Create a Query Wizard filter

Last Modified on 10/27/2026 2:28 pm CDT

Path: Reporting > Ad hoc Reporting > Filter Designer

Use Query Wizard to create dynamic Ad hoc filters that return current data based on selected fields, filter criteria, output settings, and aggregation options. This guide explains how to configure each Query Wizard screen to create a filter for reporting, exports, and other Ad hoc processes.

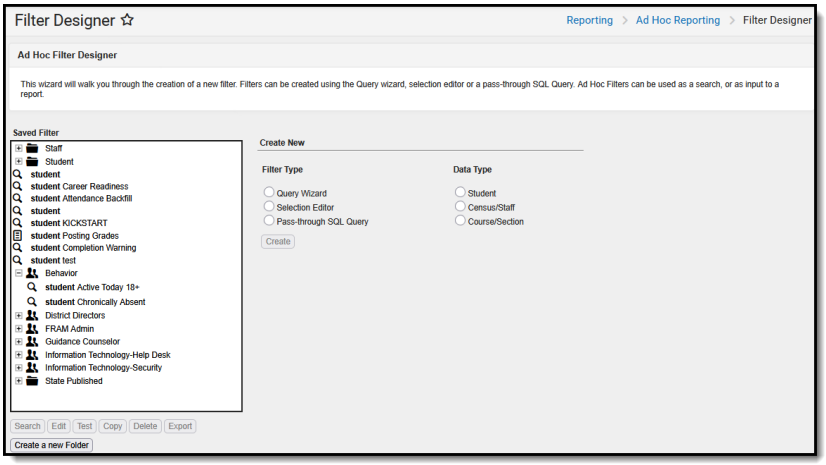
Before you start

Before creating a filter:

- Determine the data you need to report and whether it is Student, Census/Staff, or Course/Section data. Student and Course/Section queries use the calendar selected in the Campus toolbar; Census/Staff queries use data from the entire Campus database.
- Identify the fields you want to include in the filter. For complex queries, identify the fields you want to display and any criteria needed to limit results.
- Large queries can affect system performance. Apply filters whenever possible to limit the number of records returned.
- Verify you have the appropriate tool rights to create and save filters.

Step 1 - Select the filter and data type

Specify the Filter Type and Data Type to create a new filter.

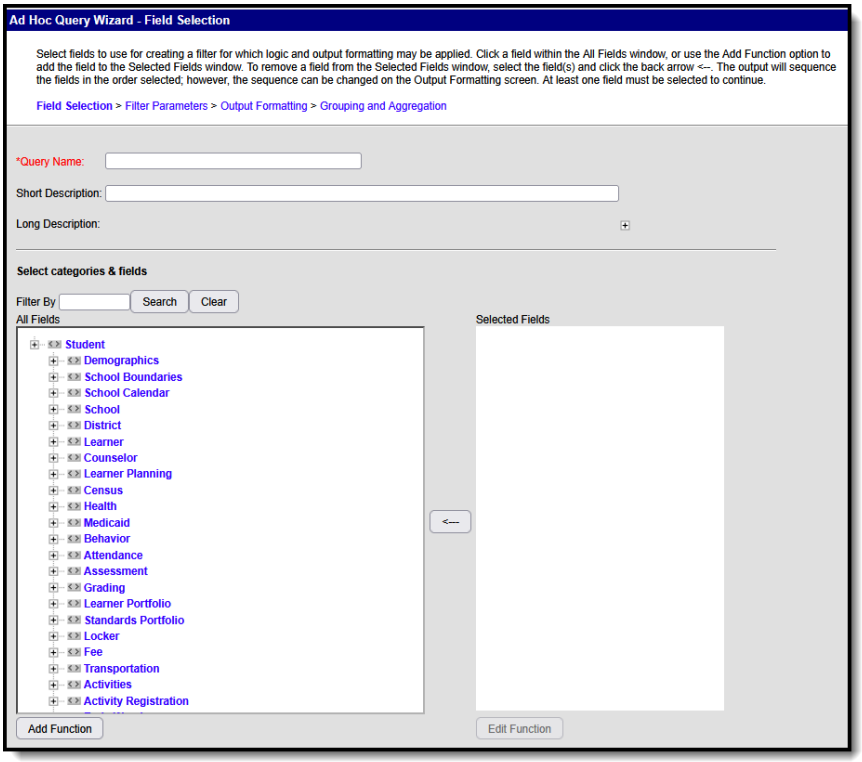


1. Select the Query Wizard **Filter Type.**
2. Select a **Data Type.** The selected data type determines which fields are available when building the filter.
    - Student
    - Census/Staff
    - Course/Section
3. Click **Create**.

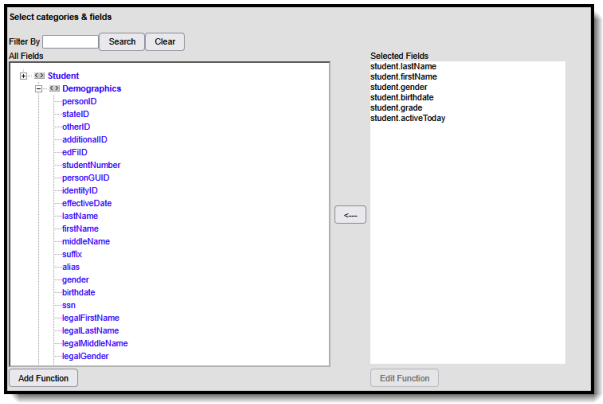
Step 2 - Select fields for the filter

The Field Selection screen determines which data elements are available for filtering, output, grouping, and aggregation.

- Only selected fields can be used later for filtering, output formatting, grouping, and aggregation.
- Short and Long Descriptions help users understand the purpose of saved filters.
- Query Wizard filters are dynamic and always return current database information.



1. Enter a **Query Name**.
2. Enter a **Short Description** and/or **Long Description** to provide additional information and context about the filter**.** Select the (**+**) icon to access the **Long Description** text box.
3. Use **Filter By** to search for a field by entering part of its name and clicking **Search** OR expand the categories to locate available fields and subcategories.
4. Click the field name displayed in blue in the **All Fields** list to move it to the **Selected Fields** list.



5. Optional: Click **Add Function** to apply functions to selected fields.

6. Click **Next** to continue configuring the filter. This may require that you scroll down the page.
7. Select **Save** or **Save and Test** to save the filter before completing the remaining configuration steps
8. Click **Return to List** to navigate back to the Filter Designer home page.
9. Click **Next** to open the Filter Parameters screen.

Custom Tool field behavior

When a filter includes fields from a **Custom Tool**, the Query Wizard returns all students in the result set, even if a student does not have a corresponding record in the custom dated tab.

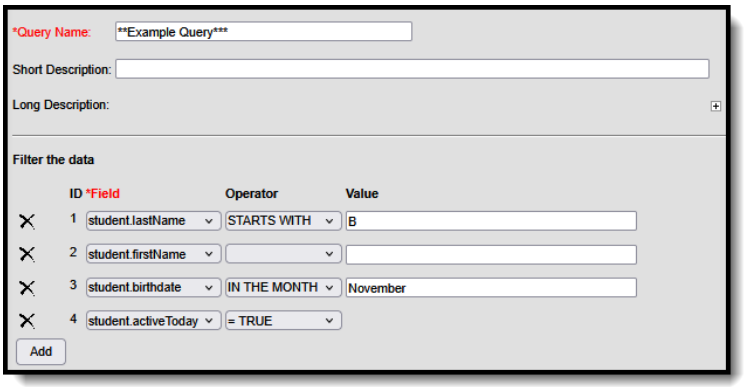
To exclude students who do not have records in a custom tab that uses the Table or List Element tab type, set the **statusDate** operator to **IS NOT NULL**. This limits results to students with associated custom-tab records.

When Query Wizard returns data from Table or List Element custom tabs, the filter logic evaluates all available date and time combinations stored for those records and returns every applicable combination. Table tab types store both date and time values, while List Element tab types always store a time value of 12:00 AM. This behavior can cause additional records to appear in the filter output when multiple dated custom-tab records exist for a student.

For more information about custom tab types and how custom data is stored, see Custom Tool Setup.

Step 3 - Define filter parameters

Use Filter Parameters to determine which records to include in the results.

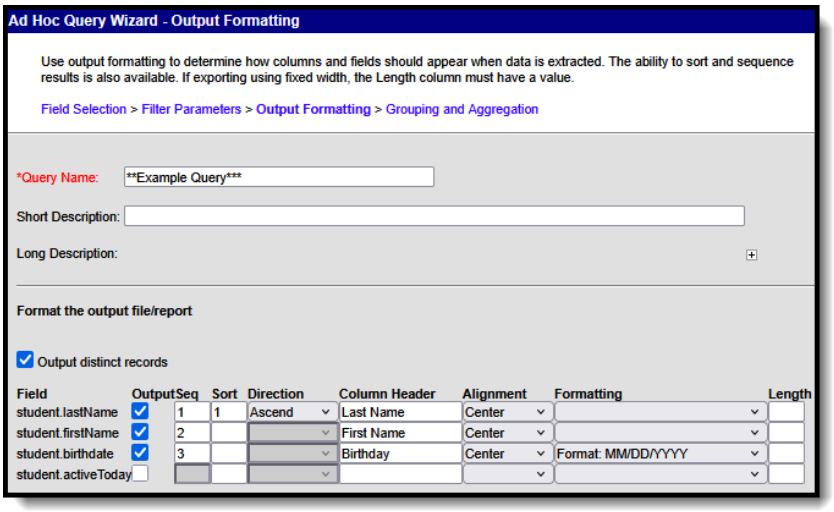


1. Review the selected fields.
2. Select an **Operator** for each of the applicable fields.
3. Enter a **Value** for each operator. The value works with the selected operator to determine which records are returned.
    - Result: *Only students enrolled as of today whose last names begin with "B" and have birthdays in November are included in the results.*
4. When using string-based operators such as LIKE, CONTAINS, STARTS WITH, or ENDS WITH,

you can use wildcard characters to broaden your search criteria. Common wildcard characters include:
    - % = zero or more characters (for example, L% finds names that start with L)
    - _ (underscore) = a single character (for example, L__ finds three-character names that start with L)
    - [token] = a range of possible characters (for example, L[ae]% finds names that start with La or Le)
    - ^ = excludes characters within a token set (for example, L[^ae]% finds names that do not start with La or Le)
5. Click **Add** to apply multiple conditions to the same field. Click the "**X**" at the start of the row to remove that data element.
6. Optional: Enter a **Logical Expression** to control how multiple conditions are evaluated.
7. Select **Save** or **Save and Test** to save the filter before completing the remaining configuration steps.
8. Click **Next** to open the **Output Formatting** screen.

Step 4 - Configure output formatting

Use Output Formatting to determine how data displays when exported or reported.

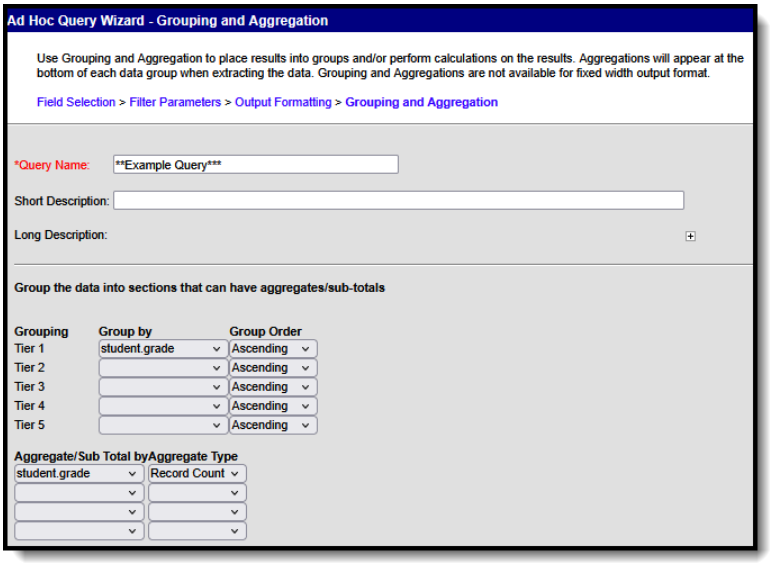


1. Select **Output distinct records** to remove duplicate records from the results.
2. Verify the **Output** checkbox is selected for any fields that should appear in the filter output.
3. Enter a **Sequence** number to determine the order of fields in the output.
4. Enter a **Sort** value to define the sort order.
5. If a **Sort** value is entered, select a **Direction** of Ascending or Descending.
6. Enter a **Column Header** for each field (for example, State ID). If a column header is not specified, the system will use the field name as the default column header (for example, student.stateID).
7. Select an **Alignment** option (Left, Right, Center).
8. Select a **Formatting** option to determine how values are reported for the field when used in reports and exported files. Formatting options are important for filters used in reports that require specific formatting to ensure the file is submitted correctly to an entity or system.
9. Enter a **Length** value if needed. Length determines the maximum number of characters returned for the field.

10. Select **Save** or **Save and Test** to save the filter before completing the remaining configuration steps.
11. Click **Next** to open the **Grouping and Aggregation** screen.

Step 5 - Configure grouping and aggregation

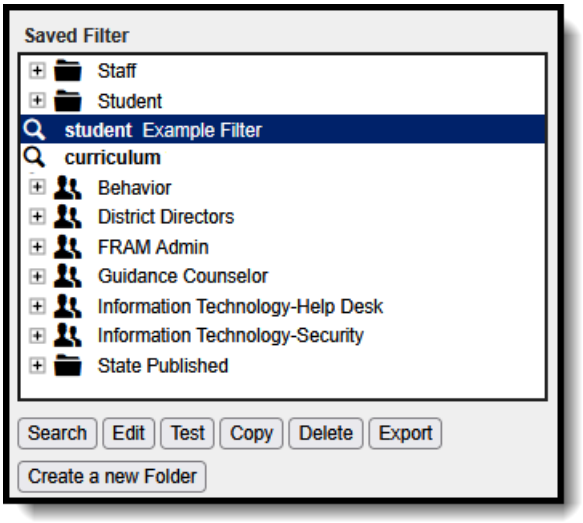
Use the Grouping and Aggregation page to organize query results into groups and calculate summary values for each group.



1. Select a **Group By** field for each grouping level. Grouping organizes related records into sections.
2. Select a **Group Order** for each grouping level to determine how group aggregate/sub-totals are reported when exported via the Data Export tool.
3. Select a field to **Aggregate/SubTotal by.** This determines which field within the filter is used for the 'Group by' fields.
4. Select an **Aggregate Type** to calculate totals or other sum values for grouped data.
5. Continue configuring additional grouping levels as needed.
6. Click **Save and Test** to save the filter and review the results.
7. Optional: Mark the **Force Order** checkbox for complex filters that require performance optimization.

Step 6 - Generate a saved filter

Filters saved to the **Saved Filter** list are available for immediate testing, reporting, and data exports.



1. Select the filter from the **Saved Filter** list.
2. Select the appropriate **Calendar**.
3. Click **Test**. The filter opens in a separate window for review.
4. For Census/Staff filters, a calendar selection is not required

Impact

Users with appropriate permissions can reuse, modify, and share the filter with authorized user groups

What's next

- Modify the filter to add additional criteria or formatting.
- Save the filter to a folder for organization and easier access.
- Share the filter with authorized user groups.
- Use the filter in Data Export, Letter Builder, and other Ad hoc reporting tools.
- Modify saved filters to add fields, update criteria, or adjust output formatting as reporting requirements change.

Appendix A - Common operators

Operator	Results	Example
= (Equals)	Returns exact match of value.	student.grade=3 Only students in grade 3 are returned.

Operator	Results	Example
< > (Does not equal)	Returns results not equal to the value.	student.gender < > M Students who have Gender = F assigned in their Identities record, or who do not have a value entered in the Grade field, are returned. This operator allows NULL values.
> (Greater than)	Returns results that are greater than the entered numeric value.	student.age > 16 All students older than 16 are returned.
> = (Greater than or equal to)	Returns results that are greater than or equal to the entered numeric value.	student.age >= 16 All students aged 16 and older are returned.
< (Less than)	Returns results that are less than the entered numeric value.	student.age < 16 All students under the age of 16 are returned.
< = (Less than or equal to)	Returns results that are less than or equal to the entered numeric value.	student.age <= 16 All students aged 16 or younger are returned.
IN	Includes value. Do not include spaces after the comma when using this format.	student.grade IN 9,10 All students in 9th and 10th grade are returned.
NOT IN	Excludes value. Do not include spaces after the comma when using this format.	student.grade NOT IN 11,12 All students not in 11th or 12th grade are returned. This operator allows NULL values.
BETWEEN	Filters data between two specified values. Works with numbers, dates, and strings. If a date field is selected, the following options are available: • **DATE** - Returns data based on the specified date range (where	For **BETWEEN**: student.stateID BETWEEN 00001 THROUGH 100000. All students with a State ID between 00001 - 100000 are returned. For **DATE**: student.birthDate BETWEEN DATE 10151995 THROUGH DATE

Operator	Results	Example
	the starting date is sub-option 1 and the ending date is sub-option 2). • **TODAY** - Filters data based on dates that occur from a specific date through today or vice versa. • **TOMORROW** - Filters data based on dates that occur from a specific date through tomorrow or vice versa. • **YESTERDAY** - Filters data based on dates that occur from a specific date through yesterday or vice versa. • **DAYS BEFORE** - Filters data based on the number of days (sub-option 1) prior to sub-option 2 through sub-option 2. • **MONTHS BEFORE** - Filters data based on the number of months (sub-option 1) prior to sub-option 2 through sub-option 2. • **DAYS AFTER** - Filters data based on sub-option 1 through the number of days (sub-option 2) after the sub-option 1 date. • **MONTHS AFTER** - Filters data based on sub-option 1 through the number of months (sub-option 2) after the sub-option 1 date.	10152010. All students with a birth date between 10/15/1995 - 10/15/2010 are returned. For **TODAY**: student.startDate BETWEEN TODAY THROUGH TODAY. All students enrolled at the school today (current date) have been returned. For **YESTERDAY**: student.startDate BETWEEN YESTERDAY THROUGH DATE 10152010. All students enrolled at the school between yesterday and 10/15/2010 are returned. For **DAYS BEFORE**: student.startDate BETWEEN DAYS BEFORE 4 THROUGH YESTERDAY. All students who began enrollment in the school 4 days before yesterday through yesterday are returned. For **MONTHS BEFORE**: student.startDate BETWEEN MONTHS BEFORE 5 THROUGH TODAY. All students who began enrollment in the school 5 months prior to today through today are returned. For **DAYS AFTER**: student.startDate BETWEEN DATE 10152010 THROUGH DAYS AFTER 5. All students who began enrollment at the school between 10/15/2010 and 10/20/2010 (5 days later) are returned. For **MONTHS AFTER**: student.startDate BETWEEN DATE 10152010 THROUGH MONTHS AFTER 5. All students enrolled at the school between 10/15/2010 and 3/15/2011 (five months

Operator	Results	Example
IS CURRENT USER	Returns the current user's ID.	later) are returned. **Learner Plan Manager** Setting learningPlan.planManagerPersonID IS CURRENT USER reports the current user's ID, along with data only applicable to that user. **Current Teacher Sections** For courseSection.personID IS CURRENT USER limits the results to students in the current teacher's section. This is useful for a report on student birthdays by homeroom, or for a Spirit Squad Advisor who needs to make locker signs and a list of participants and locker information.
LIKE	Searches for test string in the field.	course LIKE hist All courses like History 101 are returned.
NOT LIKE	Searches for test string and filters data that is not like the user-defined value.	course NOT LIKE hist All courses not like Hist are returned. This operator allows NULL values.
SOUNDS LIKE	Uses a database function to return names with similar sound patterns.	student.lastName SOUNDS LIKE Ball Names such as "Ball," "Bell" and "Boll" are returned.
CONTAINS	Searches for strings that include the same data entered by the user in the field. Any string that does not contain the user-defined value is filtered out. Any wildcard characters entered are treated as standard SQL wildcards.	student.birthCountry CONTAINS Cana All students with a Birth Country that contains "Cana" are returned.
STARTS WITH	Searches for strings that begin with the same data the user entered in the field. Any string that does not contain the user-defined value is filtered out. Any wildcard characters entered are treated as standard SQL wildcards.	student.birthCountry STARTS WITH Mexi All students with a Birth Country that begins with "Mexi" are returned.

Operator	Results	Example
ENDS WITH	Searches for strings that end with the same data entered by the user in the field. Any string that does not contain the user-defined value is filtered out. Any wildcard characters entered are treated as standard SQL wildcards.	student.birthCountry ENDS WITH many All students with a Birth Country ending in "many" are returned.
IS NULL	Returns fields that are completely NULL (0 is considered a value).	student.stateID IS NULL All students without a state ID are returned.
IS NOT NULL	Returns all fields that are not NULL (0 is considered a value).	student.ssn IS NOT NULL All students who have a stateID are returned.
IS TODAY	Returns result dates as the current date.	start.date IS TODAY Entries where the start.date is the current date are returned.
IS YESTERDAY	Returns results as of yesterday.	start.date IS YESTERDAY Results for one day previous to the current date are returned.
IS TOMORROW	Returns result dates as of tomorrow's date.	end.date IS TOMORROW Results for one day after the current date are returned.
IN THE MONTH	Returns all database field data for the month entered. This operator allows both numbered dates and spelled-out dates (e.g., 10 or October). It also allows both uppercase and lowercase letters. If spelling out a month, users must enter at least the first three characters (e.g., Oct for October).	employment.districtStartDate IN THE MONTH October All employees with a district employment Start Date in October are returned. This operator does not look at the Year or Calendar selected in the Campus toolbar. All historical and current district employment records with a Start Date in October are returned.
=TRUE	Returns checkbox values of "true" (checkbox is marked)	enrollment.stateExclude = TRUE All students with the State Exclude checkbox marked on their enrollment records are returned.

Operator	Results	Example
=FALSE	Returns checkbox values of "false" (checkbox is not marked)	enrollment.stateExclude = FALSE All students without the State Exclude checkbox checked on their enrollment records are returned.

Appendix B - Aggregation types

Aggregation Type	Description
Record Count	Counts all records
Distinct Count	Counts unique records
MIN	Returns the lowest value
MAX	Returns the highest value
SUM	Adds all values
AVG	Returns the average value