

# Package ‘tibbrConnector’

December 15, 2016

**Type** Package

**Title** R Interface to TIBCO 'tibbr'

**Version** 1.5.1

**Imports** methods, utils, RCurl, rjson

**Author** TIBCO Software Inc.

**Maintainer** Stephen Kaluzny <skaluzny@tibco.com>

**Description** Post messages to tibbr from within R.

**License** BSD\_3\_clause + file LICENSE

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2016-12-15 11:03:27

## R topics documented:

tibbr.connect . . . . .	1
tibbr.post . . . . .	3
tibbr.subject . . . . .	4
tibbr.user . . . . .	5
<b>Index</b>	<b>7</b>

---

tibbr.connect	<i>Connect To tibbr</i>
---------------	-------------------------

---

## Description

Creates a connection to the tibbr service.

**Usage**

```

tibbr.connect(server, user, password)
tibbr.disconnect(connection)

## S3 method for class 'tibbrConnection'
print(x, ...)

```

**Arguments**

server	a character string containing the hostname of the tibbr instance to connect to. Do not include the http:// or https:// prefix.
user	a character string specifying the name of the tibbr user to connect as.
password	a character string containing the password of the tibbr user to connect as.
connection	a tibbrConnection object from which to disconnect.
x	an object that inherits from class "tibbrConnection".
...	additional arguments passed to the print method.

**Details**

Errors encountered during the tibbr login or logout process are raised by these functions.

**Value**

`tibbr.connect` returns a `tibbrConnection` object containing the following items:

- `server` – The name of the server to which this object is connected.
- `user` – The authentication information returned by the tibbr API.
- `clientkey` – The tibbr API client key that references the authenticated session this object represents.

`tibbr.disconnect` returns TRUE invisibly.

**See Also**

[tibbr.post](#)

**Examples**

```

## Not run:
con <- tibbr.connect("try.tibbr.com", "myusername", "mypassword")
con
tibbr.disconnect(con)

## End(Not run)

```

---

tibbr.post	<i>Post A Message To tibbr</i>
------------	--------------------------------

---

**Description**

Post a message to tibbr. The post can include links or attached files.

**Usage**

```
tibbr.post(connection, content, subject = NULL, links = character(0),
           attachments = character(0))
```

**Arguments**

connection	a tibbrConnection object indicating the tibbr instance to post to.
content	a character string containing the content of the message to post.
subject	a character vector or list containing any combination of character strings containing the fully-qualified subject names (including the subject names of all parent subjects, separated by .) of subjects, tibbrSubject objects, and tibbrUser objects to which this message should post. If NULL, the message is posted to the wall of the user specified in the creation of the connection. Note that subject names consist of alphanumeric characters and the special characters ., -, and _, and may optionally start with the subject indicator character @.
links	a character vector containing URLs to link to in this post.
attachments	a character vector specifying the names of files to attach to this post.

**Details**

Errors encountered during the tibbr posting process will be raised by this function.

**Value**

returns TRUE invisibly.

**See Also**

[tibbr.connect](#), [tibbr.subject](#), [tibbr.user](#)

**Examples**

```
## Not run:
con <- tibbr.connect("try.tibbr.com", "myusername", "mypassword")
tibbr.post(con, "Hello, tibbr!") # Posts to your wall
tibbr.post(con, "Hello, salesmen!", subject="Sales")
tibbr.post(con, "Hello, everyone.",
           subject=c("Sales", "Marketing", "Engineering.ProjectAlpha"))
```

```
tibbr.post(con, "Here's that proposal.", subject="Sales",
           links="http://sales/Alpha_Proposal/",
           attachments=c("AlphaCoverLetter.docx", "AlphaProjections.csv"))
alpha <- tibbr.subject(con, "alpha")
tibbr.post(con, "To everyone involved in the Alpha project: Good job!", subject=alpha)

## End(Not run)
```

---

tibbr.subject	<i>Search tibbr For Subjects</i>
---------------	----------------------------------

---

## Description

Searches tibbr for subjects.

## Usage

```
tibbr.subject(connection, subject = NULL)
```

## Arguments

connection	a tibbrConnection object indicating the tibbr instance to search.
subject	a character string containing the search criteria. If NULL, tibbr returns all subjects on the tibbr instance. (Note: This can take a long time to return, depending on the number of subjects on your tibbr instance.)

tibbr searches the following fields for matches:

- Subject name
- Subject display name
- Subject description

## Details

Errors encountered during the search process are raised by this function.

## Value

returns a list of tibbrSubject objects describing the results of the search. The names of the tibbrSubject objects represent attributes of the corresponding subject in tibbr.

## See Also

[tibbr.user](#)

## Examples

```
## Not run:
con <- tibbr.connect("try.tibbr.com", "myusername", "mypassword")
subs <- tibbr.subject(con, "analytics")
subs

## End(Not run)
```

---

tibbr.user

*Search tibbr For Users*

---

## Description

Searches tibbr for users.

## Usage

```
tibbr.user(connection, user = NULL)
```

## Arguments

connection	a tibbrConnection object indicating the tibbr instance to search.
user	a character string containing the search criteria. If NULL, tibbr returns all users on the tibbr instance. (Note: This can take a long time to return, depending on the number of users on your tibbr instance.) tibbr searches the following fields for matches: <ul style="list-style-type: none"><li>• First name</li><li>• Last name</li><li>• Email address</li><li>• Login name</li><li>• Department</li><li>• Location</li><li>• Profile description</li></ul>

## Details

Errors encountered during the search process are raised by this function.

## Value

returns a list of tibbrUser objects describing the results of the search. The names of the tibbrUser objects represent fields of the corresponding user's profile in tibbr.

## See Also

[tibbr.subject](#)

**Examples**

```
## Not run:  
con <- tibbr.connect("try.tibbr.com", "myusername", "mypassword")  
users <- tibbr.user(con, "Sam")  
users  
  
## End(Not run)
```

# Index

`print.tibbrConnection (tibbr.connect), 1`

`tibbr.connect, 1, 3`

`tibbr.disconnect (tibbr.connect), 1`

`tibbr.post, 2, 3`

`tibbr.subject, 3, 4, 5`

`tibbr.user, 3, 4, 5`