

# Package ‘dkanr’

February 18, 2019

**Title** Client for the 'DKAN' API

**Version** 0.1.3

**Description** Provides functions to facilitate access to the 'DKAN' API (<<https://dkan.readthedocs.io/en/latest/apis/index.html>>), including the 'DKAN' REST API (metadata), and the 'DKAN' datastore API (data). Includes functions to list, create, retrieve, update, and delete datasets and resources nodes. It also includes functions to search and retrieve data from the 'DKAN' datastore.

**License** MIT + file LICENSE

**Depends** R (>= 3.2)

**Imports** assertthat (>= 0.2.0), dplyr (>= 0.7.4), httr (>= 1.3.1), jsonlite (>= 1.5), purrr (>= 0.2.4), stringr (>= 1.2.0), tibble (>= 1.4.2)

**Suggests** covr (>= 3.0.1), knitr (>= 1.19), rmarkdown (>= 1.8), testthat (>= 2.0.0), httpptest

**VignetteBuilder** knitr

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.1

**NeedsCompilation** no

**Author** Tony Fujs [aut, cre],  
Meera Desai [aut],  
Karthik Ramanathan [aut]

**Maintainer** Tony Fujs <[tonyfujs@gmail.com](mailto:tonyfujs@gmail.com)>

**Repository** CRAN

**Date/Publication** 2019-02-18 15:50:02 UTC

## R topics documented:

attach_file_to_node . . . . .	2
create_node . . . . .	3
delete_node . . . . .	3

dkan_logout	4
dkan_settings	5
dkan_setup	5
ds_is_available	6
ds_search	7
ds_search_all	8
err_handler	9
filters_to_text_query	9
get_resource_nids	9
get_resource_url	10
is.dkan_list	11
is.dkan_node	11
list_nodes	12
list_nodes_all	13
retrieve_node	14
update_node	14

**Index** **16**

attach\_file\_to\_node    *attach\_file\_to\_node*

**Description**

Attach a file to an existing resource node

**Usage**

```
attach_file_to_node(nid, url = get_url(), credentials = list(cookie =
  dkanr::get_cookie(), token = dkanr::get_token()), file_path,
  attach = 1)
```

**Arguments**

nid	character: DKAN resource node ID
url	character: The DKAN site URL
credentials	Optional list parameter. Default values are Cookie and Token generated by dkan_setup()
file_path	character: path to file on local machine
attach	numeric: Setting the attach parameter to 0 ensures that the file will replace any existing file on the resource. Setting it to 1 will result in a rejected request if the resource already has an attached file (but it will work if the resource file upload field is empty).

---

create_node	<i>create_node</i>
-------------	--------------------

---

**Description**

Create a DKAN node

**Usage**

```
create_node(url = get_url(), body, credentials = list(cookie =
  dkanr::get_cookie(), token = dkanr::get_token()), as = "json", ...)
```

**Arguments**

url	character: The DKAN site URL
body	json: A JSON object to be passed to the body of the request
credentials	Optional list parameter. Default values are Cookie and Token generated by dkan_setup()
as	character: Output format. Options are: 'json' or 'list'
...	Other optional parameters passed to the underlying GET request

**Value**

dkan\_node object

**Examples**

```
## Not run:
create_node(url = "http://demo.getdkan.com", body = {"title": "TEST DATASET"})

## End(Not run)
```

---

delete_node	<i>delete_node</i>
-------------	--------------------

---

**Description**

Delete an existing DKAN node

**Usage**

```
delete_node(nid, url = get_url(), credentials = list(cookie =
  dkanr::get_cookie(), token = dkanr::get_token()), as = "json", ...)
```

**Arguments**

nid	character: DKAN node ID
url	character: The DKAN site URL
credentials	Optional list parameter. Default values are Cookie and Token generated by dkan_setup()
as	character: Output format. Options are: 'json' or 'list'
...	Other optional parameters passed to the underlying GET request

**Value**

character or dkan\_node object

**Examples**

```
## Not run:
delete_node(nid = 1, url = "http://demo.getdkan.com")

## End(Not run)
```

---

dkanr\_logout

*dkanr\_logout*


---

**Description**

Close connection with DKAN API initiated by dkanr\_setup()

**Usage**

```
dkanr_logout(url = get_url(), credentials = list(cookie =
  dkanr::get_cookie(), token = dkanr::get_token()))
```

**Arguments**

url	Character: API root URL. Default value is the url returned by get_url()
credentials	Optional list parameter. Default values are Cookie and Token generated by dkan_setup()

**Value**

json

---

dkanr_settings	<i>Get or set dkanr DKAN settings</i>
----------------	---------------------------------------

---

**Description**

Get or set dkanr DKAN settings

**Usage**

```
dkanr_settings()
```

```
get_url()
```

```
get_cookie()
```

```
get_token()
```

**Value**

dkanr\_settings prints your base url, API cookie and token (if used) dkanr\_setup sets your production settings

**See Also**

[dkanr\\_setup](#), [get\\_url](#)

**Examples**

```
dkanr_settings()
```

---

dkanr_setup	<i>Configure default DKAN settings</i>
-------------	--

---

**Description**

Configure default DKAN settings

**Usage**

```
dkanr_setup(url = "http://demo.getdkan.com", username = NULL,  
password = NULL)
```

**Arguments**

url	A DKAN URL (optional), default: "http://demo.getdkan.com"
username	Username (optional, character)
password	Password (optional, character)

**Details**

`dkanr_setup` sets DKAN connection details. `dkanr` functions default to use the default URL and API key unless specified explicitly.

**Examples**

```
## Not run:
DKAN users without admin/editor privileges could run:
dkanr_setup(url = "http://demo.getdkan.com")

Privileged DKAN editor/admin users can run:
dkanr_setup(url = "http://demo.getdkan.com", username = "your-username", password = "your-password")

Not specifying the default DKAN URL will reset the DKAN URL to its default
"http://demo.getdkan.com":
dkanr_setup()

## End(Not run)
```

---

<code>ds_is_available</code>	<i>ds_is_available</i>
------------------------------	------------------------

---

**Description**

Returns TRUE if a data file has been imported in the DKAN datastore, returns FALSE otherwise

**Usage**

```
ds_is_available(metadata)
```

**Arguments**

`metadata` list: Resource level metadata

**Value**

logical vector

**Examples**

```
## Not run:
metadata <- retrieve_node(nid = '140366', as = 'list')
ds_is_available(metadata)

## End(Not run)
```

---

ds_search	<i>ds_search</i>
-----------	------------------

---

## Description

Search csv resource file stored in the datastore.

## Usage

```
ds_search(resource_id, fields = NULL, filters = NULL, limit = 100,
  offset = 0, sort = NULL, q = NULL, url = get_url(),
  credentials = list(cookie = dkanr::get_cookie(), token =
  dkanr::get_token()))
```

## Arguments

resource_id	character: DKAN resource ID
fields	character vector: fields to be returned by the request
filters	named character vector: filters to be applied when retrieving records
limit	numeric: Number of records to retrieve
offset	numeric: Number of results to skip in the beginning
sort	named character vector: field to be sorted by
q	character: full text query filter
url	character: The DKAN site URL
credentials	Optional list parameter. Default values are Cookie and Token generated by dkan_setup()

## Value

data.frame or character

## Examples

```
## Not run:
ds_search(resource_id = '10c578a6-63c4-40bd-a55d-0c27bf276283',
  fields = c('Country', 'City', 'Region', 'Population'),
  filters = list('Country'=c('co', 'my'), 'Region'=c('04', '09', '22')),
  limit = 20,
  offset = 0,
  sort = c('Country'='asc'),
  q = "puertica",
  url = get_url(),
  as = 'df')

## End(Not run)
```

---

ds_search_all	<i>ds_search_all</i>
---------------	----------------------

---

### Description

Search csv resource file stored in the datastore.

### Usage

```
ds_search_all(resource_id, fields = NULL, filters = NULL,
              num_records = NULL, offset = 0, sort = NULL, q = NULL,
              url = get_url(), credentials = list(cookie = dkanr::get_cookie(),
              token = dkanr::get_token()), as = "df")
```

### Arguments

resource_id	character: DKAN resource ID
fields	character vector: fields to be returned by the request
filters	named character vector: filters to be applied when retrieving records
num_records	numeric: Number of records to retrieve
offset	numeric: Number of results to skip in the beginning
sort	named character vector: field to be sorted by and the order of sorting
q	character: full text query filter
url	character: The DKAN site URL
credentials	Optional list parameter. Default values are Cookie and Token generated by dkan_setup()
as	character: output format - json, list or data frame

### Value

character, list, or data frame

### Examples

```
## Not run:
ds_search_all(resource_id = '10c578a6-63c4-40bd-a55d-0c27bf276283',
              fields = c('Country', 'City', 'Region', 'Population'),
              filters = list('Country'=c('co', 'my'), 'Region'=c('04', '09', '22')),
              num_records = 20,
              offset = 0,
              limit = 100,
              sort = c('Country'='asc'),
              q = "puertica",
              url = get_url(),
              as = 'df')

## End(Not run)
```



---

err_handler	<i>err_handler</i>
-------------	--------------------

---

**Description**

Helper function to capture error messages returned by the API

**Usage**

```
err_handler(x)
```

**Arguments**

x	response: The API response
---	----------------------------

---

filters_to_text_query	<i>filters_to_text_query</i>
-----------------------	------------------------------

---

**Description**

Helper function to build API query string

**Usage**

```
filters_to_text_query(filters, text)
```

**Arguments**

filters	named vector: filters to be applied
text	vector: The key word used for filtering

---

get_resource_nids	<i>get_resource_nids</i>
-------------------	--------------------------

---

**Description**

Returns all the resource node ids attached to a specific node id

**Usage**

```
get_resource_nids(metadata)
```

**Arguments**

metadata	list: Dataset level metadata
----------	------------------------------

**Value**

character vector

**Examples**

```
## Not run:  
metadata <- retrieve_node(nid = '140177', as = 'list')  
get_resource_nids(metadata)  
  
## End(Not run)
```

---

*get\_resource\_url*      *get\_resource\_url*

---

**Description**

Returns the download link or external page URL for a specific resource

**Usage**

```
get_resource_url(metadata)
```

**Arguments**

metadata      list: Resource level metadata

**Value**

character vector

**Examples**

```
## Not run:  
metadata <- retrieve_node(nid = '140366', as = 'list')  
get_resource_url(metadata)  
  
## End(Not run)
```

---

is.dkan\_list                    *dkan\_list class helpers*

---

**Description**

dkan\_list class helpers

**Usage**

is.dkan\_list(x)

**Arguments**

x                    Variety of things, character, list, or dkan\_list class object  
as.dkan\_list <- function(x, ...) UseMethod("as.dkan\_list")  
#' @export as.dkan\_list.character <- function(x, ...) retrieve\_node(x, ...)  
#' @export as.dkan\_list.dkan\_list <- function(x, ...) x  
#' @export as.dkan\_list.list <- function(x, ...) structure(x, class = "dkan\_list")

---

is.dkan\_node                    *dkan\_node class helpers*

---

**Description**

dkan\_node class helpers

**Usage**

is.dkan\_node(x)

**Arguments**

x                    Variety of things, character, list, or dkan\_node class object # create item class  
from only an item ID as.dkan\_node("22")  
# gives back itself (x <- as.dkan\_node("22")) as.dkan\_node(x) as.dkan\_node <-  
function(x, ...) UseMethod("as.dkan\_node")  
#' @export as.dkan\_node.character <- function(x, ...) retrieve\_node(x, ...)  
#' @export as.dkan\_node.dkan\_node <- function(x, ...) x  
#' @export as.dkan\_node.list <- function(x, ...) structure(x, class = "dkan\_node")

---

list_nodes	<i>list_nodes</i>
------------	-------------------

---

### Description

Returns list of nodes

### Usage

```
list_nodes(url = get_url(), credentials = list(cookie =
  dkanr::get_cookie(), token = dkanr::get_token()), fields = NULL,
  filters = NULL, pagesize = NULL, page = NULL, as = "json", ...)
```

### Arguments

url	character: The DKAN site URL
credentials	Optional list parameter. Default values are Cookie and Token generated by dkan_setup()
fields	character vector: fields to be returned by the request
filters	named character vector: filters to be applied to the search
pagesize	numeric: Number of records to get per page (max = 20).
page	numeric: The zero-based index of the page to get, defaults to 0.
as	character: Output format. Options are: 'json', 'list', or 'df' (data frame)
...	Other optional parameters to be passed to the underlying GET request

### Value

json, dkan\_node (list), or data frame

### Examples

```
## Not run:
list_nodes(url = "http://demo.getdkan.com")
list_nodes(url = "http://demo.getdkan.com",
  fields = c('nid', 'type', 'uri'),
  filters = c(type='resource'))

## End(Not run)
```

---

list_nodes_all	<i>list_nodes_all</i>
----------------	-----------------------

---

## Description

Returns list of nodes

## Usage

```
list_nodes_all(url = get_url(), credentials = list(cookie =  
  dkanr::get_cookie(), token = dkanr::get_token()), fields = NULL,  
  filters = NULL, as = "df", ...)
```

## Arguments

url	character: The DKAN site URL
credentials	Optional list parameter. Default values are Cookie and Token generated by dkan_setup()
fields	character vector: fields to be returned by the request
filters	named character vector: filters to be applied to the search
as	character: Output format. Options are: 'json', 'list' or 'df' (data frame)
...	Other optional parameters to be passed to the underlying GET request

## Value

json, dkan\_node (list), or data frame

## Examples

```
## Not run:  
list_nodes_all(url = "http://demo.getdkan.com",  
  fields = c('nid', 'type'),  
  filters = c(type = 'resource'))  
  
## End(Not run)
```

---

retrieve_node	<i>retrieve_node</i>
---------------	----------------------

---

**Description**

Returns metadata associated with an existing DKAN Node ID.

**Usage**

```
retrieve_node(nid, url = get_url(), credentials = list(cookie =
  dkanr::get_cookie(), token = dkanr::get_token()), as = "json", ...)
```

**Arguments**

nid	character: DKAN node ID
url	character: The DKAN site URL
credentials	Optional list parameter. Default values are Cookie and Token generated by dkan_setup()
as	character: Output format. Options are: 'json' or 'list'
...	Other optional parameters passed to the underlying GET request

**Value**

dkan\_node object

**Examples**

```
## Not run:
retrieve_node(nid = 1, url = "http://demo.getdkan.com")

## End(Not run)
```

---

update_node	<i>update_node</i>
-------------	--------------------

---

**Description**

Update an existing DKAN node

**Usage**

```
update_node(nid, url = get_url(), body, credentials = list(cookie =
  dkanr::get_cookie(), token = dkanr::get_token()), as = "json", ...)
```

**Arguments**

<code>nid</code>	character: DKAN node ID
<code>url</code>	character: The DKAN site URL
<code>body</code>	json: A JSON object to be passed to the body of the request
<code>credentials</code>	Optional list parameter. Default values are Cookie and Token generated by <code>dkan_setup()</code>
<code>as</code>	character: Output format. Options are: 'json' or 'list'
<code>...</code>	Other optional parameters passed to the underlying GET request

**Value**

`dkan_node` object

**Examples**

```
## Not run:  
update_node(nid = 1, url = "http://demo.getdkan.com")  
  
## End(Not run)
```

# Index

[attach\\_file\\_to\\_node](#), 2

[create\\_node](#), 3

[delete\\_node](#), 3

[dkanr\\_logout](#), 4

[dkanr\\_settings](#), 5

[dkanr\\_setup](#), 5, 5

[ds\\_is\\_available](#), 6

[ds\\_search](#), 7

[ds\\_search\\_all](#), 8

[err\\_handler](#), 9

[filters\\_to\\_text\\_query](#), 9

[get\\_cookie \(dkanr\\_settings\)](#), 5

[get\\_resource\\_nids](#), 9

[get\\_resource\\_url](#), 10

[get\\_token \(dkanr\\_settings\)](#), 5

[get\\_url](#), 5

[get\\_url \(dkanr\\_settings\)](#), 5

[is.dkan\\_list](#), 11

[is.dkan\\_node](#), 11

[list\\_nodes](#), 12

[list\\_nodes\\_all](#), 13

[retrieve\\_node](#), 14

[update\\_node](#), 14