

Package ‘datasetjson’

January 9, 2024

Type Package

Title Read and Write CDISC Dataset JSON Files

Version 0.2.0

Description Read, construct and write CDISC (Clinical Data Interchange Standards Consortium) Dataset JSON (JavaScript Object Notation) files, while validating per the Dataset JSON schema file, as described in CDISC (2023) <<https://www.cdisc.org/dataset-json>>.

URL <https://github.com/atorus-research/datasetjson>

BugReports <https://github.com/atorus-research/datasetjson/issues>

Encoding UTF-8

Language en-US

License Apache License (>= 2)

LazyData true

RoxygenNote 7.2.3

Depends R (>= 3.5)

Imports jsonlite (>= 1.8.0), jsonvalidate (>= 1.3.1)

Suggests testthat (>= 2.1.0), knitr, haven, rmarkdown, withr

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Author Mike Stackhouse [aut, cre] (<<https://orcid.org/0000-0001-6030-723X>>),
Nicholas Masel [aut]

Maintainer Mike Stackhouse <mike.stackhouse@atorusresearch.com>

Repository CRAN

Date/Publication 2024-01-09 21:33:01 UTC

R topics documented:

dataset_json	2
dataset_metadata	4
data_metadata	5
file_metadata	5
iris_items	6
read_dataset_json	7
sas_datetime_formats	8
sas_date_formats	8
sas_time_formats	8
schema_1_0_0	9
set_source_system	9
set_study_oid	10
validate_dataset_json	11
write_dataset_json	12
Index	13

dataset_json	<i>Create a Dataset JSON Object</i>
--------------	-------------------------------------

Description

Create the base object used to write a Dataset JSON file.

Usage

```
dataset_json(
  .data,
  item_id,
  name,
  label,
  items,
  dataset_meta,
  version = "1.0.0",
  data_type = c("clinicalData", "referenceData"),
  file_meta = file_metadata(),
  data_meta = data_metadata()
)
```

Arguments

.data	Input data to contain within the Dataset JSON file. Written to the itemData parameter.
item_id	ID used to label dataset with the itemGroupData parameter. Defined as "Object of Datasets. Key value is a unique identifier for Dataset, corresponding to ItemGroupDef/@OID in Define-XML."

name	Dataset name
label	Dataset Label
items	Variable metadata
dataset_meta	A dataset_metadata object holding pre-specified dataset metadata.
version	Version of Dataset JSON schema to follow.
data_type	Type of data being written. clinicalData for subject level data, and referenceData for non-subject level data (i.e. TDMs, Associated Persons)
file_meta	A file_metadata object holding pre-specified file metadata
data_meta	A data_metadata object holding pre-specified data metadata

Value

dataset_json object pertaining to the specific Dataset JSON version specific

Examples

```
# Create a basic object
ds_json <- dataset_json(iris, "IG.IRIS", "IRIS", "Iris", iris_items)

# Attach attributes directly
ds_json_updated <- set_data_type(ds_json, "referenceData")
ds_json_updated <- set_file_oid(ds_json_updated, "/some/path")
ds_json_updated <- set_metadata_ref(ds_json_updated, "some/define.xml")
ds_json_updated <- set_metadata_version(ds_json_updated, "MDV.MSGv2.0.SDTMIG.3.3.SDTM.1.7")
ds_json_updated <- set_originator(ds_json_updated, "Some Org")
ds_json_updated <- set_source_system(ds_json_updated, "source system", "1.0")
ds_json_updated <- set_study_oid(ds_json_updated, "SOMESTUDY")

# Create independent objects for metadata sections first
file_meta <- file_metadata(
  originator = "Some Org",
  sys = "source system",
  sys_version = "1.0"
)

data_meta <- data_metadata(
  study = "SOMESTUDY",
  metadata_version = "MDV.MSGv2.0.SDTMIG.3.3.SDTM.1.7",
  metadata_ref = "some/define.xml"
)

dataset_meta <- dataset_metadata(
  item_id = "IG.IRIS",
  name = "IRIS",
  label = "Iris",
  items = iris_items
)

ds_json_from_meta <- dataset_json(
  iris,
```

```
dataset_meta = dataset_meta,  
file_meta = file_meta,  
data_meta = data_meta  
)
```

dataset_metadata	<i>Generate an individual element that fills the itemGroupData field</i>
------------------	--

Description

Generate an individual element that fills the itemGroupData field

Usage

```
dataset_metadata(item_id, name, label, items, .data)
```

Arguments

item_id	Data Object ID for item in Dataset JSON object, corresponding to ItemGroupDef/@OID in Define-XML.
name	Dataset name
label	Dataset Label
items	Variable metadata
.data	Dataframe to be written to Dataset JSON file

Value

dataset_metadata object

Examples

```
dataset_meta <- dataset_metadata(  
  item_id = "IG.IRIS",  
  name = "IRIS",  
  label = "Iris",  
  items = iris_items  
)
```

data_metadata	<i>Create the data metadata container for a Dataset JSON object</i>
---------------	---

Description

Create the data metadata container for a Dataset JSON object

Usage

```
data_metadata(study = NULL, metadata_version = NULL, metadata_ref = NULL)
```

Arguments

study	Study OID value
metadata_version	Metadata version OID value
metadata_ref	Metadata reference (i.e. path to Define.xml)

Value

data_metadata object

Examples

```
# Create object directly
data_meta <- data_metadata(
  study = "SOMESTUDY",
  metadata_version = "MDV.MSGv2.0.SDTMIG.3.3.SDTM.1.7",
  metadata_ref = "some/define.xml"
)

# Use setter functions
data_meta <- data_metadata()
data_meta_updated <- set_metadata_ref(data_meta, "some/define.xml")
data_meta_updated <- set_metadata_version(data_meta_updated, "MDV.MSGv2.0.SDTMIG.3.3.SDTM.1.7")
data_meta_updated <- set_study_oid(data_meta_updated, "SOMESTUDY")
```

file_metadata	<i>Create a file metadata object</i>
---------------	--------------------------------------

Description

Create a file metadata object

Usage

```
file_metadata(
  originator = NULL,
  sys = NULL,
  sys_version = NULL,
  file_oid = NULL,
  version = "1.0.0"
)
```

Arguments

originator	originator parameter, defined as "The organization that generated the Dataset-JSON file."
sys	sourceSystem parameter, defined as "The computer system or database management system that is the source of the information in this file."
sys_version	sourceSystemVersion, defined as "The version of the sourceSystem"
file_oid	fileOID parameter, defined as "A unique identifier for this file."
version	Dataset JSON schema version being used

Value

file_metadata object

Examples

```
# Create using parameters
file_meta <- file_metadata(
  originator = "Some Org",
  sys = "source system",
  sys_version = "1.0"
)

# Set parameters after
file_meta <- file_metadata()

file_meta_updated <- set_file_oid(file_meta, "/some/path")
file_meta_updated <- set_originator(file_meta_updated, "Some Org")
file_meta_updated <- set_source_system(file_meta_updated, "source system", "1.0")
```

iris_items

Example Variable Metadata for Iris

Description

Example of the necessary variable metadata included in a Dataset JSON file based on the Iris data frame.

Usage

```
iris_items
```

Format

`iris_items` **A data frame with 5 rows and 6 columns::**

OID Unique identifier for Variable. Must correspond to ItemDef/@OID in Define-XML.

name Display format supports data visualization of numeric float and date values.

label Label for Variable

type Data type for Variable

length Length for Variable

displayFormat Display format supports data visualization of numeric float and date values.

keySequence Indicates that this item is a key variable in the dataset structure. It also provides an ordering for the keys.

read_dataset_json	<i>Read a Dataset JSON to datasetjson object</i>
-------------------	--

Description

This function validates a dataset JSON file against the Dataset JSON schema, and if valid returns a datasetjson object. The Dataset JSON file can be either a file path on disk or a URL which contains the Dataset JSON file.

Usage

```
read_dataset_json(file)
```

Arguments

`file` File path or URL of a Dataset JSON file

Value

datasetjson object

Examples

```
# Read from disk
## Not run:
dat <- read_dataset_json("path/to/file.json")
dat <- dataset_json('https://www.somesite.com/file.json')

## End(Not run)

# Read from an already imported character vector
ds_json <- dataset_json(iris, "IG.IRIS", "IRIS", "Iris", iris_items)
js <- write_dataset_json(ds_json)
dat <- read_dataset_json(js)
```

sas_datetime_formats *A List of valid SAS(c) datetime formats*

Description

Valid SAS(c) datetime formats pulled from <https://documentation.sas.com/doc/en/vdmmlcdc/8.1/ds2pg/p0bz5detpfj01qn1kz2in7>

Usage

sas_datetime_formats

Format

sas_datetime_formats:
A character vector with 7 elements

sas_date_formats *A List of valid SAS(c) date formats*

Description

Valid SAS(c) date formats pulled from <https://documentation.sas.com/doc/en/vdmmlcdc/8.1/ds2pg/p0bz5detpfj01qn1kz2in7>

Usage

sas_date_formats

Format

sas_date_formats:
A character vector with 45 elements

sas_time_formats *A List of valid SAS(c) time formats*

Description

Valid SAS(c) time formats pulled from <https://documentation.sas.com/doc/en/vdmmlcdc/8.1/ds2pg/p0bz5detpfj01qn1kz2in7>

Usage

sas_time_formats

Format

sas_time_formats:
A character vector with 4 elements

schema_1_0_0	<i>Dataset JSON Schema Version 1.0.0</i>
--------------	--

Description

This object is a character vector holding the schema for Dataset JSON Version 1.0.0

Usage

```
schema_1_0_0
```

Format

```
schema_1_0_0:
  A character vector with 1 element
```

set_source_system	<i>File Metadata Setters</i>
-------------------	------------------------------

Description

Set information about the file and source system used to generate the Dataset JSON object.

Usage

```
set_source_system(x, sys, sys_version)

set_originator(x, originator)

set_file_oid(x, file_oid)

set_data_type(x, data_type = c("clinicalData", "referenceData"))
```

Arguments

x	datasetjson object
sys	sourceSystem parameter, defined as "The computer system or database management system that is the source of the information in this file."
sys_version	sourceSystemVersion, defined as "The version of the sourceSystem"
originator	originator parameter, defined as "The organization that generated the Dataset-JSON file."
file_oid	fileOID parameter, defined as "A unique identifier for this file."
data_type	Type of data being written. clinicalData for subject level data, and referenceData for non-subject level data (i.e. TDMs, Associated Persons)

Details

The fileOID parameter should be structured following description outlined in the ODM V2.0 specification. "FileOIDs should be universally unique if at all possible. One way to ensure this is to prefix every FileOID with an internet domain name owned by the creator of the ODM file or database (followed by a forward slash, "/"). For example, FileOID="BestPharmaceuticals.com/Study5894/1" might be a good way to denote the first file in a series for study 5894 from Best Pharmaceuticals."

Value

datasetjson or file_metadata object

Examples

```
file_meta <- file_metadata()

file_meta_updated <- set_file_oid(file_meta, "/some/path")
file_meta_updated <- set_originator(file_meta_updated, "Some Org")
file_meta_updated <- set_source_system(file_meta_updated, "source system", "1.0")
```

set_study_oid	<i>Set data metadata parameters</i>
---------------	-------------------------------------

Description

This set of functions

Usage

```
set_study_oid(x, study, ...)

set_metadata_version(x, metadata_version, ...)

set_metadata_ref(x, metadata_ref)
```

Arguments

x	data metadata or datasetjson object
study	Study OID value
...	Additional parameters
metadata_version	Metadata version OID value
metadata_ref	Metadata reference (i.e. path to Define.xml)

Value

A datasetjson or data_metadata object

Examples

```
data_meta <- data_metadadata()
data_meta_updated <- set_metadadata_ref(data_meta, "some/define.xml")
data_meta_updated <- set_metadadata_version(data_meta_updated, "MDV.MSGv2.0.SDTMIG.3.3.SDTM.1.7")
data_meta_updated <- set_study_oid(data_meta_updated, "SOMESTUDY")
```

validate_dataset_json *Validate a Dataset JSON file*

Description

This function calls `jsonvalidate::json_validate()` directly, with the parameters necessary to retrieve the error information of an invalid JSON file per the Dataset JSON schema.

Usage

```
validate_dataset_json(x)
```

Arguments

`x` File path or URL of a Dataset JSON file, or a character vector holding JSON text

Value

A data frame

Examples

```
## Not run:
  validate_dataset_json('path/to/file.json')
  validate_dataset_json('https://www.somesite.com/file.json')

## End(Not run)

ds_json <- dataset_json(iris, "IG.IRIS", "IRIS", "Iris", iris_items)
js <- write_dataset_json(ds_json)

validate_dataset_json(js)
```

write_dataset_json *Write out a Dataset JSON file*

Description

Write out a Dataset JSON file

Usage

```
write_dataset_json(x, file, pretty = FALSE)
```

Arguments

x	datasetjson object
file	File path to save Dataset JSON file
pretty	If TRUE, write with readable formatting

Value

NULL when file written to disk, otherwise character string

Examples

```
# Write to character object
ds_json <- dataset_json(iris, "IG.IRIS", "IRIS", "Iris", iris_items)
js <- write_dataset_json(ds_json)

# Write to disk
## Not run:
write_dataset_json(ds_json, "path/to/file.json")

## End(Not run)
```

Index

- * **Data metadata setters**

- set_study_oid, 10

- * **File Metadata Setters**

- set_source_system, 9

- * **datasets**

- iris_items, 6

- sas_date_formats, 8

- sas_datetime_formats, 8

- sas_time_formats, 8

- schema_1_0_0, 9

data_metadata, 5

dataset_json, 2

dataset_metadata, 4

file_metadata, 5

iris_items, 6

read_dataset_json, 7

sas_date_formats, 8

sas_datetime_formats, 8

sas_time_formats, 8

schema_1_0_0, 9

set_data_type (set_source_system), 9

set_file_oid (set_source_system), 9

set_metadata_ref (set_study_oid), 10

set_metadata_version (set_study_oid), 10

set_originator (set_source_system), 9

set_source_system, 9

set_study_oid, 10

validate_dataset_json, 11

write_dataset_json, 12