

# Defining a filter

## Introduction

In order to use scope you have to define an expression as explained in [using scope and filter expressions](#).

### On this page:

- [Introduction](#)
- [Where are expressions used](#)
  - [Expressions used to define API filters](#)
  - [Expressions used to define scope](#)
  - [Expressions used to define instruction result filters](#)

## Where are expressions used

There are three main use cases for expressions.

They all share the same syntax as seen on [Using scope and filter expressions](#) page.

In each case, data type field should be null when the expression is sent to the API. Any value passed in that field will be ignored.

### Expressions used to define API filters

Many API endpoints support filtering and sorting of data they return. These often (but not always) include the word 'Search' in their URI, like '/Consumers/Search' (see [Consumers](#) page).

In this case, the attribute in the expression should be one of the columns given endpoint allows filtering on. Documentation for each API endpoint will list which columns are supported in a filter and sort expressions.

### Expressions used to define scope

Scope expressions have a pre-defined set of columns that can be used as attributes. Please see the [Defining the scope](#) page.

Scope expressions are used by endpoints like '/Consumer/Devices/Scope'.

### Expressions used to define instruction result filters

When an instruction is issued it can have a filter applied to its results to better refine the result set. Likewise, a follow up instruction can filter results of the previous instruction to make sure the follow up is executed only when necessary.

[Results filter for an instruction](#) page discusses this use case in greater details.