

# Using search filters with the HTTP API

## Usable fields

Any field in the object's search definition can be used in a filter.

## Filter operations

- equal or equals - the value is the same as the criteria (also used as an "in" operation when the criteria is an array)
- notequal or notequals - the value is different than the criteria
- lessthan - the value is less than (numerically or alphabetically)
- lessthanorequal/lessthanorequals - the value is less or equal to the criteria
- greaterthan - the value is more than (numerically or alphabetically)
- greaterthanorequal or greaterthanorequals - the value is more than or equal to the criteria
- contains/notcontain (string) - the value includes the provided string (e.g., smithe contains smith)
- blank/notblank - does not have a value
- radius - this is a special case for use with location/radius searches including animals, organizations and events. See Radius searches below.



If you want to filter by multiple values (for example, you want to retrieve all pets that are either Baby or Young), you can use the equals operation, and provide an array for the criteria. So, animalGeneralAge equals array("Baby","Young"). An array in the criteria is handled similar to an "in" clause.

## Criteria

The criteria for a filter should match the data/field type (see data types on the [HTTP API field properties and data types](#) page).



Leading and trailing spaces are trimmed from the criteria.

There are some special cases:

- rg:contactID - the user's current contactID is used
- rg:today - today's date is used (eastern time)

## Filter processing



Filter processing is an advanced feature, and is not required. If you do not include the filterProcessing property to your searches, the filters will be evaluated with "and" (i.e, 1 and 2 and 3, etc.).

The RescueGroups.org HTTP API provides a very powerful way to perform advanced searches.

The method involves sending a string representation of how the filters you provided should be evaluated.

In the following example, we want to search for either small dogs or cats. The filters below specify all of the criteria:

```
"filters" => array(
  array(
    "fieldName" => "animalSpecies",
    "operation" => "equals",
    "criteria" => "dog",
  ),
  array(
    "fieldName" => "animalGeneralSizePotential",
    "operation" => "equals",
    "criteria" => "small",
  ),
  array(
    "fieldName" => "animalSpecies",
    "operation" => "equals",
    "criteria" => "cat",
  ),
),
```

In order to tell the API to process the filters so that we get small dogs and cats, we would include the filterProcessing value:

```
"filterProcessing" => "(1 and 2) or 3",
```



The first filter is represented by "1" in the filterProcessing string, the second by "2" and so on. Even if you send the filters as 0 1, etc, the filterProcessing string must use 1, 2, etc.

## Radius searches

Some objects, including animals, organization and events, allow for a radius search. For example, a radius search for animals could find adoptable animals that are within a certain number of miles of a specific zip/postal code.



If you notice odd sorting issues (like the results don't seem to be sorting the way you requested), check the tool that you are using. Some tools, like Postman, seem to re-sort results for some reason. If you experience this issue be sure to check the raw results to see if they are sorting correctly.

The following is an example of filters that would search for organizations within 90 miles of the 90210 zip code:

```
"filters" => array(
  array(
    "fieldName" => "orgLocationDistance",
    "operation" => "radius",
    "criteria" => "90",
  ),
  array(
    "fieldName" => "orgLocation",
    "operation" => "equals",
    "criteria" => "90210",
  ),
),
```

- animals: animalLocation for the zip/postal code and animalLocationDistance for the distance in miles
- orgs: orgLocation for the zip/postal code and orgLocationDistance for the distance in miles
- events: \*\* COMING SOON



You can sort the results by the distance field so that you can show the closest results at the top of the list.