Package ‘robotoolbox’

June 9, 2023

Title  Client for the 'KoboToolbox' API
Version  1.3.2
Description  Suite of utilities for accessing and manipulating
data from the 'KoboToolbox' API. 'KoboToolbox' is a robust
platform designed for field data collection in various disciplines.
This package aims to simplify the process of fetching and handling
data from the API. Detailed documentation for the 'KoboToolbox' API
can be found at <https://support.kobotoolbox.org/api.html>.

Depends  R (>= 4.1)
License  MIT + file LICENSE
Encoding  UTF-8
Language  en_US

URL  https://dickoa.gitlab.io/robotoolbox,
     https://gitlab.com/dickoa/robotoolbox

BugReports  https://gitlab.com/dickoa/robotoolbox/-/issues

Imports  crul (>= 1.4.0), RcppSimdJson (>= 0.1.6), data.table (>=
1.14.2), dplyr (>= 1.1.2), tidyr (>= 1.3.0), purrr (>= 1.0.1),
rlang (>= 1.0.0), tidyselect (>= 1.2.0), tibble (>= 3.2.1),
stringi (>= 1.7.6), glue (>= 1.6.0), dm (>= 1.0.5), labelled
(>= 2.11.0), readr (>= 2.1.0), cli (>= 3.6.1)

Suggests  roxygen2 (>= 7.2.3), devtools (>= 2.4.3), vcr (>= 1.2.0),
knitr (>= 1.37), testthat (>= 3.1.1), covr (>= 3.6.2),
markdown (>= 2.21), DiagrammeR (>= 1.0.9), DiagrammeRsvg (>=
0.1), sf (>= 1.0.9), mapview (>= 2.11.0)

VignetteBuilder  knitr
RoxygenNote  7.2.3
LazyData  true
Config/Needs/website  vidonne/unhcrtemplate
X-schema.org-applicationCategory  Data Access
X-schema.org-keywords  open-data, kobotoolbox, odk, kpi, data, dataset
Examples of KoboToolbox assets and list of assets

Description

Examples of KoboToolbox assets and list of assets.

Usage

asset_list

asset_ml

asset_rg

asset_spatial

asset_sm_label

asset_audit
data_ml_en

Format

asset_list: a data.frame of 28 rows and 7 columns with a list of API assets
asset_ml: A kobo_asset object on a survey using multiple languages.
asset_rg: A kobo_asset object on a survey using repeat groups.
asset_spatial: A kobo_asset object on a survey showcasing gps questions.
asset_sm_label: A kobo_asset object to showcase select multiple labels.
asset_audit: A kobo_asset object on a survey with audit logging enabled.

data_ml_en

Examples of KoboToolbox submissions data

Description

Examples of KoboToolbox submissions data.

Usage

data_ml_en
data_ml_fr
data_ml_ar
data_ml_default
data_ml_vlabel
data_rg
data_spatial
data_sm
data_sm_label
data_audit

Format

data_ml: A data.frame with submissions from asset_ml in English.
data_ml_fr: A data.frame with submissions from asset_ml in French.
data_ml_ar: A data.frame with submissions from asset_ml in Arabic.
data_ml_default: A data.frame with submissions from asset_ml with the default language.
data_ml_vlabel: A data.frame with submissions from asset_ml using variable labels as column names.
data_rg: A dm object with submissions from asset_rg
data.spatial: A data.frame with submissions from the asset.spatial KoboToolbox API asset.
data.sm: A data.frame with submissions from asset.sm_label with no labels for the select_multiple question.
data.sm_label: A data.frame with submissions from asset.sm_label with labels for the select_multiple question.
data_audit: A data.frame with submissions from asset_audit.

---

**kobo_asset**

*Get a specific KoboToolbox API asset from a unique identifier*

**Description**

Get a specific KoboToolbox API asset from a unique identifier

**Usage**

```r
kobo_asset(x)
```

**Arguments**

- **x**
  
  the unique identifier of a specific asset (character) or a kobo_asset object.

**Value**

A kobo_asset object. It contains all the information about the KoboToolbox API asset associated to the unique identifier.

**Examples**

```r
## Not run:
# replace by your own url and token
kobo_setup(url = "https://kf.kobotoolbox.org", token = "abcde")
# use a valid uid
uid <- "a9cwE0cbWqWzA5h2kJRqW"
asset <- kobo_asset(uid)
asset

## End(Not run)
```
**kobo_asset_list**

---

### kobo_asset_list

List all available KoboToolbox API assets

### Description

List all available KoboToolbox API assets and their metadata.

### Usage

```r
kobo_asset_list(limit = 100L)
```

### Arguments

- **limit**: integer, the number of API assets to display per page. Default to 100.

### Value

A `data.frame` containing the list of all your KoboToolbox API assets and the following metadata:

- `uid`: the asset unique identifier
- `name`: the name of the asset
- `asset_type`: the type of asset (block, survey, question, or template)
- `owner_username`: the user account of the owner of the asset
- `date_create`: when the asset was created
- `date_modified`: when the asset was last modified
- `deployed`: whether or not the asset is currently deployed
- `submissions`: the number of submissions for the asset (survey)

### Examples

```r
## Not run:
kobo_setup()
asset_list <- kobo_asset_list(limit = 10L)
asset_list

## End(Not run)
```
**kobo_asset_version**

*Get a specific KoboToolbox API asset version from an asset unique identifier*

**Description**

Get a specific KoboToolbox Asset version from an asset unique identifier or kobo_asset object.

**Usage**

```r
kobo_asset_version(x, version)
```

**Arguments**

- `x`  
  the unique identifier of a specific asset (character) or a kobo_asset object.
- `version`  
  character, the unique identifier of the version of the asset

**Value**

A kobo_asset_version object

**Examples**

```r
## Not run:
kobo_setup()
uid <- "a9cwE0cbWzA5hzaRkUwi"
asset <- kobo_asset(uid)
asset_version_list <- kobo_asset_version_list(asset)
kobo_asset_version(asset, asset_version_list$uid[1])
## End(Not run)
```

---

**kobo_asset_version_list**

*List all available versions of a KoboToolbox API asset*

**Description**

List all available versions of a KoboToolbox API asset and their metadata.

**Usage**

```r
kobo_asset_version_list(x)
```
**kobo_audit**

Arguments

- `x` the uid or kobo_asset object.

Value

A data.frame containing the list of all the versions of a given KoboToolbox API asset with the following metadata:

- `uid` the asset version unique identifier.
- `url` the URL of the asset version.
- `deployed` whether or not the asset version is deployed.
- `date_modified` when the asset version was last modified.

a data.frame

Examples

```r
## Not run:
kobo_setup() # setup using your url and token
uid <- "a9cwEQcbWqWzA5hzkjRtU" # pick a valid uid
asset <- kobo_asset(uid)
kobo_asset_version_list(asset)
## End(Not run)
```
A data.frame. It contains survey paradata from audit logs. The following columns are available:

- **_id** This column is generated by robotoolbox, allowing you to do a mapping of the _id of the submissions in kobo_data.
- **event** the action that took place. The different event types include: form start, form exit, question, group questions, end screen, and device or metadata audit.
- **node** the name of the question or group related to the event.
- **name** This column is appended by robotoolbox to match the name of the question in the audit and the data from kobo_data.
- **start** the timestamp when the event started.
- **end** the timestamp when the event ended.
- **latitude** the latitude of the device when the event occurred.
- **longitude** the longitude of the device when the event occurred.
- **accuracy** the GPS accuracy of the location data.
- **old-value** the previous value of the question before it was changed in this event.
- **new-value** the new value of the question after it was changed in this event.
- **user** the username of the data collector.
- **change-reason** the reason before they save changes to a form.

**Examples**

```r
## Not run:
kobo_setup()
uid <- "a9cwEGcbWgZdASqkJRUMi"
asset <- kobo_asset(uid)
audit <- kobo_audit(asset)

if (require(dplyr)) {
  library(dplyr)
glimpse(audit)
}

## End(Not run)
```

---

**kobo_data**

*Get all submissions from a KoboToolbox API asset*

**Description**

Get all submissions from a KoboToolbox API asset through a kobo_asset or asset unique identifier.
Usage

kobo_data(
  x,
  lang,
  all_versions,
  colnames_label,
  select_multiple_label,
  progress,
  paginate,
  page_size
)

kobo_submissions(
  x,
  lang,
  all_versions,
  colnames_label,
  select_multiple_label,
  progress,
  paginate,
  page_size
)

## S3 method for class 'kobo_asset'
kobo_submissions(
  x,
  lang = NULL,
  all_versions = TRUE,
  colnames_label = FALSE,
  select_multiple_label = FALSE,
  progress = FALSE,
  paginate = FALSE,
  page_size = NULL
)

## S3 method for class 'character'
kobo_submissions(
  x,
  lang = NULL,
  all_versions = TRUE,
  colnames_label = FALSE,
  select_multiple_label = FALSE,
  progress = FALSE,
  paginate = FALSE,
  page_size = NULL
)

## Default S3 method:
kobo_submissions(
  x,
  lang = NULL,
  all_versions = TRUE,
  colnames_label = FALSE,
  select_multiple_label = FALSE,
  progress = FALSE,
  paginate = FALSE,
  page_size = NULL
)

Arguments

x the asset uid or the kobo_asset object.
lang character, form language used for the variable and value labels.
all_versions logical, whether or not to include submissions from all form versions. Default to TRUE. If FALSE, it uses the data from the latest version of the form.
colnames_label logical, whether or not to use variable labels in lieu of column names based on form question names. Default to FALSE.
select_multiple_label logical, whether or not to replace select_multiple columns values by labels. Default to FALSE.
progress logical, whether or not you want to see the progress via message. Default to FALSE.
paginate logical, split submissions by page_size. Default to FALSE.
page_size integer, number of submissions per page.

Details

kobo_data is the main function of robotoolbox, it is used pull submissions from your Kobotoolbox survey. The main result is a data.frame for regular form and you have a dm for a form with repeating groups of questions.

Value

A data.frame or A dm object if you have a repeating group of questions. It contains the responses from the Kobotoolbox survey.

Examples

## Not run:
# Use your own URL and token
kobo_setup(url = "https://kf.kobotoolbox.org/",
  token = "9et1814c285w094f6v9bd629df47a1a0e81x53a0")
# Use your own unique identifier
uid <- "a9cwE9QcbWqA5hzkjJRVW"
asset <- kobo_asset(uid)
subs <- kobo_data(asset)
if (require(dplyr)) {
    library(dplyr)
    glimpse(subs)
}

## End(Not run)

---

**kobo_form**

*Get a KoboToolbox survey form*

**Description**

Get a KoboToolbox survey form from a kobo_asset or an asset unique identifier.

**Usage**

```r
kobo_form(x, version)
```

**Arguments**

- **x** the unique identifier of a specific asset (character) or a kobo_asset object.
- **version** character, the unique identifier of the version of the asset.

**Value**

A data.frame with the following columns:

- name the name of the survey questions
- list_name the name of list of code used for values and labels
- type the type of KoboToolbox survey questions
- label the label of the questions
- lang the languages used in the survey
- version the survey version unique identifier
- choices a list column with the choices values and labels
- kuid the unique identifier of the question
- qpath and xpath the path of the question in JSON/XML

You can also have other columns such as relevant, calculation, etc. depending on how you structure for survey form.
Examples

```r
## Not run:
# Use your own URL and token
kobo_setup(url = "https://kf.kobotoolbox.org/",
               token = "9et1814c285w094f6v9bd629df47a1a0e81x53a0")
# Use your own API asset identifier
uid <- "a9cwEQcbWqWzA5hzkjRUWi"
asset <- kobo_asset(uid)
form <- kobo_form(asset)

## End(Not run)
```

kobo_lang

Get the languages used in a KoboToolbox survey form

Description

Get the languages used in a KoboToolbox survey form from a kobo_asset or asset unique identifier.

Usage

```r
kobo_lang(x)
```

Arguments

- `x` the unique identifier of a specific asset (character) or a kobo_asset object.

Value

A vector of character. The languages used in the form, it returns "Labels" when no language is set.

Examples

```r
## Not run:
kobo_setup()
uid <- "a9cwEQcbWqWzA5hzkjRUWi"
asset <- kobo_asset(uid)
lang <- kobo_lang(asset)
lang

## End(Not run)
```
**kobo_settings**

*Get robotoolbox settings*

**Description**

Print the KoboToolbox server URL and API token currently in use.

**Usage**

```r
kobo_settings()
```

**Value**

A list with information about your KoboToolbox server URL and token.

**Examples**

```r
## Not run:
kobo_settings()
```

**End(Not run)**

---

**kobo_setup**

*Set robotoolbox settings*

**Description**

Set the KoboToolbox server URL, API token and return invisibly a `kobo_settings` object.

**Usage**

```r
kobo_setup(  
  url = Sys.getenv("KOBOTOOLBOX_URL", ""),  
  token = Sys.getenv("KOBOTOOLBOX_TOKEN", ""))
```

**Arguments**

- `url` character, the base URL of the KoboToolbox server.
- `token` character, the API token.

**Value**

A `kobo_settings` object priting the server URL and the API token.
Examples

```r
## Not run:
# use your own URL and token
kobo_setup(url = "https://kf.kobotoolbox.org/",
    token = "9et1814c285w094f6v9bd629df47a1a0e81x53a0")
kobo_settings()

## End(Not run)
```

---

**kobo_token**

*Get your KoboToolbox API token*

Description

Get your KoboToolbox API token from your username and password.

Usage

```r
kobo_token(username = NULL, password = NULL, url = NULL, overwrite = FALSE)
```

Arguments

- `username`: character, KoboToolbox account username.
- `password`: character, KoboToolbox account password.
- `url`: character, KoboToolbox server URL.
- `overwrite`: logical, if TRUE, it overwrites the existing token. Default to FALSE.

Value

A character, the KoboToolbox API token. It also stores, as a side effect, the URL and token as the environment variables `KOBOTOOLBOX_URL` and `KOBOTOOLBOX_TOKEN` respectively.

Examples

```r
## Not run:
# use your own KoboToolbox URL, username and password
if (require(askpass)) {
  token <- kobo_setup(username = "cool_user_name",
    password = askpass::askpass(),
    url = "https://kf.kobotoolbox.org/")
  token
}

## End(Not run)
```
Index

* datasets
  asset_list, 2
  data_ml_en, 3
asset_audtt, 4
asset_audit (asset_list), 2
asset_list, 2
asset_ml, 3, 4
asset_ml (asset_list), 2
asset_rg, 4
asset_rg (asset_list), 2
asset_sm_label, 4
asset_sm_label (asset_list), 2
asset_spatial, 4
asset_spatial (asset_list), 2
data_audtt (data_ml_en), 3
data_ml_ar (data_ml_en), 3
data_ml_default (data_ml_en), 3
data_ml_en, 3
data_ml_fr (data_ml_en), 3
data_ml_vlabel (data_ml_en), 3
data_rg (data_ml_en), 3
data_sm (data_ml_en), 3
data_sm_label (data_ml_en), 3
data_spatial (data_ml_en), 3
kobo_asset, 4
kobo_asset_list, 5
kobo_asset_version, 6
kobo_asset_version_list, 6
kobo_audtt, 7
kobo_data, 8, 10
kobo_form, 11
kobo_lang, 12
kobo_settings, 13
kobo_setup, 13
kobo_submissions (kobo_data), 8
kobo_token, 14