Package ‘ONETr’

August 25, 2015

Type Package
Title Efficient Authenticated Interaction with the O*NET API
Version 1.0.3
Date 2015-08-23
Author Eric Knudsen
Maintainer Eric Knudsen <eknudsen@gc.cuny.edu>
Description Provides a series of functions designed to enable users to easily search and interact with occupational data from the O*NET API <www.onetonline.org>. The package produces parsed and listed XML data for custom interactions, or pre-packaged functions for easy extraction of specific data (e.g., Knowledge, Skills, Abilities, Work Styles, etc.).
Depends XML, RCurl, plyr
License GPL-3
NeedsCompilation no
Repository CRAN
Date/Publication 2015-08-25 01:01:23

R topics documented:

abilities ................................................. 2
cacheEnv ............................................. 3
education .............................................. 3
interests .............................................. 4
jobData ................................................. 4
jobData2 ............................................... 5
jobTitles .............................................. 6
jobZone ............................................... 6
keySearch ............................................ 7
knowledge ............................................ 8
occupation ........................................... 9
onetr ............................................... 9
relatedOccupations ................................ 10
setCreds ............................................ 11
abilities

Description

This function should be used after a socSearch has been stored. The function extracts ability information for the searched/stored occupation.

Usage

abilities(list)

Arguments

list the name of the list object that the socSearch data has been stored in

Value

A data frame with relevant data.

Note

May not work if data are not properly formatted.

Author(s)

Eric Knudsen

Examples

```r
## Not run:
# You need to set your credentials with setCreds() prior to use.
abilities(jobData)

## End(Not run)
```
cacheEnv

Environment housing API credentials

Description
This environment houses API credentials set with setCreds. It is accessed by keySearch and socSearch.

Usage
cacheEnv

Format
Environment.

education Pull education data from job list

Description
This function should be used after a socSearch has been stored. The function extracts education information for the searched/stored occupation.

Usage
education(list)

Arguments
list the name of the list object that the socSearch data has been stored in

Value
A data frame with relevant data.

Note
May not work if data are not properly formatted.

Author(s)
Eric Knudsen

Examples
data(jobData)
# You need to set your credentials with setCreds() prior to use.
education(jobData)
interests
Pull interest data from job list

Description
This function should be used after a socSearch has been stored. The function extracts interest information for the searched/stored occupation.

Usage
interests(list)

Arguments
list the name of the list object that the socSearch data has been stored in

Value
A data frame with relevant data.

Note
May not work if data are not properly formatted.

Author(s)
Eric Knudsen

Examples
data(jobData)
# You need to set your credentials with setCreds() prior to use.
interests(jobData)

jobData Sample Job Data for Clinical Psychologist

Description
This data set contains job data for "Clinical Psychologist". It is the direct output of a socSearch using the O*NET SOC code 19-3031.02, and is parsed into a list for efficient access by all package functions.

Usage
jobData
Description

This data set contains job data for 'Physical Therapist Aide'. It is the direct output of a socSearch using the O*NET SOC code 31-2022.00, and is parsed into a list for efficient access by all package functions.

Usage

jobData2

Format

A list of length 16.

Source

O*NET Online.

References

O*NET OnLine. National Center for O*NET Development.
jobTitles  

_PULL job title data from job list_

**Description**

This function should be used after a socSearch has been stored. The function extracts job title information for the searched/stored occupation.

**Usage**

```r
jobTitles(list)
```

**Arguments**

- `list`  
  the name of the list object that the socSearch data has been stored in

**Value**

A data frame with relevant data.

**Note**

May not work if data are not properly formatted.

**Author(s)**

Eric Knudsen

**Examples**

```r
data(jobData)  
# You need to set your credentials with setCreds() prior to use.  
jobTitles(jobData)
```

---

jobZone  

_PULL "Job Zone" data from job list_

**Description**

This function should be used after a socSearch has been stored. The function extracts "Job Zone" information for the searched/stored occupation.

**Usage**

```r
jobZone(list)
```
keySearch

Arguments

list

the name of the list object that the socSearch data has been stored in

Value

A data frame with relevant data.

Note

May not work if data are not properly formatted.

Author(s)

Eric Knudsen

Examples

data(jobData)

# You need to set your credentials with setCreds() prior to use.
jobZone(jobData)

keySearch(keyword)

Search O*NET by keyword

Description

This function allows you to search O*NET occupations using a keyword, and receive the results in a data frame.

Usage

keySearch(keyword)

Arguments

keyword

an occupational keyword you’d like to query the API with

Value

A data frame containing the search results.

Note

May not work if data are not properly formatted.

Author(s)

Eric Knudsen
Examples

```r
## Not run:
    # You need to set your credentials with setCreds() prior to use.
    keySearch("psychologist")

## End(Not run)
```

---

**knowledge**  
Pull knowledge data from job list

Description

This function should be used after a `socSearch` has been stored. The function extracts knowledge information for the searched/stored occupation.

Usage

```r
knowledge(list)
```

Arguments

- `list`  
  the name of the list object that the `socSearch` data has been stored in

Value

A data frame with relevant data.

Note

May not work if data are not properly formatted.

Author(s)

Eric Knudsen

Examples

```r
data(jobData)
    # You need to set your credentials with setCreds() prior to use.
    knowledge(jobData)
```
occupation

Pull occupation data from job list

Description

This function should be used after a socSearch has been stored. The function extracts occupation information for the searched/stored occupation.

Usage

occupation(list)

Arguments

list the name of the list object that the socSearch data has been stored in

Value

A data frame with relevant data.

Note

May not work if data are not properly formatted.

Author(s)

Eric Knudsen

Examples

data(jobData)
  # You need to set your credentials with setCreds() prior to use.
  occupation(jobData)

onetr

Efficient authenticated interaction with the O*NET API.

Description

This package provides a series of functions designed to enable users to easily search and interact with occupational data from the O*NET API <www.onetonline.org>. The package produces parsed and listed XML data for custom interactions, or pre-packaged functions for easy extraction of specific data (e.g., Knowledge, Skills, Abilities, Work Styles, etc.).
relatedOccupations

Details
This package should be used to explore or extract specific occupational data from the O*NET API. The setCreds function should be called with the proper arguments prior to the use of any other package functions- the function stores one’s API credentials for use by the other functions throughout the session. keySearch allows a search by keyword (e.g., "psychologist") and prints the search results, from which occupational SOC codes can be extracted. SOC codes can then be used with socSearch to print or store data about a specific occupation. For a list of functions designed for extract of specific data points (e.g., Knowledge, Skills, Abilities, etc.), please read the documentation and explore the package.

Author(s)
Eric Knudsen
Maintainer: Eric Knudsen <eknudsen@gc.cuny.edu>

References
http://www.onetonline.org/

Examples
```r
## Not run:
ssetCreds("username","password") # must have O*NET API developer account
keySearch("psychologist")
socSearch("19-3031.02")
## End(Not run)
```

relatedOccupations
Pull related occupations data from job list

Description
This function should be used after a socSearch has been stored. The function extracts related occupations information for the searched/stored occupation.

Usage
```r
relatedOccupations(list)
```

Arguments
- `list` the name of the list object that the socSearch data has been stored in

Value
A data frame with relevant data.
setCreds

Note
May not work if data are not properly formatted.

Author(s)
Eric Knudsen

Examples
data(jobData)
  # You need to set your credentials with setCreds() prior to use.
  relatedOccupations(jobData)

---

setCreds Set O*NET API credentials for functional use

Description
This function allows you to store your O*NET API HTTPS credentials for easy authentication when calling package functions. This function must be used before any other function in the package.

Usage
setCreds(user, pass)

Arguments
user O*NET API developer username (for the HTTPS API)
pass O*NET API developer password (for the HTTPS API)

Value
An list to store the API username and password for access by the package functions.

Author(s)
Eric Knudsen

Examples
# store API username and password
setCreds("sampleuser","samplepassword")
**sim.index**

*Similarity indices for job attributes*

**Description**

Computes the Sorensen-Dice and/or Jaccard indices of similarity between two jobs on the named data type (e.g., knowledge, skills, etc.).

**Usage**

```r
sim.index(list1, list2, FUN, index=c("sd", "ji", "all"))
```

**Arguments**

- `list1`: list object (from `socSearch`) of the first job
- `list2`: list object (from `socSearch`) of the second job
- `FUN`: job data type to compare (e.g., knowledge)
- `index`: the preferred index of similarity (Sorensen-Dice and/or Jaccard). Can use "all" to compute both.

**Value**

A list of the computed indices

**Note**

May not work if data are not properly formatted.

**Author(s)**

Eric Knudsen

**Examples**

```r
data(jobData)
data(jobData2)
sim.index(jobData, jobData2, knowledge, index="all")
```
skills

Pull skill data from job list

Description
This function should be used after socSearch has been stored. The function extracts skill information for the searched/stored occupation.

Usage
skills(list)

Arguments
list the name of the list object that the socSearch data has been stored in

Value
A data frame with relevant data.

Note
May not work if data are not properly formatted.

Author(s)
Eric Knudsen

Examples
data(jobData)
# You need to set your credentials with setCreds() prior to use.
skills(jobData)

socSearch

Searches and pulls occupational data based on SOC code

Description
This function should be used to extract and store data on a specific job for further analysis/manipulation by package functions.

Usage
socSearch(soc)
### tasks

**Description**

This function should be used after a `socSearch` has been stored. The function extracts task information for the searched/stored occupation.

**Usage**

```r
tasks(list)
```

**Arguments**

- `list` the name of the list object that the `socSearch` data has been stored in

**Value**

A data frame with relevant data.

**Note**

May not work if data are not properly formatted.
**technology**

**Author(s)**

Eric Knudsen

**Examples**

```r
data(jobData)
# You need to set your credentials with setCreds() prior to use.
tasks(jobData)
```

---

**Description**

This function should be used after a socSearch has been stored. The function extracts technology information for the searched/stored occupation.

**Usage**

```r
technology(list)
```

**Arguments**

- `list` the name of the list object that the socSearch data has been stored in

**Value**

A data frame with relevant data.

**Note**

May not work if data are not properly formatted.

**Author(s)**

Eric Knudsen

**Examples**

```r
data(jobData)
# You need to set your credentials with setCreds() prior to use.
technology(jobData)
```
tools

Pull tools data from job list

Description
This function should be used after a socSearch has been stored. The function extracts tools information for the searched/stored occupation.

Usage
tools(list)

Arguments
list

the name of the list object that the socSearch data has been stored in

Value
A data frame with relevant data.

Note
May not work if data are not properly formatted.

Author(s)
Eric Knudsen

Examples
data(jobData)
  # You need to set your credentials with setCreds() prior to use.
  tools(jobData)

workActivities

Pull work activity data from job list

Description
This function should be used after a socSearch has been stored. The function extracts work activity information for the searched/stored occupation.

Usage
workActivities(list)
workcontext

Arguments

list the name of the list object that the socSearch data has been stored in

Value

A data frame with relevant data.

Note

May not work if data are not properly formatted.

Author(s)

Eric Knudsen

Examples

data(jobData)
  # You need to set your credentials with setCreds() prior to use.
  workactivities(jobData)

workcontext Pull work context data from job list

Description

This function should be used after a socSearch has been stored. The function extracts work context information for the searched/stored occupation.

Usage

workcontext(list)

Arguments

list the name of the list object that the socSearch data has been stored in

Value

A data frame with relevant data.

Note

May not work if data are not properly formatted.

Author(s)

Eric Knudsen
workStyles

Examples

data(jobData)
  # You need to set your credentials with setCreds() prior to use.
  workContext(jobData)

workStyles

Description

This function should be used after a socSearch has been stored. The function extracts work style information for the searched/stored occupation.

Usage

workStyles(list)

Arguments

list

the name of the list object that the socSearch data has been stored in

Value

A data frame with relevant data.

Note

May not work if data are not properly formatted.

Author(s)

Eric Knudsen

Examples

data(jobData)
  # You need to set your credentials with setCreds() prior to use.
  workStyles(jobData)
workValues

Pull work value data from job list

Description

This function should be used after a `socSearch` has been stored. The function extracts work value information for the searched/stored occupation.

Usage

```r
workValues(list)
```

Arguments

- `list` the name of the list object that the `socSearch` data has been stored in

Value

A data frame with relevant data.

Note

May not work if data are not properly formatted.

Author(s)

Eric Knudsen

Examples

```r
data(jobData)
# You need to set your credentials with setCreds() prior to use.
workValues(jobData)
```
Index

* Topic `\texttt{abilities}`
  abilities, 2
* Topic `\texttt{authentication}`
  setCreds, 11
* Topic `\texttt{credentials}`
  setCreds, 11
* Topic `\texttt{education}`
  education, 3
* Topic `\texttt{interests}`
  interests, 4
* Topic `\texttt{jobtitles}`
  jobTitles, 6
* Topic `\texttt{jobzone}`
  jobZone, 6
* Topic `\texttt{keyword}`
  keySearch, 7
* Topic `\texttt{knowledge}`
  knowledge, 8
* Topic `\texttt{occupation}`
  occupation, 9
* Topic `\texttt{relatedoccupations}`
  relatedOccupations, 10
* Topic `\texttt{research}`
  research, 7
  socSearch, 13
* Topic `\texttt{sim.index}`
  sim.index, 12
* Topic `\texttt{skills}`
  skills, 13
* Topic `\texttt{soccode}`
  socSearch, 13
* Topic `\texttt{tasks}`
  tasks, 14
* Topic `\texttt{technology}`
  technology, 15
* Topic `\texttt{tools}`
  tools, 16
* Topic `\texttt{workactivities}`
  workActivities, 16
* Topic `\texttt{workcontext}`
  workContext, 17
* Topic `\texttt{workstyles}`
  workStyles, 18
* Topic `\texttt{values}`
  workValues, 19
* Topic `\texttt{datasets}`
  jobData, 4
  jobData2, 5
* Topic `\texttt{environment}`
  cacheEnv, 3
* Topic `\texttt{jobs}`
  onetr, 9
* Topic `\texttt{occupations}`
  onetr, 9
* Topic `\texttt{package}`
  onetr, 9

  abilities, 2
  cacheEnv, 3
  education, 3
  interests, 4
  jobData, 4
  jobData2, 5
  jobTitles, 6
  jobZone, 6
  keySearch, 7
  knowledge, 8
  occupation, 9
  onetr, 9
  relatedOccupations, 10
  setCreds, 11
INDEX

sim.index, 12
skills, 13
socSearch, 13

tasks, 14
technology, 15
tools, 16

workActivities, 16
workContext, 17
workStyles, 18
workValues, 19