

Package ‘LendingClub’

June 5, 2018

Type Package

Date 2018-06-04

Title A Lending Club API Wrapper

Version 2.0.0

URL <https://github.com/kuhnr130/LendingClub>

BugReports <https://github.com/kuhnr130/LendingClub/issues>

Description Functions to access Lending Club's API and assist the investor manage their account. Lending Club is a peer-to-peer lending service where loans are broken up into \$25 notes that investors buy with the expectation of earning a return on the interest. You can learn more about the API here: <http://www.lendingclub.com/developers/lc-api.action>.

License MIT + file LICENSE

LazyData TRUE

BuildVignettes TRUE

Suggests knitr, rmarkdown, testthat, covr

VignetteBuilder knitr

RoxygenNote 6.0.1

Imports dplyr, plyr, httr, jsonlite, utils, readr

NeedsCompilation no

Author Ryan Kuhn [aut, cre] (<<https://orcid.org/0000-0001-6827-3492>>)

Maintainer Ryan Kuhn <kuhnr130@gmail.com>

Repository CRAN

Date/Publication 2018-06-05 05:25:48 UTC

R topics documented:

AccountSummary	2
FolioBuy	3
FolioListing	4

FolioOrders	4
lc_AccountSummary	5
lc_AddFunds	6
lc_AvailableCash	7
lc_CancelTransfer	7
lc_CreatePortfolio	8
lc_Dashboard	8
lc_DetailedNotesOwned	9
lc_ListedLoans	9
lc_MakeCredential	10
lc_PendingTransfers	11
lc_PortfoliosOwned	11
lc_SubmitOrder	12
lc_WithdrawFunds	12
LendingClub	13
LendingClub-deprecated	13
print.AccountSummary	14
print.LendingClub_api	14
print.LendingClub_credential	15
summary holdings	15

Index **17**

AccountSummary	<i>Return the investor's Account Summary</i>
----------------	--

Description

Access the user's account and return the summary data. The data points include available cash, total account balance, and the number of notes in the portfolio.

Usage

```
AccountSummary(LC_CRED = NULL)
```

Arguments

LC_CRED	Credential object having class LendingClub_credential. This object can be created using the MakeCredential() function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".
---------	--

Value

Object of class LendingClub_api. The object has two components. The first is \$content with a data.frame of attributes of the investor's account such as available cash and value of the portfolios. The attributes are arranged by rows in the first column and the values are in the second column. The second component is the \$response including a timestamp of when the data was retrieved.

See Also

[LendingClub-deprecated](#)

Examples

```
## Not run:
LC_CRED<- lc_MakeCredential(InvestorID, APIKey)
AccountSummary()
## End(Not run)
```

FolioBuy

Transact with the secondary market

Description

Buy or sell notes on the FOLIOfn secondary market

Usage

```
FolioBuy(loanId = NULL, orderId = NULL, noteId, Price = NULL,
         LC_CRED = NULL, quiet = T)
```

```
FolioSell(loanId = NULL, orderId = NULL, noteId, expireDate = NULL,
          Price = NULL, LC_CRED = NULL, quiet = T)
```

Arguments

loanId	Deprecated. As of Jan 2018, the folio API was revised and the loan ID is no longer required.
orderId	Deprecated. As of Jan 2018, the folio API was revised and the order ID is no longer required. #' @inheritParams AccountSummary
noteId	The loan's note ID
Price	Price you wish to sell your loan for. Remember to consider
LC_CRED	Credential object having class <code>LendingClub_credential</code> . This object can be created using the <code>lc_MakeCredential()</code> function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".
quiet	Should the API response be printed to the console the accrued interest in your offering.
expireDate	The date the sell offer will expire. Can be as short as the current date or up to 7 days from posting the offer. Character string of the format "YYYY-MM-DD".

FolioListing	<i>Notes listed on Folio marketplace</i>
--------------	--

Description

Returns a dataframe of notes listed on the secondary market.

Usage

```
FolioListing(updatedSince = NULL, LC_CRED = NULL)
```

Arguments

updatedSince	filter listings to include only notes that were modified/created within the last X minutes.
LC_CRED	Credential object having class LendingClub_credential. This object can be created using the <code>lc_MakeCredential()</code> function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".

Details

This function can return the complete listing of notes on the secondary market or just those that have been recently updated. The complete listing of loans may take a significant amount of time to download depending on the users internet connection speed.

Notes listed on the market automatically expire in 7 days. The seller may reset the expiration date by changing the price on the note.

Value

A data.frame. If the entire current listings are returned, the data frame will have 22 columns. If a subset of the listings are returned, via the updatedSince argument, then the returned data frame will have 21 columns.

FolioOrders	<i>Folio orders submitted in the current month</i>
-------------	--

Description

View either buy or sell orders placed in the current month

Usage

```
FolioOrders(includeDetails = TRUE, orderType, status = NULL,
  LC_CRED = NULL)
```

Arguments

includeDetails	Include note level details
orderType	Return 'buy' or 'sell' orders
status	Used to filter results. Can be NULL or one of 9 possible status: <ul style="list-style-type: none"> • IN_LISTING • SETTLED • CANCELED_BY_SELLER • CANCELED_TIME_EXPIRED • PENDING_TRADE_SETTLEMENT • CANCELED_BY_PAYMENT • CANCELED_BY_DEFAULT_OR_CHARGE_OFF • REPRICED • CANCELED_BY_BANKRUPTCY
LC_CRED	Credential object having class LendingClub_credential. This object can be created using the <code>lc_MakeCredential()</code> function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".

lc_AccountSummary	<i>Return the investor's Account Summary</i>
-------------------	--

Description

Access the user's account and return the summary data. The data points include available cash, total account balance, and the number of notes in the account

Usage

```
lc_AccountSummary(LC_CRED = NULL)
```

Arguments

LC_CRED	Credential object having class LendingClub_credential. This object can be created using the <code>lc_MakeCredential()</code> function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".
---------	--

Value

Object of class `LendingClub_api`. The object has two components. The first is `$content` with a `data.frame` of attributes of the investor's account such as available cash and value of the portfolios. The data arranged as a 1 row `data.frame` with an attribute in each column. The second component is the `$response` including a timestamp of when the data was retrieved.

Examples

```
## Not run:
LC_CRED<- lc_MakeCredential(InvestorID, APIKey)
lc_AccountSummary()
## End(Not run)
```

lc_AddFunds

Add funds to the investor's account

Description

Sets up a recurring or one-time funds transfer. The investor must have already setup the bank account from which the funds are transferred. Transfers can one-time or recurring.

Usage

```
lc_AddFunds(amount, freq = "LOAD_NOW", start = NULL, end = NULL,
  quiet = T, LC_CRED = NULL)
```

Arguments

amount	Numeric. Amount to be transferred
freq	String. Frequency of transfers. Optional. The default value is LOAD_NOW so if a value is not supplied the transfer will be processed immediately. If a value is supplied, it must be one of: <ul style="list-style-type: none"> • LOAD_NOW (default) • LOAD_ONCE • LOAD_WEEKLY • LOAD_BIWEEKLY • LOAD_ON_DAY_1_AND_15 • LOAD_MONTHLY
start	String. Defaults to NULL and is not required for immediate transfers. For future transfers, the date must be a string formatted as YYYY/MM/DD.
end	String. Defaults to NULL and is not required for immediate transfers. For future transfers, the date must be a string formatted as YYYY/MM/DD.
quiet	Should the response be printed to the console
LC_CRED	Credential object having class LendingClub_credential. This object can be created using the lc_MakeCredential() function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".

lc_AvailableCash	<i>Return investor's available cash</i>
------------------	---

Description

Retrieves the available cash in the user's account. Cash committed to open buy orders is excluded from the total.

Usage

```
lc_AvailableCash(LC_CRED = NULL)
```

Arguments

LC_CRED	Credential object having class LendingClub_credential. This object can be created using the <code>lc_MakeCredential()</code> function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".
---------	--

lc_CancelTransfer	<i>Cancel a scheduled transfer</i>
-------------------	------------------------------------

Description

Cancel a scheduled transfer. You can get a listing of scheduled transfers by using the `PendingTransfers()` function.

Usage

```
lc_CancelTransfer(transferId, LC_CRED = NULL)
```

Arguments

transferId	Numeric. Unique ID for the scheduled transfer
LC_CRED	Credential object having class LendingClub_credential. This object can be created using the <code>lc_MakeCredential()</code> function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".

lc_CreatePortfolio *Create a new portfolio*

Description

Create a new portfolio to assign loans to. Notes can be assigned to a portfolio using the API when they are purchased from the primary market.

Usage

```
lc_CreatePortfolio(port_name, port_desc, LC_CRED = NULL, quiet = T)
```

Arguments

port_name	String. Name of the new portfolio
port_desc	String. Portfolio description
LC_CRED	Credential object having class LendingClub_credential. This object can be created using the <code>lc_MakeCredential()</code> function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".
quiet	Should the response be printed to the console

lc_Dashboard *Creates an HTML dashboard*

Description

Knits a document with summary statistics needed to evaluate an account.

Usage

```
lc_Dashboard(filename = "Dashboard.html", out_dir = getwd(),
  ID = LC_CRED$investorID, key = LC_CRED$key, ...)
```

Arguments

filename	Name of the file to create
out_dir	Directory where to save the html file
ID	User's ID from the Account Summary page
key	User's API key from the account settings page
...	Arguments passed on to <code>rmarkdown::render</code>

lc_DetailedNotesOwned *Listing of investor's loans*

Description

Access the user's account and return a dataframe of the investor's notes owned.

Usage

```
lc_DetailedNotesOwned(LC_CRED = NULL)
```

```
lc_NotesOwned(LC_CRED = NULL)
```

Arguments

LC_CRED	Credential object having class <code>LendingClub_credential</code> . This object can be created using the <code>lc_MakeCredential()</code> function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".
---------	--

Details

The `lc_DetailedNotesOwned()` function returns nearly the same variables as `lc_NotesOwned()`. As the name suggests, the function returns additional information about the notes owned including datapoints on the purpose, portfolio, and the payment history.

lc_ListedLoans *Listing of all loans on the primary market*

Description

Uses Lending Club's API to return all current listed loans

Usage

```
lc_ListedLoans(showAll = TRUE, LC_CRED = NULL)
```

Arguments

showAll	logical, indicate whether all loans should be returned. Use false to limit the results to those loans listed in the most recent listing period.
LC_CRED	Credential object having class <code>LendingClub_credential</code> . This object can be created using the <code>lc_MakeCredential()</code> function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".

Value

LendingClub_api object. There are two components, the first is the content of the request, the second is the response from the API. The content can be further subdivided into a 2 element list. The first element is the date of the request and second is a dataframe of the listed loans.

lc_MakeCredential	<i>Create the Lending Club credential</i>
-------------------	---

Description

Creates the credential needed to access the other functions in this package.

Usage

```
lc_MakeCredential(investorID, APIkey)
```

Arguments

investorID	User's ID from the Account Summary page
APIkey	User's API key from the account settings page
...	See the details section for more information.

Details

This is the first step to accessing the API. Pass the function your investor ID and API key. The investor ID can be found on your account summary page and the API Key can be found on your account settings page. The function will create an object of class LendingClub_credential. If the credential is stored to the variable LC_CRED, then the other functions in the package will be able to read the credential from the global environment. Else, you may pass the credential to the other functions as an argument.

Examples

```
LC_CRED<- lc_MakeCredential("investorID", "APIkey")
```

lc_PendingTransfers *Return investor's pending transfers*

Description

Uses Lending Club's API to access the user's account and return the summary

Usage

```
lc_PendingTransfers(LC_CRED = NULL)
```

Arguments

LC_CRED	Credential object having class LendingClub_credential. This object can be created using the <code>lc_MakeCredential()</code> function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".
---------	--

lc_PortfoliosOwned *Listing of investor's portfolios*

Description

Provides a dataframe of all portfolios owned by the investor.

Usage

```
lc_PortfoliosOwned(LC_CRED = NULL)
```

Arguments

LC_CRED	Credential object having class LendingClub_credential. This object can be created using the <code>lc_MakeCredential()</code> function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".
---------	--

lc_SubmitOrder *Buy a note on the primary market*

Description

Create an order to purchase a note. If the account has sufficient funding, the note will be purchased once the loan has passed through the funding stage.

Usage

```
lc_SubmitOrder(loanId, amount = 25, portfolioId = NULL, LC_CRED = NULL,
               quiet = T)
```

Arguments

loanId	Unique ID for loans. These can be found using the ListedLoans() function.
amount	Amount to be purchased
portfolioId	Unique ID for the portfolio. It can be found using the PortfoliosOwned() function.
LC_CRED	Credential object having class LendingClub_credential. This object can be created using the lc_MakeCredential() function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".
quiet	Should the response be printed to the console

Examples

```
## Not run:
SubmitOrder(12345678, 25)
SubmitOrder(12345678, 25, "myPortfolio")

#For multiple loans in a single order
SubmitOrder(loanId=c(123, 456, 789), amount= c(rep(25,3)))

## End(Not run)
```

lc_WithdrawFunds *Withdraw funds*

Description

Withdraw funds from the investor's account

Usage

```
lc_WithdrawFunds(amount, LC_CRED = NULL)
```

Arguments

amount	Amount to be transferred
LC_CRED	Credential object having class <code>LendingClub_credential</code> . This object can be created using the <code>lc_MakeCredential()</code> function. You can pass the object directly as an argument to the function or alternatively, it can be read from the global environment if it was stored as "LC_CRED".

LendingClub

Lending Club

Description

Providing the tools to work with the Lending Club API. Both the primary and secondary markets can be accessed with this package.

Details

Lending Club is a peer-to-peer lending service where loans are broken up into \$25 notes that investors buy with expectation of earning a return on the interest. This package is built to assist the investor manage their account and maximize their return.

To learn more about the LendingClub package, start with the vignette: `vignette("LendingClub")`

Author(s)

Ryan Kuhn, CPA

See Also

- <http://ryankuhn.net/LendingClub>
- <http://github.com/kuhnr130/LendingClub/issues> for bug reports

LendingClub-deprecated

*Deprecated functions in the **LendingClub** package.*

Description

The functions listed below are provided for compatibility with older versions of the LendingClub package. They are deprecated and may be removed (defunct) in later versions.

Arguments

... Parameters to be passed to the modern versions of the function

Details

Deprecated Function	Modern Function
AccountSummary	lc_AccountSummary
AccountSummary_	lc_AccountSummary
AddFunds	lc_AddFunds
AvailableCash	lc_AvailableCash
CreatePortfolio	lc_CreatePortfolio
Dashboard	lc_Dashboard
DetailedNotesOwned	lc_DetailedNotesOwned
MakeCredential	lc_MakeCredential
NotesOwned	lc_NotesOwned
ListedLoans	lc_ListedLoans
SubmitOrder	lc_SubmitOrder

```
print.AccountSummary Print the output from an API call
```

Description

This is a general function to print the content from an API call. The API call returns a list of 2 elements: The first is the content from the call and the second element is the actual response.

Usage

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

Arguments

x an object of class `LendingClub_api`
 ... To pass additional arguments to the generic print function

```
print.LendingClub_api Print the output from an API call
```

Description

This is a general function to print the content from an API call. The API call returns a list of 2 elements: The first is the content from the call and the second element is the actual response.

Usage

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

Arguments

x an object of class LendingClub_api
... To pass additional arguments to the generic print function

print.LendingClub_credential
Print the LC_CRED object

Description

This function prints the content from the MakeCredential call.

Usage

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

Arguments

x an object of class LendingClub_api
... To pass additional arguments to the generic print function

summary.holdings *Prepare a dashboard of key metrics*

Description

Creating a summary of matrices an investor would find useful in evaluating their account.

Usage

```
## S3 method for class 'holdings'  
summary(object, ...)
```

Arguments

object data.frame of loans from DetailedNotesOwned()
... currently not used

Details

At Risk Ratio calculates the percentage of the loans with principal 16 or more days late.

Examples

```
## Not run:
summary( lc_DetailedNotesOwned( LC_CRED ))

# split notes on a variable and run summary for each subset
holdings<- lc_DetailedNotesOwned(LC_CRED)$content
x<- lapply(split(holdings, f= holdings$portfolioName), summary)
y<- Reduce(function(...) merge(..., by= 'col_labs', all.x= TRUE), x)
names(y)[-1]<- names(x)
y

## End(Not run)
```


Index

AccountSummary, [2](#)
AccountSummary_
 (LendingClub-deprecated), [13](#)
AddFunds (LendingClub-deprecated), [13](#)
AvailableCash (LendingClub-deprecated),
 [13](#)

CancelTransfer
 (LendingClub-deprecated), [13](#)
CreatePortfolio
 (LendingClub-deprecated), [13](#)

Dashboard (LendingClub-deprecated), [13](#)
DetailedNotesOwned
 (LendingClub-deprecated), [13](#)

FolioBuy, [3](#)
FolioListing, [4](#)
FolioOrders, [4](#)
FolioSell (FolioBuy), [3](#)

lc_AccountSummary, [5](#), [14](#)
lc_AddFunds, [6](#), [14](#)
lc_AvailableCash, [7](#), [14](#)
lc_CancelTransfer, [7](#)
lc_CreatePortfolio, [8](#), [14](#)
lc_Dashboard, [8](#), [14](#)
lc_DetailedNotesOwned, [9](#), [14](#)
lc_ListedLoans, [9](#), [14](#)
lc_MakeCredential, [10](#), [14](#)
lc_NotesOwned, [14](#)
lc_NotesOwned (lc_DetailedNotesOwned), [9](#)
lc_PendingTransfers, [11](#)
lc_PortfoliosOwned, [11](#)
lc_SubmitOrder, [12](#), [14](#)
lc_WithdrawFunds, [12](#)
LendingClub, [13](#)
LendingClub-deprecated, [13](#)
LendingClub-deprecated-package
 (LendingClub-deprecated), [13](#)

LendingClub-package (LendingClub), [13](#)
ListedLoans (LendingClub-deprecated), [13](#)

MakeCredential
 (LendingClub-deprecated), [13](#)

NotesOwned (LendingClub-deprecated), [13](#)

PendingTransfers
 (LendingClub-deprecated), [13](#)
PortfoliosOwned
 (LendingClub-deprecated), [13](#)
print.AccountSummary, [14](#)
print.LendingClub_api, [14](#)
print.LendingClub_credential, [15](#)

SubmitOrder (LendingClub-deprecated), [13](#)
 summary.holdings, [15](#)

WithdrawFunds (LendingClub-deprecated),
 [13](#)