

Package ‘RForcecom’

March 5, 2012

Type Package

Title RForcecom provides the connection to Force.com (Salesforce.com) from R.

Version 0.1

Date 2012-02-29

Author Takekatsu Hiramura

Maintainer Takekatsu Hiramura <thira@plavox.info>

Description RForcecom provides the connection to Force.com (Salesforce.com) from R.

License BSD

LazyLoad yes

Depends XML, RCurl

Repository CRAN

URL <http://rforcecom.plavox.info/>

R topics documented:

RForcecom-package	2
rforcecom.api	3
rforcecom.create	4
rforcecom.delete	5
rforcecom.login	5
rforcecom.query	6
rforcecom.retrieve	7
rforcecom.search	8
rforcecom.update	9
rforcecom.upsert	9

Index	11
--------------	-----------

RForcecom-package *RForcecom provides the connection to Force.com (Salesforce.com) from R.*

Description

RForcecom provides the connection to Force.com (Salesforce.com) from R.

Details

Package:	RForcecom
Type:	Package
Version:	0.1
Date:	2012-02-29
License:	BSD
LazyLoad:	yes

Author(s)

Takekatsu Hiramura <thira@plavox.info>

References

Force.com REST API Developer's Guide http://www.salesforce.com/us/developer/docs/api_rest/
Web Services API Developer's Guide <http://www.salesforce.com/us/developer/docs/api/>

See Also

XML RCurl

Examples

```
## Not run:
# Sign in to the Force.com
username <- "yourname@yourcompany.com"
password <- "YourPasswordSECURITY_TOKEN"
instanceURL <- "https://xxx.salesforce.com/"
apiVersion <- "24.0"
session <- rforcecom.login(username, password, instanceURL, apiVersion)

# Execute a SOQL
soqlQuery <- "SELECT Id, Name, Industry, AnnualRevenue FROM Account"
rforcecom.query(session, soqlQuery)

# Execute a SOSL
queryString <- "United"
rforcecom.search(session, queryString)

# Creating a record
```

```
objectName <- "Account"
fields <- c(Name="R Analytics Service Ltd", Phone="5555-5555-5555")
rforcecom.create(session, objectName, fields)

# Retrieving a record
objectName <- "Account"
fields <- c("name", "Industry", "AnnualRevenue")
rforcecom.retrieve(session, objectName, fields)

# Updating a record
objectName <- "Account"
id <- "999x000000xxxxxZZZ"
fields <- c(Phone="9999-9999-9999")
rforcecom.update(session, objectName, id, fields)

# Upsert a record
objectName <- "Account";
externalIdField <- "AccountMaster__c"
externalId <- "AM-00000151"
fields <- c(Name="ABC Network Company", Phone="3333-3333-3333")
rforcecom.upsert(session, objectName, externalIdField, externalId, fields)

# Deleting a record
objectName <- "Account";
id <- "999x000000xxxxxZZZ"
rforcecom.delete(session, objectName, id)

## End(Not run)
```

rforcecom.api

Functions to connect API.

Description

For internal use.

Usage

```
rforcecom.api.getSoapEndpoint(apiVersion)

rforcecom.api.getSoqlEndpoint(apiVersion)

rforcecom.api.getSoslEndpoint(apiVersion)

rforcecom.api.getObjectEndpoint(apiVersion, objectName)

rforcecom.api.getRecordEndpoint(apiVersion, objectName, id)

rforcecom.api.getFieldEndpoint(apiVersion, objectName, field)

rforcecom.api.getExternalIdFieldEndpoint(apiVersion, objectName, field, id)
```

Arguments

apiVersion	An REST API or SOAP API version to connect. (e.g. "24.0". As of Feb 2012)
objectName	An object name. (e.g. Account, Contact, CustomObject__c)
field	A field name. (e.g. Id, Name, Industry, AnnualRevenue)
id	A record ID. (e.g. "999x000000xxxxxZZZ")

Value

Return values depends on the function used.

Author(s)

Takekatsu Hiramura <thira@plavox.info>

rforcecom.create *Creating a record*

Description

Creating a record

Usage

```
rforcecom.create(session, objectName, fields)
```

Arguments

session	Session data. It can be retrieve from rforcecom.login .
objectName	An object name. (e.g. "Account", "Contact", "CustomObject__c")
fields	Field names and value. (e.g. Name="CompanyName", Phone="000-000-000")

Value

No data.

Author(s)

Takekatsu Hiramura <thira@plavox.info>

Examples

```
## Not run:
objectName <- "Account"
fields <- c(Name="R Analytics Service Ltd", Phone="5555-5555-5555")
rforcecom.create(session, objectName, fields)

## End(Not run)
```

rforcecom.delete	<i>Deleting a record</i>
------------------	--------------------------

Description

Deleting a record

Usage

```
rforcecom.delete(session, objectName, id)
```

Arguments

session	Session data. It can be retrieve from rforcecom.login .
objectName	An object name. (e.g. "Account", "Contact", "CustomObject__c")
id	Record ID to retrieve. (e.g. "999x000000xxxxxZZZ")

Value

No data.

Author(s)

Takekatsu Hiramura <thira@plavox.info>

Examples

```
## Not run:
# Deleting a record
objectName <- "Account";
id <- "999x000000xxxxxZZZ" # Record's Id
rforcecom.delete(session, objectName, id)

## End(Not run)
```

rforcecom.login	<i>Sign in Force.com (Salesforce.com) and retrieve a session ID.</i>
-----------------	--

Description

Sign in Force.com (Salesforce.com) and retrieve a session ID.

Usage

```
rforcecom.login(username, password, instanceURL, apiVersion)
```

Arguments

username	Your username for login to Salesforce.com. In many cases, username is your E-mail address.
password	Your password for login to Salesforce.com. Note that DO NOT FORGET your Security Token. (Ex.) If your password is "Pass1234" and your security token is "XYZXYZXYZXYZ", you should set "Pass1234XYZXYZXYZXYZ".
instanceURL	Your SFDC domain name.
apiVersion	Version of the REST API and SOAP API that you want to use. Current API version is "24.0" (As of Feb. 2012).

Value

sessionID	Session ID.
instanceURL	Instance URL.
apiVersion	API Version.

Author(s)

Takekatsu Hiramura <thira@plavox.info>

See Also

[rforcecom.query](#) [rforcecom.search](#) [rforcecom.create](#) [rforcecom.delete](#) [rforcecom.retrieve](#)
[rforcecom.update](#) [rforcecom.upsert](#)

Examples

```
## Not run:
# Sign in to the Force.com
username <- "yourname@yourcompany.com"
password <- "YourPasswordSECURITY_TOKEN"
instanceURL <- "https://xxx.salesforce.com/"
apiVersion <- "24.0"
session <- rforcecom.login(username, password, instanceURL, apiVersion)

## End(Not run)
```

rforcecom.query	<i>Execute a SOQL</i>
-----------------	-----------------------

Description

Execute a SOQL

Usage

```
rforcecom.query(session, soqlQuery)
```

Arguments

session	Session data. It can be retrieve from rforcecom.login .
soqlQuery	A SOQL query. (e.g. "SELECT Id, Name FROM Account")

Value

Result dataset.

Author(s)

Takekatsu Hiramura <thira@plavox.info>

Examples

```
## Not run:
# Execute a SOQL
soqlQuery <- "SELECT Id, Name, Industry, AnnualRevenue FROM Account"
rforcecom.query(session, soqlQuery)

## End (Not run)
```

rforcecom.retrieve *Retrieving a record*

Description

Retrieving a record

Usage

```
rforcecom.retrieve(session, objectName, fields, limit=NULL, id=NULL)
```

Arguments

session	Session data. It can be retrieve from rforcecom.login .
objectName	An object name. (e.g. "Account", "Contact", "CustomObject__c")
fields	Field names. (e.g. "Id", "Name", "Industry", "AnnualRevenue")
limit	Number of the records to retrieve. (e.g. 5)
id	Record ID to retrieve. (e.g. "999x000000xxxxxZZZ")

Value

Result dataset.

Author(s)

Takekatsu Hiramura <thira@plavox.info>

See Also

[rforcecom.login](#)

Examples

```
## Not run:
# Retrieving a record
objectName <- "Account"
fields <- c("name", "Industry", "AnnualRevenue")
rforcecom.retrieve(session, objectName, fields)
rforcecom.retrieve(session, objectName, fields, limit = 5)
rforcecom.retrieve(session, objectName, fields, id = "999x000000xxxxxZZZ")

## End(Not run)
```

rforcecom.search	<i>Execute a SOSL</i>
------------------	-----------------------

Description

Execute a SOSL

Usage

```
rforcecom.search(session, queryString)
```

Arguments

`session` Session data. It can be retrieve from [rforcecom.login](#).
`queryString` Query strings to search. (e.g. "United", "Electronics")

Value

Result dataset.

Author(s)

Takekatsu Hiramura <thira@plavox.info>

Examples

```
## Not run:
# Execute a SOSL
queryString <- "United"
rforcecom.search(session, queryString)

## End(Not run)
```

rforcecom.update	<i>Updating a record</i>
------------------	--------------------------

Description

Updating a record

Usage

```
rforcecom.update(session, objectName, id, fields)
```

Arguments

session	Session data. It can be retrieve from rforcecom.login .
objectName	An object name. (e.g. "Account", "Contact", "CustomObject__c")
id	Record ID to update.
fields	Sets of field name and its value. (e.g. "999x000000xxxxxZZZ")

Value

No data.

Author(s)

Takekatsu Hiramura <thira@plavox.info>

Examples

```
## Not run:
# Updating a record
objectName <- "Account"
id <- "999x000000xxxxxZZZ"
fields <- c(Phone="9999-9999-9999")
rforcecom.update(session, objectName, id, fields)

## End(Not run)
```

rforcecom.upsert	<i>Upsert a record</i>
------------------	------------------------

Description

Upsert a record

Usage

```
rforcecom.upsert(session, objectName, externalIdField, externalId, fields)
```

Arguments

session	Session data. It can be retrieve from rforcecom.login .
objectName	An object name. (e.g. "Account", "Contact", "CustomObject__c")
externalIdField	An External Key's field name. (e.g. "AccountMaster__c")
externalId	An External Key's ID.
fields	Sets of field name and its value.

Value

No data.

Author(s)

Takekatsu Hiramura <thira@plavox.info>

Examples

```
## Not run:
# Upsert a record
objectName <- "Account";
externalIdField <- "AccountMaster__c"
externalId <- "AM-00000151"
fields <- c(Name="ABC Network Company", Phone="3333-3333-3333")
rforcecom.upsert(session, objectName, externalIdField, externalId, fields)

## End(Not run)
```

Index

*Topic **connection**

- RForcecom-package, 2
- rforcecom.api, 3
- rforcecom.create, 4
- rforcecom.delete, 5
- rforcecom.login, 5
- rforcecom.query, 6
- rforcecom.retrieve, 7
- rforcecom.search, 8
- rforcecom.update, 9
- rforcecom.upsert, 9

*Topic **package**

- RForcecom-package, 2

RForcecom (*RForcecom-package*), 2

RForcecom-package, 2

rforcecom.api, 3

rforcecom.create, 4, 6

rforcecom.delete, 5, 6

rforcecom.login, 4, 5, 5, 7–10

rforcecom.query, 6, 6

rforcecom.retrieve, 6, 7

rforcecom.search, 6, 8

rforcecom.update, 6, 9

rforcecom.upsert, 6, 9