

# Package ‘xlm’

October 22, 2018

**Type** Package

**Title** A 'Stellar' Client

**Version** 0.1.1

**Author** Oliver Frost

**Maintainer** Oliver Frost <oliverfrost@protonmail.com>

**Description** An R wrapper for interacting with the Stellar network, mostly via the Horizon API.

The Horizon API is a RESTful API that allows applications to query the Stellar blockchain, or to stream data, potentially in real time.

A full overview of Horizon can be found at <<https://www.stellar.org/developers/reference/>>.

**License** GPL-2

**Encoding** UTF-8

**LazyData** true

**Imports** httr, RJSONIO, methods, data.table, stats, R6

**RoxygenNote** 6.1.0

**Suggests** testthat, knitr, rmarkdown

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2018-10-22 17:30:02 UTC

## R topics documented:

Account . . . . .	2
distribution . . . . .	3
findPaymentPath . . . . .	4
friendbot . . . . .	5
getAccountDetail . . . . .	5
getAssets . . . . .	6
getEffects . . . . .	7
getEffects_Account . . . . .	7
getEffects_Ledger . . . . .	8
getEffects_Transaction . . . . .	9

getLedgerDetail . . . . .	10
getLedgers . . . . .	10
getOffers_Account . . . . .	11
getOperations . . . . .	12
getOperations_Account . . . . .	13
getOperations_Ledger . . . . .	14
getOperations_Transaction . . . . .	14
getOrderbook . . . . .	15
getPayments . . . . .	16
getPayments_Account . . . . .	17
getPayments_Ledger . . . . .	18
getPayments_Transaction . . . . .	19
getTradeAggregations . . . . .	20
getTrades . . . . .	21
getTransactionDetail . . . . .	22
getTransactions . . . . .	23
getTransactions_Account . . . . .	23
getTransactions_Ledger . . . . .	24
Ledger . . . . .	25
listToDF . . . . .	26
Orderbook . . . . .	26
postTransaction . . . . .	27
price . . . . .	28
Transaction . . . . .	28
<b>Index</b>	<b>30</b>

---

Account	<i>Account object.</i>
---------	------------------------

---

### Description

An R6 class representing an Account on the Stellar network. It is initialized from the `getAccountDetail()` function.

### Usage

Account

### Format

An object of class `R6ClassGenerator` of length 24.

**Fields**

- pk - character representing the public key of the account.
- sequence - the current transaction sequence number of the account.
- balances - a list of balances, including the native XLM and any other assets the account holds.
- signers - get information about the signers who authorize activity on the account.
- data - get any data key-value pairs (not secret keys) that have been assigned to the account.
- response - list containing the raw JSON response.

**Note**

<https://www.stellar.org/developers/guides/concepts/accounts.html>

**Examples**

```
binance <- Account$new("GC02IP3MJNUOKS4PUDI4C7LGGMQDJGXG3COYX3WSB4HHNAHKYV5YL3VC")
binance$effects()
```

---

distribution

*Gets the current XLM distribution stats.*

---

**Description**

Uses an API call from the Stellar dashboard to get the distribution figures.

**Usage**

```
distribution()
```

**Value**

list

**Note**

<https://dashboard.stellar.org/>

**Examples**

```
xlm_stats = distribution()
```

---

findPaymentPath	<i>Find a payment path between a source and a destination.</i>
-----------------	--

---

### Description

Calculate a path for a payment based on the assets the source/target owns.

### Usage

```
findPaymentPath(destination_account, destination_asset_type,
  destination_asset_code, destination_asset_issuer, destination_amount,
  source_account, domain = "public")
```

### Arguments

destination_account	character - The destination account that any returned path should use.
destination_asset_type	character - The type of the destination asset credit_alphanum4
destination_asset_code	character - The code for the destination, if destination_asset_type is not "native".
destination_asset_issuer	character - The issuer for the destination, if destination_asset_type is not "native".
destination_amount	character - The amount, denominated in the destination asset, that any returned path should be able to satisfy.
source_account	character - The sender's account id. Any returned path must use a source that the sender can hold.
domain	- character - signalling to query the testnet or public network. Can be either "testnet" or "public".

### Value

list

### Note

<https://www.stellar.org/developers/horizon/reference/endpoints/path-finding.html>

---

friendbot	<i>Creates and fund a new account on the test network.</i>
-----------	--

---

**Description**

Make a call to the Friendbot service on the test network and fund an account. Returns a JSON response.

**Usage**

```
friendbot(account)
```

**Arguments**

account	character - the public key of the account you wish to fund.
---------	---

**Value**

list

**Note**

<https://www.stellar.org/developers/guides/get-started/create-account.html>

**Examples**

```
friendbot("GAKYVLEN4MFPQ2R5Y4BA5RSJ7U5R2HGOK3VJ5PEUTLOHKTCBGCJZSH0S")
```

---

getAccountDetail	<i>Get an overview of an account on the Stellar ledger.</i>
------------------	---

---

**Description**

Query the accounts endpoint, specifying no extra resources. Converts the JSON response to a list.

**Usage**

```
getAccountDetail(pk, data = NA, domain = "public")
```

**Arguments**

pk	character - your Stellar account address (also known as a public key.)
data	character - if the Stellar account has data key-value pairs assigned to it, pass the key name to get the value. The internal request builder will attempt to parse the key to ASCII if it contains any non-ASCII characters.
domain	character - signalling to query the testnet or public network. Can be either "test-net" or "public".

**Value**

list or Account class.

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/accounts-single.html>

**Examples**

```
getAccountDetail("GCO2IP3MJNUOKS4PUDI4C7LGGMQDJGXG3COYX3WSB4HHNAHKYV5YL3VC")
```

---

getAssets	<i>Returns all assets.</i>
-----------	----------------------------

---

**Description**

Return all assets in the system with statistics.

**Usage**

```
getAssets(limit, cursor = NULL, asset_code = "", asset_issuer = "",
  order = "asc", data.table = TRUE, domain = "public")
```

**Arguments**

limit	numeric - the number of records to return. Default is 10.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
asset_code	character - (optional) code of the asset to filter by
asset_issuer	character - (optional) issuer of the asset to filter by
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public". boolean - if TRUE, a data.table is overwritten and the server-side streaming capability is utilised. A list will be returned.

**Value**

list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/assets-all.html>

**Examples**

```
getAssets(10, domain = "testnet")
```



**Arguments**

pk	character - your Stellar account/wallet address.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
limit	numeric - optional, the number of records to return. Default is 10.
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

data.table (by default) or list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/accounts-single.html>

**Examples**

```
## Not run: getEffects_Account("GC02IP3MJNUOKS4PUDI4C7LGGMQDJGXG3COYX3WSB4HHNAHKYV5YL3VC")
```

---

getEffects\_Ledger      *Get an overview of account effects on the Stellar ledger.*

---

**Description**

Returns effects from the a specific ledger. Converts the JSON response to a list.

**Usage**

```
getEffects_Ledger(ledger, cursor = NULL, limit = 10, order = "asc",
  data.table = TRUE, domain = "public")
```

**Arguments**

ledger	numeric - a ledger ID.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
limit	numeric - optional, the number of records to return. Default is 10.
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".



**Value**

data.table (by default) or list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/effects-for-ledger.html>

**Examples**

```
getEffects_Ledger("25000")
```

---

getEffects\_Transaction

*Get an overview of transaction effects for a specific transaction.*

---

**Description**

Returns effect details for a specific transaction on the ledger. Converts the JSON response to a list.

**Usage**

```
getEffects_Transaction(hash, cursor = NULL, limit = 10,  
  order = "asc", data.table = TRUE, domain = "public")
```

**Arguments**

hash	character - the transaction id.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
limit	numeric - optional, the number of records to return. Default is 10.
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

data.table (by default) or list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/effects-for-transaction.html>

**Examples**

```
getEffects_Transaction("SomeTxnHash")
```

---

getLedgerDetail      *Returns a single ledger.*

---

**Description**

Return a single ledger by providing a sequence id.

**Usage**

```
getLedgerDetail(sequence, domain = "public")
```

**Arguments**

sequence	character - (required) a ledger id. Must be 1 or greater.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/ledgers-single.html>

**Examples**

```
## Not run: getLedgerDetail("1", "testnet")
```

---

getLedgers      *Returns all ledgers.*

---

**Description**

Return all valid ledgers in pages or stream them to R.

**Usage**

```
getLedgers(limit, cursor = NULL, order = "asc", data.table = TRUE,  
           domain = "public")
```

**Arguments**

limit	numeric, the number of records to return. Default is 10.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public". boolean - if TRUE, a data.table is overwritten and the server-side streaming capability is utilised. A list will be returned.

**Value**

list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/ledgers-all.html>

**Examples**

```
## Not run: getLedgers(limit = 50)
```

---

getOffers\_Account      *Get an overview of offers on the Stellar ledger.*

---

**Description**

Returns offers with a particular account. Converts the JSON response to a list.

**Usage**

```
getOffers_Account(pk, cursor = NULL, limit = 10, order = "asc",
  data.table = TRUE, domain = "public")
```

**Arguments**

pk	character - your Stellar account/wallet address.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
limit	numeric - optional, the number of records to return.
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/offers-for-account.html>

**Examples**

```
getOperations_Account("GC02IP3MJNUOKS4PUDI4C7LGGMQDJGXG3COYX3WSB4HHNAHKYV5YL3VC")
```

---

getOperations	<i>Returns all operations.</i>
---------------	--------------------------------

---

**Description**

Return all operations that are a part of a valid transaction.

**Usage**

```
getOperations(limit, cursor = NULL, order = "asc", data.table = TRUE,
              domain = "public")
```

**Arguments**

limit	numeric - the number of records to return. Default is 10.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public". boolean - if TRUE, a data.table is overwritten and the server-side streaming capability is utilised. A list will be returned.

**Value**

data.table or list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/operations-all.html>

**Examples**

```
getOperations(33, domain = "testnet")
```

---

getOperations\_Account *Get an overview of account operations on the Stellar ledger.*

---

## Description

Returns all operations that were included in valid transactions that affected a particular account. Converts the JSON response to a list.

## Usage

```
getOperations_Account(pk, cursor = NULL, limit = 10, order = "asc",
  data.table = TRUE, domain = "public")
```

## Arguments

pk	character - your Stellar account/wallet address (public key.)
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
limit	numeric - optional, the number of records to return. Default is 10.
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

## Value

data.table (default) or list

## Note

<https://www.stellar.org/developers/horizon/reference/endpoints/operations-for-account.html>

## Examples

```
getOperations_Account("GC02IP3MJNUOKS4PUDI4C7LGGMQDJG3COYX3WSB4HHNAHKYV5YL3VC")
```

---

getOperations\_Ledger *Get an overview of account operations on the Stellar ledger.*

---

### Description

Returns all operations on a specific ledger. Converts the JSON response to a list.

### Usage

```
getOperations_Ledger(ledger, cursor = NULL, limit = 10,
  order = "asc", data.table = TRUE, domain = "public")
```

### Arguments

ledger	numeric - a ledger ID.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
limit	numeric - optional, the number of records to return. Default is 10.
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

### Value

data.table (default) or list

### Note

<https://www.stellar.org/developers/horizon/reference/endpoints/operations-for-ledger.html>

### Examples

```
getOperations_Ledger("1000")
```

---

getOperations\_Transaction  
*Get an overview of operations for a transaction on the Stellar ledger.*

---

### Description

Returns all operations that were included in a specific transactions. Converts the JSON response to a list.

**Usage**

```
getOperations_Transaction(hash, cursor = NULL, limit = 10,
  order = "asc", data.table = TRUE, domain = "public")
```

**Arguments**

hash	character = the transaction id.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
limit	numeric - optional, the number of records to return. Default is 10.
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

data.table (default) or list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/operations-for-transaction.html>

**Examples**

```
getOperations_Transaction("someHash")
```

---

getOrderbook

*Returns details on the current state of the order book*

---

**Description**

Return a summary of assets bought and sold on the ledger.

**Usage**

```
getOrderbook(selling_asset_type, selling_asset_code = "",
  selling_asset_issuer = "", buying_asset_type, buying_asset_code = "",
  buying_asset_issuer = "", domain = "public")
```

**Arguments**

selling_asset_type	required, character - Type of the Asset being sold native
selling_asset_code	optional, character - code of the Asset being sold USD
selling_asset_issuer	optional, character - account ID of the issuer of the Asset being sold GA2HGBJIJKI6O4XEM7CZWY5PS
buying_asset_type	required, character - type of the Asset being bought credit_alphanum4
buying_asset_code	optional, character - code of the Asset being bought BTC
buying_asset_issuer	optional, character - account ID of the issuer of the Asset being bought
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/orderbook-details.html>

**Examples**

```
getOrderbook(selling_asset_type = "native", buying_asset_type = "credit_alphanum4")
```

---

getPayments

*Returns all payments.*

---

**Description**

Return all payments operations that are a part of a valid transaction.

**Usage**

```
getPayments(limit, cursor = NULL, order = "asc", data.table = TRUE,
  domain = "public")
```



**Arguments**

limit	numeric - the number of records to return. Default is 10.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public". boolean - if TRUE, a data.table is overwritten and the server-side streaming capability is utilised. A list will be returned.

**Value**

data.table or list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/payments-all.html>

**Examples**

```
getPayments(20, domain = "testnet")
```

---

getPayments\_Account    *Get an overview of payments on the Stellar ledger.*

---

**Description**

Returns all payments from valid transactions that affected a particular account. Converts the JSON response to a list.

**Usage**

```
getPayments_Account(pk, cursor = NULL, limit = 10, order = "asc",
  data.table = TRUE, domain = "public")
```

**Arguments**

pk	character - your Stellar account/wallet address.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
limit	numeric - optional, the number of records to return. Default is 10.
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

data.table (by default) or list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/payments-for-account.html>

**Examples**

```
## Not run: getPayments_Account("GC02IP3MJNUOKS4PUDI4C7LGGMQDJGXG3COYX3WSB4HHNAHKYV5YL3VC")
```

---

getPayments\_Ledger      *Get an overview of payments on the Stellar ledger.*

---

**Description**

Returns all payments from valid transactions that affected a particular ledger. Converts the JSON response to a list.

**Usage**

```
getPayments_Ledger(ledger, cursor = NULL, limit = 10, order = "asc",
  data.table = TRUE, domain = "public")
```

**Arguments**

ledger	numeric - a ledger ID.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
limit	numeric - optional, the number of records to return. Default is 10.
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

data.table (by default) or list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/payments-for-ledger.html>

**Examples**

```
getPayments_Ledger("10000")
```

---

`getPayments_Transaction`*Get an overview of payments for a specific transaction.*

---

**Description**

Returns all payment operations that are part of a given transaction. Converts the JSON response to a list.

**Usage**

```
getPayments_Transaction(hash, cursor = NULL, limit = 10,  
  order = "asc", data.table = TRUE, domain = "public")
```

**Arguments**

<code>hash</code>	character = the transaction id.
<code>cursor</code>	numeric - optional, a paging token - where to start from. Can also be "now".
<code>limit</code>	numeric - optional, the number of records to return. Default is 10.
<code>order</code>	character - optional, "asc" or "desc"
<code>data.table</code>	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
<code>domain</code>	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

data.table (by default) or list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/payments-for-transaction.html>

**Examples**

```
getPayments_Transaction("someHash")
```

---

getTradeAggregations *Returns trade aggregations.*

---

### Description

Gather historical trading data directly from the API.

### Usage

```
getTradeAggregations(start_time, end_time, resolution, limit = 10,
  order = "asc", base_asset_type, base_asset_code = "",
  base_asset_issuer = "", counter_asset_type, counter_asset_code = "",
  counter_asset_issuer = "", data.table = FALSE, domain = "public")
```

### Arguments

start_time	numeric - milliseconds since epoch
end_time	numeric - milliseconds since epoch
resolution	numeric - segment duration as millis since epoch. Supported values are 5 minutes (300000), 15 minutes (900000), 1 hour (3600000), 1 day (86400000) and 1 week (604800000).
limit	numeric - optional, the number of records to return. Default is 10.
order	character - optional, "asc" or "desc"
base_asset_type	string - type of base asset
base_asset_code	string - code of base asset, not required if type is native
base_asset_issuer	string - issuer of base asset, not required if type is native
counter_asset_type	string - type of counter asset
counter_asset_code	string - code of counter asset, not required if type is native
counter_asset_issuer	string - issuer of counter asset, not required if type is native
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	- character - signalling to query the testnet or public network. Can be either "testnet" or "public".

### Value

data.table or list

**Note**

[https://www.stellar.org/developers/horizon/reference/endpoints/trade\\_aggregations.html](https://www.stellar.org/developers/horizon/reference/endpoints/trade_aggregations.html)

**Examples**

```
start_time="1512689100000"
end_time="1534924458000"
resolution="300000"
base_asset_type="native"
counter_asset_type="credit_alphanum4"
counter_asset_code="BTC"
counter_asset_issuer = "GATEMHCKCY67ZUCKTROYN24ZYT5GK4EQZ65JJLDHKHRUZI3EUEKMTCH"

getTradeAggregations(start_time, end_time, resolution,
base_asset_type = base_asset_type, counter_asset_type = counter_asset_type,
counter_asset_code = counter_asset_code, counter_asset_issuer = counter_asset_issuer)
```

---

getTrades	<i>Returns data on trades.</i>
-----------	--------------------------------

---

**Description**

Return all partially fulfilled trades to buy or sell assets on the ledger.

**Usage**

```
getTrades(limit, cursor = NULL, order = "asc", domain = "public",
base_asset_type = "", base_asset_code = "", base_asset_issuer = "",
counter_asset_type = "", counter_asset_code = "",
counter_asset_issuer = "", offer_id = "", data.table = FALSE)
```

**Arguments**

limit	numeric - the number of records to return.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
order	character - optional, "asc" or "desc"
domain	- character - signalling to query the testnet or public network. Can be either "testnet" or "public".
base_asset_type	optional, string - type of base asset
base_asset_code	optional, string - code of base asset, not required if type is native
base_asset_issuer	optional, string - issuer of base asset, not required if type is native
counter_asset_type	optional, string - type of counter asset

counter_asset_code	optional, string - code of counter asset, not required if type is native
counter_asset_issuer	optional, string - issuer of counter asset, not required if type is native
offer_id	optional, string - filter for by a specific offer id
data.table	boolean - if true, a data.table is returned. FALSE by default.

**Value**

data.table or list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/trades.html>

**Examples**

```
## Not run: getTrades(20)
```

---

getTransactionDetail *Returns a single valid transactions.*

---

**Description**

Return a single transaction using a transaction id.

**Usage**

```
getTransactionDetail(hash, domain = "public")
```

**Arguments**

hash	character - a transaction id/hash.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/transactions-single.html>

**Examples**

```
getTransactionDetail("someHash")
```

---

getTransactions	Returns all valid transactions.
-----------------	---------------------------------

---

### Description

Return all valid transactions in pages or stream them to R.

### Usage

```
getTransactions(limit, cursor = NULL, order = "asc",
  data.table = TRUE, domain = "public")
```

### Arguments

limit	numeric - the number of records to return. Default is 10.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public". boolean - if TRUE, a data.table is overwritten and the server-side streaming capability is utilised. A list will be returned.

### Value

data.table (by default) or list

### Note

<https://www.stellar.org/developers/horizon/reference/endpoints/transactions-all.html>

### Examples

```
getTransactions(10, domain = "testnet")
```

---

getTransactions_Account
-------------------------

---

*Get an overview of transactions on the Stellar ledger.*

---

### Description

Returns all valid transactions that affected a particular account. Converts the JSON response to a list.

**Usage**

```
getTransactions_Account(pk, cursor = NULL, limit = 10, order = "asc",  
  data.table = TRUE, domain = "public")
```

**Arguments**

pk	character - your Stellar account/wallet address.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
limit	numeric - optional, the number of records to return.
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

data.table (by default) or list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/transactions-for-account.html>

**Examples**

```
## Not run: getTransactions_Account("GC02IP3MJNUOKS4PUDI4C7LGGMQDJGXG3COYX3WSB4HHNAHKYV5YL3VC")
```

---

getTransactions\_Ledger

*Get an overview of transactions on the Stellar ledger.*

---

**Description**

Returns all valid transactions on a specific ledger. Converts the JSON response to a list.

**Usage**

```
getTransactions_Ledger(ledger, cursor = NULL, limit = 10,  
  order = "asc", data.table = TRUE, domain = "public")
```



**Arguments**

ledger	numeric - a ledger ID.
cursor	numeric - optional, a paging token - where to start from. Can also be "now".
limit	numeric - optional, the number of records to return.
order	character - optional, "asc" or "desc"
data.table	boolean - if TRUE, a data.table is returned. If FALSE or NULL, a list is returned.
domain	character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

data.table (by default) or list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/transactions-for-ledger.html>

**Examples**

```
getTransactions_Ledger("300000")
```

---

Ledger	<i>Ledger object.</i>
--------	-----------------------

---

**Description**

An R6 class representing a Ledger on the Stellar network. It is initialized from the `getLedgerDetail()` function.

**Usage**

```
Ledger
```

**Format**

An object of class `R6ClassGenerator` of length 24.

**Fields**

hash - a cryptographic signature of all of the data inside the ledger.  
sequence - the current sequence number of the ledger  
transaction\_count - how many transactions in the ledger.  
closed\_at - when the transaction was confirmed.  
total\_coins - how many lumens existed on the network at the time the ledger was confirmed.  
response - list containing the raw JSON response.

**Note**

<https://www.stellar.org/developers/guides/concepts/ledger.html>

**Examples**

```
genesis <- Ledger$new("1")
genesis$effects()
```

---

listToDF

*Convert parsed JSON responses to a tabular format.*

---

**Description**

Takes a successful response containing multiple records and converts it into a table format. It works by flattening semi-structured lists into a table where it can. For example, if there are multiple signatures to a transaction then it will warn the user before moving those values to a separate table.

**Usage**

```
listToDF(response)
```

**Arguments**

response      list - the list object returned by the other xlm functions.

**Value**

one or more data.tables

---

Orderbook

*Order book object.*

---

**Description**

An R6 class representing an orderbook on the Stellar network. It is initialized from the getOrderbook() function.

**Usage**

```
Orderbook
```

**Format**

An object of class R6ClassGenerator of length 24.

**Fields**

- bids - a cryptographic signature of all of the data inside the ledger.
- asks - the current sequence number of the ledger
- base - how many transactions in the ledger.
- counter - when the transaction was confirmed.
- domain - character - signalling to query the testnet or public network. Can be either "testnet" or "public".
- response - list containing the raw JSON response.

**Note**

<https://www.stellar.org/developers/horizon/reference/resources/orderbook.html>

---

postTransaction	<i>Post a transaction.</i>
-----------------	----------------------------

---

**Description**

Posts a new transaction to the Stellar Network.

**Usage**

```
postTransaction(body, domain = "public")
```

**Arguments**

- body                    character - base64 representation of transaction envelope XDR.
- domain                 - character - signalling to query the testnet or public network. Can be either "testnet" or "public".

**Value**

list

**Note**

<https://www.stellar.org/developers/horizon/reference/endpoints/transactions-create.html>

**Examples**

```
## Not run: postTransaction("aaaaafds098as0d9")
```

---

price	<i>Gets the current XLM price.</i>
-------	------------------------------------

---

**Description**

Make a call to the Binance API and get the current XLM market price.

**Usage**

```
price(currency = "USDT", live = FALSE)
```

**Arguments**

currency	character - current pairings are USD, ETH, BTC and BNB. USD is worked out indirectly via BTC with two API calls.
live	boolean - if true, a while loop will continuously get the latest price and print it to the screen.

**Value**

numeric

**Note**

Details of API can be found here: <https://github.com/binance-exchange/binance-official-api-docs/blob/master/rest-api.md>

**Examples**

```
price("USD")
```

---

Transaction	<i>Transaction object.</i>
-------------	----------------------------

---

**Description**

An R6 class representing a transaction on the Stellar network. It is initialized from the `getTransactionDetail()` function.

**Usage**

```
Transaction
```

**Format**

An object of class `R6ClassGenerator` of length 24.

**Fields**

ledger - the sequence number of the ledger from which the transaction was confirmed.

hash - the cryptographic signature of the transaction (sha256)

source\_account - the account that originates the transaction.

created\_at - when the transaction was submitted.

fee\_paid - how much fee was paid (in stroops.)

response - list containing the raw JSON response.

**Note**

<https://www.stellar.org/developers/guides/concepts/transactions.html>

**Examples**

```
## Not run: t = Transaction$new("afbe6f687c85d1f34ea18b73629be006518a2a3f5891760086340b4f1e834ccb")
```

# Index

## \*Topic **datasets**

- Account, [2](#)
- Ledger, [25](#)
- Orderbook, [26](#)
- Transaction, [28](#)

Account, [2](#)

distribution, [3](#)

findPaymentPath, [4](#)

friendbot, [5](#)

getAccountDetail, [5](#)

getAssets, [6](#)

getEffects, [7](#)

getEffects\_Account, [7](#)

getEffects\_Ledger, [8](#)

getEffects\_Transaction, [9](#)

getLedgerDetail, [10](#)

getLedgers, [10](#)

getOffers\_Account, [11](#)

getOperations, [12](#)

getOperations\_Account, [13](#)

getOperations\_Ledger, [14](#)

getOperations\_Transaction, [14](#)

getOrderbook, [15](#)

getPayments, [16](#)

getPayments\_Account, [17](#)

getPayments\_Ledger, [18](#)

getPayments\_Transaction, [19](#)

getTradeAggregations, [20](#)

getTrades, [21](#)

getTransactionDetail, [22](#)

getTransactions, [23](#)

getTransactions\_Account, [23](#)

getTransactions\_Ledger, [24](#)

Ledger, [25](#)

listToDF, [26](#)

Orderbook, [26](#)

postTransaction, [27](#)

price, [28](#)

Transaction, [28](#)