

Package ‘OpenMindat’

June 5, 2025

Type Package

Title Quickly Retrieve Datasets from the 'Mindat' API

Version 1.0.1

Imports httr, jsonlite (>= 1.8.4), readxl (>= 1.4.3), utils, stringi,
stringr, usethis

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Maintainer Xiang Que <xiangq@uidaho.edu>

Description Provide functions for users or machines to quickly and easily re-
trieve datasets from the 'mindat.org' API (<<https://api.mindat.org/schema/redoc/>>).

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.3.1

URL <https://github.com/quexiang/OpenMindat>,
<https://quexiang.github.io/OpenMindat/>

BugReports <https://github.com/quexiang/OpenMindat/issues>

License MIT + file LICENSE

NeedsCompilation no

Author Xiang Que [aut, cre] (ORCID: <<https://orcid.org/0000-0002-5687-8627>>),
Xiaogang Ma [aut] (ORCID: <<https://orcid.org/0000-0002-9110-7369>>)

Repository CRAN

Date/Publication 2025-06-05 07:40:02 UTC

Contents

ConvertDF2JsonLD	4
ConvertDF2TTL	5
crystalclasses_symbols	6
crystalclasses_systems	6
Dana8_groups	7

Dana8_subgroups	8
geomaterials_bi_greater_than	8
geomaterials_bi_less_than	9
geomaterials_bi_range	10
geomaterials_by_groupid	10
geomaterials_cleavagetype	11
geomaterials_colour	12
geomaterials_contain_all_but_not_elems	13
geomaterials_contain_all_elems	14
geomaterials_contain_any_but_not_elems	14
geomaterials_contain_any_elems	15
geomaterials_contain_only_elems	16
geomaterials_crystal_system	17
geomaterials_dens_greater_than	18
geomaterials_dens_less_than	18
geomaterials_dens_range	19
geomaterials_diapheny	20
geomaterials_entrytype	21
geomaterials_expand	21
geomaterials_field_exists	22
geomaterials_fracturetype	23
geomaterials_hardness_gt	24
geomaterials_hardness_lt	24
geomaterials_hardness_range	25
geomaterials_ima	26
geomaterials_ima_notes	27
geomaterials_ima_status	28
geomaterials_lustretype	28
geomaterials_meteoritical_code	29
geomaterials_name	30
geomaterials_not_contain_elems	31
geomaterials_optical2v_max	31
geomaterials_optical2v_min	32
geomaterials_optical2v_range	33
geomaterials_opticalsign	34
geomaterials_opticaltype	34
geomaterials_polytypeof	35
geomaterials_ri_gt	36
geomaterials_ri_lt	37
geomaterials_ri_range	37
geomaterials_search_name	38
geomaterials_streak	39
geomaterials_synid	40
geomaterials_updated_at	40
geomaterials_varietyof	41
geomeaterials_non_utf	42
geomeaterials_ordering	43
getExtension	44

localities_list_all	44
localities_list_country	45
localities_list_description	46
localities_list_elems_exc	47
localities_list_elems_inc	47
localities_list_elems_inc_exc	48
localities_list_expand	49
localities_list_txt	50
localities_list_updated_at	50
localities_retrieve_id	51
localities_status_list	52
localities_status_retrieve	53
locality_type_retrieve	53
locality_age	54
locality_age_list	55
locality_type_list	55
Locentries_list	56
Locentries_retrieve	57
Locentries_statistics_list	57
Locentries_stat_retrieve	58
mindat_build_querystring	59
mindat_cache_delete	59
mindat_cache_empty	60
mindat_cache_get	60
mindat_cache_has	61
mindat_cache_return_or_setup	61
mindat_cache_set	62
mindat_connection	62
mindat_countries	63
mindat_country	63
mindat_crystalclasses	64
mindat_crystalclass_list	64
mindat_dana8_groups	65
mindat_dana8_subgroups	66
mindat_extract_response_body	66
mindat_geomaterial	67
mindat_geomaterial_list	68
mindat_geomaterial_search	68
mindat_geomaterial_varieties	69
mindat_get_data_from_uri	69
mindat_localities_list	70
mindat_locality	71
mindat_locality_status	71
mindat_locality_status_list	72
mindat_locality_type	73
mindat_locality_type_list	73
mindat_locentries_list	74
mindat_locentries_lstm_id	74

mindat_locentries_retrieve	75
mindat_locentries_stat	76
mindat_make_data_frame	76
mindat_mineral_ima	77
mindat_mineral_ima_list	78
mindat_nickel_strunz10_classes	78
mindat_nickel_strunz10_families	79
mindat_nickel_strunz10_subclasses	79
mindat_parse_raw_data	80
mindat_query	81
mindat_setup	81
mindat_spacegroups	82
mindat_spacegroupsets	82
mindat_spacegroupsets_list	83
mindat_spacegroups_list	84
minerals_ima_list	84
minerals_ima_list_expand	85
minerals_ima_list_ima	86
minerals_ima_retrieve	86
minerals_ima_updated_at	87
Nickel_strunz10_classes	88
Nickel_strunz10_families	88
Nickel_strunz10_subclasses	89
params_to_string	90
saveMindatDataAs	90
set_api_base	91
set_api_token	91
set_page_size	92
spacegroupsets_by_id	92
spacegroupsets_cclass	93
spacegroupsets_list	94
spacegroupsets_sgtext	94
spacegroups_by_id	95
spacegroups_cclass	96
spacegroups_list	96
spacegroups_sgtext	97

Index	99
--------------	-----------

ConvertDF2JsonLD	<i>Output file as a given format</i>
------------------	--------------------------------------

Description

Convert the mindat R dataframe to JSON-LD string

Usage

```
ConvertDF2JsonLD(inputdata, template = NULL)
```

Arguments

inputdata	R dataframe of retrieved data from Mindat database.
template	filepath to the template

Examples

```
## Not run:  
df <- geomaterials_search_name("Quartz")  
df_out <- ConvertDF2JsonLD(df)  
  
## End(Not run)
```

ConvertDF2TTL	<i>Convert a dataframe to a string of TTL format</i>
---------------	--

Description

Convert the mindat R dataframe to TTL string

Usage

```
ConvertDF2TTL (inputdata, template = NULL)
```

Arguments

inputdata	R dataframe of retrieved data from Mindat database.
template	filepath to the template

Examples

```
## Not run:  
df <- geomaterials_search_name("Quartz")  
df_out <- ConvertDF2TTL(df)  
  
## End(Not run)
```

crystalclasses_symbols

crystalclasses that match a given vector of symbols (case-insensitive)

Description

: Queries a list of crystalclasses that match a given list of symbols

Usage

```
crystalclasses_symbols(symbols, ...)
```

Arguments

symbols	vector of given crystals (array of strings or null). The field "symbol" is to describe the symbol of Crystal class dictionary.
...	Further parameters like "system"(crystal system) .Other optional arguments-Additional arguments.

Details

This function filter data by a given list of symbols of crystal class dictionary case-insensitive

Value

df, a data frame of crystalclasses

Examples

```
## Not run:
df <-crystalclasses_symbols(c("2/m", "mm2"))

## End(Not run)
```

crystalclasses_systems

crystalclasses that match a given vector of crystal system (case-insensitive)

Description

: Queries a list of crystalclasses that match a given list of crystal system.

Usage

```
crystalclasses_systems(systems, ...)
```

Arguments

systems vector of given system. "crystal system of the mineral; "Amorphous","Hexagonal","Icosahedral","Isometr
... Further named parameters.Other optional arguments.

Details

This function filter data by a given list of crystal_system of crystal class dictionary case-insensitive

Value

df, a data frame of crystalclasses

Examples

```
## Not run:  
df <-geomaterials_crystal_system(c("Icosahedral"))  
  
## End(Not run)
```

Dana8_groups	<i>dana_8 classification</i>
--------------	------------------------------

Description

: Queries a list of Dana 8th edition classifications.

Usage

```
Dana8_groups(...)
```

Arguments

... Further parameters.Other optional arguments-Additional arguments.

Details

This function return a list of dana8 groups case-insensitive

Value

df, a data frame of dana8 groups

Examples

```
## Not run:  
df <-Dana8_groups()  
  
## End(Not run)
```

Dana8_subgroups	<i>dana_8 subgroups</i>
-----------------	-------------------------

Description

: Queries a list of subgroups of the Dana 8th edition classifications.

Usage

Dana8_subgroups(...)

Arguments

... Further parameters.Other optional arguments-Additional arguments.

Details

This function return a list of dana8 subgroups case-insensitive

Value

df, a data frame of dana8 subgroups

Examples

```
## Not run:
df <-Dana8_subgroups()

## End(Not run)
```

geomaterials_bi_greater_than	<i>retrieve the geomaterials whose birifrigence are higher than the given value.</i>
------------------------------	--

Description

: Queries the list of geomaterials that minmum value of the given birifrigence value.

Usage

geomaterials_bi_greater_than(gt, ...)

Arguments

gt float value.Birifrigence is calculated from refractive index as (rimax-rimin).Range: bi_min - bi_max.

... Further named parameters.Other optional arguments.

Details

This function related to the field "bi_min" of geomaterials. retrieve all the geomaterials that has higher birifrigence than the given value(gt).

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_bi_greater_than(0.2)

## End(Not run)
```

```
geomaterials_bi_less_than
```

retrieve the geomaterials whose birifrigence are lower density than the given value.

Description

: Queries the list of geomaterials that have lower birifrigence than lt.

Usage

```
geomaterials_bi_less_than(lt, ...)
```

Arguments

lt	float value.Birifrigence is calculated from refractive index as (rimax-rimin).Range: bi_min - bi_max.
...	Further named parameters.Other optional arguments.

Details

This function related to the field "bi_max" of geomaterials. retrieve all the geomaterials that has higher birifrigence than the given value(lt).

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_bi_less_than(0.3)

## End(Not run)
```

`geomaterials_bi_range` *retrieve the geomaterials whose birifrigence are higher and lower than the given value.*

Description

: Queries the list of geomaterials that have lower birifrigence than lt.

Usage

```
geomaterials_bi_range(gt,lt, ...)
```

Arguments

<code>gt</code>	float value.Birifrigence is calculated from refractive index as (rimax-rimin).Range: bi_min - bi_max.
<code>lt</code>	float value.Birifrigence is calculated from refractive index as (rimax-rimin).Range: bi_min - bi_max.
<code>...</code>	Further named parameters.Other optional arguments.

Details

This function releated to the fields "bi_min"and "bi_max" of geomaterials. retrieve all the geoma-terials that has the birifrigence within the given range of (gt,lt).

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_bi_range(0.2,0.3)

## End(Not run)
```

`geomaterials_by_groupid` *retrieve the geomaterials by an given value of groupid.*

Description

: Queries the list of geomaterials that match an given groupid.

Usage

```
geomaterials_by_groupid(gid,...)
```

Arguments

```
gid          integer value. The id of the group to which this mineral belongs
...          Further named parameters.Other optional arguments.
```

Details

This function related to the field "groupid" of geomaterials. retrieve all the geomaterials that match an given groupid.

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_by_groupid(0)

## End(Not run)
```

```
geomaterials_cleavagetype
```

geomaterials that match an given cleavagetype

Description

: Queries the list of geomaterials that match an given cleavagetype

Usage

```
geomaterials_cleavagetype(types, ...)
```

Arguments

```
types        vector of given cleavagetype (array of strings or null). The field "cleavage" is
              used to describe the crystallographic orientation of cleavage directions or planes
              and quality.
...          Further named parameters.Other optional arguments-Additional arguments.
```

Details

This function related to the field "cleavagetype" of geomaterials. Items Enum: "Distinct/Good" "Imperfect/Fair" "None Observed" "Perfect" "Poor/Indistinct" "Very Good"

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_cleavagetype(c("Poor/Indistinct"))

## End(Not run)
```

geomaterials_colour *geomaterials that have the given colors*

Description

: Queries the list of geomaterials that match a given colors.

Usage

```
geomaterials_colour(colors, ...)
```

Arguments

colors	vector of given colors. colors of the mineral or rock - individual minerals at localities can also have color information.
...	Further named parameters.Other optional arguments-Additional arguments.

Details

This function related to the field "colour" of geomaterials. For example: "Brown", "Yellow", "green", "Pink","White","Orange","Blue","Gold","Dark brown","Purple".

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_colour(c("bright blue"))

## End(Not run)
```

```
geomaterials_contain_all_but_not_elems
```

geomaterials that contain all of some given elements but without any of some other given elements.

Description

Queries the list of geomaterials that contain all the given elements listed in `icl_elms`, but do not contain the given elements listed in `ecl_elms`

Usage

```
geomaterials_contain_all_but_not_elems(icl_elms, ecl_elms, ...)
```

Arguments

<code>icl_elms</code>	vector of elements.
<code>ecl_elms</code>	vector of elements.
<code>...</code>	Further named parameters. Other optional arguments-Additional arguments.

Details

This function related to the field "elements_inc" and "elements_exc" of geomaterials. This function queries the list of geological materials that contain an given list of elements (`icl_elms`), but not contain the other list of elements (`ecl_elms`). It performs the query operation by calling the `mindat_geomaterial_list` function.

Value

`df`, a data frame of geomaterials.

Examples

```
## Not run:
geomaterials_contain_all_but_not_elems(c('Fe','S'), c('O'),fields ="id,name")

## End(Not run)
```

```
geomaterials_contain_all_elems
      geomaterials_contain_all_elems
```

Description

retrieve the geomaterials that contain all of the elements. This function queries the list of geomaterials that contain all the given elements. It performs the query operation by calling the `mindat_geomaterial_list` function

Usage

```
geomaterials_contain_all_elems(icl_elms_vector,...)
```

Arguments

`icl_elms_vector` vector of elements.

`...` Further named parameters. Other optional arguments-Additional arguments that can be passed to the `mindat_geomaterial_list` function.

Details

This function related to the field "elements_inc" of geomaterials.

Value

`df`, a data frame of geomaterials list.

Examples

```
## Not run:
df <- geomaterials_contain_all_elems (c('Fe', 'S'), fields = "id,name,mindat_formula,elements")

## End(Not run)
```

```
geomaterials_contain_any_but_not_elems
      geomaterials that contain any of some given elements but with out any
      of some other given elements
```

Description

: Queries the list of geological materials that contain any one of the given elements.

Usage

```
geomaterials_contain_any_but_not_elems(any_elems_vector, ecl_elms_vector, ...)
```

Arguments

`any_elems_vector` vector of elements. vector of any elements contained.

`ecl_elms_vector` vector of elements. vector of any elements excluded.

`...` Further named parameters. Other optional arguments-Additional arguments.

Details

This function related to the field "elements_inc" of geomaterials. This function queries the list of geological materials that contain any element of an given list (`any_elems`). It performs the query operation by looping through each given element and calling the `mindat_geomaterial_list` function.

Value

`df`, a data frame of geomaterials.

Examples

```
## Not run:
df <- geomaterials_contain_any_but_not_elems(c('Fe', 'S'), c('O'))

## End(Not run)
```

```
geomaterials_contain_any_elems
```

geomaterials that contain any one of the given elements

Description

: Queries the list of geological materials that contain any one of the given elements.

Usage

```
geomaterials_contain_any_elems(any_elems, ...)
```

Arguments

`any_elems` vector of elements.

`...` Further named parameters. Other optional arguments-Additional arguments.

Details

This function related to the field "elements_inc" of geomaterials. This function queries the list of geological materials that contain any element of an given list (any_elems). It performs the query operation by looping through each given element and calling the mindat_geomaterial_list function.

Value

df, a data frame of geomaterials.

Examples

```
## Not run:
df <- geomaterials_contain_any_elems (c('Fe', 'S'), fields = "id,name,mindat_formula,elements")

## End(Not run)
```

```
geomaterials_contain_only_elems
      geomaterials_contain_only_elems
```

Description

retrieve the geomaterials that only contain elements in an given list (icl_only_elms_vector).

Usage

```
geomaterials_contain_only_elems (icl_only_elms_vector,...)
```

Arguments

```
icl_only_elms_vector
      vector of elements.

...
      Further named parameters.Other optional arguments-Additional arguments that
      can be passed to the mindat_geomaterial_list function.
```

Details

This function related to the fields "elements_inc" and "elements_exc" of geomaterials. Here is a list of all elements that can make up geomaterials:'H','Li', 'Be', 'B', 'C', 'N', 'O', 'F', 'Na', 'Mg', 'Al','Si', 'P', 'S','Cl',K', 'Ca', 'Sc', 'Ti', 'V', 'Cr', 'Mn', 'Fe', 'Co', 'Ni', 'Cu', 'Zn', 'Ga', 'Ge','As','Se', 'Br', 'Rb', 'Sr', 'Y', 'Zr', 'Nb', 'Mo', 'Ru', 'Rh', 'Pd', 'Ag', 'Cd','In', 'Sn', 'Sb', 'Te', 'I', 'Cs',Ba', 'La', 'Ce', 'Nd', 'Sm', 'Gd', 'Dy', 'Er','Yb', 'Hf', 'Ta', 'W', 'Re', 'Os', 'Ir', 'Pt', 'Au','Hg','Tl', 'Pb', 'Bi', 'Th', 'U' It performs the query operation by calling the mindat_geomaterial_list function

Value

df, a data frame of geomaterials.

Examples

```
## Not run:  
df <-geomaterials_contain_only_elems(c('Fe','S'),fields ="id,name,mindat_formula,elements")  
  
## End(Not run)
```

geomaterials_crystal_system
geomaterials that have the given crystal

Description

: Queries the list of geomaterials that have the given crystal system.

Usage

```
geomaterials_crystal_system(crystals, ...)
```

Arguments

crystals	vector of given crystals. "crystal system of the mineral; "Amorphous","Hexagonal","Icosahedral","Isomet
...	Further named parameters.Other optional arguments.

Details

This function related to the field "crystal_system" of geomaterials. Items Enum: "Amorphous" "Hexagonal" "Icosahedral" "Isometric" "Monoclinic" "Orthorhombic" "Tetragonal" "Triclinic" "Trigonal"

Value

df, a data frame of geomaterials

Examples

```
## Not run:  
df <-geomaterials_crystal_system(c("Icosahedral"))  
  
## End(Not run)
```

```
geomaterials_dens_greater_than
```

retrieve the geomaterials whose density are higher than a given value.

Description

: Queries the list of geomaterials that have higher density than gt.

Usage

```
geomaterials_dens_greater_than(gt, ...)
```

Arguments

gt	float value. dmeas: measured density of the mineral. This is either the lower limit (if there is a dmeas2) or average (if there is no dmeas2). dmeas2:measured maximum density of mineral
...	Further named parameters.Other optional arguments.

Details

This function releated to the field "density_min" of geomaterials. retrieve all the geomaterials that has higher density than the given density(gt).

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_dens_greater_than(2)

## End(Not run)
```

```
geomaterials_dens_less_than
```

retrieve the geomaterials whose density are lower density than a given value.

Description

: Queries the list of geomaterials that have lower density than lt.

Usage

```
geomaterials_dens_less_than(lt, ...)
```

Arguments

`lt` float value. `dmeas`: measured density of the mineral. This is either the lower limit (if there is a `dmeas2`) or average (if there is no `dmeas2`). `dmeas2`: measured maximum density of mineral

`...` Further named parameters. Other optional arguments.

Details

This function related to the field "density_max" of geomaterials. retrieve all the geomaterials that has higher density than the given density(`lt`).

Value

`df`, a data frame of geomaterials

Examples

```
## Not run:
df <- geomaterials_dens_less_than(3)

## End(Not run)
```

`geomaterials_dens_range`

retrieve the geomaterials whose density are within an given value.

Description

: Queries the list of geomaterials that match an given range.

Usage

```
geomaterials_dens_range(gt,lt, ...)
```

Arguments

`gt` float value

`lt` float value `dmeas`: measured density of the mineral. This is either the lower limit (if there is a `dmeas2`) or average (if there is no `dmeas2`). `dmeas2`: measured maximum density of mineral

`...` Further named parameters. Other optional arguments.

Details

This function related to the fields "density_min" and "density_max" of geomaterials. retrieve all the geomaterials records that has the density within an given range of (`gt`,`lt`).

Value

df, a data frame of geomaterials

Examples

```
## Not run:  
df <-geomaterials_dens_range(2,3)  
  
## End(Not run)
```

geomaterials_diapheny *retrieve the geomaterials that have an given diapheny.*

Description

: Queries the list of geomaterials that have an given diapheny.

Usage

```
geomaterials_diapheny(diapheny, ...)
```

Arguments

diapheny	string. The diaphany of the mineral - transparent; translucent; opaque
...	Further named parameters.Other optional arguments.

Details

This function releated to the field "diapheny" of geomaterials. The diaphany of the mineral(Items Enum): "Opaque" "Translucent" "Transparent"

Value

df, a data frame of geomaterials

Examples

```
## Not run:  
df <-geomaterials_diapheny("Transparent")  
  
## End(Not run)
```

geomaterials_entrytype

retrieve the geomaterials that have the given entrytype

Description

: Queries the list of geomaterials that have the given entrytype

Usage

```
geomaterials_entrytype(types,...)
```

Arguments

types	list of entry types.
...	Further named parameters.Other optional arguments.

Details

This function related to the field "entrytype" of geomaterials. Items Enum: 0 1 2 3 4 5 6 7 8
Multiple choice: 0- mineral; 1-synonym; 2-variety; 3-mixture; 4-series; 5-grouplist; 6-polytype;
7-rock; 8-commodity Related field: entrytype_text (description of the entrytype).

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_entrytype(c('1'))

## End(Not run)
```

geomaterials_expand

retrieve the geomaterials that have the given expand.

Description

: Queries the list of geomaterials that have the given expand.

Usage

```
geomaterials_expand(expanded_fields,...)
```

Arguments

expanded_fields list of expand (Array of strings (Expanded fields)).Select fields to expand.
 ... Further named parameters.Other optional arguments.

Details

This function related to the field "expand" of geomaterials. The field expand(Items Enum): "description" "type_localities" "localities" "relations" "~all" "*" "

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_expand("~all")

## End(Not run)
```

geomaterials_field_exists

retrieve the geomaterials records of empty or not empty of a given field.

Description

: Queries the list of geomaterials with an empty or not empty of a given field.

Usage

```
geomaterials_field_exists(fieldname,bexists,...)
```

Arguments

fieldname string
 bexists bool
 ... Further named parameters.Other optional arguments.

Details

This function related to all the fields of geomaterials. e.g. meteoritical_code_exists.Meteoritical code exists. Include non-empty (true) / include empty only (false) retrieve the geomaterial list with an empty or not empty of a given field.

Value

df, a list of geomaterials

Examples

```
## Not run:
df <-geomaterials_field_exists("meteoritical_code")

## End(Not run)
```

geomaterials_fracturetype

retrieve the geomaterials that have the given fracturetype.

Description

: Queries the list of geomaterials that have the given fracturetype.

Usage

```
geomaterials_fracturetype(types, ...)
```

Arguments

types	list of types.fracturetype(Array of strings or null): How the mineral breaks-"Conchoidal" "Fibrous" "Hackly" "Irregular/Uneven" "Micaceous" "None observed" "Splintery" "Step-Like" "Sub-Conchoidal".
...	Further named parameters.Other optional arguments.

Details

This function related to the field "fracturetype" of geomaterials. fracturetype(Items Enum): "Conchoidal" "Fibrous" "Hackly" "Irregular/Uneven" "Micaceous" "None observed" "Splintery" "Step-Like" "Sub-Conchoidal"

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_fracturetype(c("Step-Like"))

## End(Not run)
```

```
geomaterials_hardness_gt
```

retrieve the geomaterials whose hardness are higher than an given value.

Description

: Queries the list of geomaterials that have higher hardness than an given value(hmin).

Usage

```
geomaterials_hardness_gt(hmin, ...)
```

Arguments

hmin	float value of the Mohs scale of mineral hardness, which ranging from 0 to 10.
...	Further named parameters.Other optional arguments.

Details

This function related to the field "hardness_min" of geomaterials. retrieve all the geomaterials that has higher hardness than the given value(hmin). hmin:the given value of minimum Moh's hardness

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_hardness_gt(8)

## End(Not run)
```

```
geomaterials_hardness_lt
```

retrieve the geomaterials whose hardness are lower than an given value.

Description

: Queries the list of geomaterials that have lower hardness than an given vlaue(hmax).

Usage

```
geomaterials_hardness_lt(hmax, ...)
```


Arguments

hmax float value of the Mohs scale of mineral hardness, which ranging from 0 to 10.
 ... Further named parameters.Other optional arguments.

Details

This function related to the field "hardness_max" of geomaterials. retrieve all the geomaterials that has lower hardness than an given value(hmax). hamx: maximum Moh's hardness

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_hardness_lt(2)

## End(Not run)
```

geomaterials_hardness_range

retrieve the geomaterials whose hardness is within the given range.

Description

: Queries the list of geomaterials that have hardness within the given range.

Usage

```
geomaterials_hardness_range(hmin,hmax, ...)
```

Arguments

hmin float value of the Mohs scale of mineral hardness, which ranging from 0 to 10.
 hmax float value of the Mohs scale of mineral hardness, which ranging from 0 to 10.
 ... Further named parameters.Other optional arguments.

Details

This function related to the fields "hardness_min" and "hardness_max" of geomaterials. retrieve all the geomaterials that has the hardness within an given range(hmin,hmax). hmin:the given value of minimum Moh's hardness hamx: maximum Moh's hardness

Value

df, a data frame of geomaterials

Examples

```
## Not run:  
df <-geomaterials_hardness_range(2,3)  
  
## End(Not run)
```

geomaterials_ima	<i>retrieve the geomaterials approved by IMA or not.</i>
------------------	--

Description

: Queries the geomaterials within or without the ima.

Usage

```
geomaterials_ima(btrue,...)
```

Arguments

btrue	boolean value.TRUE IMA approved, otherwise not approved.
...	Further named parameters.Other optional arguments.

Details

This function related to the field "ima" of geomaterials. retrieve all the geomaterials that are approved by the IMA or not.

Value

df, a data frame of geomaterials

Examples

```
## Not run:  
df <-geomaterials_ima(TRUE)  
  
## End(Not run)
```

`geomaterials_ima_notes`*retrieve the geomaterials match given notes.*

Description

: Queries the geomaterials with an given .

Usage

```
geomaterials_ima_notes(enum_item,...)
```

Arguments

`enum_item` Array of integers or null. Ima notes: multiple choice (OR) : "GROUP" "INTERMEDIATE" "NAMED_AMPHIBOLE" "PENDING_APPROVAL" "PUBLISHED_WITHOUT_APPROVAL" "REDEFINED" "REJECTED" "RENAMED" "UNNAMED_INVALID" "UNNAMED_VALID"

`...` Further named parameters.Other optional arguments.

Details

This function related to the field "ima_notes" of geomaterials. Rejected by the IMA; Pending approval; IMA Approved Group Name; Redefined by the IMA; Renamed by the IMA; Intermediate member of a solid-solution series; Published without approval; Unnamed (probably valid); Unnamed (probably invalid); Named Amphibole

retrieve all the geomaterials that match the input IMA notes.

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_ima_notes(c("PENDING_APPROVAL"))

## End(Not run)
```

```
geomaterials_ima_status
```

retrieve the geomaterials matched given IMA status.

Description

: Queries the geomaterials with an given ima status.

Usage

```
geomaterials_ima_status(enum_status,...)
```

Arguments

enum_status	Ima status: multiple choice (OR): "APPROVED" "DISCREDITED" "GRAND-FATHERED" "PENDING_PUBLICATION" "QUESTIONABLE"
...	Further named parameters. Other optional arguments.

Details

This function related to the field "ima_status" of geomaterials. retrieve all the geomaterials that match the input IMA notes.

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_ima_status(c("APPROVED"))

## End(Not run)
```

```
geomaterials_lustretype
```

retrieve the geomaterials that match an given lustretype.

Description

: Queries the geomaterials that match an given lustretype.

Usage

```
geomaterials_lustretype(types, ...)
```

Arguments

types string of the type name (Array of strings or null). adamantine, subadamantine, vitreous, subvitreous, resinous, waxy, greasy, silky, pearly, metallic, submetallic, dull, earthy

... Further named parameters. Other optional arguments.

Details

This function related to the field "lustretype" of geomaterials. lustretype(Items Enum): "Adamantine" "Dull" "Earthy" "Greasy" "Metallic" "Pearly" "Resinous" "Silky" "Sub-Adamantine" "Sub-Metallic" "Sub-Vitreous" "Vitreous" "Waxy" multiple choice (AND)

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <- geomaterials_lustretype(c("Adamantine"))

## End(Not run)
```

geomaterials_meteoritical_code

retrieve the geomaterials matched a given string in its meteoritical code.

Description

: Queries the geomaterials with a given string matched its given meteoritical_code.

Usage

```
geomaterials_meteoritical_code(str_meteoritical_code,...)
```

Arguments

str_meteoritical_code boolean, meteoritical code exists. Include non-empty (TRUE) / include empty only (FALSE).

... Further named parameters. Other optional arguments.

Details

This function related to the field "meteoritical_code_exists" of geomaterials. Meteoritical code exists. Include non-empty (true) / include empty only (false). retrieve all the geomaterials that match the input str_meteoritical_code.

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_meteoritical_code(TRUE)

## End(Not run)
```

geomaterials_name	<i>retrieve the geomaterials matched a given string in its name.</i>
-------------------	--

Description

: Queries the geomaterials with a given name.

Usage

```
geomaterials_name(str_name,...)
```

Arguments

str_name	Text search supporting: _ as wildcards, e.g. "qu_rtz", "bario*"
...	Further named parameters.Other optional arguments.

Details

This function related to the field "name" of geomaterials. retrieve all the geomaterials that match the input IMA notes.

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_name("qu_rtz")

## End(Not run)
```

geomaterials_not_contain_elems

geomaterials that do not contain the elements

Description

retrieve the geomaterials that do not contain any of the given elements.

Usage

```
geomaterials_not_contain_elems (ecl_elms_vector, ...)
```

Arguments

ecl_elms_vector

vector of elements.

... Further named parameters. Other optional arguments-Additional arguments.

Details

This function related to the field "elements_exc" of geomaterials.

Value

df, a data frame of geomaterials list.

Examples

```
## Not run:
df <- geomaterials_not_contain_elems (c('Fe', 'S', 'O'), fields = "id,name,min_dat_formula,elements")

## End(Not run)
```

geomaterials_optical2v_max

retrieve the geomaterials that less than the given optical 2v.

Description

: Queries the geomaterials have the lower optical 2v value than the given lt.

Usage

```
geomaterials_optical2v_max(lt, ...)
```

Arguments

- lt list of the signs.Please refer to the details.
- ... Further named parameters.Other optional arguments.

Details

This function releated to the field "optical2v_max" of geomaterials. optical2vcalc:The calculated 2V angle minimum or average of biaxial minerals optical2vcalc2:The calculated 2V angle maximum of biaxial minerals optical2vmeasured:The measured 2V angle minimum or average of biaxial minerals optical2vmeasured2:The measured 2V angle maximum of biaxial minerals

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_optical2v_max(0.2)

## End(Not run)
```

geomaterials_optical2v_min	
	<i>retrieve the geomaterials that has higher value than the given optical 2v.</i>

Description

: Queries the geomaterials have the higher optical 2v value than the given gt.

Usage

```
geomaterials_optical2v_min(gt, ...)
```

Arguments

- gt given value of optical 2v of mineral.Please refer to the details.
- ... Further named parameters.Other optional arguments.

Details

This function releated to the field "optical2v_mix" of geomaterials. optical2vcalc:The calculated 2V angle minimum or average of biaxial minerals optical2vcalc2:The calculated 2V angle maximum of biaxial minerals optical2vmeasured:The measured 2V angle minimum or average of biaxial minerals optical2vmeasured2:The measured 2V angle maximum of biaxial minerals

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_optical2v_min(0.1)

## End(Not run)
```

geomaterials_optical2v_range

retrieve the geomaterials that has the given range of optical 2v.

Description

: Queries the geomaterials have the higher optical 2v value than the given lt.

Usage

```
geomaterials_optical2v_range(gt,lt,...)
```

Arguments

gt	given value of minimum of optical 2v of mineral.Please refer to the details.
lt	an given value of maximum of optical 2v of mineral.Please refer to the details.
...	Further named parameters.Other optional arguments.

Details

This function related to the field "optical2v_min" and "optical2v_max" of geomaterials. optical2vcalc:The calculated 2V angle minimum or average of biaxial minerals optical2vcalc2:The calculated 2V angle maximum of biaxial minerals optical2vmeasured:The measured 2V angle minimum or average of biaxial minerals optical2vmeasured2:The measured 2V angle maximum of biaxial minerals

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_optical2v_range(0.1,0.2)

## End(Not run)
```

```
geomaterials_opticalsign
```

retrieve the geomaterials that match an given optical signs.

Description

: Queries the geomaterials match an given optical signs.

Usage

```
geomaterials_opticalsign(signs, ...)
```

Arguments

<code>signs</code>	list of the signs(string or null). sign for uniaxial and biaxial minerals: +;+/- .Please refer to the details.
<code>...</code>	Further named parameters.Other optional arguments.

Details

This function related to the field "opticalsign" of geomaterials. Optical sign: single choice (Enum): "+", "+/-", "-"

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_opticalsign("-")

## End(Not run)
```

```
geomaterials_opticaltype
```

retrieve the geomaterials that match an given optical type.

Description

: Queries the geomaterials match an given optical type.

Usage

```
geomaterials_opticaltype(types, ...)
```

Arguments

types list of the types for the field of opticaltype. Please refer to the details.
 ... Further named parameters.Other optional arguments.

Details

This function related to the field "opticaltype" of geomaterials. transparent mineral. optical-type(Enum) : "Biaxial" "Isotropic" "Uniaxial"

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_opticaltype("Isotropic")

## End(Not run)
```

geomaterials_polytypeof

retrieve the geomaterials by an given id of polytype of (the id of the mineral that this record is the polytype of.)

Description

: Queries the geomaterials by an given id for its polytype. A mineral that differs from another only in the stacking of similar structural units in its atomic structure

Usage

```
geomaterials_polytypeof(ptype, ...)
```

Arguments

ptype integer. an mindat id of the mineral that this record is the polytype of
 ... Further named parameters.Other optional arguments.

Details

This function related to the field "polytypeof" of geomaterials. retrieve the geomaterials with an given id of polytypeof.

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_polytypeof(0,fields = "id,name,polytypeof")

## End(Not run)
```

geomaterials_ri_gt	<i>retrieve the geomaterials that refractive index higher than an given value(gt).</i>
--------------------	--

Description

: Queries the geomaterials have the higher refractive index than an given value(gt).

Usage

```
geomaterials_ri_gt(gt, ...)
```

Arguments

gt	float value. Refractive index, from (rimax>=).
...	Further named parameters.Other optional arguments.

Details

This function related to the field "ri_min" of geomaterials. retrieve the geomaterials with the refractive index higher than an given value(gt).

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_ri_gt(0.3)

## End(Not run)
```

geomaterials_ri_lt	<i>retrieve the geomaterials that refractive index lower than an given value(lt).</i>
--------------------	---

Description

: Queries the geomaterials have the lower refractive index than an given value(lt).

Usage

```
geomaterials_ri_lt(lt,...)
```

Arguments

lt	float value. Refractive index, to (rimin<=)
...	Further named parameters.Other optional arguments.

Details

This function related to the field "ri_max" of geomaterials. retrieve the geomaterials with the refractive index lower than an given value(lt).

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_ri_lt(0.5)

## End(Not run)
```

geomaterials_ri_range	<i>retrieve the geomaterials whose refractive index is within an given range(gt,lt).</i>
-----------------------	--

Description

: Queries the list of geomaterials that have refractive index within an given range(gt,lt).

Usage

```
geomaterials_ri_range(gt,lt, ...)
```

Arguments

gt	float value. Refractive index, from (rimax>=).
lt	float value. Refractive index, to (rimin<=)
...	Further named parameters.Other optional arguments.

Details

This function related to the fields "ri_min" and "ri_max" of geomaterials. retrieve all the geomaterials that has the refractive index within the range of (gt,lt).

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_ri_range(0.2,0.5)

## End(Not run)
```

geomaterials_search_name

retrieve the geomaterials by a given name.

Description

: Queries the list of geomaterials by a given name.

Usage

```
geomaterials_search_name(name,...)
```

Arguments

name	string. Text search supporting wildcards, e.g. qu_rtz, bario*"
...	Further named parameters.Other optional arguments.

Details

This function related to the fields "name" of geomaterials. retrieve the geomaterial list that match the given name.

Value

df, a list of geomaterials

Examples

```
## Not run:  
df <-geomaterials_search_name("Quartz")  
  
## End(Not run)
```

geomaterials_streak	<i>retrieve the geomaterials that match an given streak.</i>
---------------------	--

Description

: Queries the list of geomaterials that match an given steak.

Usage

```
geomaterials_streak(str,...)
```

Arguments

str	string. The color of the streak (color of powdered mineral)
...	Further named parameters.Other optional arguments.

Details

This function releated to the fields "steak" of geomaterials. The color of the streak (color of powdered mineral). retrieve the geomaterials that has the given steak.

Value

df, a data frame of geomaterials

Examples

```
## Not run:  
df <-geomaterials_streak("orange")  
  
## End(Not run)
```

geomaterials_synid	<i>retrieve the geomaterials by an given synid.</i>
--------------------	---

Description

: Queries the list of geomaterials that match an given synid.

Usage

```
geomaterials_synid(idnum,...)
```

Arguments

idnum	integer,an given synonym id.
...	Further named parameters.Other optional arguments.

Details

This function releated to the fields "synid" of geomaterials. The id of the geomaterial that is the synonym of this record (this geomaterial cannot be added to a locality). retrieve the geomaterials that has an given synid.

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_synid(3777)

## End(Not run)
```

geomaterials_updated_at	<i>retrieve the geomaterials updated at an given time.</i>
-------------------------	--

Description

: Queries the list of geomaterials that were updated at an given time

Usage

```
geomaterials_updated_at(strDate,...)
```


Arguments

strDate string(date-time>), Last updated datetime in format %Y-%m-%d %H:%M:%S
 ... Further named parameters.Other optional arguments.

Details

This function related to the fields "updated_at" of geomaterials. Last updated datetime in format %Y-%m-%d %H:%M:%S retrieve the geomaterials that have the latest updated at the given time.

Value

df, a data frame of geomaterials

Examples

```
## Not run:
df <-geomaterials_updated_at("2020-2-12 10:15:12")

## End(Not run)
```

geomaterials_varietyof

retrieve the geomaterials that are varieties of an given id of geomaterials.

Description

: Queries the list of geomaterials that match the given varietyof.

Usage

```
geomaterials_varietyof(intvalue,...)
```

Arguments

intvalue integer, id of mineral that has this variety.
 ... Further named parameters.Other optional arguments.

Details

This function related to the fields "varietyof" of geomaterials. Varieties are geomaterials that have a special distinction from the main geomaterial ie. amethyst var. quartz retrieve the geomaterials that are varieties of an given id of geomaterials.

Value

df, a data frame of geomaterials

Examples

```
## Not run:  
df <-geomaterials_varietyof(3337)  
  
## End(Not run)
```

`geomeaterials_non_utf` *retrieve the geomaterials that include non-utf mineral names or not.*

Description

: Queries the geomaterials include non-utf mineral names or not.

Usage

```
geomeaterials_non_utf(btrue =TRUE,...)
```

Arguments

<code>btrue</code>	boolean. Include non-UTF mineral names?.Default is TRUE.
<code>...</code>	Further named parameters.Other optional arguments.

Details

This function releated to the field "non_utf" of geomaterials. retrieve the geomaterials that contain (or not contain) the non-utf name.

Value

df, a data frame of geomaterials

Examples

```
## Not run:  
df <-geomeaterials_non_utf(TRUE,fields = "id,name,non_utf")  
  
## End(Not run)
```

`geomeaterials_ordering`*retrieve the geomaterials by an given ordering.*

Description

: Queries the geomaterials by an given ordering.

Usage

```
geomeaterials_ordering(ord, ...)
```

Arguments

<code>ord</code>	string of field. Prepend "-" to the field name for descending order. Enum: "approval_year" "id" "minstats__ms_locentries" "minstats__ms_photos" "name" "updttime" "weighting".
<code>...</code>	Further named parameters.Other optional arguments.

Details

This function releated to the field "ordering" of geomaterials. ordering=-id - sort by id descending. Prepend "-" to the field name for descending order. fields:"approval_year" "id" "minstats__ms_locentries" "minstats__ms_photos" "name" "updttime" "weighting". retrieve the geomaterials by an given ordering.

Value

df, a data frame of geomaterials

Examples

```
## Not run:  
df <-geomeaterials_ordering(-id)  
  
## End(Not run)
```

getExtension	<i>Output the file extension of a filename</i>
--------------	--

Description

Convert the mindat R dataframe to JSON-LD string

Usage

```
getExtension (filename)
```

Arguments

filename R dataframe of retrieved data from Mindat database.

Examples

```
filename<- "fname.txt"
fname_extension<- getExtension(filename)
```

localities_list_all	<i>retrieve the localities list.</i>
---------------------	--------------------------------------

Description

: Queries the list of localities.

Usage

```
localities_list_all(...)
```

Arguments

... Further named parameters. Other optional arguments.

Details

This function related to the fields "ids" of localities. retrieve all the localities.

Value

df, a data frame of localities

Examples

```
## Not run:
df <-localities_list_all(fields = "id,name,latitude,longitude")

## End(Not run)
```

localities_list_country

retrieve the localities list that are belong to a given country.

Description

: Queries the list of localities that are within a given country.

Usage

```
localities_list_country(country,...)
```

Arguments

country	name of country,
...	Further named parameters.Other optional arguments.

Details

This function related to the field "country" of localities. Enum: "Afghanistan" "Albania" "Algeria" "American Samoa" "Andorra" "Angola" "Anguilla" "Antigua and Barbuda" "Argentina" "Armenia" "Aruba" "Ashmore and Cartier Islands" "Australia" "Austria" "Azerbaijan" "Bahamas" "Bahrain" "Bangladesh" "Barbados" "Belarus" "Belgium" "Belize" "Benin" "Bermuda" "Bhutan" "Bolivia" "Bosnia And Herzegovina" "Botswana" "Bouvet Island" "Brazil" "British Indian Ocean Territories" "British Solomon Islands" "British Virgin Islands" "Brunei" "Bulgaria" "Burkina Faso" "Burundi" "Cambodia" "Cameroon" "Canada" "Cape Verde" "Cayman Islands" "Central African Republic" "Chad" "Chile" "China" "Christmas Island" "Cocos Islands" "Colombia" "Comoro Islands" "Cook Islands" "Costa Rica" "Croatia" "Cuba" "Cyprus" "Czech Republic" "Democratic Republic of the Congo" "Denmark" "Djibouti" "Dominica" "Dominican Republic" "East Timor" "Ecuador" "Egypt" "El Salvador" "Equatorial Guinea" "Estonia" "Ethiopia" "Faeroe Islands" "Falkland Islands" "Federated States of Micronesia" "Fiji" "Finland" "France" "French Guiana" "French Polynesia" "Gabon" "Gambia" "Georgia" "Germany" "Ghana" "Gibraltar" "Greece" "Greenland" "Grenada" "Guadeloupe" "Guam" "Guatemala" "Guernsey" "Guinea" "Guinea-Bissau" "Guyana" "Haiti" "Honduras" "Hong Kong" "Hungary" "Iceland" "India" "Indonesia" "Iran" "Iraq" "Ireland" "Isle of Man" "Israel" "Italy" "Ivory Coast (Côte d'Ivoire)" "Jamaica" "Japan" "Jersey" "Jordan" "Kazakhstan" "Kenya" "Kiribati" "Kosovo" "Kuwait" "Kyrgyzstan" "Laos" "Latvia" "Lebanon" "Lesotho" "Liberia" "Libya" "Liechtenstein" "Lithuania" "Luxembourg" "Macao" "Madagascar" "Malawi" "Malaysia" "Maldives" "Mali" "Malta" "Martinique" "Mauritania" "Mauritius" "Mexico" "Moldova" "Monaco" "Mongolia" "Montenegro" "Montserrat" "Morocco" "Mozambique" "Myanmar" "Namibia" "Nauru" "Nepal" "Netherlands" "Netherlands Antilles" "New Caledonia" "New Zealand" "Nicaragua" "Niger" "Nigeria" "North Korea" "Norway" "Oman" "Pakistan" "Panama" "Papua New Guinea" "Paraguay" "Peru" "Philippines" "Poland" "Portugal" "Puerto Rico" "Qatar" "Republic of Congo (Brazzaville)" "Republic of Macedonia" "Reunion Island" "Romania" "Russia" "Rwanda" "Saint Helena" "Saint Lucia" "Saint Vincent and the Grenadines" "San Marino" "Sao Tome And Principe" "Saudi Arabia" "Senegal" "Serbia" "Seychelles" "Sierra Leone" "Singapore" "Slovakia" "Slovenia" "Solomon Islands" "Somalia" "South Africa" "South Korea" "Spain"

"Sri Lanka" "St Christopher-Nevis Islands" "Sudan" "Suriname" "Swaziland" "Sweden" "Switzerland" "Syria" "Taiwan" "Tajikistan" "Tanzania" "Thailand" "Togo" "Tonga" "Trinidad And Tobago" "Tunisia" "Turkey" "Turkmenistan" "Turks And Caicos Islands" "Tuvalu" "U.S. Virgin Islands" "Uganda" "Ukraine" "United Arab Emirates" "United Kingdom" "United States" "Uruguay" "Uzbekistan" "Vanuatu (Republic of Vanuatu; New Hebrides)" "Venezuela" "Vietnam" "Western Sahara" "Western Samoa" "Yemen" "Zambia" "Zimbabwe"

Value

df, a data frame of localities

Examples

```
## Not run:
df <-localities_list_country ("Norway")

## End(Not run)
```

localities_list_description

retrieve the localities that contain the given description

Description

: Queries the list of localities that contain the given description.

Usage

```
localities_list_description(desc,...)
```

Arguments

desc string,
... Further named parameters.Other optional arguments.

Details

This function related to all the field "description" of localities. retrieve the localities that contain the given description

Value

df, a data frame of localities

Examples

```
## Not run:
df <-localities_list_description("volcano")

## End(Not run)
```

`localities_list_elems_exc`*localities that do not contain the given elements*

Description

Queries the list of localities that do not contain the given elements.

Usage

```
localities_list_elems_exc(exc_elems_list, ...)
```

Arguments

`exc_elems_list` vector of elements.

`...` Further named parameters. Other optional arguments-Additional arguments.

Details

This function related to the field "elements_exc" of localities. This function queries the list of localities that do contain the specified elements.

Value

`df`, a data frame of localities

Examples

```
## Not run:  
df<-localities_list_elems_exc(c("H","O","Si","Al","Fe","Ca","Na","K","P","C","Mn","F","Mg","S"))  
  
## End(Not run)
```

`localities_list_elems_inc`*localities that contain the given elements*

Description

Queries the list of localities that contain the given elements.

Usage

```
localities_list_elems_inc(inc_elems_list, ...)
```

Arguments

`inc_elems_list` vector of elements.
 ... Further named parameters. Other optional arguments-Additional arguments.

Details

This function related to the field "elements_inc" of localities. This function queries the list of localities that contain the given elements.

Value

df, a data frame of localities

Examples

```
## Not run:
df <-localities_list_elems_inc(c("Dy"))

## End(Not run)
```

localities_list_elems_inc_exc

localities that contain the given elements but not contain some other given elements.

Description

Queries the list of localities that contain the given elements, but not contain some other given elements.

Usage

```
localities_list_elems_inc_exc(inc_elems_list, exc_elems_list, ...)
```

Arguments

`inc_elems_list` vector of elements.
`exc_elems_list` vector of elements.
 ... Further named parameters. Other optional arguments-Additional arguments.

Details

This function related to the fields "elements_inc" and "elements_exc" of localities. This function queries the list of localities that contain the given elements, but not contain some other given elements.

Value

df, a data frame of localities

Examples

```
## Not run:  
df <-localities_list_elems_inc_exc(c("Dy"), c("Li"))  
  
## End(Not run)
```

localities_list_expand

localities that contain the given expands.

Description

Queries the list of localities that contain the given expands.

Usage

```
localities_list_expand(expands,...)
```

Arguments

expands	vector of expands.
...	Further named parameters.Other optional arguments-Additional arguments.

Details

This function related to the fields "expand" of localities. Items Enum: "geomaterials" "~all" "*"
This function queries the list of localities that contain the given expands.

Value

df, a data frame of localities

Examples

```
## Not run:  
df <-localities_list_expand("~all")  
  
## End(Not run)
```

localities_list_txt	<i>localities that contain the given txt name.</i>
---------------------	--

Description

Queries the list of localities that contain the given txt name.

Usage

```
localities_list_txt(txt,...)
```

Arguments

txt	string.
...	Further named parameters.Other optional arguments.

Details

This function related to the fields "txt" of localities. This function queries the list of localities that contain the given txt name.

Value

df, a data frame of localities

Examples

```
## Not run:
df <-localities_list_txt("lava")

## End(Not run)
```

localities_list_updated_at	<i>retrieve the localities list updated at the given time.</i>
----------------------------	--

Description

: Queries the list of localities that have the given time

Usage

```
localities_list_updated_at(updateDate,...)
```

Arguments

updateDate string (date-time), Last updated datetime in format %Y-%m-%d %H:%M:%S
 ... Further named parameters.Other optional arguments.

Details

This function related to all the fields "updated_at" of localities. retrieve the localities that have the latest updated at the given time.

Value

df, a data frame of localities

Examples

```
## Not run:
df <-localities_list_updated_at("lava")

## End(Not run)
```

localities_retrieve_id

retrieve the localities by a given mindat id.

Description

: Queries the localitiy by given id.

Usage

```
localities_retrieve_id(id,...)
```

Arguments

id integer. the mindat localitiy id.
 ... Further named parameters.Other optional arguments.

Details

This function related to all the fields "id" of localities. retrieve the localities by a given id.

Value

df, a data frame of localities

Examples

```
## Not run:  
df <-localities_retrieve_id(3337)  
  
## End(Not run)
```

localities_status_list

localities_status_list

Description

retrieve all locality status list.

Usage

```
localities_status_list (...)
```

Arguments

... Further named parameters.

Details

This function is to retrieve all the locality_status list.

Value

df, data frame of locality status

Examples

```
## Not run:  
df <-localities_status_list()  
  
## End(Not run)
```

```
localities_status_retrieve  
    localities_status_retrieve
```

Description

retrieve locality status by its id.

Usage

```
localities_status_retrieve (id,...)
```

Arguments

id	the mindat localitiy status id
...	Further named parameters.

Details

This function is to retrieve the locality_status by an given id of locality.

Value

df, data frame of locality status.

Examples

```
## Not run:  
df <-localities_status_retrieve(10)  
  
## End(Not run)
```

```
localitiy_type_retrieve  
    localitiy_type_retrieve
```

Description

retrieve locality type by an given id of locality.

Usage

```
localitiy_type_retrieve (id,...)
```

Arguments

id the mindat localitiy id
... Further named parameters.

Details

This function is to retrieve the locality types by an given id of locality. @export

Value

df, data frame of locality status.

Examples

```
## Not run:  
df <-localitiy_type_retrieve(50)  
  
## End(Not run)
```

locality_age	<i>locality_age</i>
--------------	---------------------

Description

retrieve locality age by its id

Usage

locality_age (id,...)

Arguments

id the mindat localitiy age id.
... Further named parameters.

Details

This function releated to the fields "id" of locality_age and locality.

Value

df, data frame of locality age.

Examples

```
## Not run:  
df <-locality_age(3337)  
  
## End(Not run)
```

locality_age_list	<i>locality_age_list</i>
-------------------	--------------------------

Description

retrieve all locality age list or by its conditions

Usage

```
locality_age_list (...)
```

Arguments

... Further named parameters.

Details

This function is to retrieve all the locality_age list.

Value

df, data frame of locality age.

Examples

```
## Not run:  
df <-locality_age_list()  
  
## End(Not run)
```

locality_type_list	<i>locality_type_list</i>
--------------------	---------------------------

Description

retrieve all locality type list.

Usage

```
locality_type_list (...)
```

Arguments

... Further named parameters.

Details

This function is to retrieve the locality types list.

Value

df, data frame of locality type.

Examples

```
## Not run:  
df <-locality_type_list()  
  
## End(Not run)
```

Locentries_list	<i>retrieve a full list of locacentries.</i>
-----------------	--

Description

: Queries locacentries.

Usage

```
Locentries_list(...)
```

Arguments

... Further named parameters.Other optional arguments.

Details

This function return a list of locacentries.

Value

df, a data frame of localities

Examples

```
## Not run:  
df <-Locentries_list()  
  
## End(Not run)
```

Locentries_retrieve *retrieve locacentries by given a Locentry ID.*

Description

: Queries locacentries by given a ID.

Usage

```
Locentries_retrieve(id,...)
```

Arguments

id	mindat Locentry id (Integer)
...	Further named parameters.Other optional arguments.

Details

This function return a list of locacentries.

Value

df, a data frame of localities

Examples

```
## Not run:  
df <-Locentries_retrieve(2)  
  
## End(Not run)
```

Locentries_statstics_list
 retrieve a full list of locacentries statstics.

Description

: Queries locacentries statstics list.

Usage

```
Locentries_statstics_list(...)
```

Arguments

... Further named parameters.Other optional arguments.

Details

This function return a list of locacentries.

Value

df, a data frame of localities

Examples

```
## Not run:  
df <-Locentries_statstics_list()  
  
## End(Not run)
```

Locentries_stat_retrieve

retrieve locacentries statistics by given a Locentry ID.

Description

: Queries locacentries locacentries statistics by given a ID.

Usage

```
Locentries_stat_retrieve(id,...)
```

Arguments

id	mindat Locentry id (Integer)
...	Further named parameters.Other optional arguments.

Details

This function return a list of locacentries.

Value

df, a data frame of localities

Examples

```
## Not run:  
df <-Locentries_stat_retrieve(2)  
  
## End(Not run)
```

```
mindat_build_querystring
      mindat_build_querystring
```

Description

Build query string based on the query conditions.

Usage

```
mindat_build_querystring (args)
```

Arguments

args query args.

Value

qs. generated query string.

Examples

```
## Not run:
mindat_cache_set('page-size',800)
ids<-c("")
hardness_min<- 9.3
fields<- c("name,hardness")
args<- list(ids,hardness_min,fields)
querystring<-mindat_build_querystring(args)

## End(Not run)
```

```
mindat_cache_delete      Delete a cached value by the users input varname
```

Description

Remove (clear) the cache named varname in current environment.

Usage

```
mindat_cache_delete(varname)
```

Arguments

varname string input a cached name.Set a cached value empty by the given varname. A string, list or other objects.

Examples

```
mindat_cache_delete('api_token')
```

mindat_cache_empty	<i>Remove all cached values</i>
--------------------	---------------------------------

Description

Clear all current cached values. Set current environment cache empty.

Usage

```
mindat_cache_empty()
```

Examples

```
mindat_cache_empty()
```

mindat_cache_get	<i>Get cache value</i>
------------------	------------------------

Description

Retrieve the value of the cache named varname in current environment.

Usage

```
mindat_cache_get(varname)
```

Arguments

varname	string
---------	--------

Value

cached value. A string, list or other objects.

Examples

```
token<- mindat_cache_get('api_token')
```

mindat_cache_has	<i>Check if the current environment has the cached value of varname.</i>
------------------	--

Description

Check whether or not the current environment has the cache named varname.

Usage

```
mindat_cache_has(varname)
```

Arguments

varname string.

Value

Boolean value. if the varname is found in current environment cache, return True otherwise return False.

Examples

```
b_has <- mindat_cache_has('api_token')
```

mindat_cache_return_or_setup	<i>Check if the current environment has the cached function named varname.</i>
------------------------------	--

Description

Check whether the current environment has the cached function named varname,if has, return it. if not, setup up a new cache function named varname.

Usage

```
mindat_cache_return_or_setup(varname,setupfun)
```

Arguments

varname string.
 setupfun boolean, if the cached is a setup function.

Value

If the varname is found in current environment cache, return cached function. If not, eval the function and return cached function.

Examples

```
aep<- api_end_points<-mindat_cache_return_or_setup('api_end_points', function(){return (list()) })
```

mindat_cache_set	<i>Set cache name and value</i>
------------------	---------------------------------

Description

Assigns the value to the cache named varname in current environment.

Usage

```
mindat_cache_set(varname, value)
```

Arguments

varname	string. The cached varname.
value	string.

Examples

```
mindat_cache_set('api_token', "9ce67655d74bcd981e937be80dcea9cb")
```

mindat_connection	<i>Initializing Mindat API</i>
-------------------	--------------------------------

Description

Initializing API Call. Setup the base_url, token and format.

Usage

```
mindat_connection(token, base_url = "https://api.mindat.org/v1", page_size = 800)
```

Arguments

token	string. You can apply a token from Mindat.org.
base_url	string. The base url of mindat API, default is "https://api.mindat.org/v1".
page_size	integer, setting the page size of responded data from the API server.

Examples

```
mindat_connection("9ce67655d74bcd981e937be80dcea9cb", page_size = 1500)
```

mindat_countries	<i>mindat_countries</i>
------------------	-------------------------

Description

retrieve all countries list or the contries by given conditions.

Usage

```
mindat_countries (...)
```

Arguments

... Further named parameters.

Value

df, data frame of countries list

Examples

```
## Not run:  
df<- mindat_countries()  
  
## End(Not run)
```

mindat_country	<i>mindat_country</i>
----------------	-----------------------

Description

retrieve the country by given its id.

Usage

```
mindat_country (id,...)
```

Arguments

id country id in mindat.
... Further named parameters.

Value

df, a data frame of country

Examples

```
## Not run:  
df<- mindat_country(1)  
  
## End(Not run)
```

`mindat_crystalclasses` *mindat_crystalclasses*

Description

retrieve crystalclasses by its id

Usage

```
mindat_crystalclasses (id,...)
```

Arguments

<code>id</code>	crystalclasses id
<code>...</code>	Further named parameters.

Value

df, data frame of crystalclasses

Examples

```
## Not run:  
df<- mindat_crystalclasses(4)  
  
## End(Not run)
```

`mindat_crystalclass_list`
mindat_crystalclass_list

Description

retrieve all the crystalclasses list or the crystalclasses list by given conditions.

Usage

```
mindat_crystalclass_list(...)
```


Arguments

... Further named parameters.

Value

df, data frame of crystalclasses list

Examples

```
## Not run:  
df<- mindat_crystalclass_list()  
  
## End(Not run)
```

mindat_dana8_groups	<i>mindat_dana8_groups</i>
---------------------	----------------------------

Description

retrieve all the classifications of dana8.

Usage

```
mindat_dana8_groups(...)
```

Arguments

... Further named parameters.

Value

df, data frame of dana8 classification list

Examples

```
## Not run:  
df<- mindat_dana8_groups()  
  
## End(Not run)
```

```
mindat_dana8_subgroups  
mindat_dana8_subgroups
```

Description

retrieve all the subgroups of dana8.

Usage

```
mindat_dana8_subgroups(...)
```

Arguments

... Further named parameters.

Value

df, data frame of subgroups of dana8 classification.

Examples

```
## Not run:  
df<- mindat_dana8_subgroups()  
  
## End(Not run)
```

```
mindat_extract_response_body  
mindat_extract_response_body
```

Description

.

Usage

```
mindat_extract_response_body (response)
```

Arguments

response response json

Value

if status of the response is success (200), return the all_data_text (the content of response). Otherwise, report the errors.

Examples

```
## Not run:
library(httr)
uri<- "https://api.mindat.org/v1/geomaterials/?id_in=&hardness_min=9.3&fields=name,+
hardness&page_size=1500"
api_token<- "9ce67655d74bcd981e937be80dcea9cb"
response <- GET(uri,add_headers('Authorization'= paste('Token ',api_token,sep = "")))
raw_data <- mindat_extract_response_body(response)

## End(Not run)
```

mindat_geomaterial	<i>mindat_geomaterial</i>
--------------------	---------------------------

Description

retrieve geomaterial by its id

Usage

```
mindat_geomaterial (id,...)
```

Arguments

id	geomaterial id
...	Further named parameters.

Value

df, data frame of locality type list

Examples

```
## Not run:
df<- mindat_geomaterial(3337)

## End(Not run)
```

```
mindat_geomaterial_list
      mindat_geomaterial_list
```

Description

retrieve all the geomaterial list or the geomaterial by given conditions.

Usage

```
mindat_geomaterial_list(...)
```

Arguments

... Further named parameters.

Value

df, data frame of locality type list

Examples

```
## Not run:
df<- mindat_geomaterial_list()

## End(Not run)
```

```
mindat_geomaterial_search
      mindat_geomaterial_search
```

Description

retrieve all the geomaterial list or the geomaterial by given conditions.

Usage

```
mindat_geomaterial_search (...)
```

Arguments

... Further named parameters (e:Exact.If 1 returns only exact match;ima:if 1 returns only ima-approved minerals;size:limit of returned records).

Value

df, data frame of geomaterials match the search

Examples

```
## Not run:  
df<- mindat_geomaterial_search(q="Quartz")  
  
## End(Not run)
```

```
mindat_geomaterial_varieties  
mindat_geomaterial_varieties
```

Description

retrieve the geomaterial varieties by the id of geomaterial.

Usage

```
mindat_geomaterial_varieties (id,...)
```

Arguments

id	geomaterial id
...	Further named parameters.

Value

df, data frame of locality type list

Examples

```
## Not run:  
df<- mindat_geomaterial_varieties(3337)  
  
## End(Not run)
```

```
mindat_get_data_from_uri  
mindat_get_data_from_uri
```

Description

retrieve data from the uri.

Usage

```
mindat_get_data_from_uri (uri)
```

Arguments

uri request uri

Value

df. R data frame of the request uri.

Examples

```
## Not run:
library(httr)
uri <- "https://api.mindat.org/geomaterials/?id__in=&hardness_min=9.3&fields=name,+
hardness&page-size=1500"
mindat_cache_set('api_token',"9ce67655d74bcd981e937be80dcea9cb")
df <- mindat_get_data_from_uri(uri)

## End(Not run)
```

mindat_localities_list
mindat_localities_list

Description

retrieve localities list

Usage

mindat_localities_list (...)

Arguments

... Further named parameters.

Value

df. data frame of localities list.

Examples

```
## Not run:
df<- mindat_localities_list()

## End(Not run)
```

mindat_localitiy	<i>mindat_localitiy</i>
------------------	-------------------------

Description

retrieve locality by its id

Usage

```
mindat_localitiy (id,...)
```

Arguments

id	the mindat localitiy id
...	Further named parameters.

Value

df, data frame of locality

Examples

```
## Not run:  
df<- mindat_localitiy(3337)  
  
## End(Not run)
```

mindat_locality_status	<i>mindat_locality_status</i>
------------------------	-------------------------------

Description

retrieve all locality status by its id

Usage

```
mindat_locality_status (id,...)
```

Arguments

id	the mindat localitiy status id.
...	Further named parameters.

Value

df, data frame of locality status

Examples

```
## Not run:  
df<- mindat_locality_status(10)  
  
## End(Not run)
```

<code>mindat_locality_status_list</code>
<i><code>mindat_locality_status_list</code></i>

Description

retrieve all locality status list

Usage

```
mindat_locality_status_list (...)
```

Arguments

... Further named parameters.

Value

df, data frame of locality status list

Examples

```
## Not run:  
df<- mindat_locality_status_list()  
  
## End(Not run)
```

`mindat_locality_type` *mindat_locality_type*

Description

retrieve locality type by its id

Usage

```
mindat_locality_type (id,...)
```

Arguments

<code>id</code>	locality type id
<code>...</code>	Further named parameters.

Value

df, data frame of locality type list

Examples

```
## Not run:  
df<- mindat_locality_type(50)  
  
## End(Not run)
```

`mindat_locality_type_list`
 mindat_locality_type_list

Description

retrieve all locality type list

Usage

```
mindat_locality_type_list (...)
```

Arguments

<code>...</code>	Further named parameters.
------------------	---------------------------

Value

df, data frame of locality type list

Examples

```
## Not run:
df<- mindat_locality_type_list()

## End(Not run)
```

```
mindat_locentries_list
      mindat_locentries_list
```

Description

retrieve Mindat locentries. A 'locentry' is a record of specific geomaterial (mineral, etc) at a specific locality.

Usage

```
mindat_locentries_list (...)
```

Arguments

```
...          Further named parameters.
```

Value

df, data frame of locentries list.

Examples

```
## Not run:
df<- mindat_locentries_list()

## End(Not run)
```

```
mindat_locentries_lstm_id
      mindat_locentries_lstm_id
```

Description

retrieve Mindat locality-geomaterial pairs and some statistics list.

Usage

```
mindat_locentries_lstm_id (id,...)
```

Arguments

id A unique integer value identifying this locstatsmin.
... Further named parameters.

Value

df, data frame of locentries statistics list.

Examples

```
## Not run:  
df<- mindat_locentries_lstm_id(2)  
  
## End(Not run)
```

mindat_locentries_retrieve
mindat_locentries_retrieve

Description

retrieve Mindat locentries by a given ID. A 'locentry' is a record of specific geomaterial (mineral, etc) at a specific locality.

Usage

```
mindat_locentries_retrieve (id,...)
```

Arguments

id Locentry id (Integer)
... Further named parameters.

Value

df, data frame of locentries list.

Examples

```
## Not run:  
df<- mindat_locentries_retrieve(2)  
  
## End(Not run)
```

```
mindat_locentries_stat  
mindat_locentries_stat
```

Description

retrieve Mindat locality-geomaterial pairs and some statistics list.

Usage

```
mindat_locentries_stat (...)
```

Arguments

... Further named parameters.

Value

df, data frame of locentries statistics list.

Examples

```
## Not run:  
df<- mindat_locentries_stat()  
  
## End(Not run)
```

```
mindat_make_data_frame  
mindat_make_data_frame
```

Description

convert the response json to dataframe of R

Usage

```
mindat_make_data_frame (reg_list)
```

Arguments

reg_list response json data to list format obj.

Value

df_out, R data frame

Examples

```
## Not run:
id<- c('42155','9300','11282','48322')
name<- c('Cuarzo opalescente', 'Cupromagnesite', 'Cuprozippeite', 'Quartz-anorthosite')
ima_status <- c(0,0,0,0)
synid <- c(42133, 9281, 0, 0)
list_cvt <- list(id=id, name=name, ima_status=ima_status, synid=synid)
df<- mindat_make_data_frame(list_cvt)

## End(Not run)
```

<code>mindat_mineral_ima</code>	<code>mindat_mineral_ima</code>
---------------------------------	---------------------------------

Description

retrieve ima mineral by its id.

Usage

```
mindat_mineral_ima (id,...)
```

Arguments

<code>id</code>	mindat id
<code>...</code>	Further parameters.

Value

df. query results in data frame format.

Examples

```
## Not run:
df<- mindat_mineral_ima(3337)

## End(Not run)
```

```
mindat_mineral_ima_list  
mindat_mineral_ima_list
```

Description

retrieve ima mineral list

Usage

```
mindat_mineral_ima_list (...)
```

Arguments

... , Further named parameters.

Value

df, data frame of mineral list.

Examples

```
## Not run:  
df<- mindat_mineral_ima_list()  
  
## End(Not run)
```

```
mindat_nickel_strunz10_classes  
mindat_nickel_strunz10_classes
```

Description

retrieve the class list of Nickel-Strunz 10th edition classifications.

Usage

```
mindat_nickel_strunz10_classes(...)
```

Arguments

... Further named parameters.

Value

df, data frame of classes of Nickel-Strunz 10th edition classifications.

Examples

```
## Not run:  
df<- mindat_nickel_strunz10_classes()  
  
## End(Not run)
```

```
mindat_nickel_strunz10_families  
      mindat_nickel_strunz10_families
```

Description

retrieve the families list of Nickel-Strunz 10th edition classifications.

Usage

```
mindat_nickel_strunz10_families(...)
```

Arguments

```
...          Further named parameters.
```

Value

df, data frame of families of Nickel-Strunz 10th edition classifications.

Examples

```
## Not run:  
df<- mindat_nickel_strunz10_families()  
  
## End(Not run)
```

```
mindat_nickel_strunz10_subclasses  
      mindat_nickel_strunz10_subclasses
```

Description

retrieve the subclass list of Nickel-Strunz 10th edition classifications.

Usage

```
mindat_nickel_strunz10_subclasses(...)
```

Arguments

... Further named parameters.

Value

df, data frame of subclasses of Nickel-Strunz 10th edition classifications.

Examples

```
## Not run:
df<- mindat_nickel_strunz10_subclasses()

## End(Not run)
```

`mindat_parse_raw_data` *mindat_parse_raw_data*

Description

parse the raw response of json to dataframe of R. If the raw_data obtained from the response is paged, request all the pages and then add them into the df_out data frame.

Usage

```
mindat_parse_raw_data (raw_data)
```

Arguments

raw_data content of the response body

Value

df_out, R data frame

Examples

```
## Not run:
rd<-"{"count":5,"next":null,"previous":null,+
"results":[{"name":"Diamond"},{"name":"Khamrabaevite"},+
{"name":"Moissanite"},{"name":"Qingsongite"},{"name":"Uakitite"}]}"
df<- mindat_parse_raw_data(rd)

## End(Not run)
```

mindat_query	<i>mindat_query</i>
--------------	---------------------

Description

Basic function for query dataset at a specified endpoint.

Usage

```
mindat_query (endpoint, query = list())
```

Arguments

endpoint	query endpoint, e.g.'minerals_ima'.
query	list for query conditions.

Value

df query results in data frame format.

Examples

```
## Not run:  
df <-mindat_query("geomaterials_list",list(ids="",hardness_min=9))  
  
## End(Not run)
```

mindat_setup	<i>mindat_setup</i>
--------------	---------------------

Description

set up of the mindat basic uri, endpoints, and cache

Usage

```
mindat_setup(base_uri = 'https://api.mindat.org/v1',page_size = 800)
```

Arguments

base_uri	base uri of mindat API.
page_size	integer,setting the page size of responded data from the API server.

Examples

```
## Not run:
mindat_setup()

## End(Not run)
```

mindat_spacegroups	<i>mindat_spacegroups</i>
--------------------	---------------------------

Description

retrieve spacegroups by its id

Usage

```
mindat_spacegroups (id,...)
```

Arguments

id	spacegroups id
...	Further named parameters.

Value

df, data frame of spacegroups

Examples

```
## Not run:
df<- mindat_spacegroups(4)

## End(Not run)
```

mindat_spacegroupsets	<i>mindat_spacegroupsets</i>
-----------------------	------------------------------

Description

retrieve spacegroups by its id

Usage

```
mindat_spacegroupsets (id,...)
```

Arguments

id spacegroupsets id
... Further named parameters.

Value

df, data frame of spacegroupsets

Examples

```
## Not run:  
df<- mindat_spacegroupsets(4)  
  
## End(Not run)
```

mindat_spacegroupsets_list
mindat_spacegroupsets_list

Description

retrieve all the spacegroups list or the spacegroups list by given conditions.

Usage

```
mindat_spacegroupsets_list(...)
```

Arguments

... Further named parameters.

Value

df, data frame of spacegroupsets list

Examples

```
## Not run:  
df<- mindat_spacegroups_list()  
  
## End(Not run)
```

mindat_spacegroups_list	
	<i>mindat_spacegroups_list</i>

Description

retrieve all the spacegroups list or the spacegroups list by given conditions.

Usage

```
mindat_spacegroups_list(...)
```

Arguments

... Further named parameters.

Value

df, data frame of spacegroups list

Examples

```
## Not run:  
df<- mindat_spacegroups_list()  
  
## End(Not run)
```

minerals_ima_list	<i>minerals_ima_list</i>
-------------------	--------------------------

Description

retrieve all mineral ima list.

Usage

```
minerals_ima_list (...)
```

Arguments

... Further named parameters.

Details

This function is to retrieve the IMA minerals list.

Value

df, data frame of minerals.

Examples

```
## Not run:  
df <-minerals_ima_list()  
  
## End(Not run)
```

minerals_ima_list_expand
<i>minerals_ima_list_expand</i>

Description

retrieve mineral ima list with the given expand.

Usage

```
minerals_ima_list_expand (expand,...)
```

Arguments

expand	description
...	Further named parameters.

Details

This function is related to the filed "expand" of ima mineral. Items Enum: "~all" "*" "

Value

df, data frame of ima minerals with expanded fields.

Examples

```
## Not run:  
df <-minerals_ima_list_expand("~all")  
  
## End(Not run)
```

```
minerals_ima_list_ima  minerals_ima_list_ima
```

Description

retrieve mineral ima list with the given intValue.

Usage

```
minerals_ima_list_ima (intValue,...)
```

Arguments

intValue	Integer
...	Further named parameters.

Details

This function is related to the filed "ima" of ima minerals. Integer. 0: "PENDING_PUBLICATION"
1: "APPROVED"

Value

df, data frame of locality type.

Examples

```
## Not run:
df <-minerals_ima_list_ima(1)

## End(Not run)
```

```
minerals_ima_retrieve  minerals_ima_retrieve
```

Description

retrieve mineral ima by its id.

Usage

```
minerals_ima_retrieve (id,...)
```

Arguments

id	the mindat ima id
...	Further named parameters.

Details

This function is related to the filed "id" of ima minerals.

Value

df, data frame of ima mineral by a given id.

Examples

```
## Not run:
df <-minerals_ima_retrieve(3337)

## End(Not run)
```

minerals_ima_updated_at

retrieve the mineral_ima list updated at the given time.

Description

: Queries the list of mineral_ima that have the given time

Usage

```
minerals_ima_updated_at(updateDate,...)
```

Arguments

updateDate	string (date-time), Last updated datetime in format %Y-%m-%d %H:%M:%S
...	Further named parameters.Other optional arguments.

Details

This function is related to the filed "updated_at" of ima minerals. retrieve the localities that have the latest updated at the given time.

Value

df, a data frame of localities

Examples

```
## Not run:
df <-minerals_ima_updated_at("2020-11-10 10:12:20")

## End(Not run)
```

Nickel_strunz10_classes

Nickel-strunz10-classes

Description

: Queries a list of Nickel-Strunz 10th edition classifications.

Usage

```
Nickel_strunz10_classes(...)
```

Arguments

... Further parameters. Other optional arguments-Additional arguments.

Details

This function return a list of Nickel-Strunz-10 classes. case-insensitive

Value

df, a data frame of the classes of Nickel-Strunz 10th edition classifications.

Examples

```
## Not run:
df <- Nickel_strunz10_classes()

## End(Not run)
```

Nickel_strunz10_families

Nickel-strunz10-families

Description

: Queries a list of the families of Nickel-Strunz 10th edition classifications.

Usage

```
Nickel_strunz10_families(...)
```

Arguments

... Further parameters. Other optional arguments-Additional arguments.

Details

This function return a list of Nickel-Strunz-10 families

Value

df, a data frame of the families of Nickel-Strunz 10th edition classifications.

Examples

```
## Not run:  
df <-Nickel_strunz10_families()  
  
## End(Not run)
```

Nickel_strunz10_subclasses
<i>Nickel-strunz10-subclasses</i>

Description

: Queries a list of the subclasses of Nickel-Strunz 10th edition classifications.

Usage

```
Nickel_strunz10_subclasses(...)
```

Arguments

... Further parameters.Other optional arguments-Additional arguments.

Details

This function return a list of Nickel-Strunz-10 subclasses. case-insensitive

Value

df, a data frame of the subclasses of Nickel-Strunz 10th edition classifications.

Examples

```
## Not run:  
df <-Nickel_strunz10_subclasses()  
  
## End(Not run)
```

params_to_string	<i>params_to_string</i>
------------------	-------------------------

Description

Prase params to string,so that the query function can deal with the other exteranl condition set by the users.

Usage

```
params_to_string (params)
```

Arguments

params	convert params to string,which is used by the mindat query function.
--------	--

Value

str .

Examples

```
## Not run:
  params_to_string("")

## End(Not run)
```

saveMindatDataAs	<i>Output file as a given format</i>
------------------	--------------------------------------

Description

Save the mindat R dataframe to a specify format

Usage

```
saveMindatDataAs (inputdata,outputfname)
```

Arguments

inputdata	R dataframe of retrieved data from Mindat database.
outputfname	string. the output file name.

Examples

```
## Not run:
df <- geomaterials_search_name("Quartz")
saveMindatDataAs(df, "test.jsonld")

## End(Not run)
```

set_api_base	<i>set_api_base</i>
--------------	---------------------

Description

set base uri of current environment

Usage

```
set_api_base (api_base)
```

Arguments

api_base string. The base uri of mindat api.

Examples

```
set_api_base("9ce67655d74bcd981e937be80dcea9cb")
```

set_api_token	<i>set_api_token</i>
---------------	----------------------

Description

set the token of current environment

Usage

```
set_api_token (api_token)
```

Arguments

api_token string. The token of mindat api.

Examples

```
set_api_token("9ce67655d74bcd981e937be80dcea9cb")
```

set_page_size	<i>set_page_size</i>
---------------	----------------------

Description

set the page_size of response records.

Usage

set_page_size (page_size)

Arguments

page_size string. The token of mindat api.

Examples

set_page_size(800)

spacegroupsets_by_id	<i>spacegroupsets that match a given spacegroupets ID (integer)</i>
----------------------	---

Description

: Queries a list of spacegroupsets that match a given spacegroupets ID

Usage

spacegroupsets_by_id(spacegroupsets_id, ...)

Arguments

spacegroupsets_id integer spacegroup ID . The field "spacegroupets_id" is a integer of spacegroupets ID.

... Further parameters like "sgtext"(space group text) .Other optional arguments- Additional arguments.

Details

This function filter data by a given given spacegroup ID.

Value

df, a data frame of spacegroupsets

Examples

```
## Not run:
df <-spacegroupsets_by_id(2)

## End(Not run)
```

spacegroupsets_cclass *spacegroupsets that match a given crystalclass ID (integer)*

Description

: Queries a list of spacegroupsets that match a given crystalclass ID

Usage

```
spacegroupsets_cclass(crystalclass_id, ...)
```

Arguments

crystalclass_id	integer crystalclass ID . The field "crystalclass_id" is a integer of crystalclass ID.
...	Further parameters like "sgtext"(space group text) .Other optional arguments-Additional arguments.

Details

This function filter data by a given given crystalclass ID.

Value

df, a data frame of spacegroup

Examples

```
## Not run:
df <-spacegroupsets_cclass(2)

## End(Not run)
```

spacegroupsets_list *return a full list of spacegroupsets*

Description

: Queries a full list of spacegroupsets

Usage

```
spacegroupsets_list(...)
```

Arguments

... Further parameters like "sgtext"(space group text) .Other optional arguments-Additional arguments.

Details

This function return a full list of spacegroupsets.

Value

df, a data frame of the full list of spacegroupsets

Examples

```
## Not run:
df <-spacegroupsets_list()

## End(Not run)
```

spacegroupsets_sgtext *spacegroupsets that match a given sgtext (string)*

Description

: Queries a list of spacegroupsets that match a given sgtext (string)

Usage

```
spacegroupsets_sgtext(sgtext, ...)
```

Arguments

sgtext string space group text (case-insensitive). The field "sgtext" is a string of space group text.

... Further parameters like "cclass"(Crystalclass) .Other optional arguments-Additional arguments.

Details

This function filter data by a given given crystalclass ID.

Value

df, a data frame of spacegroupsets

Examples

```
## Not run:
df <-spacegroupsets_sgtext("P1")

## End(Not run)
```

spacegroups_by_id	<i>spacegroups that match a given spacegroup ID (integer)</i>
-------------------	---

Description

: Queries a list of spacegroup that match a given spacegroup ID

Usage

```
spacegroups_by_id(spacegroup_id, ...)
```

Arguments

spacegroup_id	integer spacegroup ID . The field "spacegroup_id" is a integer of spacegroup ID.
...	Further parameters like "sgtext"(space group text) .Other optional arguments-Additional arguments.

Details

This function filter data by a given given spacegroup ID.

Value

df, a data frame of spacegroup

Examples

```
## Not run:
df <-spacegroups_by_id(2)

## End(Not run)
```

spacegroups_cclass	<i>spacegroups that match a given crystalclass ID (integer)</i>
--------------------	---

Description

: Queries a list of spacegroup that match a given crystalclass ID

Usage

```
spacegroups_cclass(crystalclass_id, ...)
```

Arguments

crystalclass_id	integer crystalclass ID . The field "crystalclass_id" is a integer of crystalclass ID.
...	Further parameters like "sgtext"(space group text) .Other optional arguments-Additional arguments.

Details

This function filter data by a given given crystalclass ID.

Value

df, a data frame of spacegroup

Examples

```
## Not run:
df <-spacegroups_cclass(2)

## End(Not run)
```

spacegroups_list	<i>return a full list of spacegroups</i>
------------------	--

Description

: Queries a full list of spacegroup

Usage

```
spacegroups_list(...)
```


Arguments

... Further parameters like "sgtext"(space group text) .Other optional arguments-Additional arguments.

Details

This function return a full list of spacegroups.

Value

df, a data frame of the full list of spacegroups

Examples

```
## Not run:
df <-spacegroups_list()

## End(Not run)
```

spacegroups_sgtext	<i>spacegroups that match a given sgtext (string)</i>
--------------------	---

Description

: Queries a list of spacegroups that match a given sgtext (string)

Usage

```
spacegroups_sgtext(sgtext, ...)
```

Arguments

sgtext string space group text (case-insensitive). The field "sgtext" is a string of space group text.

... Further parameters like "cclass"(Crystalclass) .Other optional arguments-Additional arguments.

Details

This function filter data by a given given crystalclass ID.

Value

df, a data frame of spacegroups

Examples

```
## Not run:  
df <-spacegroups_sgttext("P1")  
  
## End(Not run)
```

Index

ConvertDF2JsonLD, [4](#)
ConvertDF2TTL, [5](#)
crystalclasses_symbols, [6](#)
crystalclasses_systems, [6](#)

Dana8_groups, [7](#)
Dana8_subgroups, [8](#)

geomaterials_bi_greater_than, [8](#)
geomaterials_bi_less_than, [9](#)
geomaterials_bi_range, [10](#)
geomaterials_by_groupid, [10](#)
geomaterials_cleavagetype, [11](#)
geomaterials_colour, [12](#)
geomaterials_contain_all_but_not_elems, [13](#)
geomaterials_contain_all_elems, [14](#)
geomaterials_contain_any_but_not_elems, [14](#)
geomaterials_contain_any_elems, [15](#)
geomaterials_contain_only_elems, [16](#)
geomaterials_crystal_system, [17](#)
geomaterials_dens_greater_than, [18](#)
geomaterials_dens_less_than, [18](#)
geomaterials_dens_range, [19](#)
geomaterials_diapheny, [20](#)
geomaterials_entrytype, [21](#)
geomaterials_expand, [21](#)
geomaterials_field_exists, [22](#)
geomaterials_fracturetype, [23](#)
geomaterials_hardness_gt, [24](#)
geomaterials_hardness_lt, [24](#)
geomaterials_hardness_range, [25](#)
geomaterials_ima, [26](#)
geomaterials_ima_notes, [27](#)
geomaterials_ima_status, [28](#)
geomaterials_lustretype, [28](#)
geomaterials_meteoritical_code, [29](#)
geomaterials_name, [30](#)
geomaterials_not_contain_elems, [31](#)
geomaterials_optical2v_max, [31](#)
geomaterials_optical2v_min, [32](#)
geomaterials_optical2v_range, [33](#)
geomaterials_opticalsign, [34](#)
geomaterials_opticaltype, [34](#)
geomaterials_polytypeof, [35](#)
geomaterials_ri_gt, [36](#)
geomaterials_ri_lt, [37](#)
geomaterials_ri_range, [37](#)
geomaterials_search_name, [38](#)
geomaterials_streak, [39](#)
geomaterials_synid, [40](#)
geomaterials_updated_at, [40](#)
geomaterials_varietyof, [41](#)
geomeaterials_non_utf, [42](#)
geomeaterials_ordering, [43](#)
getExtension, [44](#)

localities_list_all, [44](#)
localities_list_country, [45](#)
localities_list_description, [46](#)
localities_list_elems_exc, [47](#)
localities_list_elems_inc, [47](#)
localities_list_elems_inc_exc, [48](#)
localities_list_expand, [49](#)
localities_list_txt, [50](#)
localities_list_updated_at, [50](#)
localities_retrieve_id, [51](#)
localities_status_list, [52](#)
localities_status_retrieve, [53](#)
localitiy_type_retrieve, [53](#)
locality_age, [54](#)
locality_age_list, [55](#)
locality_type_list, [55](#)
Locentries_list, [56](#)
Locentries_retrieve, [57](#)
Locentries_stat_retrieve, [58](#)
Locentries_statistics_list, [57](#)

mindat_build_querystring, [59](#)

mindat_cache_delete, [59](#)
mindat_cache_empty, [60](#)
mindat_cache_get, [60](#)
mindat_cache_has, [61](#)
mindat_cache_return_or_setup, [61](#)
mindat_cache_set, [62](#)
mindat_connection, [62](#)
mindat_countries, [63](#)
mindat_country, [63](#)
mindat_crystalclass_list, [64](#)
mindat_crystalclasses, [64](#)
mindat_dana8_groups, [65](#)
mindat_dana8_subgroups, [66](#)
mindat_extract_response_body, [66](#)
mindat_geomaterial, [67](#)
mindat_geomaterial_list, [68](#)
mindat_geomaterial_search, [68](#)
mindat_geomaterial_varieties, [69](#)
mindat_get_data_from_uri, [69](#)
mindat_localities_list, [70](#)
mindat_locality, [71](#)
mindat_locality_status, [71](#)
mindat_locality_status_list, [72](#)
mindat_locality_type, [73](#)
mindat_locality_type_list, [73](#)
mindat_locentries_list, [74](#)
mindat_locentries_lstm_id, [74](#)
mindat_locentries_retrieve, [75](#)
mindat_locentries_stat, [76](#)
mindat_make_data_frame, [76](#)
mindat_mineral_ima, [77](#)
mindat_mineral_ima_list, [78](#)
mindat_nickel_strunz10_classes, [78](#)
mindat_nickel_strunz10_families, [79](#)
mindat_nickel_strunz10_subclasses, [79](#)
mindat_parse_raw_data, [80](#)
mindat_query, [81](#)
mindat_setup, [81](#)
mindat_spacegroups, [82](#)
mindat_spacegroups_list, [84](#)
mindat_spacegroupsets, [82](#)
mindat_spacegroupsets_list, [83](#)
minerals_ima_list, [84](#)
minerals_ima_list_expand, [85](#)
minerals_ima_list_ima, [86](#)
minerals_ima_retrieve, [86](#)
minerals_ima_updated_at, [87](#)

Nickel_strunz10_classes, [88](#)

Nickel_strunz10_families, [88](#)
Nickel_strunz10_subclasses, [89](#)

params_to_string, [90](#)

saveMindatDataAs, [90](#)
set_api_base, [91](#)
set_api_token, [91](#)
set_page_size, [92](#)
spacegroups_by_id, [95](#)
spacegroups_cclass, [96](#)
spacegroups_list, [96](#)
spacegroups_sgtext, [97](#)
spacegroupsets_by_id, [92](#)
spacegroupsets_cclass, [93](#)
spacegroupsets_list, [94](#)
spacegroupsets_sgtext, [94](#)