Package ‘PetfindeR’

October 12, 2022

Type Package
Title 'Petfinder' API Wrapper
Version 2.1.0
Author Aaron Schlegel
Maintainer Aaron Schlegel <aaron@aaronschlegel.me>
Description Wrapper of the 'Petfinder API' <https://www.petfinder.com/developers/v2/docs/> that implements methods for interacting with and extracting data from the 'Petfinder' database. The 'Petfinder REST API' allows access to the 'Petfinder' database, one of the largest online databases of adoptable animals and animal welfare organizations across North America.
Encoding UTF-8
LazyData true
RoxygenNote 7.1.1
License MIT + file LICENSE
URL https://github.com/aschleg/PetfindeR
VignetteBuilder knitr
Imports R6
Suggests lubridate, testthat, knitr, rmarkdown, dplyr, httr, jsonlite
NeedsCompilation no
Repository CRAN
Date/Publication 2020-07-07 16:10:02 UTC

R topics documented:

.Petfinder.class ................................................................. 2
animals ........................................................................ 3
animal_types .................................................................. 5
breeds ............................................................................ 6
organizations .................................................................... 7
Petfinder ........................................................................... 8

Index 10
Description

Internal R6 class

Methods

Public methods:

• `.Petfinder.class$new()`
• `.Petfinder.class$animal_types()`
• `.Petfinder.class$breeds()`
• `.Petfinder.class$animals()`
• `.Petfinder.class$organizations()`
• `.Petfinder.class$clone()`

Method `new()`:

Usage:

`.Petfinder.class$new(key, secret)`

Method `animal_types()`:

Usage:

`.Petfinder.class$animal_types(types = NULL)`

Method `breeds()`:

Usage:

`.Petfinder.class$breeds(animal_types = NULL, return_df = FALSE)`

Method `animals()`:

Usage:

`.Petfinder.class$animals(`

  animal_id = NULL,
  animal_type = NULL,
  breed = NULL,
  size = NULL,
  gender = NULL,
  age = NULL,
  color = NULL,
  coat = NULL,
  status = NULL,
  name = NULL,
  organization = NULL,`
animals

Returns adoptable animal data from Petfinder based on specified criteria.

Description

Returns adoptable animal data from Petfinder based on specified criteria.
Usage

animals(
  animal_id = NULL,
  animal_type = NULL,
  breed = NULL,
  size = NULL,
  gender = NULL,
  age = NULL,
  color = NULL,
  coat = NULL,
  status = NULL,
  name = NULL,
  organization_id = NULL,
  location = NULL,
  distance = NULL,
  sort = NULL,
  pages = 1,
  results_per_page = 20,
  return_df = FALSE
)

Arguments

animal_id Integer or vector or list of integers representing animal IDs obtained from Petfinder. When animal_id is specified, the other function parameters are overridden. If animal_id is not specified, a search of animals on Petfinder matching given criteria is performed.

animal_type Character vector representing desired animal type to search. Must be one of 'dog', 'cat', 'rabbit', 'small-furry', 'horse', 'bird', 'scales-fins-other', or 'barnyard'.

breed Character vector or vector or list of character strings of desired animal type breed to search. Available animal breeds in the Petfinder database can be found using the :code:`breeds()` method.

size Character vector or vector or list of character strings of desired animal sizes to return. The specified size(s) must be one of 'small', 'medium', 'large', or 'xlarge'.

gender Character vector or vector or list of strings representing animal genders to return. Must be of 'male', 'female', or 'unknown'.

age Character or vector or list of strings specifying animal age(s) to return from search. Must be of 'baby', 'young', 'adult', 'senior'.

color String representing specified animal 'color' to search. Colors for each available animal type in the Petfinder database can be found using the animal_types() method.

coat Desired coat(s) to return. Must be of 'short', 'medium', 'long', 'wire', 'hairless', or 'curly'.
### status
Animal status to filter search results. Must be one of 'adoptable', 'adopted', or 'found'.

### name
Searches for animal names matching or partially matching name.

### organization_id
Returns animals associated with given code:organization_id. Can be a vector or a vector or list of character strings representing multiple organizations.

### location
Returns results by specified location. Must be in the format 'city, state' for city-level results, 'latitude, longitude' for lat-long results, or 'postal code'.

### distance
Returns results within the distance of the specified location. If not given, defaults to 100 miles. Maximum distance range is 500 miles.

### sort
Sorts by specified attribute. Leading dashes represents a reverse-order sort. Must be one of 'recent', '-recent', 'distance', or '-distance'.

### pages
Specifies which page of results to return. Defaults to the first page of results. If set to NULL, all results will be returned.

### results_per_page
Number of results to return per page. Defaults to 20 results and cannot exceed 100 results per page.

### return_df
If TRUE, a data.frame will be returned. Does not currently do anything.

## Value
List of JSON data of resulting animals.

## Examples
```r
## Not run:
pf <- Petfinder(key=key, secret=secret) # Initialize Petfinder class
pf$animals(results_per_page = 100, pages = 10)
pf$animals(location='Seattle, WA', distance = 100, results_per_page = 50, pages = 5)
pf$animals(animal_id=c(animalid1, animalid2, animalid3))
## End(Not run)
```

---

### animal_types
Returns data on an animal type, or types available from the Petfinder API. This data includes the available type’s coat names and colors, gender and other specific information relevant to the specified type(s). The animal type must be of 'dog', 'cat', 'rabbit', 'small-furry', 'horse', 'bird', 'scales-fins-other', 'barnyard'.

## Description
Returns data on an animal type, or types available from the Petfinder API. This data includes the available type’s coat names and colors, gender and other specific information relevant to the specified type(s). The animal type must be of 'dog', 'cat', 'rabbit', 'small-furry', 'horse', 'bird', 'scales-fins-other', 'barnyard'.

---
Usage

animal_types(types = NULL)

Arguments

types

Specifies the animal type or types to return. Can be a character vector representing a single animal type, or a vector or list of animal types if more than one type is desired. If not specified, all animal types are returned.

Value

List of returned JSON data for each specified animal type from the Petfinder API.

Examples

```r
## Not run:
pf <- Petfinder(key=key, secret=secret) # Initialize Petfinder class
cat <- pf$animal_types(types='cat')
cat_dog <- pf$animal_types(types=c('cat', 'dog'))
all_types <- pf$animal_types()
## End(Not run)
```

---

breeds

Returns breed names of specified animal type, or types.

Description

Returns breed names of specified animal type, or types.

Usage

breeds(types, return_df = TRUE)

Arguments

types

Specifies the animal type or types. Can be a character vector representing a single animal type, or a vector or list of animal types if more than one type is desired. If not specified, all available breeds for each animal type is returned. The animal type must be `dog`, `cat`, `rabbit`, `small-furry`, `horse`, `bird`, `scales-fins-other`, `barnyard`.

return_df

If TRUE, the result set will be coerced into a pandas data.frame with two columns, breed and name.

Value

List of returned JSON data of available breeds for each specified animal type from the Petfinder API. If the parameter return_df is TRUE, a data.frame is returned instead.
Examples

```r
## Not run:
pf <- Petfinder(key=key, secret=secret) # Initialize Petfinder class
cat_breeds <- pf$breeds('cat')
cat_dog_breeds <- pf$breeds(c('cat', 'dog'))
all_breeds <- pf$breeds()
all_breeds_df <- pf$breeds(return_df=TRUE)

## End(Not run)
```

### organizations

Returns data on an animal welfare organization, or organizations, based on specified criteria.

#### Description

Returns data on an animal welfare organization, or organizations, based on specified criteria.

#### Usage

```r
organizations(
  organization_id = NULL,
  name = NULL,
  location = NULL,
  distance = NULL,
  state = NULL,
  country = NULL,
  query = NULL,
  sort = NULL,
  results_per_page = 20,
  pages = 1,
  return_df = FALSE
)
```

#### Arguments

- `organization_id` Returns results for specified `organization_id`. Can be a character vector or a vector or list of character vectors representing multiple organizations.
- `name` Returns results matching or partially matching organization name.
- `location` Returns results by specified location. Must be in the format 'city, state' for city-level results, 'latitude, longitude' for lat-long results, or 'postal code'.
- `distance` Returns results within the distance of the specified location. If not given, defaults to 100 miles. Maximum distance range is 500 miles.
- `state` Filters the results by the selected state. Must be a two-letter state code abbreviation of the state name, such as 'WA' for Washington or 'NY' for New York.
country
Filters results to specified country. Must be a two-letter abbreviation of the
country and is limited to the United States and Canada.

query
Search matching and partially matching name, city or state.

sort
Sorts by specified attribute. Leading dashes represents a reverse-order sort.
Must be one of 'recent', '-recent', 'distance', or '-distance'.

results_per_page
Number of results to return per page. Defaults to 20 results and cannot exceed
100 results per page.

pages
Specifies which page of results to return. Defaults to the first page of results. If
set to NULL, all results will be returned.

return_df
If TRUE, the results will be returned as a data.frame. Currently does not do
anything.

Value
List of returned JSON data of organizations matching given search criteria.

Examples
## Not run:
```r
pf <- Petfinder(key=key, secret=secret) # Initialize Petfinder class
pf$organizations(state='WA', results_per_page = 100, pages = 2)
pf$organizations(organization_id=c(orgid1, orgid2, orgid3))
## End(Not run)
```

---

**Petfinder**

Creates an authenticated connection with the Petfinder API. The stored
authentication is then used to call the Petfinder API methods. An API
key can be obtained from Petfinder by creating an account on their
developer page (https://www.petfinder.com/developers/api-key). The
function wraps the .Petfinder.class R6 class.

---

**Description**

Creates an authenticated connection with the Petfinder API. The stored authentication is then used to
call the Petfinder API methods. An API key can be obtained from Petfinder by creating an account on
their developer page (https://www.petfinder.com/developers/api-key). The function wraps the
.Petfinder.class R6 class.

**Usage**

```r
Petfinder(key, secret)
```

**Arguments**

- **key**
  The API key received from Petfinder

- **secret**
  The secret key received from Petfinder along with the API key.
Value

Initialized Petfinder object that is then used to access the API.

Examples

```r
## Not run:
pf <- Petfinder(key) # Creates the connection with the Petfinder API.
pf$breeds('cat') # The connection can now be used to access the Petfinder API methods.

## End(Not run)
```
Index

.Petfinder.class, 2
animal_types, 5
animals, 3
breeds, 6
organizations, 7
Petfinder, 8