation if they don't have complete information for all columns relevant to the indicator definition, or if filters are applied.

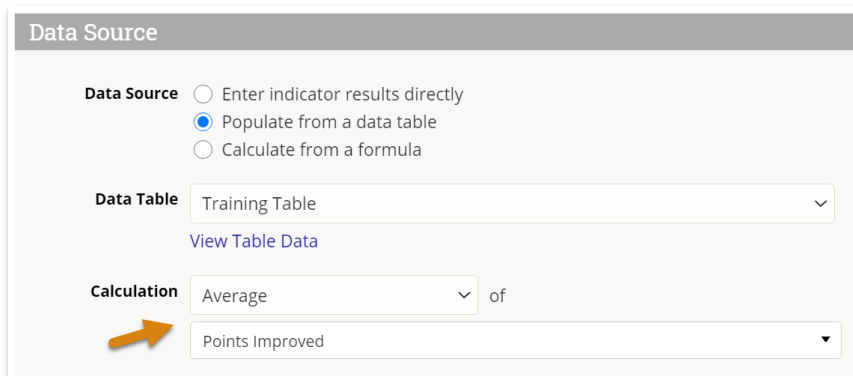
[back to top](#)

## Data table indicator calculation: Average

---


The **Average** calculation gives an average of a numeric column in your data table.



To create an average indicator, set the data table **Calculation** to **Average** and select the column of data to be averaged. The dropdown only includes columns defined with numeric formats.



**Data Source**

**Data Source** ☐ Enter indicator results directly  
☒ Populate from a data table  
☐ Calculate from a formula

**Data Table** Training Table   
[View Table Data](#)

**Calculation** Average  of  
Points Improved 

### What result do I get from Average?

The result for this indicator mapping will be a average of the values in the **Points Improved** column. In reports, the results can be subdivided by reporting period and geographic place, plus by activity and by any other disaggregations if relevant.

Rows of data are excluded from the calculation if they don't have complete information for all columns relevant to the indicator definition, or if filters are applied.

[back to top](#)

## Apply filters

---

Indicators populated from a data table can filter rows to include only those relevant to your indicator. For example, an indicator that counts *number of women* should have the filter "Sex = Female" in a table that includes both females and males. You might not need any filters, or you might need one or more filters depending on your data table and your indicator.

A filter should express the data that you want to *include*. You can create a filter based on any field from a data table, though it is not advisable to use "Other Text" fields as spelling errors, typos, or bounding quotation marks (" ") will yield unexpected results.

Filters can use one of four operators:

- = (equal-to)
- ≠ (not equal-to)
- **has a value** (not null)
- **has no value** (null)



In this example, the indicator will only count trainees who have a final test score reported. You could also count only young adult trainees with the filter "Age = Young Adult", or select only those trained in business administration with the filter "Training Type = business administration".

A screenshot of a filter configuration interface. It shows a single filter: "Final Test Score" with the operator "has a value" and a dropdown arrow. Below this is a text input field with the placeholder "Add a new filter..." and a dropdown arrow. A trash icon is visible to the right of the filter.

## Multiple filters

When combining multiple filters, you can use either AND or OR logic (but not both).

By default, multiple filters will be combined using 'AND' logic, meaning that only a row of data that meets ALL of that criteria will be included in the count. In the example below, all trainees will be excluded except for:

- Those who are female,
- *and* where the training type was filled out,
- *and* the training topic was "Technology",
- *and* the trainee improved more than 10 percentage points.

A screenshot of the "Data Source" configuration interface. It shows the following settings: "Data Source" is set to "Populate from a data table"; "Data Table" is "Training Table" with a "View Table Data" link; "Calculation" is "Count All"; "Geography Column" is "Location of Training"; "Date Column" is "Date". Below these, there is a list of filters enclosed in an orange box: "Sex = Female", "and Training Type has a value", "and Training Topic = Technology", and "and Points Improved > 10 %". Each filter has a trash icon to its right. At the bottom is a text input field "Add a new filter..." with a dropdown arrow.

Alternatively, 'OR' logic can be enabled by checking the option to " **Include data matching any of the above filters** ". Rows of data that meet ANY of the criteria will be included in the count. In the example below, trainees would only be excluded if they were *neither*:

- Female
- Attending a training in a well-supplied classroom

A screenshot of the "Data Source" configuration interface showing "OR" logic. The first filter is "Sex = Female". Below it, there is a radio button labeled "or" (which is selected) and a second filter "Classroom ID: Well-supplied? = Yes". Below the filters, there is a checkbox labeled "Include data matching any of the above filters" which is checked. At the bottom is a text input field "Add a new filter..." with a dropdown arrow.

In other words:

- A **female** trainee in a classroom that is **well-supplied** would be counted (meets both criterion)
- A **female** trainee in a classroom that is **not well-supplied** would be counted (meets first criterion)
- A **male** trainee in a classroom that is **well-supplied** would be counted (meets second criterion)
- A **male** trainee in a classroom that is **not well-supplied** *would not be counted* (does not meet either criteria)

## Troubleshooting

A common error arises when using multiple indicator filters on the same data table column. For example, if you want to create an indicator that counted rows where the service type is "nutrition" or "health", you might include two filters:

- Service Type = Nutrition
- Service Type = Health

The screenshot shows the 'Data Source' configuration window. It has several sections: 'Data Source' with radio buttons for 'Enter indicator results directly', 'Populate from a data table' (selected), and 'Calculate from a formula'; 'Data Table' with a dropdown set to 'Beneficiaries' and a 'View Table Data' link; 'Calculation' with a dropdown set to 'Count All'; 'Geography Column' with a dropdown set to 'Location' and a note 'Data will be associated with places from this column.'; and 'Date Column' with a dropdown set to 'Date' and a note 'Data will be associated with dates from this column.'.

The 'Filters' section contains two filter rules, both for the 'Service Type' column:

Filters	Service Type	=	Nutrition	
and	Service Type	=	Health	

Below the filters is a button 'Add a new filter...'. An orange arrow points to a warning box at the bottom:

**⚠ These filters exclude all data because it is impossible for all filters to be true for a single row of the data table. Please re-write the filters such that a row of data could meet all filter requirements.**

The problem is that **no rows** could have **both** nutrition and health for the service type **at the same time**. Any individual row could only have either nutrition or health as the service type, but not both. Since all of the filters must be true for a row in order to count that row toward the indicator result, this configuration would give you zero results.

How do you fix it? Look at the disaggregation categories for this disaggregation.

## Disaggregation by Service Type

Details			
Name	Service Type		
Description	Disaggregation definition.		
Disaggregation categories		Category	Description
	⌵	Nutrition	
	⌵	Education	
	⌵	Health	
	⌵	Finance	

The indicator needs to count rows where the service type is equal to two of these categories, which is the same as **not equal** to the other two categories. Instead of filtering using the categories you want to include, filter out the categories that you need to exclude:

Data Source			
Data Source	<input type="radio"/> Enter indicator results directly <input checked="" type="radio"/> Populate from a data table <input type="radio"/> Calculate from a formula		
Data Table	Beneficiaries <span>▼</span> <a href="#">View Table Data</a>		
Calculation	Count All <span>▼</span>		
Geography Column	Location <span>▼</span> <small>Data will be associated with places from this column.</small>		
Date Column	Date <span>▼</span> <small>Data will be associated with dates from this column.</small>		
Filters	Service Type	≠	Education <span>▼</span> <span>🗑️</span>
	and Service Type	≠	Finance <span>▼</span> <span>🗑️</span>
<a href="#">Add a new filter...</a> <span>▼</span>			

Both of these filters can be true for a single row of data **at the same time**. This configuration will give you the intended results: a count of clients who received nutrition or health services.

[back to top](#)

Didn't answer your question? Please email us at [help@devresults.com](mailto:help@devresults.com).

---

### Related Articles

---