

# Package ‘ONETr’

September 13, 2014

**Type** Package

**Title** Efficient authenticated interaction with the O\*NET API.

**Version** 1.0

**Date** 2014-09-08

**Author** Eric Knudsen

**Maintainer** Eric Knudsen <eknudsen@gc.cuny.edu>

## 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.).

**Depends** XML, RCurl

**License** GPL-3

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2014-09-13 07:55:41

## R topics documented:

abilities	2
cacheEnv	3
education	3
interests	4
jobData	5
jobTitles	5
jobZone	6
keySearch	7
knowledge	7

occupation . . . . .	8
onetr . . . . .	9
relatedOccupations . . . . .	10
setCreds . . . . .	10
skills . . . . .	11
socSearch . . . . .	12
tasks . . . . .	12
technology . . . . .	13
tools . . . . .	14
workActivities . . . . .	15
workContext . . . . .	15
workStyles . . . . .	16
workValues . . . . .	17
<b>Index</b>	<b>18</b>

---

abilities	<i>Pull ability data from job extract stored as a list.</i>
-----------	---

---

### 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**

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

---

cacheEnv	<i>Environment housing API credentials</i>
----------	--

---

**Description**

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

**Usage**

```
cacheEnv
```

**Format**

Environment.

---

education	<i>Pull education data from job extract stored as a list.</i>
-----------	---

---

**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 extract stored as a 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`**Format**

A list of length 15.

**Source**

O\*NET Online.

**References**

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

---

`jobTitles`*Pull job title data from job extract stored as a 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**`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**

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

---

jobZone

*Pull "Job Zone" data from job extract stored as a 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**

```
jobZone(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.
jobZone(jobData)
```

---

keySearch	<i>Search O*NET by keyword.</i>
-----------	---------------------------------

---

**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**

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

---

knowledge	<i>Pull knowledge data from job extract stored as a list.</i>
-----------	---

---

**Description**

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

**Usage**

```
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**

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

---

occupation

*Pull occupation data from job extract stored as a 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](http://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.).

## 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](mailto:eknudsen@gc.cuny.edu)>

## References

<http://www.onetonline.org/>

## Examples

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

## End(Not run)
```

---

relatedOccupations	<i>Pull related occupations data from job extract stored as a list.</i>
--------------------	---

---

**Description**

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

**Usage**

```
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.

**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	<i>Set O*NET API credentials for functional use.</i>
----------	--

---

**Description**

This function allows you to store your O\*NET API developer 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  
pass            O\*NET API developer password

**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")
```

---

skills                            *Pull skill data from job extract stored as a 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	<i>Searches and pulls occupational data based on SOC code.</i>
-----------	--

---

**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)
```

**Arguments**

soc                    occupation SOC code (if necessary, use keySearch to find SOC code)

**Value**

A list (parsed from XML) of all existing O\*NET data on queried occupation.

**Note**

May not work if data are not properly formatted.

**Author(s)**

Eric Knudsen

**Examples**

```
## Not run:  
# You need to set your credentials with setCreds() prior to use.  
socSearch("19-3031.02")  
  
## End(Not run)
```

---

tasks	<i>Pull task data from job extract stored as a list.</i>
-------	--

---

**Description**

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

**Usage**

```
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.

**Author(s)**

Eric Knudsen

**Examples**

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

---

`technology`

*Pull technology data from job extract stored as a list.*

---

**Description**

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

**Usage**

```
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**

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

---

**tools***Pull tools data from job extract stored as a 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	<i>Pull work activity data from job extract stored as a list.</i>
----------------	---

---

**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)
```

**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	<i>Pull work context data from job extract stored as a list.</i>
-------------	--

---

**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

**Examples**

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

---

workStyles

*Pull work style data from job extract stored as a list.*

---

**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 extract stored as a 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**

```
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**

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

# Index

- \*Topic **\textasciitildeabilities**
    - abilities, [2](#)
  - \*Topic **\textasciitildeauthentication**
    - setCreds, [10](#)
  - \*Topic **\textasciitildecredentials**
    - setCreds, [10](#)
  - \*Topic **\textasciitildeeducation**
    - education, [3](#)
  - \*Topic **\textasciitildeinterests**
    - interests, [4](#)
  - \*Topic **\textasciitildejobtitles**
    - jobTitles, [5](#)
  - \*Topic **\textasciitildejobzone**
    - jobZone, [6](#)
  - \*Topic **\textasciitildekeyword**
    - keySearch, [7](#)
  - \*Topic **\textasciitildeknowledge**
    - knowledge, [7](#)
  - \*Topic **\textasciitildeoccupation**
    - occupation, [8](#)
  - \*Topic **\textasciitilderelatedoccupations**
    - relatedOccupations, [10](#)
  - \*Topic **\textasciitildesearch**
    - keySearch, [7](#)
    - socSearch, [12](#)
  - \*Topic **\textasciitildeskills**
    - skills, [11](#)
  - \*Topic **\textasciitildesocode**
    - socSearch, [12](#)
  - \*Topic **\textasciitildetasks**
    - tasks, [12](#)
  - \*Topic **\textasciitildetechnology**
    - technology, [13](#)
  - \*Topic **\textasciitildetools**
    - tools, [14](#)
  - \*Topic **\textasciitildeworkactivities**
    - workActivities, [15](#)
  - \*Topic **\textasciitildeworkcontext**
    - workContext, [15](#)
  - \*Topic **\textasciitildeworkstyles**
    - workStyles, [16](#)
  - \*Topic **\textasciitildeworkvalues**
    - workValues, [17](#)
  - \*Topic **datasets**
    - jobData, [5](#)
  - \*Topic **environment**
    - cacheEnv, [3](#)
  - \*Topic **jobs**
    - onetr, [9](#)
  - \*Topic **occupations**
    - onetr, [9](#)
  - \*Topic **package**
    - onetr, [9](#)
- abilities, [2](#)
- cacheEnv, [3](#)
- education, [3](#)
- interests, [4](#)
- jobData, [5](#)
- jobTitles, [5](#)
- jobZone, [6](#)
- keySearch, [7](#)
- knowledge, [7](#)
- occupation, [8](#)
- onetr, [9](#)
- relatedOccupations, [10](#)
- setCreds, [10](#)
- skills, [11](#)
- socSearch, [12](#)
- tasks, [12](#)

technology, [13](#)  
tools, [14](#)

workActivities, [15](#)  
workContext, [15](#)  
workStyles, [16](#)  
workValues, [17](#)