

Package ‘ENCODEExplorer’

October 9, 2015

Name ENCODEExplorer

Type Package

Title A compilation of ENCODE metadata

Version 1.0.6

Date 2015-02-25

Author Charles Joly Beauparlant <charles.joly-beauparlant@crchul.ulaval.ca>, Audrey Lemacon <lemacon.audrey@ulaval.ca> and Arnaud Droit <arnaud.droit@crchudequebec.ulaval.ca >

Author@R c(person(`Charles`,`Joly Beauparlant`, email = `charles.joly-beauparlant@crchul.ulaval.ca`, role = `aut,cre`), person(`Audrey`,`Lemacon`, email = `lemacon.audrey@ulaval.ca`, role = `aut`), person(`Arnaud`,`Droit`, email = `arnaud.droit@crchudequebec.ulaval.ca`, role = `aut`), person(`Astrid-Louise`,`Deschenes`, email = `Astrid-Louise.Deschenes@crchudequebec.ulaval.ca`, role = `ctb`))

Maintainer Charles Joly Beauparlant <charles.joly-beauparlant@crchul.ulaval.ca>

Description This package allows user to quickly access ENCODE project files metadata and give access to helper functions to query the ENCODE rest api, download ENCODE datasets and save the database in SQLite format.

License Artistic-2.0

BugReports <https://github.com/CharlesJB/ENCODEExplorer/issues>

VignetteBuilder knitr

Depends R (>= 3.2)

Imports tools, jsonlite, RSQLite

Suggests RUnit,BiocGenerics,knitr, curl, http

LazyData true

biocViews Infrastructure, DataImport

NeedsCompilation no

R topics documented:

clean_table	2
downloadEncode	3
ENCODEExplorer	4
encode_df	4
export_ENCODEdb_matrix	5
extract_table	5
get_encode_types	6
get_schemas	6
prepare_ENCODEdb	7
queryEncode	7
searchEncode	8
Index	10

clean_table	<i>Clean a data.frame that was produced by extract_table</i>
-------------	--

Description

data.frames produced when converting JSON to data.frame with the fromJSON function will sometime have columns that are lists and/or columns that are data.frames.

Usage

```
clean_table(table)
```

Arguments

table The table produced by the extract_table function.

Details

This function will either remove columns that are not relevant and convert columns to a vector.

Value

a data.frame corresponding to the cleaned version of the input data.frame.

downloadEncode	<i>Download files from the Internet.</i>
----------------	--

Description

After processing to a basic search with the searchEncode function or a precise search thanks to the queryEncode function, you can proceed to the downloading of all the corresponding files.

Usage

```
downloadEncode(df = NULL, resultSet = NULL, resultOrigin = NULL,  
              format = "all", dir = ".", force = TRUE)
```

Arguments

df	list of two data.frame containing ENCODE experiment and dataset meta-data.
resultSet	the results set.
resultOrigin	name of the function used to generate the result set (searchEncode or queryEncode)
format	file format, default = all
dir	the name of the directory where the downloaded file will be saved. Default = current directory
force	Download file is it already exists and md5sums is valid? Default: TRUE.

Details

This function can be used to download a set of files by providing the results set, its origin (searchEncode or queryEncode), the file format and finally the destination directory.

Value

The downloaded file names, if download worked correctly.

Examples

```
resultSet <- queryEncode(biosample = "A549", file_format = "bam")  
## Not run:  
downloadEncode(resultSet = resultSet, dir = ".")  
  
## End(Not run)
```

ENCODEExplorer	<i>ENCODEExplorer</i>
----------------	-----------------------

Description

ENCODEExplorer

encode_df	<i>Metadata from the ENCODE database in a R object. The tables were generated with the prepare_ENCODEdb function.</i>
-----------	---

Description

Metadata from the ENCODE database in a R object. The tables were generated with the prepare_ENCODEdb function.

Usage

```
data(encode_df)
```

Format

A list of data frames

Value

A list of data frames

See Also

[get_encode_types](#) to get a list of possible types. Note that some of the types are empty tables that are not included in the database created with [prepare_ENCODEdb](#) function.

`export_ENCODEdb_matrix`

Extract essential informations from the RSQLite databse in a list of data.frames

Description

Extract essential informations from the RSQLite databse in a list of data.frames

Usage

```
export_ENCODEdb_matrix(database_filename)
```

Arguments

database_filename

The name of the file to save the database into. Default: ENCODEdb.sqlite.

Value

a list containing two elements. The first one 'experiment' is a data.frame containing essential informations for each file part of an experiment ; the second one 'dataset' is a data.frame containing essential informations for each file part of a dataset.

Examples

```
database_filename <- system.file("extdata/ENCODEdb.sqlite",
                                package = "ENCODEdb")
## Not run:
  export_ENCODEdb_matrix(database_filename = database_filename)
## End(Not run)
```

`extract_table`

Extract a data.frame corresponding to a table in ENCODE database

Description

Extract a data.frame corresponding to a table in ENCODE database

Usage

```
extract_table(type)
```

Arguments

type

The type of table to extract from ENCODE rest api.

Value

a `data.frame` corresponding to the table asked. If no match is found, returns an empty `data.frame`

get_encode_types	<i>A list of known tables from ENCODE database.</i>
------------------	---

Description

The type (table) names are extracted from the schema list from ENCODE-DCC github repository: <https://github.com/ENCODE-DCC/encoded/tree/master/src/encoded/schemas>

Usage

```
get_encode_types()
```

Details

The data is extracted using the github api: <https://developer.github.com/guides/getting-started/>

Value

a vector of character with the names of the known tables in the ENCODE database.

get_schemas	<i>Extract the schemas from ENCODE's github</i>
-------------	---

Description

The JSONs are fetched from: <https://github.com/ENCODE-DCC/encoded/tree/master/src/encoded/schemas>

Usage

```
get_schemas()
```

Details

The data is extracted using the github api: <https://developer.github.com/guides/getting-started/>

The data is then converted in `data.frame` using the `jsonlite` package.

Value

a list of JSON converted in `data.frame`.

prepare_ENCODEdb	<i>Create the RSQLite database for the tables in ENCODE</i>
------------------	---

Description

Create the RSQLite database for the tables in ENCODE

Usage

```
prepare_ENCODEdb(database_filename = "inst/extdata/ENCODEdb.sqlite",
  types = get_encode_types(), overwrite = FALSE)
```

Arguments

database_filename	The name of the file to save the database into. Default: ENCODEdb.sqlite.
types	The names of the tables to extract from ENCODE rest api.
overwrite	Should tables already present in database be overwritten? Default: FALSE.

Value

is a list with selected tables from ENCODE that were used to create the RSQLite database.

Examples

```
prepare_ENCODEdb(database_filename = "platform.sql", types = "platform")
file.remove("platform.sql")
## Not run:
  prepare_ENCODEdb("ENCODEdb.sqlite")

## End(Not run)
```

queryEncode	<i>Produce a subset of data following predefined criteria</i>
-------------	---

Description

After running the prepare_ENCODEdb function, this function will allow you to extract a subset of data encoding to the following criteria : accession, assay name, biosample, dataset accession, file accession, file format, laboratory, donor organism, target and treatment.

Usage

```
queryEncode(df = NULL, set_accession = NULL, assay = NULL,
  biosample = NULL, dataset_accession = NULL, file_accession = NULL,
  file_format = NULL, lab = NULL, organism = NULL, target = NULL,
  treatment = NULL, file_status = "released", status = "released",
  fixed = TRUE)
```

Arguments

<code>df</code>	list of two <code>data.frame</code> containing ENCODE experiment and dataset metadata
<code>set_accession</code>	character string to select the experiment or dataset accession
<code>assay</code>	character string to select the assay type
<code>biosample</code>	character string to select the biosample name
<code>dataset_accession</code>	character string to select the dataset accession
<code>file_accession</code>	character string to select the file accession
<code>file_format</code>	character string to select the file format
<code>lab</code>	character string to select the laboratory
<code>organism</code>	character string to select the donor organism
<code>target</code>	character string to select the experimental target
<code>treatment</code>	character string to select the treatment
<code>file_status</code>	character string to select the file status ("released", "revoked", "all"). Default "released"
<code>status</code>	character string to select the dataset/experiment status
<code>fixed</code>	logical. If TRUE, <code>pattern</code> is a string to be matched as it is.

Details

By default, the query can be made on an exact match term. This behaviour can be modified by setting the `fixed` argument at TRUE

Value

a list of two `data.frames` containing data about ENCODE experiments and datasets

Examples

```
queryEncode(biosample = "A549", file_format = "bam")
```

<code>searchEncode</code>	<i>Simulate a query on ENCODE website and return the result as a <code>data.frame</code></i>
---------------------------	--

Description

`data.frames` produced when converting JSON to `data.frame` with the `fromJSON` function will sometime have columns that are lists and/or columns that are `data.frames`.

Usage

```
searchEncode(searchTerm = NULL, limit = 10)
```


Arguments

<code>searchTerm</code>	a search term
<code>limit</code>	the maximum number of return entries, default 10. <code>limit = all</code> will return all the result. It can generate large results set.

Details

This function simulates a basic query on ENCODE website

Value

a `data.frame` corresponding Every object that matches the search term

Examples

```
searchEncode("ChIP-Seq+H3K4me1")
```

Index

*Topic **datasets**

encode_df, [4](#)

clean_table, [2](#)

downloadEncode, [3](#)

encode_df, [4](#)

ENCODEExplorer, [4](#)

ENCODEExplorer-package (ENCODEExplorer), [4](#)

export_ENCODEdb_matrix, [5](#)

extract_table, [5](#)

get_encode_types, [4](#), [6](#)

get_schemas, [6](#)

prepare_ENCODEdb, [4](#), [7](#)

queryEncode, [7](#)

searchEncode, [8](#)